Internet Engineering Task Force
INTERNET-DRAFT
Valid for six months
Category: Best Current Practice

Steve Christey
MITRE
Chris Wysopal
@stake, Inc.
February 2002

# Responsible Vulnerability Disclosure Process
draft-christey-wysopal-vuln-disclosure-00.txt

## Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026. Internet Drafts are working documents of the Internet Engineering Task Force (IETF), its Areas, and its Working Groups. Note that other groups may also distribute working documents as Internet Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/1id-abstracts.html

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

## Copyright Notice

Copyright (C) The Internet Society (2002). All Rights Reserved.

## Abstract

New vulnerabilities in software and hardware products are discovered and publicized on a daily basis. The disclosure of vulnerability information has been a divisive topic for years. During the process of disclosure, many vendors, security researchers, and other parties follow a variety of unwritten or informal guidelines for how they interact and share information. Some parties may be unaware of these guidelines, or they may intentionally ignore them. This state of affairs can make it difficult to achieve a satisfactory outcome for everyone who uses or is affected by vulnerability information.

The purpose of this document is to describe best practices for a responsible disclosure process that involves vulnerability reporters, product vendors or maintainers, third parties, the security community, and ultimately customers and users.

Table of Contents

1 Introduction and Purpose ..... 3
1.1 Background ..... 3
1.2 Major Roles in Disclosure ..... 3
1.3 Motivations ..... 4
1.4 Goals of Responsible Disclosure ..... 5
2 Phases of Responsible Disclosure ..... 6
3 Responsibilities in the Phases of Vulnerability Disclosure ..... 7
3.1 Latent Flaw ..... 7
3.2 Discovery ..... 7
3.3 Notification Phase: Initial Notification ..... 8
3.3.1 Vendor Responsibilities ..... 8
3.3.2 Reporter Responsibilities ..... 9
3.4 Notification Phase: Vendor Receipt ..... 11
3.4.1 Vendor Responsibilities ..... 11
3.5 Validation Phase ..... 11
3.5.1 Vendor Responsibilities ..... 11
3.5.2 Reporter Responsibilities ..... 13
3.5.3 Coordinator Responsibilities ..... 14
3.6 Resolution Phase ..... 14
3.6.1 Vendor Responsibilities ..... 14
3.6.2 Reporter Responsibilities ..... 15
3.7 Release Phase ..... 16
3.7.1 Vendor Responsibilities ..... 16
3.7.2 Reporter Responsibilities ..... 18
3.7.3 Coordinator Responsibilities ..... 18
3.7.4 Customer Responsibilities ..... 19
3.7.5 Security Community Responsibilities ..... 19
3.8 Follow-Up Phase ..... 20
4 Policy Publication ..... 20
4.1 Vendor Policy ..... 20
4.2 Reporter Policy ..... 20
4.3 Coordinator Policy ..... 21
5 References ..... 21
5.1 Disclosure Policies ..... 21
5.2 Commentary on Disclosure Details ..... 21
5.3 Commentary on Disclosure Process ..... 22
5.4 Commentary on Advisories ..... 24
5.5 Commentary on Vendor Accessibility ..... 24
5.6 Discovery of Issues in the Wild ..... 25
5.7 Researcher Credibility and Vulnerability Reproduction ..... 26
5.8 Miscellaneous ..... 26
6 Acknowledgements ..... 26
7 Security Considerations ..... 26
8 Authors' Addresses ..... 27
9 Full Copyright Statement ..... 27

Document Conventions

The key words "REQUIRED", "MUST", "MUST NOT", "SHOULD", "SHOULD NOT", and "MAY" in this document are to be interpreted as described in "Key words for use in RFCs to Indicate Requirement Levels" [RFC2119].

## 1 Introduction and Purpose

This document provides guidance and recommendations for the community of developers, vendors, end users, researchers and security professionals who wish to perform responsible vulnerability disclosure within the information technology arena. For purposes of this document, the term "responsible" refers to the recognition of a formal, repeatable process for the reporting, evaluation, resolution and publication of vulnerability information. "Vulnerability" refers to any bug, flaw, behavior, output, outcome or event within an application, system, device, or service that could lead to increased risk or security exploit. For purposes of this document, we have standardized on the term "product" to encompass the full suite of products that are addressed by this document.

### 1.1 Background

Vulnerabilities are an inherent and unfortunate part of the design and development process. Vulnerability detection may occur during any phase of the product lifecycle, to include design, development, testing, implementation or operation. Ideally, vulnerabilities are largely prevented through a design process that considers security. However, due to a variety of reasons, many vulnerabilities are detected after a product is implemented in an operational environment and supporting customer objectives. A variety of legislative and social issues directly influences the process for vulnerability research, detection and response. Developers, customers and the security community all have divergent perspectives on the impact of vulnerabilities. Currently, vulnerability release is inconsistent and largely driven from the perspective of the party who has the greatest ability to control the process. In an effort to create a common framework by which objectives are met to the benefit of all parties, this document communicates a formal, repeatable process for addressing vulnerability disclosure in a responsible manner. This document provides a means to address the common goal of providing more secure products while reducing the risk to customers.

### 1.2 Major Roles in Disclosure

Several types of individuals or organizations may play a role in the process of vulnerability disclosure. These roles may overlap.

A Vendor is an individual or organization who provides, develops, or maintains software, hardware, or services, possibly for free.

A Customer is the end user of the software, hardware, or service that may be affected by the vulnerability.

A Reporter is the individual or organization that informs (or attempts to inform) the Vendor of the vulnerability. Note that the Reporter may not have been the initial discoverer of the problem.

A Coordinator is an individual or organization who works with the Reporter and the Vendor to analyze and address the vulnerability. Coordinators are often well-known third parties. Coordinators may have resources, credibility, or working relationships that exceed those of the reporter or vendors. Coordinators may serve as proxies for reporters, help to verify the reporter's claims, resolve conflicts, and work with all parties to resolve the vulnerability in a satisfactory manner. Note: while Coordinators can facilitate the responsible disclosure process for a vulnerability, the use of Coordinators by other parties is not a requirement.

The Security Community includes individuals or organizations whose primary goals include improving overall information technology security. The community includes security administrators and analysts, system administrators who are responsible for the security of their systems, commercial or non-profit organizations who provide security-related products or services, researchers and academics, informal groups, and individuals.

## 1.3 Motivations

Individuals and organizations have a wide variety of motivations (some in direct conflict with each other) that make the disclosure process more complex.

Vendors may have one or more of the following motivations. Some vendors believe that public notification may help their customers address vulnerabilities, at the possible cost of negative publicity. Some vendors may be unresponsive, or secretly fix vulnerabilities, for fear of negative publicity. Some vendors may not have the technical skills to understand the nature of the vulnerability and the risk that it poses.

Customers often wish to have secure products, but security features can make it more difficult to use those products. Many customers do not care about the nature of the vulnerability. However, there is a small percentage of customers for whom vulnerability information plays a critical role in their usage of products. Some vendors may be customers of other vendors.

Reporters have a variety of motivations. Because reporters are often the means through which vulnerability information is communicated, they have a major impact on how the disclosure process is followed. Reporters may be motivated by altruism ("to make computers more secure"), recognition or fame, marketing to highlight technical skills (for individuals as well as companies), forcing unresponsive vendors to address a vulnerability, curiosity or the challenge of vulnerability analysis, or malicious intent to damage the reputations of specific vendors, wreak havoc, or cause financial damage to customers. The vague goals of altruism are often open to different interpretations by different reporters. Reporters may be inexperienced, malicious, or have insufficient resources to follow the full process of disclosure. Reporters are seldom compensated for their important role in enhancing Internet security.

The motivations for members of the security community may vary depending on the specific tasks that are being undertaken by the members. Community members may have motivations that include those of vendors, customers, and/or reporters. In addition, members of the security community may wish to track trends in vulnerabilities, identify new types of vulnerabilities, or design new products and processes to reduce the impact of vulnerabilities.

Coordinators are often members of the security community, and as such may share the same motivations. Coordinators may also be required by their mission or contract to perform this role.

## 1.4 Goals of Responsible Disclosure

The goals of responsible disclosure include:

1) Ensure that vulnerabilities can be identified and eliminated effectively and efficiently for all parties.

2) Minimize the risk to customers from vulnerabilities that could allow damage to their systems.

3) Provide customers with sufficient information for them to evaluate the level of security in vendors' products.

4) Provide the security community with the information necessary to develop tools and methods for identifying, managing, and reducing the risks of vulnerabilities in information technology.

5) Minimize the amount of time and resources required to manage vulnerability information.

6) Facilitate long-term research and development of techniques, products, and processes for avoiding or mitigating vulnerabilities.

7) Minimize the amount of antagonism that often exists between parties as a result of different assumptions and expectations, due to the lack of consistent and explicit disclosure practices.

## 2 Phases of Responsible Disclosure

Following are the basic phases of the responsible vulnerability disclosure process. Some of these phases may be bypassed in specific situations with agreement across all parties. In other cases, one or more parties may not be responsible, skipping some phases.

1) Latent Flaw. A flaw is introduced into a product during its design, specification, development, installation, or default configuration.

2) Discovery. One or more individuals or organizations discover the flaw through casual evaluation, by accident, or as a result of focused analysis and testing. In some cases, knowledge of the flaw may be kept within a particular group. A vulnerability report or an exploit program may be discovered "in the wild," i.e., in use by malicious attackers or made available for use and distribution.

3) Notification. A reporter or coordinator notifies the vendor of the vulnerability ("Initial Notification"). In turn, the vendor provides the reporter or coordinator with assurances that the notification was received ("Vendor Receipt").

4) Validation. The vendor or other parties verify and validate the reporter’s claims ("Reproduction").

5) Resolution. The vendor and other parties also try to identify where the flaw resides ("Diagnosis"). The vendor develops a patch or workaround that eliminates or reduces the risk of the vulnerability ("Fix Development"). The patch is then tested by other parties (such as reporter or coordinator) to ensure that the flaw has been corrected ("Patch Testing").

6) Release. The vendor, coordinator, and/or reporter release the information about the vulnerability, along with its resolution. The vendor may initially release this information to its customers and other organizations with which it may have special relationships ("Limited release"). The vendor or other parties may then release the information - possibly with additional details - to the security community.

7) Follow-up. The vendor, customer, coordinator, reporter, or security community may conduct additional analysis of the vulnerability or the quality of its resolution.

# 3 Responsibilities in the Phases of Vulnerability Disclosure

## 3.1 Latent Flaw

The following recommendations identify how most latent flaws can be avoided.

1) The Vendor SHOULD ensure that programmers, designers, and testers are knowledgeable about common flaws in the design and implementation of products.

Rationale: Some classes of vulnerabilities are well-known and can be easily exploited using repeatable techniques. Educated programmers, designers, and testers can identify and eliminate vulnerabilities before the product is provided to customers, or prevent their introduction into the product in the first place.

2) Customers SHOULD configure their products and systems in ways that eliminate latent flaws or reduce the impact of latent flaws, including (1) removing default services that are not necessary for the operation of the affected systems, (2) limiting necessary services only to networks or systems that require access, (3) using the minimal amount of access and privileges necessary for proper functioning of the products, and (4) using security features of the product or operating system that reduce the chance that a flaw can be successfully exploited.

Rationale: Many computer intrusions involve the exploitation of vulnerabilities in network services that are unnecessary for typical operating environments. In some cases, system configuration can reduce the overall risk of vulnerabilities (known and unknown). For example, the Code Red and Nimda worms of 2001 were largely successful because of these factors.

3) The Security Community SHOULD track all known vulnerabilities to identify new classes of vulnerabilities, educate the public about these types of vulnerabilities, and find ways to detect or prevent them in the development, testing, and deployment of products.

## 3.2 Discovery

1) The Reporter SHOULD make a reasonable effort to ensure that: - the vulnerability is real - the process of getting the product into a known exploitable state is repeatable - the vulnerability has not already been reported by the vendor or well-established vulnerability information sources

Rationale: Some vulnerabilities are re-discovered after they have already been fixed, or the reporter has introduced the problem due to

misconfiguration, or the reporter identifies the symptoms of the vulnerability without determining the cause. If the reporter ensures that the problem is new and real, then the reporter will will avoid unnecessarily consuming the time and resources spent by vendors and other parties in investigating the problem.

Note: in some cases, a reporter may not be able to make a reasonable effort due to limitations of time, resources, access to the product, or expertise. In some cases, the problem may only appear intermittently, or the product is only temporarily accessible to the reporter (e.g., when the reporter is a consultant who discovers the problem in products that a customer uses). In other cases, the reporter may discover information about the vulnerability without having any access to the product.

Note: in some cases, the reporter may be able to coerce the product into a state that is known to be exploitable, without creating a fully working exploit program (e.g., a buffer overflow with a long string of 'A' characters may produce a result that shows that the instruction pointer has been overwritten). This is considered a reasonable effort.

### 3.3 Notification Phase: Initial Notification

To facilitate the disclosure process, Vendors need to be accessible to Reporters, and Reporters need to find and use the appropriate communication channels for notifying Vendors.

#### 3.3.1 Vendor Responsibilities

1) The Vendor MUST make it as easy as possible for Reporters, Coordinators, Customers, and the Security Community to notify the Vendor of vulnerabilities.

Rationale: It is often difficult for reporters or other parties to notify vendors of vulnerabilities, especially if the reporters are not customers. This may cause the parties to bypass other phases of the disclosure process, or adopt a policy that avoids vendor notification because of previous bad experiences with vendors.

2) The Vendor SHOULD establish a Security Response Capability (SRC) that consists of one or more individuals or groups that are responsible for responding to vulnerability reports, verifying vulnerabilities, releasing bulletins, etc.

3) The Vendor SHOULD ensure that its staff knows how to recognize a reported security issue and direct it to the Security Response Capability. This recommendation applies to staff who provide support online, over the telephone, in person, or through some other means by which reporters may interact with the Vendor.

4) If the Vendor can control the e-mail addresses that it uses (e.g., it has its own domain name), then the Vendor SHOULD define and publish the "secalert" alias for use in vulnerability notification.

Rationale: Currently, Vendors use a variety of aliases for notification, including "security-alert," "security," and "support." Some Vendors may use the "security" alias for physical security facilities. The "security" alias is also defined in RFC2142 for use in incident handling. The "security-alert" alias is longer than 8 characters and contains a dash, which could make it more difficult to use or locate in search engines. The "secalert" alias is not commonly used at this time, and as such it does not have the types of issues that some commonly-used aliases have.

Note: smaller vendors may not be able to control which e-mail addresses they use.

5) If the Vendor operates a web site or other means of distributing information regarding its product, then the Vendor SHOULD create and publish a "security" page or folder that identifies how vulnerability reports should be made. The Vendor SHOULD make this page easy to find from other locations, such as a separate contact page or index.

6) The Vendor MUST provide a facility for individuals or organizations who are not Customers to report vulnerabilities. The Vendor SHOULD NOT require (1) an active technical support number, (2) telephone access that is not toll-free, or (3) user registration for a web site or other facility that would be used for reporting.

Rationale: As described earlier, some reporters or coordinators are not necessarily customers of the Vendor. If the Vendor is not accessible to them, then they will be more likely to bypass other aspects of this process.

7) The Vendor SHOULD recognize that inexperienced or malicious reporters may not use proper notification, and define its own procedures for handling such cases.

### 3.3.2 Reporter Responsibilities

1) The Reporter SHOULD make reasonable efforts to use the appropriate channels for notifying the Vendor of the vulnerability:

(a) The Reporter SHOULD attempt to notify the vendor through the channels described in this section.

(b) If the Vendor is not accessible through those channels, then the Reporter MAY attempt to contact the vendor through technical support.

Note: in some cases, a reporter may not be able to make a reasonable effort due to time limitations, lack of proper access to the vendor, inexperience, expense, prohibitions by the reporter's own organization, or the reporter does not meet some criteria for notification (e.g., a support contract number).

2) If the Reporter is unable to notify the Vendor, then the Reporter SHOULD ask a Coordinator to notify the Vendor. The Reporter SHOULD provide the Coordinator with a list of contacts or mechanisms that were used to attempt to notify the Vendor.

Rationale: a Coordinator may appear more credible than the Reporter, or have a previously established relationship with the Vendor. The Reporter may be prohibited from disclosing the vulnerability directly to the Vendor.

Note: the Coordinator will not necessarily have a different way of reaching the Vendor than the Reporter does.

3) The Reporter and/or Coordinator SHOULD record the date of notification.

Rationale: This helps Customers, Reporters, Coordinators, and the Security Community track how long it takes for a Vendor to resolve a vulnerability after the initial notification.

4) The Reporter SHOULD provide the Vendor, and the Coordinator (if any), with all known details of the issue, including any programs, scripts, or pseudo-code that would allow the Vendor to reproduce and/or confirm the vulnerability.

Rationale: such details make it easier for the Vendor and Coordinator to reproduce and diagnose the vulnerability, which then allows the Vendor to identify or develop a resolution more quickly.

Note: some vulnerabilities may be theoretical or not well-understood in this phase of the disclosure process, and the Reporter may not have developed programs that exploit the problem. In other cases, the Reporter may be using proprietary programs to demonstrate the vulnerability.

## 3.4 Notification Phase: Vendor Receipt

### 3.4.1 Vendor Responsibilities

1) The Vendor MUST notify the Reporter and involved Coordinators that the Vendor has received the notification. This Receipt does not necessarily imply that the Vendor has researched or reproduced the vulnerability, only that the Vendor is aware of the notification.

Rationale: if the Vendor does not respond, then the Reporter or Coordinator may not be sure if the Vendor is truly aware of the reported vulnerability, and/or if the Vendor intends to resolve the vulnerability. This often causes Reporters or Coordinators to bypass later phases of the disclosure process in order to warn customers of the risks to their systems.

2) The Vendor MUST provide the Reporter and involved Coordinators with a Receipt within 7 days.

Rationale: Other time frames (such as 5 business days) were considered but deemed unworkable due to international issues (e.g., "work weeks" may fall on different days in different countries, there are different national or religious holidays). Defining a time frame relative to the Vendor or Reporter could not work without some form of communication between both parties.

Note: small but responsible Vendors or individuals may not be able to provide this degree of responsiveness, especially during vacation periods. Reporters and Coordinators SHOULD take this into account during the notification phase. Small, responsible Vendors SHOULD post some clear notification when it is known that such delays will occur.

3) If the Vendor’s receipt message is automatically generated, then it SHOULD include a time period or date by which an individual (or the Security Response Capability) will provide follow-up on the reported vulnerability. The time period SHOULD NOT exceed 10 days.

4) Within 10 days of initial notification, the Vendor’s Security Response Capability SHOULD provide a clear response to the Reporter and any involved Coordinators.

## 3.5 Validation Phase

### 3.5.1 Vendor Responsibilities

1) If the vulnerability is found in a supported product, the Vendor MUST either (1) reproduce the vulnerability, (2) determine if there is enough evidence for the existence of the vulnerability when it

cannot be reproduced, (3) determine if the vulnerability is already known (and possibly resolved), or (4) work with the Reporter to determine if the vulnerability is related to the specific environment in which it was discovered (including configuration errors or interactions with other products).

2) If the vulnerability is found in an unsupported or discontinued product, the Vendor MAY refuse to validate the vulnerability. However, the Vendor MUST ensure that the reported vulnerability does not exist in supported product versions or other supported products based on the vulnerable product.

3) The Vendor SHOULD NOT assume that the risk or impact of the vulnerability is limited to what has been identified by the Reporter or involved Coordinator.

Rationale: The Reporter or involved Coordinator may not have sufficient experience or time to identify the full scope of the problem. Sometimes, a theoretical vulnerability is later found to be more easily exploitable as a result of follow-on analysis or the creation of a tool. For example, it may be easy for a Reporter to find evidence of a buffer overflow vulnerability by sending a long argument that causes a product to crash. It is an indicator that a carefully crafted program could be used to execute arbitrary code. The Reporter and Vendor may not have the skills or resources to create such a program, but such a program could be created in the future.

4) The Vendor SHOULD examine its product to ensure that it is free of other problems that are similar to the reported vulnerability.

Rationale: some Vendors reproduce and resolve the specific issue that is identified by the Reporter without extending their analysis to see if similar mistakes were made elsewhere in the product. The Reporter, others in the Security Community, or hackers may conduct follow-on research to find these other vulnerabilities. This can result in a cycle in which vulnerabilities are discovered and patched so often that it becomes difficult for customers to manage the volume of resolutions that they need to apply.

5) The Vendor MUST consult with the Reporter and involved Coordinators when more information or analysis is needed.

6) The Vendor SHOULD provide status updates to the Reporter and any involved Coordinators every 7 days. The Vendor MAY negotiate with the parties for less frequent updates.

7) The Vendor MUST notify the Reporter and any involved Coordinators when the Vendor is able to reproduce the vulnerability.

8) The Vendor SHOULD attempt to resolve the vulnerability within 30 days of initial notification.

9) If the Vendor cannot resolve the vulnerability within 30 days, then the Vendor MUST provide the Reporter and involved Coordinators with specific reasons why the vulnerability cannot be resolved.

10) If the Vendor is aware of other vendors that share the same codebase as the affected product, then the Vendor MUST either (1) notify those vendors, or (2) notify a Coordinator that other Vendors may be affected by the reported vulnerability.

### 3.5.2 Reporter Responsibilities

1) The Reporter SHOULD work with the Vendor in a timely fashion to explain the vulnerability and conduct further analysis.

Rationale: if a problem is sufficiently complex or only appears in a portion of deployed systems, then the Vendor may not be able to reproduce the issue. In other cases, the Vendor may not understand the problem. If the Reporter is slow to respond, then this can extend the time window during which Customers are at risk.

2) If the Vendor does not understand the nature, risk, or resolution of the vulnerability, then the Reporter or involved Coordinators SHOULD provide the Vendor with resources that help to explain the vulnerability.

Note: Some Vendors may require - or insist - upon extensive consultation to identify the vulnerability. Reporters and Coordinators may not have the time or resources to provide such assistance.

3) If the Reporter does not have the time or resources to conduct such analysis, then the Reporter SHOULD notify the Vendor and suggest alternate contacts (such as Coordinators) who may be able to assist the Vendor. The Reporter SHOULD NOT attempt to bypass later phases.

4) If the Reporter finds that the Reporter is in error, then the Reporter SHOULD notify the Vendor and involved Coordinators.

Rationale: if a Reporter does not perform this notification, then the Vendors or Coordinators may continue to spend unnecessary resources on further analysis of the issue.

5) The Reporter SHOULD grant time extensions to the Vendor if there is evidence that the Vendor is acting in good faith to resolve the vulnerability.

6) If the Vendor is unresponsive or disagrees with the Reporter's findings, then the Reporter SHOULD involve a Coordinator.

### 3.5.3 Coordinator Responsibilities

1) The Coordinator MUST attempt to resolve any conflicts or technical disagreements that arise between the Reporter and the Vendor.

2) If a Vendor is unresponsive or does not appear to be acting in good faith to resolve the vulnerability, then the Coordinator SHOULD attempt to convince the Vendor to follow the proper process.

3) If a Reporter is unresponsive or does not appear to be acting in good faith to resolve the vulnerability, then the Coordinator SHOULD attempt to convince the Reporter to follow the proper process.

4) The Coordinator SHOULD work with the Vendor and Reporter to determine if other Vendors are affected by the same problem.

5) The Coordinator SHOULD work with the Vendor and Reporter to identify time extensions (if any) that are acceptable to all parties.

## 3.6 Resolution Phase

The "Resolution" of a vulnerability involves action regarding one or more of the following:

- patch creation
- recommendation of configuration change
- design change
- workaround
- no action

If the Vendor does not participate or is unresponsive, then the Reporter and Coordinator might not be able to create a patch or change the design of the product.

### 3.6.1 Vendor Responsibilities

1) The Vendor MUST identify the fundamental nature of the flaw within the source code or in the design of the product ("Diagnosis").

2) The Vendor MUST either (1) provide a patch, configuration change, or workaround that appropriately reduces or eliminates the risk of the vulnerability ("Fix Development"), or (2) provide the Reporter and involved Coordinators with specific reasons for its inaction.

3) The Vendor SHOULD request time extensions from the Reporter and involved Coordinators when necessary.

4) The Vendor SHOULD test the patches, configuration changes, and workarounds sufficiently to ensure that either (1) they do not adversely affect the operation of the product, or (2) it is clear which conditions may adversely affect the operation of the product.

Rationale: Vendors may be pressured to quickly resolve vulnerabilities without sufficient testing, especially when Reporters have bypassed the Notification or Validation phases. As a result, the resolution may adversely affect more systems than necessary.

5) The Vendor MUST provide the Reporter and involved Coordinators with all known configuration changes or workarounds that address the vulnerability ("Fix Development").

6) The Vendor SHOULD provide the Reporter and involved Coordinators with any patches ("Patch Testing").

Rationale: this helps the Reporter and Coordinator to confirm that the vulnerability has been reduced or eliminated.

Note: the Vendor’s business model may require that only supported Customers can have access to a patch, which could exclude Reporters or Coordinators. Such Vendors should recognize that this practice may result in an incomplete patch that does not address the vulnerability in question.

7) If the Reporter is unresponsive or uncooperative, or a dispute arises, then the Vendor SHOULD work with a Coordinator to identify the best available resolution for the vulnerability.

### 3.6.2 Reporter Responsibilities

1) The Reporter SHOULD recognize that it may be difficult for a Vendor to resolve a vulnerability within 30 days if (1) the problem is related to insecure design, (2) the Vendor has a diverse set of hardware, operating systems, and/or product versions to support, or (3) the Vendor is not skilled in security.

2) The Reporter SHOULD grant time extensions to the Vendor if the Vendor is acting in good faith to resolve the vulnerability.

3) If the Vendor is unresponsive or uncooperative, or a dispute arises, then the Reporter SHOULD work with a Coordinator to identify the best available resolution for the vulnerability.

## 3.7 Release Phase

### 3.7.1 Vendor Responsibilities

1) The Vendor SHOULD work with the Reporter and involved Coordinators to arrange a date after which the vulnerability information may be released.

2) The Vendor MAY ask the Reporter and Coordinator to allow a "Grace Period" up to 30 days, during which the Reporter and Coordinator do not release details of the vulnerability that could make it easier for hackers to create exploit programs.

Rationale: a grace period provides Customers with a time period in which they can fix their systems. During this time, the lack of details may make it more difficult or resource-intensive for attackers to determine the nature of the vulnerability and craft an exploit. However, some security-aware Customers desire such details so that they can better decide whether the resolution of the vulnerability is appropriate for their environment. In addition, some members of the Security Community desire such details in order to (1) enhance tools or techniques to detect vulnerable systems on Customer networks (such as vulnerability scanners), (2) enhance tools or techniques to detect attempts to exploit vulnerabilities on Customer networks (such as intrusion detection systems), (3) provide databases or other information that Customers use to identify and prioritize vulnerabilities that may affect the Customer’s enterprise, and (4) perform research and trend analysis.

3) If the Reporter has not properly followed the process and publicly announces the vulnerability, then the Vendor SHOULD post its awareness of the vulnerability, and the Vendor’s progress in its resolution, to appropriate forums.

Rationale: this allows Customers and the Security Community to know that the Vendor is aware of the problem and working to resolve it.

Note: some Vendors may not wish to acknowledge such vulnerabilities until a patch is available.

4) If a Reporter has properly followed the process, then the Vendor MUST provide credit to that reporter.

5) If a Coordinator has properly followed the process, then the Vendor SHOULD provide credit to the Coordinator.

6) If a Reporter has not properly followed the process and publicly announces the vulnerability, then the Vendor MAY provide credit to the reporter.

Rationale: Some people believe that even if a reporter has not followed the procedures properly, the reporter has still provided valuable information that is useful to the Vendor, Customers, Coordinators, and the Security Community, and academic integrity would dictate that reporters should be credited. However, since credit is a motivation for some reporters, others believe that irresponsible reporters should not be encouraged to bypass the process and still get credit.

7) The Vendor MUST NOT assume that the lack of vulnerability details will prevent the creation of an exploit.

Rationale: If the Vendor provides source code for the product, then any entity who has access to the product could easily determine the specific locations of the vulnerability and identify possible attack vectors that reach the vulnerable code. If the Vendor does not provide source code, then any entity who has access to a patch could use reverse engineering techniques to determine how the code was changed, then infer the nature of the vulnerability.

8) The Vendor SHOULD cryptographically sign all patches using a method that is commonly accessible on the platforms for the Vendor's product. The Vendor should clearly advertise its cryptographic key and provide cryptographic checksums for its patches.

Rationale: This increases the assurance that the patches from the Vendor are authentic.

9) The Vendor SHOULD provide an easily accessible mechanism for Customers and the Security Community to obtain all security advisories, such as a web page. The most recent advisory SHOULD be listed first.

10) The Vendor SHOULD provide a mechanism for notifying Customers and the Security Community when new advisories are published.

11) The Vendor SHOULD provide a means for the Security Community to identify which reported vulnerabilities are genuine, but are not regarded by the Vendor as important enough to merit a security advisory.

Rationale: Vendors are often unwilling to release security advisories unless the security issue is critical for its Customers. This can reduce operating expenses for the Vendor and most Customers. However, some members of the Security Community, and some Customers, also prefer to protect themselves against less serious vulnerabilities. If a Vendor does not at least indicate to its security-aware Customers that a security-related resolution is available, then those Customers may remain at risk for

vulnerabilities that they would otherwise wish to resolve.

12) The Vendor SHOULD provide an easily accessible indicator that allows a Customer to determine if the resolution has been applied to a system, e.g., by modifying the product's version number or providing the Customer with a tool that identifies the resolutions that have been applied to a product.

### 3.7.2 Reporter Responsibilities

1) The Reporter SHOULD work with the Vendor and involved Coordinators to arrange a date after which the vulnerability information may be released.

2) If the Vendor has not resolved the vulnerability within a time frame that is allowed by this process, then the Reporter SHOULD work with a Coordinator to announce the vulnerability to Customers and the Security Community.

3) If another reporter has not properly followed the process and publicly announced the vulnerability, then the Reporter MAY announce that the Reporter was responsibly following the disclosure process with the Vendor and involved Coordinators.

4) If a Vendor requests a Grace Period, then the Reporter SHOULD follow the Grace Period before releasing details of the vulnerability.

5) After the Grace Period, the Reporter MAY release additional details. The Reporter SHOULD carefully consider how much detail is needed by Customers and the Security Community.

Note: in some cases, the nature of the vulnerability could make it difficult or impossible to release vulnerability details that do not allow someone to exploit the vulnerability.

6) The Reporter SHOULD provide credit to any Vendor and/or Coordinator who has followed the process.

### 3.7.3 Coordinator Responsibilities

1) The Coordinator SHOULD work with the Vendor and Reporter to arrange a date after which the vulnerability information may be released.

2) If the Vendor requests a Grace Period, the Coordinator SHOULD follow the Grace Period and encourage the Reporter to follow the Grace Period.

3) The Coordinator SHOULD provide credit to any Vendor and/or Reporter who properly follows the process.

4) The Coordinator MAY provide credit to a reporter who has not properly followed the process.

### 3.7.4 Customer Responsibilities

1) The Customer MUST NOT assume that the lack of details will prevent the creation of an exploit.

2) If the Vendor has released information regarding the vulnerability, then the Customer SHOULD assume that the information is credible. The Customer SHOULD NOT require that the vulnerability be demonstrated before applying the resolution.

3) If the Vendor has not released such information, but a well-established Reporter or Coordinator has, then the Customer SHOULD assume that the information is credible. The Customer SHOULD NOT require that the vulnerability be demonstrated before applying the resolution.

4) If vulnerability information has been released and a Grace Period exists, then the Customer SHOULD apply the resolution to its systems during the Grace Period.

5) Where possible, the Customer SHOULD test any patches, configuration changes, or workarounds on test systems before making the changes in an operational environment.

6) The Customer SHOULD inform the Vendor and the Security Community if a patch, configuration change, or workaround does not appear to work properly.

7) The Customer SHOULD give preference to products whose Vendors follow responsible disclosure practices.

### 3.7.5 Security Community Responsibilities

1) The Security Community SHOULD publicly recognize all Vendors, Reporters, and Coordinators who follow responsible vulnerability disclosure.

2) The Security Community SHOULD adopt a set of terms that allows all parties to describe the inherent risk or impact of a vulnerability that can be interpreted in various environments, threat levels, and policies.

Rationale: Customers have varying operational needs at different levels of security, which can make it difficult to define a "one size fits all" risk level for any vulnerability. Current terminology often uses a "High, Medium, Low" breakdown, but there are no formal definitions. As such, this terminology is used inconsistently, partially because it is based on the perspective of the entity who is using it. It is also insufficient to capture the complexity and tradeoffs of vulnerabilities in today's environment.

### 3.8 Follow-Up Phase

1) The Vendor SHOULD clearly notify Customers and the Security Community when a resolution is (a) faulty, or (b) revised.

2) The Vendor SHOULD NOT re-release the same advisory for newly discovered, closely related vulnerabilities.

Rationale: The re-release of an advisory may not be noticed as well by Customers, which could cause the Customers to believe that their systems are secure because they applied the resolution that was identified in the original advisory.

## 4 Policy Publication

### 4.1 Vendor Policy

A Vendor SHOULD publish a policy and procedures statement that includes the following information:

1) Where it complies (and does not comply) with the process outlined in this document.

2) The typical amount of time after notification that the Vendor requires to produce a resolution.

3) The Grace Period, if any, that the Vendor wishes to observe.

4) How the Vendor determines whether a reported problem is serious enough to merit a security advisory.

### 4.2 Reporter Policy

If a Reporter is a member of the Security Community and the Reporter frequently finds new vulnerabilities, then the Reporter SHOULD publish a policy and procedures statement that includes the following information:

1) Where it complies (and does not comply) with the process outlined in this document.

2) The maximum Grace Period that the Reporter is willing to follow.

### 4.3 Coordinator Policy

A Coordinator SHOULD publish a policy and procedures statement that includes the following information:

1) Where the Coordinator complies (and does not comply) with the process outlined in this document.

2) The maximum Grace Period that the Coordinator is willing to follow.

## 5 References

Note: many of these references identify posted messages to security-related mailing lists. These messages often resulted in long threads that explore the related issues in more depth.

### 5.1 Disclosure Policies

RFPolicy 2.0
http://www.wiretrip.net/rfp/policy.html

Bugtraq Frequently Asked Questions
http://www.securityfocus.com/popups/forums/bugtraq/faq.shtml

NTBugtraq Disclosure Policy
http://ntbugtraq.ntadvice.com/default.asp?sid=1&pid=47&aid=48

CERT/CC Vulnerability Disclosure Policy
http://www.kb.cert.org/vuls/html/disclosure/

ACROS ASPR Notification and Publishing Policy
http://www.acros.si/aspr_policy.html

NMRC policy
http://www.nmrc.org/advise/policy.txt

@stake Security Advisory Disclosure Policy
http://www.atstake.com/research/policy/index.html

### 5.2 Commentary on Disclosure Details

"Full Disclosure is a necessary evil"
Elias Levy
SecurityFocus web site
August 16, 2001
http://www.securityfocus.com/news/238

"It's Time to End Information Anarchy"
Scott Culp
Microsoft web site
October 2001
http://www.microsoft.com/technet/columns/security/noarch.asp

"Security in an Open Electronic Society"
Elias Levy
SecurityFocus web site
October 21, 2001.
http://www.securityfocus.com/news/270

"Full Disclosure"
Bruce Schneier
Crypto-Gram Newsletter
November 15, 2001
http://www.counterpane.com/crypto-gram-0111.html#1

"Script Kiddies Suck"
Marcus Ranum
Black Hat Briefings presentation
July 2000
http://web.ranum.com/usenix/blackhat-2000-keynote.mp3

"The Network Police Blotter: The Slaughter of the Innocents"
Marcus Ranum
;Login: magazine
October 2000
http://web.ranum.com/usenix/ranum_5_temp.pdf

## 5.3 Commentary on Disclosure Process

"Bugs in the Disclosure Process"
Ivan Arce
TISC Insight, Volume 3, Issue 3
February 9, 2001
http://tisc.corecom.com/newsletters/33.html

"SUMMARY: Bug announcement rule of thumb."
Bill Stout
NTBugtraq mailing list
August 13, 1998
http://marc.theaimsgroup.com/?l=ntbugtraq&m=90310164223252&w=2

"Microsoft admits IE security alert lapse"
Wendy McAuliffe
ZDNet
November 19, 2001
http://www.zdnet.com/filters/printerfriendly/
0,6061,2825716-2,00.html

"RFP2K03: Contemplations on dvwssr.dll and its affects on life"
Rain Forest Puppy
Bugtraq mailing list
April 20, 2000
http://www.securityfocus.com/archive/1/56394

"Xato Advisory: Win2k/XP Terminal Services IP Spoofing"
Xato
Bugtraq mailing list
November 14, 2001
http://www.securityfocus.com/archive/1/240248

"Vulnerability Escrow (was: Extreme Hacking)"
Crispin Cowan
NFR Wizards mailing list
July 7, 1999
http://archives.neohapsis.com/archives/nfr-wizards/1999_2/0416.html

"Can we afford full disclosure of security holes?"
Richard M. Smith
Bugtraq mailing list
August 10, 2001
http://www.securityfocus.com/archive/1/203499

"Anti-Web 'Vulnerability' is a false alarm"
Doug Hoyte
Vuln-Dev mailing list
December 1, 2001
http://marc.theaimsgroup.com/?l=vuln-dev&m=100732828128718&w=2

"Windows of Vulnerability: A Case Study Analysis"
William A. Arbaugh, William L. Fithen, John McHugh
IEEE Computer
December 2000

"Sun denies Unix flaw"
John Geralds
vnunet.com
November 20, 2001
http://www.vnunet.com/News/1126973

"Open Response To Microsoft Security - RE: It's Time to End Information Anarchy"
Steve Manzuik
Vuln-Dev mailing list
October 17, 2001
http://archives.neohapsis.com/archives/vuln-dev/2001-q4/0195.html

"A Step Towards Information Anarchy: A Call To Arms"
hellNbak
Nomad Mobile Research Center
November 2, 2001
http://www.nmrc.org/InfoAnarchy/InfoAnarchy.htm

"To Disclose or Not to Disclose, That Is the Question"
Mark Joseph Edwards
Windows 2000 Magazine
June 27, 2001
http://www.windowsitsecurity.com/Articles/Index.cfm?ArticleID=21618

"Towards a responsible vulnerability process"
David LeBlanc
NTBugtraq mailing list
November 3, 2001
http://archives.neohapsis.com/archives/ntbugtraq/2001-q4/0097.html

### 5.4 Commentary on Advisories

"Writing security advisories"
Kurt Seifried
September 10, 2001
http://www.seifried.org/security/articles/
20010910-writing-security-advisories.html

"Xato commentary on MS security bulletins"
Xato
Bugtraq mailing list
December 7, 2000
http://marc.theaimsgroup.com/?l=bugtraq&m=97626305317046&w=2

### 5.5 Commentary on Vendor Accessibility

"Getting to the Third Wave of Security Responsiveness"
Scott Culp
January 2001
http://www.microsoft.com/technet/columns/security/thrdwave.asp

"An informal analysis of vendor acknowledgement of vulnerabilities"
Steve Christey, Barbara Pease
Bugtraq mailing list
March 11, 2001
http://marc.theaimsgroup.com/?l=bugtraq&m=98438570915835&w=2

"Shockwave Flash buffer overflow"
Neal Krawetz
Bugtraq mailing list
December 29, 2000
http://marc.theaimsgroup.com/?l=bugtraq&m=97845942432045&w=2

"Re: Shockwave Flash buffer overflow"
Peter Santangeli
Bugtraq mailing list
January 5, 2001
http://marc.theaimsgroup.com/?l=bugtraq&m=97897439808223&w=2

"Re: SafeWord Agent for SSH (secure shell) vulnerability"
Leif Nixon
Bugtraq mailing list
November 29, 2001
http://marc.theaimsgroup.com/?l=bugtraq&m=100706579514862&w=2

### 5.6 Discovery of Issues in the Wild

"sadmind"
Nancy Lin
SF-INCIDENTS mailing list
December 9, 1999
http://marc.theaimsgroup.com/?l=incidents&m=94476722417209&w=2

"sadmind exploits (remote sparc/x86)"
Marcy Abene
Bugtraq mailing list
December 10, 1999
http://marc.theaimsgroup.com/?l=bugtraq&m=94486731225359&w=2

"IIS %c1%1c remote command execution"
Rain Forest Puppy
Bugtraq mailing list
October 17, 2000
http://marc.theaimsgroup.com/?l=bugtraq&m=97180137413891&w=2

### 5.7 Researcher Credibility and Vulnerability Reproduction

"vCard DoS on Outlook 2000"
Joel Moses
Bugtraq mailing list
August 31, 2000
http://marc.theaimsgroup.com/?l=bugtraq&m=96774764029236&w=2

"Microsoft Outlook 2000 vCard Buffer Overrun"
@stake
February 26, 2001
http://www.atstake.com/research/advisories/2001/a022301-1.txt

"Re: Microsoft Security Bulletin MS01-012"
Joel Moses
Bugtraq mailing list
February 23, 2001
http://marc.theaimsgroup.com/?l=bugtraq&m=98322714210100&w=2

### 5.8 Miscellaneous

"Vulnerability disclosure publications and discussion tracking"
University of Oulu
November 20, 2001
http://www.ee.oulu.fi/research/ouspg/sage/disclosure-tracking/

"Devil in the details - why package signing matters"
Kurt Seifried
October 24, 2001
http://www.seifried.org/security/articles/
20011023-devil-in-details.html

## 6 Acknowledgements

We gratefully acknowledge the constructive comments received from several contributors. Any errors or inconsistencies in this document are solely the responsibility of the authors, and not of the reviewers. This document does not necessarily reflect the opinion of the reviewers or their parent organizations.

We would like to thank Andy Balinsky, Mary Ann Davidson, Elias Levy, Russ Cooper, Scott Blake, Seth Arnold, Rain Forest Puppy, Marcus Ranum, Lori Woeler, Adam Shostack, Mark Loveless, Scott Culp, and Shawn Hernan for their valuable input.

## 7 Security Considerations

This entire document discusses security issues.

## 8 Authors' Addresses

Steve Christey
The MITRE Corporation
202 Burlington Road
Bedford, MA 01730
USA

E-Mail: coley@mitre.org

Chris Wysopal
@stake, Inc.
196 Broadway
Cambridge, MA 02139-1902
USA

E-Mail: cwysopal@atstake.com

## 9 Full Copyright Statement

Copyright (C) The Internet Society (2002). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organisations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

This document expires August 12, 2002.