



# **NISCC Technical Note 05/04**

**Issued 25 March 2004**

## **A Vulnerability Management Process for IT Product Vendors**

### **Key Points**

- **This technical note is aimed at software developers and vendors**
- **High level guidance aimed at minimising vulnerabilities is provided**
- **Security maintenance advice for products advice**
- **The use of co-ordination centres for multi-vendor vulnerabilities is recommended**

**National Infrastructure  
Security Co-Ordination Centre**  
PO Box 832  
London  
SW1P 1BG

Tel: 020 7821 1330 Ext 4511  
Fax: 020 7821 1686  
Email: [enquiries@nisc.gov.uk](mailto:enquiries@nisc.gov.uk)  
Web: [www.nisc.gov.uk](http://www.nisc.gov.uk)

## Introduction

1. This technical note addresses the processes that software product developers should put in place to minimise the occurrence of vulnerabilities in their products. Its intended audience is therefore developers and vendors of software products. The emphasis of the paper is on the process of identifying and remediating IT security vulnerabilities, but an overview of security related development practices is also included.

## Secure Development

2. Although software vulnerabilities cannot always be eliminated in advance from product development, some practices will help minimise the occurrence of vulnerabilities. These practices are as follows:

- Creation of a development environment which is secure from physical and electronic attack
- Following a structured development methodology
- Use of a configuration management system
- Development and use of backup and restoration processes for the configuration management system
- Development and use of coding standards, with guidance on secure programming and details of language-specific constructs that can lead to insecurities
- Ensuring that all external program interfaces validate data to ensure that it is the correct data type and cannot be interpreted by program as executable content
- Attempting to identify all of the errors that might occur and providing countermeasures provided by the product or its configuration documentation
- Vulnerability assessment of the design of the software and subsequent redesign if needed
- Peer review of all source code by developers in the same team
- Reviewing the source code against the coding standards by a developer in the same development team
- Further review of the source code against the coding standards by a developer not in the same development team
- Quality assurance testing of all software including third-party libraries through vulnerability testing
- Use of secure production and customer delivery mechanisms
- Use of a database to track known software flaws
- Development and use of a process to identify flaws that affect security
- Use of a process to fix all known software flaws systematically with high priority given to those that affect security
- Regression testing of all software fixes on supported versions of the software

3. Vulnerability testing should include testing all external software interfaces to check that the product validates all input correctly and that all errors are adequately handled.

4. The Common Criteria provides additional guidance on secure development environment classified according to level of assurance. Common Criteria provides a framework for formal IT security evaluation and certification. See NISCC Technical Note 01/03 for further details about Common Criteria evaluation and certification.

### **Basic Security Maintenance**

- It is as important to maintain security in a product during its lifecycle as it is to develop a secure product. The following practices will help maintain product security:
- Make a commitment to fix vulnerabilities in a timely manner
- Use of a database to track known software flaws
- Use of a process for handling reports of software flaws from customers and third-parties
- Active monitoring of vulnerabilities from public mailing lists and web sites
- Use of a process to fix all known software flaws systematically with high priority given to those that affect security
- Assess the criticality of the vulnerability (based on likelihood of exploitation and impact of the vulnerability if exploited)
- Determine how long the vulnerability will take to fix
- Regression testing of all software fixes on supported versions of the software
- Use of a process for validation of identity of software fixes (for example cryptographic checksums)
- Design and implement a framework for the secure management of information relating to software flaws
- Advertising a method of reporting vulnerabilities
- Use of a process for alerting customers to the appearance of a fix (for example a security advisory sent by email)
- Use of a secure process for issuing software fixes to customers
- Establish relationships with peer organisations for exchange of security related information
- Establish relationships with Computer Security Incident Response Teams (CSIRTs) and Critical National Infrastructure Protection organisations in your nation who may be able to provide information of vulnerabilities and exploits related to your products

5. These practices should enable a vendor to identify, track and fix software flaws as they arise. However, the practices do not address vulnerabilities which also affect other vendors' products.

### **Advanced Security Maintenance**

6. In the case where a vulnerability is assessed as being critical to the product, the following additional practices may be useful:

- Restrict vulnerability information on a need-to-know basis
- Inform customers as soon as a workaround is available

- Consider using public key certificates authenticated by a trusted third party to identify and provide privacy for communications relating to software flaws
- Consider a staged release of vulnerability information and fixes and workarounds if some customers are likely to be more affected than others

7. In the case where other vendors' products are also likely to be affected, ideally an independent vulnerability co-ordination centre should be involved. If an independent vulnerability co-ordination centre is involved, then the follow steps should also be taken:

- Inform the co-ordination centre of the criticality of the vulnerability and timescales to remediate the vulnerability
- Provide the co-ordination centre with details of the vulnerability, its impact, mitigation advice and remedial steps
- Agree with the co-ordination centre when a patch or workaround and an advisory can be released

8. The disclosure process will need to be co-ordinated with the co-ordination centre and may be subject to change depending on other vendors requirements. The appearance of exploits or publicity will lead to a shortening of the vulnerability disclosure timescales.

9. In the case where there is no co-ordination centre, vendors should co-ordinate with each other through their response teams or through trade associations or forums.