ITSS_10  IT Security Standard – IT System Acquisition & Development

Version 1.0 Effective 7 June 2016

Standard Statement

Maintaining a strong information security posture and managing information security risks relies on many disparate controls within infrastructure, operating environments and applications. The threats facing UNSW are changing and security attacks are focussed on security vulnerabilities in software applications as opposed to infrastructure devices, hence there is an increased focus on the development of applications.

The purpose of this standard sets out the baseline requirements for information security within the “System Acquisition and Development” lifecycle, in order to reduce the risk of vulnerabilities being introduced by applications acquired or developed internally by UNSW.

The methods that UNSW can adopt to implement information systems are as follows:

- In-house development.
- Acquisition of an implemented solution (commercial off the shelf package).
- Assigning the development and management to a specialised IT company (outsourcing).

Scope

This standard applies to all application development within UNSW, but is supplemented by a separate standard for the development of web applications – the ITSS_11 Web Applications Security Standard.

This standard is not intended to be an exhaustive list of security consideration for development – many sources of good practice are available such as Carnegie Mellon University Software Engineering Institute (SEI) Capability Maturity Model, Agile and other traditional Waterfall Models. UNSW software development teams must seek appropriate guidance, especially when using new application languages, tools and frameworks.

Are Local Documents on this subject permitted?  ☐ Yes  ☐ Yes, subject to any areas specifically restricted within this Document  ☐ No

Standard

1. Controls..................................................................................................................................... 2
   1.1 Security within UNSW Processes........................................................................................... 2
   1.2 Phase 1 & 2 - Feasibility & Analysis......................................................................................... 2
   1.3 Phase 3 - Design ...................................................................................................................... 2
   1.4 Security requirements mapping and solutions documentation............................................. 2
   1.5 Integration with UNSW security services.................................................................................. 2
   1.6 Phase 4 – Build.......................................................................................................................... 3
   1.7 Source code security ................................................................................................................. 3
   1.8 Segregation of development, test and production environments........................................... 3
   1.9 Phase 5 - Test ............................................................................................................................. 3
   1.10 Phase 6 - Deploy ....................................................................................................................... 3
2. Control Exceptions...................................................................................................................... 4
3. ISMS Mapping with Industry Standards.................................................................................... 4
4. Document Review, Approval & History...................................................................................... 4
   4.1 Quality Assurance................................................................................................................... 4
   4.2 Sign Off...................................................................................................................................... 5
1. Controls

1.1 Security within UNSW Processes

This standard is developed and structured into a number of sections in alignment with UNSW Processes. It is compatible with both traditional and agile methods of software development.

1.1.1 Process must include formal security checkpoints throughout its lifecycle:
- Phase 1 & 2: Feasibility & Analysis.
- Phase 3: Design.
- Phase 4: Build.
- Phase 5: Test.
- Phase 6: Deploy.

The key requirements for security checkpoints are outlined in the following sections of this document.

Phase 1 & 2 - Feasibility & Analysis

1.1.2 IT security requirements must be explicitly defined during the acquisition or development of an information system. Security Requirements must be based on:
   a) The Data Classification Standard and Handling Guidelines.
   b) A security threat and risk assessment relating to the information system and the controls required to reduce those risks to an acceptable level must be performed. The assessment must be conducted by an appropriately qualified professional. The assessment must include the following:
      - Model security threats to identify the threat actor (e.g. cyber attacker etc), the outcome in case the threat is realised (e.g. loss of information) and the impact on UNSW’s business operations (e.g. litigation, brand damage etc). evaluated against the UNSW Risk Evaluation Criteria.

1.1.3 Design and agree appropriate security controls for risk mitigation.

1.1.4 Development teams which utilise agile development methods must formally address the security requirements detailed above.

Phase 3 - Design

1.2 Security requirements mapping and solutions documentation

1.2.1 Solution designs must be documented for all information systems. A description of how the solution’s architecture meets IT security requirements defined during the Feasibility & Analysis stage must be included. Where there are gaps (e.g. end user impact) in meeting requirements analysis of the risk arising from these gaps, and the measures put in place to address this risks, must be documented.

1.3 Integration with UNSW security services

Flaws in applications often arise when applications implement bespoke security measures, for example user authentication or encryption of sensitive information. These flaws are avoided when applications make use of existing enterprise class security services to perform key security functions, for example by integrating with existing corporate directory, email or messaging systems that are scalable, robust and resilient.

UNSW IT maintains a number of enterprise security services. These include core security capabilities such as:
- Authentication and authorisation.
- Security logging and monitoring.
- Encryption.
- Anti-virus and malware.
- Intrusion detection and prevention.
- Backup.

1.3.1 Where possible and justified on a risk basis all applications must integrate and leverage existing enterprise security services. This drives enterprise capability, by maximising the value and use of these services and minimising the risks of projects creating security functionality that is not of adequate quality or robustness.
Phase 4 – Build

1.3.2 Where the project involves the acquisition of an off-the-shelf product, the Internal team / external Integrator (if integration is conducted by an external partner) must follow the solution design documentation to integrate the product with UNSW.

1.3.3 This standard recognises that application development methodologies are rapidly changing. Application development teams must:

a) Seek external security expertise and guidance when required.

b) Have access to appropriate security training and awareness materials when required, including annual awareness training on matters relevant to their development role.

c) Create a security culture by encouraging code review, development of security standards and guidelines for programming languages, both within and between development teams at UNSW and partner development communities.

d) Seek alignment with external sources of good practice so common mistakes and vulnerabilities are avoided, for example SQL injection, session hijacking or buffer overflow attacks.

e) Utilise ‘off the shelf’ libraries with proven security functionality and avoid recreating and developing this functionality, for example for cryptography, authentication, authorisation or logging.

1.4 Source code security

1.4.1 Development projects must employ code analysis techniques to identify potential security vulnerabilities or bad practices, during the development and build process. Where possible, this analysis should be automated and conducted on an ongoing basis, for example immediately after source code is checked into a source repository.

1.4.2 All significant issues found by source code analysis tools must be resolved before the code enters the release or production environment. It must be possible to track and report on issue resolution.

1.4.3 Applications that use sensitive information must be subject to an enhanced level of source code review, including peer review, if deemed necessary.

1.4.4 All source code must be maintained within a source code repository / configuration control system or equivalent. This system must be able to control access to named individuals and produce an auditable record of changes to the configuration of source code.

1.4.5 UNSW source code must be treated as “Sensitive” and adequately protected from unauthorised access, modification or deletion and have the ability to be rebuilt to any production version of an executable from the source code. Only executables of the source code must reside on the production environment.

1.4.6 Source code security patterns must be used to build in application security controls that effectively address well-known information security risks and vulnerabilities.

1.5 Segregation of development, test and production environments

1.5.1 Appropriate requirements and controls must be in place to separate the development, test and production environments:

a) Development, test, and production environments must be separated (physically or logically).

b) Where possible (if not document from a risk perspective) segregation of duties must exist between personnel assigned to the development, test environments and those assigned to the production environment. Access rights of test accounts must be managed according to the ITSS_05 User Access Management Standard.

c) Development and test environments must emulate the production environment from a security architecture perspective as closely as possible, including the use of a common operating system, database, web application server, and similar hardware. Only authorised release managers and system administrators have access to the production environment where production executable code for an application resides. Application
developers may have read-only access to production log and configuration files as deemed necessary.

1.5.2 All data classified as “Sensitive” used in any non-production environments must be masked and must not be used for development or testing. Data must be masked in accordance with the ITSS_02 Data Security Standard.

Phase 5 - Test

1.5.3 Once the acquisition or development of an information system has been completed and before it is placed in the production environment, a security review / testing must be conducted to ensure that the security requirements defined during the Feasibility & Analysis phase have been successfully designed and operating as intended.

1.5.4 A risk based security testing strategy (e.g., source code review, application business logic testing, vulnerability assessment, penetration testing) must be developed and executed for all applications. In the event that testing cannot be performed in a suitable production testing environment, it must be performed against the production environment.

1.5.5 Following the information system development and prior to its placement in production, the Directors of Solution Delivery and Service Delivery must verify that all predefined acceptance criteria has been met as agreed during the Feasibility & Analysis phase.

Phase 6 - Deploy

1.5.6 A migration plan must be developed for promoting a newly developed/acquired information system and data to the production environment. The migration plan must be signed off by management.

1.5.7 Information systems must be released in such a way that it is possible to perform a roll-back in a controlled, timely manner to restore the integrity of the target system to a previous state.

2. Control Exceptions

All exemption requests must be reviewed assessed, and approved by the relevant business stakeholder. Please refer to the ISMS Base Document for more detail.

3. ISMS Mapping with Industry Standards


<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>14 - System acquisition, development and maintenance</td>
<td>Software Security</td>
</tr>
</tbody>
</table>

4. Document Review, Approval & History

This section details the initial review, approval and ongoing revision history of the standard. Post initial review the standard will be presented to the ISSG recommending the formal UNSW policy consultation and approval process commence.

A review of this standard will be managed by the Chief Digital Officer on an annual basis.

4.1 Quality Assurance

This document was designed and created by external and internal consultants in consultation with internal key technical subject matter experts, business and academic stakeholders.
4.2 Sign Off

<table>
<thead>
<tr>
<th>Endorsement</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISSG - Information Security Steering Group</td>
<td>30th July 2015</td>
</tr>
<tr>
<td>ITC - Information Technology Committee</td>
<td>27th August 2015</td>
</tr>
<tr>
<td>CDO – Chief Digital Officer</td>
<td>7th June 2016</td>
</tr>
</tbody>
</table>

Accountabilities

<table>
<thead>
<tr>
<th>Responsible Officer</th>
<th>Chief Digital Officer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contact Officer</td>
<td><a href="mailto:ITpolicy@unsw.edu.au">ITpolicy@unsw.edu.au</a></td>
</tr>
</tbody>
</table>

Supporting Information

<table>
<thead>
<tr>
<th>Parent Document (Policy)</th>
<th>IT Security Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supporting Documents</td>
<td>Nil</td>
</tr>
<tr>
<td>Superseded Documents</td>
<td>Nil</td>
</tr>
<tr>
<td>UNSW Statute and / or Regulation</td>
<td>Nil</td>
</tr>
<tr>
<td>Relevant State / Federal Legislation</td>
<td>Nil</td>
</tr>
<tr>
<td>File Number</td>
<td>2016/16925 [ITSS_10]</td>
</tr>
</tbody>
</table>

Definitions and Acronyms

No terms have been defined

Revision History

<table>
<thead>
<tr>
<th>Version</th>
<th>Approved by</th>
<th>Approval date</th>
<th>Effective date</th>
<th>Sections modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>Vice-President, Finance and Operations</td>
<td>7 June 2016</td>
<td>7 June 2016</td>
<td>This is a new document</td>
</tr>
</tbody>
</table>