



OIT 62.407: Application Security

Responsible Executive: Information Security Officer
Responsible Office: Office of Information Technology
Related Policy:
Approved-On Date: 4/19/2011
Effective Date: 7/1/2011
Revision Date: 4/11/2011

I. Policy Statement

The Application Security Policy defines the steps necessary to provide adequate and effective protection for University applications.

II. Purpose

The Application Security Policy defines the high-level specifications for securely developing and deploying University applications.

III. Scope

This policy applies to all Commonwealth of Virginia (COV) systems and to the University IT network infrastructure.

IV. Requirements

The University ISO is accountable for ensuring the following steps are documented and followed:

Application Planning

1. Data Classification - Data used, processed or stored by the proposed application shall be classified according to the sensitivity of the data.
2. Risk Assessment – If the data classification identifies the system as sensitive, a risk assessment shall be conducted before development begins and after planning is complete.
3. Security Requirements – Identify and document the security requirements of the application early in the development lifecycle. For a sensitive system, this shall be done after a risk assessment is completed and before development begins.

4. Security Design – Use the results of the Data Classification process to assess and finalize any encryption, authentication, access control, and logging requirements. When planning to use, process or store sensitive information in an application, the University must address the following design criteria:
 - a. Encrypted communication channels shall be established for the transmission of sensitive information;
 - b. Sensitive information shall not be visibly transmitted between the client and the application; and
 - c. Sensitive information shall not be stored in hidden fields that are part of the application interface.

Application Development

The following requirements represent a minimal set of coding practices, which shall be applied to all applications under development.

5. Authentication – Application-based authentication and authorization shall be performed for access to data that is available through the application but is not considered publicly accessible.
6. Session Management - Any user sessions created by an application shall support an automatic inactivity timeout function.
7. Data storage shall be separated either logically or physically, from the application interface (i.e., design two or three tier architectures where possible).
8. The University shall not use or store sensitive data in non-production environments (*i.e., a development or test environment that does not have security controls equivalent to the production environment*).
9. Input Validation – All application input shall be validated irrespective of source. Input validation should always consider both expected and unexpected input, and not block input based on arbitrary criteria.
10. Default Deny – Application access control shall implement a default deny policy, with access explicitly granted.
11. Principle of Least Privilege – All processing shall be performed with the least set of privileges required.
12. Quality Assurance – Internal testing shall include at least one of the following: penetration testing, fuzz testing, or a source code auditing technique. Third

party source code auditing and/or penetration testing should be conducted commensurate with sensitivity and risk.

Note: Source code auditing techniques include, but are not limited to:

- a. Manual code review can identify vulnerabilities as well as functional flaws, but most agencies do not have the skilled security resources or time available within the software life cycle that a manual code review requires, and therefore, many agencies who decide to perform manual code reviews can only analyze a small portion of their applications;
- b. Application penetration testing tries to identify vulnerabilities in software by launching as many known attack techniques as possible on likely access points in an attempt to bring down the application or the entire system; and
- c. Automated source code analysis tools make the process of manual code review more efficient, affordable, and achievable. This technique of code audit results in significant reduction of analysis time, actionable metrics, significant cost savings, and can be integrated into all points of the development life cycle.

13. Configure applications to clear the cached data and temporary files upon exit of the application or logoff of the system.

Production and Maintenance

14. Production applications shall be hosted on servers compliant with the University Security requirements for IT system hardening.
15. Internet-facing applications classified as sensitive shall have periodic vulnerability scans run against the applications and supporting server infrastructure, and always when any significant change to the environment or application has been made. Any remotely exploitable vulnerability shall be remediated immediately. Other vulnerabilities should be remediated without undue delay.

Note: It is strongly recommended that the University adopt application vulnerability scanning and remediation for all internal sensitive applications as well.

Note: The Code of Virginia § 2.2-3803 (B) requires every public body in the COV that has an Internet web site to develop an Internet privacy policy and an Internet privacy policy statement that explains the policy to the public and is consistent with the requirements of the Code and is displayed on the public body's web site in a conspicuous manner.

V. Violations

Violations of this policy will be addressed in accordance relevant University and Commonwealth of Virginia policies, including University Policy 60.201 and Department of Human Resources Management Policy 1.75. The appropriate level of disciplinary action will be determined on an individual case basis by the appropriate executive or designee, with sanctions up to or including termination or expulsion depending upon the severity of the offense.

VI. Interpretation

The Information Security Officer is responsible for official interpretation of this policy. Questions regarding the application of this policy should be directed to the Office of Information Technology. The Information Security Officer reserves the right to revise or eliminate this policy.