IT Security Vulnerability and Incident Response Management

Wim Hafkamp

Rabobank Nederland
w.h.m.hafkamp@rn.rabobank.nl

Abstract

This paper summarises the results of a Dutch PhD research project on IT security vulnerability and incident response management, which is supervised by the University of Twente in the Netherlands and which is currently in its final stage. Vulnerabilities are 'failures or weaknesses in computer (application) system design, implementation or operation which can be exploited to violate the security policy defined for that system'. Incidents are defined as 'events that have actual or potentially adverse effects on computer or network operations resulting in fraud, waste or abuse, compromise of information or loss or damage of property of information'. Hacking, denial-of-service attacks and computer viruses are examples of such events. The research project identifies a number of shortcomings in IT service management processes which affect the speed and quality of IT security vulnerability and incident response processes in enterprises. To shorten the lifecycle of vulnerabilities organizations should implement three basic process elements: (1) filtering and analyzing of vulnerability announcements and alerts, (2) prioritizing of vulnerability response activities and (3) scanning of infrastructure components. Each of these steps can be related to specific IT service management processes and to IT security incident management in particular. Using checklists, procedures and dedicated response capabilities, IT organizations are able to faster detect and respond to incidents.

1. Context

In a few years time, the amount of time available to fix known security vulnerabilities in software[1] has decreased tremendously. Time-to-patch is critical because programs that take opportunity of software vulnerabilities, the so-called exploits, are nowadays available for download from the Internet within a few days after such vulnerabilities are discovered[2]. Security experts have already warned for the appearance of so-called zero day exploits; these are exploits for which for which the IT branch of trade does not yet have patches available.

[1] Also known as 'security patching'.

[2] According to Symantec's Internet Security Threat Report, Trends for July 05 – December 05, volume IX (March 2006), the average time between the announcement of a vulnerability and the appearance of exploit code is merely 6.9 days.

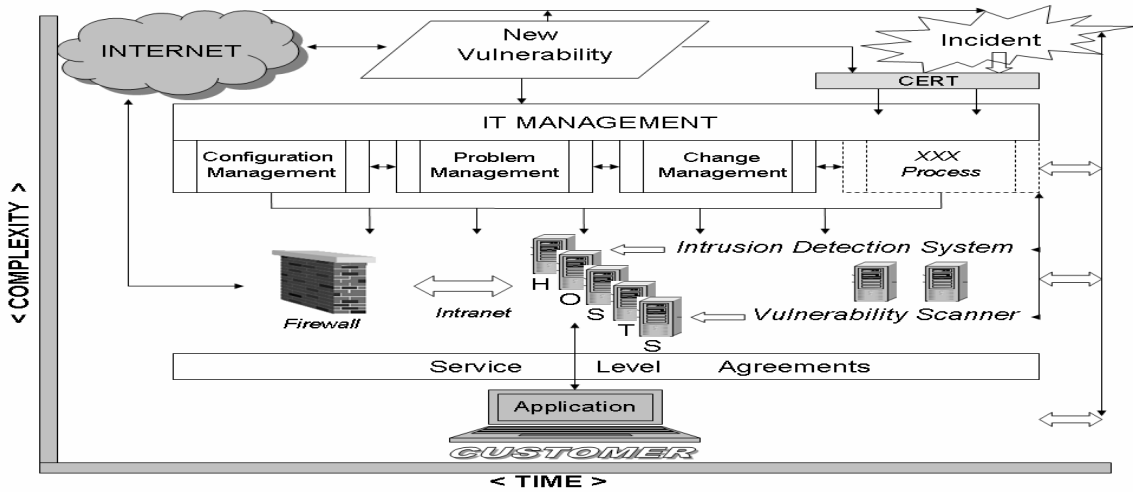

Fig. 1: IT service management processes related to vulnerabilities and incidents

The time-to-patch issue is of major significance to IT security vulnerability and incident management, the aim of which is to counter newly discovered vulnerabilities and associated exploits in a time efficient manner. Nonetheless, enterprises with complex IT infrastructures have usually organised IT service management according to an integrated process approach, which is often based on standards such as ISO/IEC 20000 (Information technology – Service management). In such organisations, it is well understood that solid IT change management procedures are essential for complying with contractual service level agreements. However, enforcing solid IT change management procedures could stand in the way of providing the quick responses that are required for effective IT security vulnerability and incident management.

2. State-of-the-Practise

2.1 IT Management

By the end of the 1990's enterprises in the Netherlands and in other European countries began organizing their IT management processes according to the principles defined by the IT Infrastructure Library (ITIL). As the abbreviation implies, ITIL is a library of best practices ('standards') describing knowledge and experience accumulated over many years with various aspects of development, maintenance and management of IT processes. The first set of ITIL books only described the IT viewpoint. From 2000 on, a 'Business Perspective Set' was developed to narrow the gap between business and IT, addressing subjects such as outsourcing and Business Continuity Management (BCM). The core of ITIL consists of two sets of books named 'Service Delivery Set [BART01] and 'Service Support Set'[BERK00].

Table 1: Examples of ITIL processes

What is needed to deliver IT services to customers (Service Delivery)	Access to IT services and to IT service providers (Service Support)
Examples: - Customer Relationship Management - Service Level Management - Availability Management - IT Service Continuity Management	Examples: - Incident Management - Problem Management - Configuration Management - Change Management

IT Security Vulnerability and Incident Response activities are related to several of these standardized IT service management processes. Take for example the ITIL Change Management process. This process aims at effective and efficient handling and execution of IT related changes[3], in such a way that the impact to the quality of service is minimized. IT organizations often appoint two authorities: the Change Manager who orders for a Request-For-Change (RFC) and a Change Advisory Board (CAB) which approves submitted RFCs. In many organizations the CAB is composed of IT management and business representatives and usually meets once or twice a week. According to ITIL, an RFC should contain at a minimum: details about the (specific) configuration items involved, the description of the proposed change, the foreseen impact on IT services, a fallback scenario and the proposed implementation date and time.

Within the change process three basic steps can be recognized. The first step is preparation. During this step an RFC has been initialized and drawn up and the impact for the IT organization, e.g. needed resources, and the urgency of the request has been determined and approved by the CAB. The second step is the testing and implementation of the change. After testing, the change is implemented in the IT production environment. Because IT service delivery cannot always be guaranteed during implementation of a change, organizations sometimes use a maintenance window for this purpose[4]. During the third and last step the implementation of the change is finally accepted by the IT manager. Important questions here are ‘does the change work?’ and ‘are there any problems left?’. A back-out plan is needed in case problems do occur that are related to the implementation of the change.

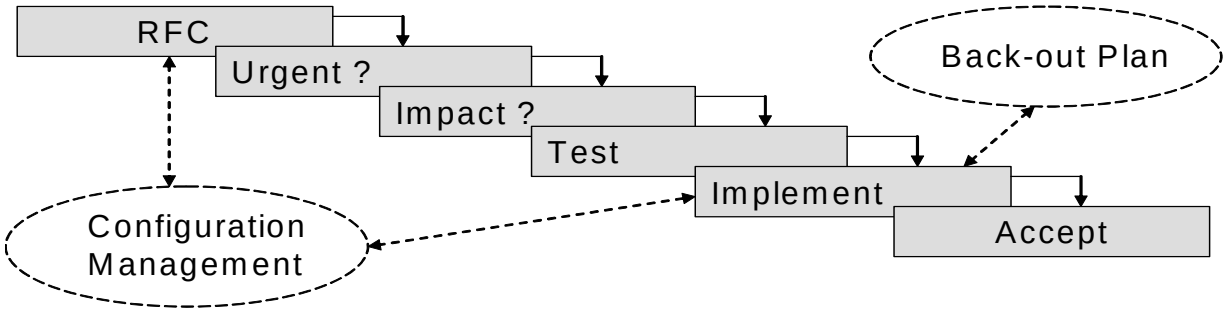

Fig. 2: a standardized change management process

[3] E.g. a new software release or a program temporary fix (patch) provided by a vendor.
[4] Non (or restricted) service hours used for testing and/or maintenance purposes; maintenance windows are often scheduled during weekends.

3.2 Computer Emergency Response Teams

A major virus incident on the Internet in 1988, the outbreak of the Morris worm, has lead to the foundation of the first Internet related Computer Security Incident Response Team (CSIRT): CERT/CC[5]. Later on, many organizations, including universities, military and commercial product and service providers, etc. founded their own incident response teams. CSIRTs vary in size and often handle a variety of mission statements and operational frameworks. The main reason for differences between CSIRTs is that they work for different user groups (called 'constituencies'). What binds them is the type of services they provide, such as artifact analysis, computer security incident response, vulnerability analysis, announcement services, educational services, etc. [WESK03]. Forums like Terena's TF-CSIRT and FIRST offer CSIRT teams a platform for sharing knowledge and experiences about the services they deliver. One of the issues that CSIRTs[6] have to deal with nowadays is the so-called 'vulnerability disclosure problem' [OUUN03]. After having received information about a newly discovered IT security vulnerability, a CSIRT has to decide the time and level of detail of the announcements to their constituency. This is of particular importance when there is no strong solution, such as a security patch, available for solving the problem at the time of announcement.

3.3 Technology push

3.3.1 Intrusion detection systems

Some fifteen years ago, many organizations realized that firewall protection alone was not sufficient to mitigate the risks associated with external connections. One of the reasons for this is that firewalls do neither provide protection against system administration errors nor against insider attacks. For this reason, IT network departments began implementing real-time intrusion detection systems.

Intrusion Detection Systems (IDS) are either network based or host based and designed for anomaly detection or misuse detection.

The difference between anomaly and misuse detection is the way in which the IDS system interprets potential intrusions. An anomaly detection based IDS identifies (statistical) deviations from 'normal use' of a guarded system or network [Denn87]. An IDS system based on misuse detection compares data in system logs or data packets transported over the network with known attack signatures stored in a database. The outcome of the IDS analysis

[5] Computer Emergency Response Team Co-ordination Center, Software Engineering Institute of Carnegie Mellon University, Pittsburgh U.S.A.
[6] In particular CSIRTs working for IT software vendors.

processes may lead to an 'active' or a 'passive' response. An example of a passive response would be sending alerts to a system console. Rebecca Gurley Base [Base00] distinguishes three forms of active responses:

1. searching for and creating extra information;
2. changing the environment, e.g. changing parameters;
3. generating automatic counteractions, e.g. back tracing the attack and denial-of-service an external IP address.

Dealing with false positives, wrongly automated responses and lack of generally accepted best practices are currently major challenges for organizations that want to implement intrusion detection systems [CMUS00].

3.3.2 Vulnerability Scanning

Vulnerability scanning is a structured way to find known vulnerabilities, such as misconfiguration and missing patches, in software components of the IT infrastructure. A variety of scanners is used today. Vendors offer specific (web) application scanners but also generic IT system scanners. Freeware scanners are available for download on the Internet [7].

Most scanners work with databases which contain 'fingerprints' of known vulnerabilities. This database should be updated on a regular basis to account for recently detected vulnerabilities.

A local scanner is implemented on a particular system and which is activated by an administrator. A remote scanner scans systems which belong to a defined range of network addresses. The remote scanner software and the remote scan results are often stored on a dedicated scan server. Remote scanning needs some fine tuning. Firstly, the effects of scanning on network and system performance needs to be addressed. Secondly, firewalls in the (internal) network may block the scanner's network packets and therefore, the firewall rules must be extended to allow specific network traffic from the remote scan server. Thirdly, one of the preconditions of remote vulnerability scanning is that all systems to be scanned should be online during the actual scan. For example, organizations where a large numbers of laptops is deployed face the risk that many of these laptops will not be scanned on a regular basis.

Although vulnerability scanning is usually associated with IT operations, more and more software developers use scanning tools to be able to discover security vulnerabilities during various stages of software development [Nico03].

[7] NESSUS is a well known IT infrastructure vulnerability scanner; until recently it could be freely downloaded from the Internet.

4. Case studies

The research project started by the end of 2002. The research covers the three topics mentioned above and relates them to each other. In 2003 and 2004, three case studies were executed in the Netherlands:

1. a global study on IT security incident and vulnerability response management processes implemented by ten banks;
2. a study about the organization and operational framework of three CSIRTs;
3. a study about IT forensic methods used by the Dutch National Police.

Key findings in the case studies are described below.

4.1 ITIL

ITIL is a de facto IT management standard within the financial sector in the Netherlands. Every bank has implemented ITIL Service Support oriented processes like Incident Management and Change Management. Most of the IT organizations of the visited banks use service level agreements to formalize IT services provided to the business, including the performance and availability of systems and/or applications. The banks have a reserved attitude towards software patching. During interviews we often heard that software patching is considered a time consuming issue, due to formal test procedures and fixed maintenance windows. The case study uncovered that none of the banks had developed a specific security patch policy as part of an overall change management system. Some banks indicated the existence of emergency change procedures for urgent changes but use of these specific procedures was often quite intuitive. A lot of ‘security media attention’ stimulates the decision process for such changes but the IT Change Advisory Board and business representatives still decide on the implementation of urgent changes in much the same way as for normal changes.

4.2 IT security incidents

Crime related IT security incidents are in most cases difficult to handle. During one of the case studies an insider attack was analysed by interviewing and studying the victim organization and by interviewing the police investigators. The results show a poor recognition of incidents that are difficult (or impossible) to detect by automated scanners, segregation of incident response capabilities within large organizations and utmost emphasis on maintaining a ‘chain-of-evidence’ by using specific forensic procedures and tools.

4.3 Computer Security Incident Response Teams

Computer Security Incident Response Teams (CSIRTs) are common in the Dutch university IT environment. Surfnet-CERT is one of the oldest incident response teams worldwide. The

members of Surfnet-CERT are computer specialists with a strong Unix, NT or computer networking background. The Surfnet CSIRT acts as a coordinating CSIRT [Kill03] for many other research or university related CSIRTs. The three visited CSIRTs vary in size and services offered. Incident response handling and communication of vulnerability announcements constitutes their major activity. Most CSIRTs operate conform an operational framework which prioritizes the handling of certain types of incidents, formulates a code of conduct for CSIRT employees and gives rules on how to register incidents. Surfnet-CERT is a member of the Forum of Incident Response and Security Teams (FIRST) and the TERENA Taskforce CSIRT (TF-CSIRT). Due to its connection to the worldwide network of specialists Surfnet-CERT often receives incident and information about newly discovered vulnerabilities in a (very) early stage. The external network also provides opportunities to discuss incident handling activities.

5. Conclusions & Recommendations

5.1 Conclusions

Generally speaking, the conclusion is that the effectiveness of IT security vulnerability and incident response of enterprises with complex IT infrastructures nowadays is rather poor. IT security patch implementation periods of two or more weeks are not exceptional. Furthermore, forensic evidence is sometimes destroyed or neglected due to shortcomings in the incident response process.

5.2 Recommendations

5.1 Vulnerability lifecycle management

The handling of unstructured vulnerability announcements which are received from various information sources is the first challenge of every managed vulnerability response process. Therefore, the process should start with the selection of appropriate vulnerability announcement sources. In the scheme below, four types of sources are included.

Having established that the information originates from a reliable source, it should be subjected to some level of quality control before being further analyzed. Structured layout, detailed impact and complexity level descriptions, CVE (Common Vulnerabilities and Exposures) number[8] and references to available fixes are minimum requirements. For further handling, the receiver of the information, which is often the IT security manager, should ticket a new vulnerability in the incident registration database (with the CVE number), to ensure handling of the information obtained according to the established processes.

[8] See http://cve.mitre.org.

Having completed the registration, the organization must determine the importance (relevance) of the newly discovered vulnerability. Two important questions are to be answered: (1) is a vulnerable component present within the IT infrastructure and if so, (2) how important is that component for the organisation? To answer these questions the ITIL configuration management process can be of help. The first question can be answered by checking the Configuration Management Database System (CMDS) that contains all IT configuration items of the infrastructure. The incident ticket can be closed, if the answer is to the second question is a simple 'not important' but in general, this question is a bit more difficult to answer. Some organizations record a criticality code for every configuration item in the CMDS and this will provide the answer. If this is not the case, a criticality classification of the infrastructure is advised, e.g. determination of the criticality of application servers, infrastructure components and workstations.

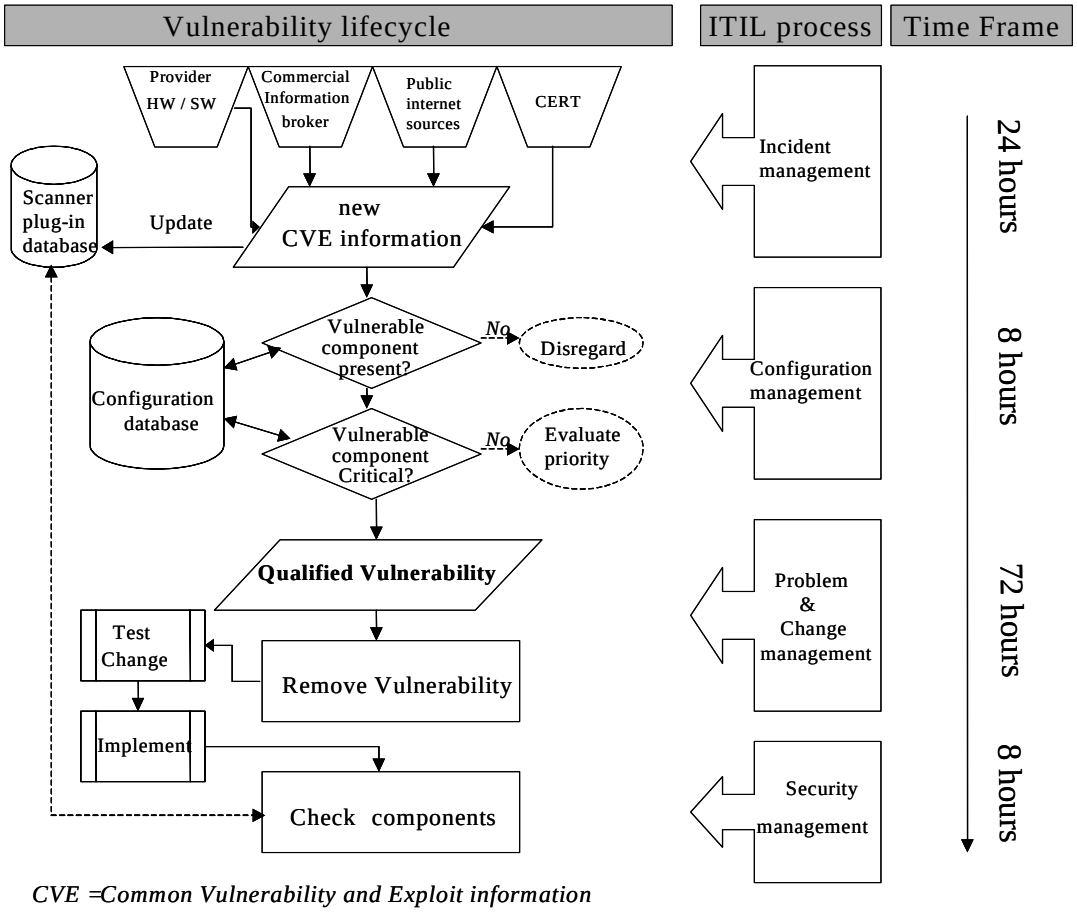

Fig. 3: Vulnerability lifecycle

After having answered both questions, the organization knows whether or not it should trigger a (prioritized) change process. The change process will often take some time. I have

averaged this towards 72 hours for ‘qualified’ urgent changes. During this time fixes or workarounds that are available have to be downloaded and tested and an implementation plan has to be set-up. The existence of a standardized infrastructure and patch management tools will of course accelerate the process.

Finally, IT security management should check within 8 hours after the implementation of the prioritized security change whether the vulnerability has been removed by scanning (parts of) the application systems or infrastructure.

In the model I have indicated a controlled vulnerability lifecycle management of less than five days for qualified vulnerabilities by using implemented ITIL oriented IT service management processes.

5.2 IT security incident responses

The second recommendation is about the handling of IT security incidents. As stated before, recognition of IT security incidents which are not automatically detected by tools[9] is a main problem. ITIL oriented organizations often deploy a structured incident management process which defines several support and escalation levels. An indicator checklist can help the IT organizations first level support to determine whether or whether not additional IT security support and/or escalation levels have to be involved once an incident has been reported.

The checklist contains three indicators. The first indicator is about the information source. Scanner or detector alerts for instance should not be analyzed by IT first level support but should be directly forwarded to the department that bears the responsibility for these tools. Secondly, first level support should analyse the symptoms of the reported incidents to determine whether confidentiality, integrity or availability of information is affected [MAPR01]. The third and last step is about causality. By correlating the incident with other known issues, like a recently implemented change, technical system problems, etc. a possible cause can be determined. After the last step first level support may ticket the incident with a label ‘possible IT security incident’ and inform IT security management for further instructions.

IT security management should proceed according to a procedure which defines what actions have to be taken in case a potential IT security incident is detected. If the organization has a CSIRT in place, it is advised that in this stage the CSIRT takes over, investigates the incident and advises IT problem management which measures should be taken to start an investigation and to maintain a ‘chain-of-evidence’ [ADKR04]. Status reports should be send to the IT problem manager involved. This manager is also accountable for the implementation of any preventive measures that would be deemed necessary after having carried out the investigation.

[9] E.g. virus scanners or intrusion detection systems.

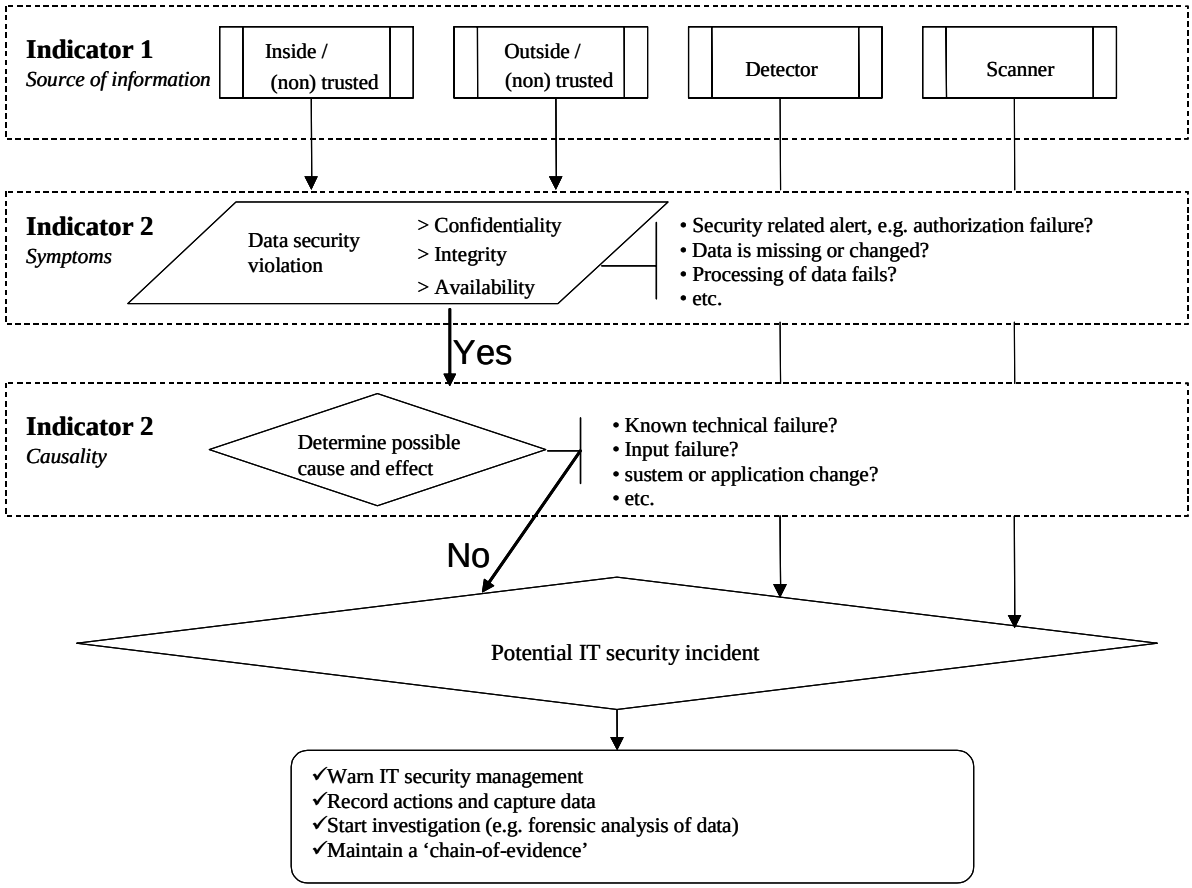

Fig. 3: IT security indicator checklist

References

[ADKR04] Alberts C. et al., *Defining Incident Management Processes for CSIRTs : A Work in Progress*, Carnegy Mellon University / Software Engineering Institute, Pittburgh U.S.A., 2004

[BART01] Bartlett J. et al, *Best Practice for Service Delivery*, The Stationary Office, Norwich U.K., 2001

[Base00] Bace R., *Intrusion Detection*, MacMillan Technical Publishing, Indianapolis U.S.A., 2000

[BERK00] Berkhout M. et al, *Best Practise for Service Support*, The Stationary Office, Norwich U.K., 2000

[CMUS00] Allen J. et al., *State of the Practise of Intrusion Detection Technologies*, Technical Report, Carnegy Mellon University / Software Engineering Institute, Pittsburgh U.S.A., 2000

[Denn87] Denning D., An *Intrusion-Detection Model*, IEEE Transactions on Software Engineering, Vol. SE-13, No. 2, February 1987

[Kill03] Killcrece G. et al., *Organizational models for Computer Security Incident Teams*, Carnegie Mellon Sofware Engineering Institute, Pittsburgh U.S.A., 2003

[MAPR01] Mandia K. & Prosise C., *Incident Response, investigating computer crime*, McGraw-Hill, U.S.A., 2001

[Nico03] Nicolett M., *Vulnerability Management Defined*, Gartner research note, available at www.gartner.com, September 2003

[OUUN03] Oulu University Secure Programming Group, *Communication in the Software Vulnerability Reporting Process*, available at www.ee.oulu.fi/research/ouspg, 2003

[WESK03] West-Brown M.J. et al, *Handbook for Computer Security Incident Response Teams, Second Edition*, Carnegie Mellon University / Software Engineering Institute, Pittsburgh U.S.A, 2003