Interested in learning more about security?

SANS Institute InfoSec Reading Room

A Taxonomy of Information Systems Audits, Assessments and Reviews

There are two sections to this dissertation. The first is an arrangement and classification of the various types and classes of IT security assessment and testing strategies. This section continues with a proposed learning and development strategy for the IT Risk Assessor to develop their testing and assessment skills. The second delivers the results of a process of experimentation designed to quantitatively assess the variation across the classes and definitively determine if there was in fact a quantitative variation...

A Taxonomy of Information Systems Audits, Assessments and Reviews

GIAC Systems and Network Auditor (GSNA)

Auditing Networks, Perimeters & Systems, AUD-507 Gold Certification

Author: Craig S Wright MMgt, MNSA, CISSP, CISA, CISM, GCFA, GNSA, G7799

BDO Kendalls

Adviser: Joey Niem

SANS

Accepted: …

Table of Contents

List of Figures

I. Abstract

Common misconceptions plague information systems audit as to the nature of security, audit and assessment types and definitions. The dissertation aims at being a definitive guide to define the terminology and detail the related methodologies across the range of information assurance services. The idea is to not only detail and define the types of audit, assessment inspections [etc], but to compare and evaluate the various strengths and benefits of each in a simple and referential critique that may remove an abstraction of error and confusion surrounding these services. The paper will cover the types, history and basis for each type of service. The paper statistically compares the strengths and weaknesses of each and sets out a scientifically repeatable foundation for the deterministic nomenclature used in the industry.

Figure 1 – Security, the puzzle…

II. Document Conventions

When you read this practical assignment, you will see the representation of certain words in different fonts and typefaces. The representation of these types of words in this manner includes the following:

command	The representation of operating system commands uses this font style. This style indicates a command entered at a command prompt or shell.
filename	The representation of filenames, paths, and directory names use this style.
`computer output`	The results of a command and other computer output are in this style
URL	Web URL's are shown in this style.
Quotation	A citation or quotation from a book or web site is in this style.

III. Executive Summary

There are two sections to this dissertation. The first is an arrangement and classification of the various types and classes of IT security assessment and testing strategies. This section continues with a proposed learning and development strategy for the IT Risk Assessor to develop their testing and assessment skills.

The second delivers the results of a process of experimentation designed to quantitatively assess the variation across the classes and definitively determine if there was in fact a quantitative variation in the results achieved using the separate processes.

In particular, this research was designed to test the hypothesis that white-box audit techniques and tools based external penetration testing differ quantitatively. The results of this experimental process demonstrate that there is in fact a significant variation in the outcomes and that an audit-based approach is far more effective of noting and finding a large range of systems vulnerabilities.

Part 1. The Nomenclature or Classification

The need to develop a structured taxonomy (naming system) of the terms or services used within the realm of IT Security is nothing new. All these services have been provided for as long as business and government have used computers. They were definitely employed as far back as the 70's.

In order to understand the terms used in our discipline, we need to understand the process to be tested and the procedure we have performed.

AUDIT

There are two definitive classes of Audit, internal and external (AICPA). An audit consists of the evaluation of an organisation's systems processes and controls and is performed against a set standard or documented process. Audits are designed to provide an independent assessment through testing and evaluation of a series of representations about the system or process. An audit may also provide a gap analysis of the operating effectiveness of the internal controls.

External audits are commonly conducted (or at least should be) by independent parties with no rights or capability to alter or update the system they are auditing (AICPA). In many cases, the external auditor is precluded from even advising their client. They are limited to reporting any control gaps and leading the client to a source of accepted principles. Due to these restrictions, an indication of the maturity or a system against an external standard (such as COBIT) is often engaged.

Internal audits involve a feedback process where the auditor may not only audit the system but also potentially provide advice in a limited fashion. They differ from the external audit in allowing the auditor to discuss mitigation strategies with the owner of the system that is being audited.

Neither an internal or external auditor can validly become involved in the implementation or design process. They may assess the level to which a design or implementation meets its desired outcomes, but must be careful not to offer advice on how to design or

implement a system. Most crucially, an auditor should never be involved with the audit of a system they have designed and/or implemented.

There is a large variety of audit types. Some examples include SAS 70 (part 1 or 2) audits, audits of ISO 9001,17799:2/27001 controls, and audits of HIPPA controls. There are many different types of audits and many standards that an audit may be applied to.

An audit must follow a rigorous program (Winkler, 1999). A vulnerability assessment as it is commonly run is more correctly termed as a controls assessment. A controls assessment may also be known as a security controls review.

INSPECTION AND REVIEWS

An audit differs from an inspection in that an audit makes representations about past results and/or performance. An inspection evaluates results at the current point in time. For an audit to be valid, it must be conducted according to accepted principles. In this, the audit team and individual auditors must be certified and qualified for the engagement. Numerous "audits" are provided without certification, these however are in consequence qualified reviews.

PENETRATION TESTS AND RED TEAMING

A Penetration test is an attempt to bypass controls and gain access to a single system. The goal of the Penetration test is to prove that the system may be compromised. A Penetration test does not assess the relative control strength nor the system or processes deployed, rather, it is a "red teaming" styled exercise designed to determine if illicit access can be obtained, but with a restricted scope. The issue is that it is infeasible to prove a negative. As such, there is no scientifically valid manner to determine if all vulnerabilities have been found and this point needs to be remembered when deciding on whether to use a Penetration test process.

Cohen (1998-2) notes in respect to red-teaming organisations "*one of the teams I work with routinely asks whether they are allowed to kidnap anyone to get the job done. They usually get turned down, and they are rarely allowed to torture anyone they kidnap*". Red teaming is based on nearly anything goes.

The greatest strength of the Penetration test lies in its being able to market the need to improve internal controls to internal management. This may seem contradictory, but it is based on perception. Being that the Internet is seen as the greatest threat to an organisation's security, management are often focused on the firewall and Internet gateway to the exclusion of the applicable security concerns and risks. As such, Penetration tests do help in selling the need for an increased focus on information security, but often at the expense of an unfocused application of these efforts.

A Penetration test is of limited value in the greater scheme of a systems information security audit programme due to the restricted nature of the test and the lack of inclusion of many key controls. Contrary to popular opinion, penetration testing does not simulate the process used by an attacker. The attacker is not limited in the level of time or funds in the manner that restricts the Penetration tester. Whereas a successful Penetration test may note vulnerabilities, an unsuccessful Penetration test does not prove the security of a system (Dijkstra, 1976).

"Red Teaming" differs from penetration testing in that it is designed to compromise or penetrate a site at all costs. It is not limited to any particular attack vector (such as a VPN or Internet) but rather is an attempt to access the systems in any feasible manner (including physical access). A typical red teaming goals would include objectives such as "steal 100,000 for Big Bank without being caught and deliver the report of how to do this to the executive of Big Bank" or "Copy file X which is marked as secret".

Both government and business have used red teaming for many decades in a variety of areas including physical and logical based testing. At its simplest, it is a peer review concept. Another way to look at it is a method of assessing vulnerabilities. In cases where red teaming refers to the provision of adversarial perspectives, and the design of the red team is not hampered in the matter is that ethical attacks are. There is a little correlation between a red team exercise and an ethical attack.

The formation of red teams (or cells) is a situation unlikely to occur in any ethical attack. Further, internal intelligence is unlikely to be gathered as part of an ethical attack. In this instance is more likely that the ethical attack will consist of an attack against the Internet

gateway. An engagement to red team is wider in scope, areas including internal subversion and associated control checks cannot be ignored in this type of test.

Penetration testing, if done correctly, can provide some value in its free-form approach if the limitations to scope inherent in this type of test are understood. When correctly implemented, a Penetration test adds a level of uncertainty to the testing. The benefit of this uncertainty is that it might uncover potential flaws in the system or controls that had not been taken into account when designing the control system. To be of value, a Penetration test needs to do more than a simple tool based scan of a system.

Penetration Testing needs to do something novel and unexpected.

There is little similarity between a penetration test, vulnerability assessment, risk assessment or audit. The lack of understanding of these differences often impedes the implementation of effective security controls.

ETHICAL ATTACKS

Ethical Attacks are a subset of penetration testing. They are designed to externally validate a set of controls in a manner that is thought to simulate an attack against the system. It should be noted that ethical attackers are not actually testing system security in the manner of an attacker due to a variety of restraints. It has been demonstrated (Cohen, 1997) that ethical attacks do far less to categorically qualify security risks than many other forms of testing. They do not for instance take note of internal controls. Many potential vulnerabilities cannot be discovered in a penetration test by the nature of the testing. Next, it needs to be remembered that there is an economic cost associated with ethical attack styled penetration testing. The Ethical attacker is constrained by a budget of time and thus money, the real attacker is not.

Blind testing by its very nature will take longer to complete than auditing a site with access and knowledge of all the systems (Dijstra, 1976) if any level of assurance is required. The review undertaken by the ethical attacker is thus hobbled from the start. It is infeasible to state that the contractor will have more knowledge at the end of a review if it is done as an ethical attack with limited knowledge over a systems review with full information.

Being a black box test format, the lack of foreknowledge as to the qualification of value associated with any particular asset negates the possible assessment of a vulnerability status by an ethical attack process (Dodson, 2005). Rather, the process is designed to determine a subset of all possible control failures, which may lead to a system breach or compromise. This subset can never equal the entire control set of possible hazards and vulnerabilities.

This said ethical attacks do have value. In particular, they are useful for process testing. If the systems and security team go through the internal processes, they can use the ethical attack process as a means of determining an estimate of the levels of protection using time based security. This is achieved by measuring the detection time and the response time. These times may than be compared at different periods (such as weekends and nights) to determine the level of protection over the system.

Unfortunately, most ethical attacks are not used as an exercise to quantify the level of protection or risk to a system. Rather they are used as a simple de facto vulnerability assessment.

VULNERABILITY ASSESSMENT

A vulnerability assessment is an assessment and gap analysis of a site's or a system's control strengths. A vulnerability assessment is a risk-based process. The process involves the identification and classification of the primary vulnerabilities that may result in a system impact. Often, methodologies such as fault tree analysis or CCA (cause consequence analysis) are employed in this process.

A vulnerability assessment is a critical component of any threat risk assessment (Keong, 2004). Following the vulnerability assessment, an impact analysis is conducted to be used in conjunction with a threat report to provide for an estimation of the organisation's risk to selected attack vectors.

There are various processes and procedures used to provide vulnerability assessments and threat/risk determinations. Some standards such as AS/NZS 4360:2006 are commonly mandated by government organisations (such as the NSW State government in Australia).

Vulnerability assessments are part of a complete risk analysis program (Moore, 2001).

Vulnerability assessments involve the cataloguing of assets and capabilities. The lack of internal knowledge provided in the typical ethical attack process precludes this phase. A vulnerability assessment helps to quantify and discern the level of risk to a system (Linde, 1975).

Vulnerabilities, and potentially threats to these resources are determined in this process, which is not limited to external attacks. This process needs to take into account not only external attacks and even internal attacks, but a necessarily must also consider physical threats and many other tests outside the reach of the ethical attack or basic penetration test

BLACK AND WHITE BOX TESTING

Both vulnerability assessments and penetration tests may be conducted as a white box or black box analysis. A black box analysis is instigated with little or no knowledge of the system being tested. A white box analysis is conducted with all details of the system provided to the tester in advance of the testing process (Dijstra, 1976).

TOOLS BASED SCANNING

The common perception that running an automated scanner such as Nessus or one of its commercial cohorts is in itself a vulnerability or penetration test is false. The belief that these services act as an audit is even further from the truth.

Most of the so-called penetration tests that are provided are no more than a system scan using tools. A penetration test, if correctly designed and implemented will attempt the use of various methodologies to bypass controls. In some instances, this may involve the creation of new or novel scripts/programs.

The issue is not that many people commonly use the words interchangeably but that so-called professionals fail to differentiate the terms. Of particular concern is the use of audit and the designation, auditor. This is as these terms are often restricted in legislation as most jurisdictions have statutory requirements surrounding their use and application.

AGREED PROCEDURES REVIEW

Information security systems provide many of the functions that construct a control system. Of particular concern are controls that limit access to accounting and financial records. This includes records held by systems that provide an e-commerce transaction path. In many jurisdictions, it is an offence to sign off an audit report when you are not a certified auditor. Traditionally the path around this has been not to call the process of testing the system an audit, but rather to call it an agreed procedures review.

An agreed procedures review or simply a review is an analysis of controls performed against an agreed process.

ACCEPTANCE TESTING

Acceptance testing is one of the final occasions to recognise any risk or exposure in a system (Myagmar, 2005). The development and implementation of an approved, inclusive and prescribed plan will support the successful execution of a solution, with the least interruption to critical systems. The process of acceptance testing is to garnish an acceptance of the changes or introduction of a system.

Acceptance testing is more correctly an audit or qualified review of a set of implementation objectives to ensure that the system meets the required levels of performance or security.

DATA CONVERSION

Testing a Data Conversion is a two-stage process (AICPA). Initially the planning process associated with the data conversion is reviewed to determine the sufficiency of any proposed controls. The subsequent stage occurs after the conversion process. The aims of this process are to present an independent evaluation as to the completeness and accuracy of the data after the conversion.

Any conversion of data into another form or to another system bears an elevated risk of error, omission or other deviations to the completeness and accuracy of that data. Standard input and process controls are frequently not maintained in the data conversion process. To be successful, any project, which includes a data conversion process, requires that the accuracy and completeness of the conversion process be preserved.

The Taxonomy

Class	Definition	Categories	Sub-Categories
Audit	An audit, consisting of an evaluation of an organisation's systems processes and controls, is performed against a set standard or documented process. Audits are designed to provide an assessment through a qualified appraisal of the representations, which have been made concerning the system or process.	Internal	• Financial • Controls • Audit against Policy and Procedures • Audit against a Standard or legislative Requirement • Contract • Service Delivery • Application • System
		External	
Assessment	Numerous "*audits*" are provided without certification, these however are qualified reviews.	Vulnerability Assessment	• Tools Based System Scan • Vulnerability Analysis
		Qualified Review	• Ethical Attack • penetration test
		• Gap Analysis • Controls Assessment • Threat / Risk Assessment	
Inspection	An inspection captures the state of		

	security at a point in time. An inspection is generally used as a part of the audit process to test controls.	
penetration testing	A penetration test is an attempt to bypass controls and gain access to a single system. The goal of the penetration test is to determine vectors over which a system may be compromised.	• Ethical Attack • Grey Hat Verification • penetration test The nature of the testing is such that a failure to uncover any vulnerabilities does not imply that the system is secure

VULNERABILITY

A vulnerability is any weakness to a system that can be triggered (either by accident or intent) to exploit a weakness in a system (NIST, 800-42).

Although it is commonly called a vulnerability, an unpatched system or "hole" does not in itself create a vulnerability. What is being noted is a potential vulnerability. Other information needs to be associated with this potential vulnerability before it may be classified as a vulnerability. There is great difference between a potential vulnerability and a vulnerability. Before this determination can be made, it is necessary to understand the system being tested.

The limited knowledge provided in blind testing or other black box test processes are seldom adequate to provide this information. Although the ethical attacker or even penetration tester may stumble across a potential vulnerability with possibly serious consequences, it is rarely likely that they will be able to determine this without additional internal information.

THREAT-SOURCE

A Threat-Source is either (NIST, 800-30):

1. Intent and method targeted at the intentional exploitation of a vulnerability, or

2. A situation and method that may accidentally trigger an exposure to a system vulnerability.

THREAT

A threat is the potential for a threat-source to exercise or exploit a specific vulnerability. A threat may be either accidental or intentional in nature.

RISK

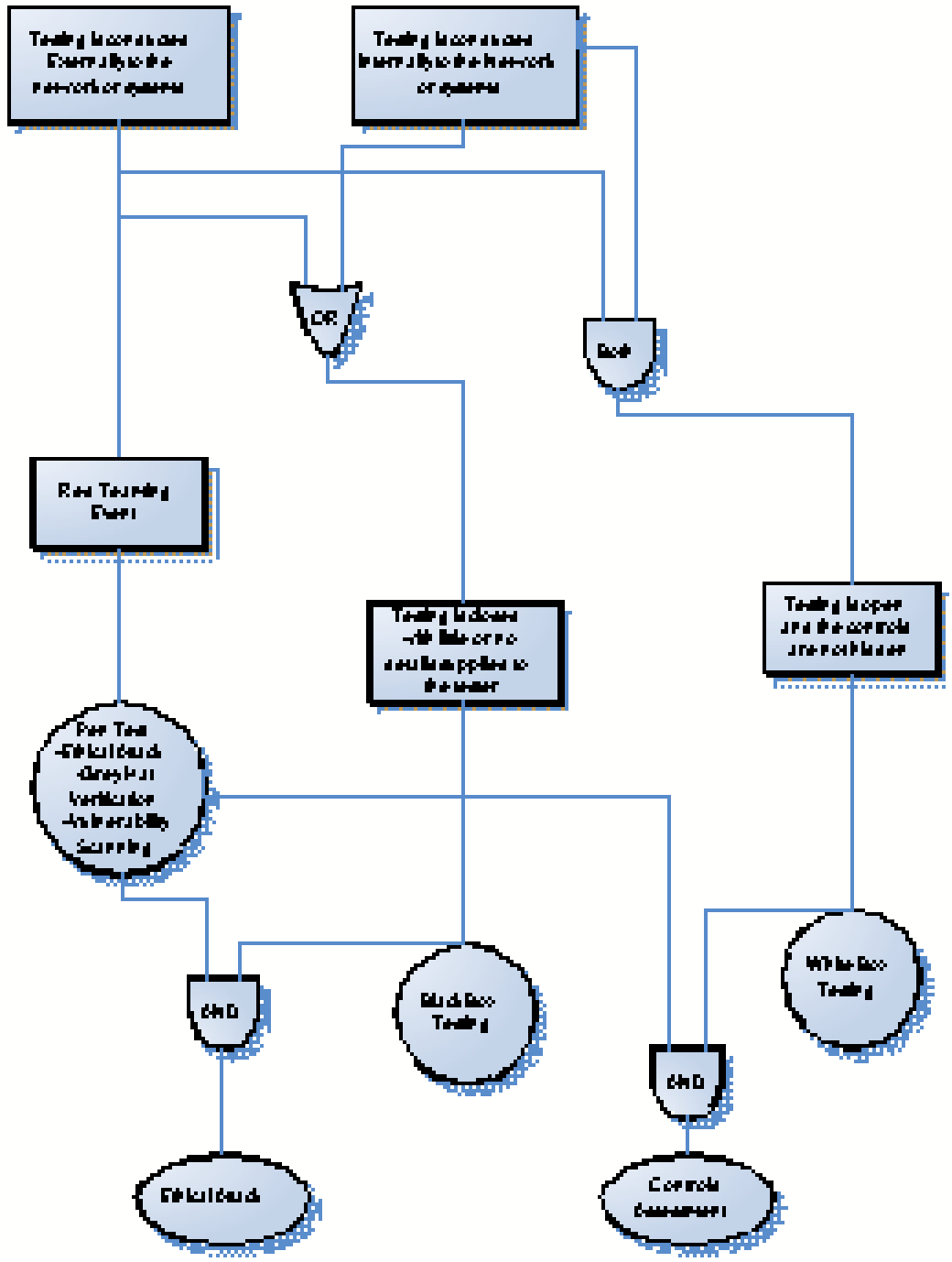
Risk is “a function of the likelihood of a given threat-source’s exercising a particular potential vulnerability and the resulting impact of that adverse event on the organisation”. A risk is a probabilistic event that may be modelled quantifiably using survival and hazard functions.

RISK MANAGEMENT

This is the process of identifying, assessing and controlling risk. Risk management is the process where the level of risk is maintained within accepted bounds. It is not possible to mitigate all risk and cost constraints due to the economic law of diminishing returns always leave some risk.

As commerce is about risk, being that all profit is determined through the taking of risk above the base bond rate, risk will continue to exist in all aspects of business and endeavour. This includes security.

The Decision Test of the process



Controls

To have an effect on an assessment of any system, it is essential that the auditor have a good understanding of controls as applied to information systems (COSO).

Controls as used within the field of information systems incorporate the policies, procedures, practices and organisational structures, which the undertaking has implemented in order to provide for a reasonable level of assurance that their objectives will be accomplished. The controls implemented within a computer system are intended to provide an efficacy and effectiveness of operations, consistency and compliance with the laws, rules and regulations with which the undertaking needs to adhere.

There are two principal control types that the Information Systems auditor needs to be aware of and understand. These are general controls and application controls, each of which will be covered in further detail below.

Controls range from the "soft" controls such as the integrity and ethical values of staff, the philosophy and operating style of management, the competence and professionalism of employees and the effectiveness of communication through to “hard” controls such as segregation of duties, network choke points and authorisation processes.

Soft controls are a more difficult area to assess, as there are no generally agreed and defined approaches to the conduct of an appraisal of these controls. For this reason, many auditors fail to assess them adequately.

Definition of Internal Control

The Committee of Sponsoring Organizations of the Treadway Commission [COSO] defines an Internal Control as follows:

Internal control is a process, affected by an entity’s board of directors, management and other personnel, designed to provide reasonable assurance regarding the achievement of objectives in the following categories:

- *Effectiveness and efficiency of operations*
- *Reliability of financial reporting*
- *Compliance with applicable laws and regulations*

KEY CONCEPTS

- *Internal control is a process. It is a means to an end, not an end in itself.*
- *Internal control is influenced by people. It is not merely policy manuals and forms, but people at every level of an organization.*
- *Internal control can be expected to provide only reasonable assurance, not absolute assurance, to an entity's management and board.*
- *Internal control is geared to the achievement* of objectives *in one or more separate but overlapping categories* (COSO, Key Concepts).

When applied to Information Systems in totality as used within an undertaking, controls encompass not only the domain associated with financial reporting as used by COSO, but rather all aspects of the undertakings operations. The Key Concepts expressed within COSO surmise the wider objectives associated with Information Systems in an efficient means.

Controls (both general and application) are processes designed to deliver an objective. The auditor is generally concerned with the controls that provide for confidentiality, integrity and availability of information systems.

From a wider view than information security, information systems controls can cover such diverse goals as systems efficiency, speed and cost effectiveness or economy. The important note to remember is that a control is a process to achieve an objective. The aim in assessing a control is to test if the undertaking can achieve its desired objective effectively.

Both general and application IT controls are designated as either "key" or "operational".

Key Controls

Key controls are those upon which the undertaking holds reliance. They warrant that objectives such as access rights, the integrity of operations and data and reporting are both valid and consistent. Key controls are at times confused with good practice. They are however not the same. A common example is the use of modular, structured and well-documented programme code in application development. This is an excellent

practice but is not a key control. Key controls generally require accuracy and reliability of processing. They do not for instance consider operational efficiency.

Operational Controls

Operational Controls are focused on the day-to-day operation of the undertaking to make certain that all of the undertakings objectives are achieved in the most efficient method. It is common for operational controls to slowly become an impediment to business over time and one of the key areas that needs to be monitored in both maintaining and reviewing operational controls is whether they still provide for the objectives they where intended to meet.

Systems efficiency and effectiveness are examples of the areas addressed within the scope of operational control.

General controls

General controls include the processes that are applied generically across the undertaking or in sections of the undertaking's Information Systems. Common general controls within an undertaking include both the organisational and administrative structure of the undertaking and its information systems processing areas.

Policies, operational procedures, systems standards, the availability of staff, their skill and training and the "*tone from the top*" given by management are just a few of the many aspects that encompass an undertakings general control framework.

The auditor needs to gain an overall impression of the controls present in the Information Systems environment. General controls form the foundation on which all other controls within the organisation are built upon. If the Information Systems General controls are not sound, it is highly unlikely that the organisation will be able to maintain an effective control structure or to achieve any level of system security.

In reviewing general controls, the auditors should include any infrastructure and environmental controls in the review. The adequacy of air conditioning (both for temperature and for humidity), smoke detectors or preferably fire suppression systems, well maintained power supply systems (uninterruptible power supplies, generators, and

surge arrestors) and an uncontaminated grime and particulate free situation are all controls. Even something as (seemingly) simple as orderly and identifiable electrical and network cabling all add to the continuing operation of an undertaking is Information Systems.

It is important to consider not only the logical access to a system, but also physical access controls. It is often the case that logical access to computer systems is tightly monitored and regulated, but physical access is left wide open. Considering there are many commands and settings that can be executed only from the physical console on many systems, physical controls are often of key importance.

In reviewing physical controls, it is necessary to conserve not only the individual systems but also the overall access control measures. For instance, facility controls such as having security guards at entry gates, displayed identification badges, the logging of visitor access to a site and enclosing all servers in a secure location will aid in increasing the level of assurance one can take over an undertaking's control framework.

Application Controls

Application controls are interconnected transversely within both the transactions and data, which may be either manual or programmed.

The objective of an application control is to warrant the completeness and accuracy of the records and the validity of the entries created or processed in the system.

Application controls incorporate data input validation, agreement of batch totals, hashing and control checks as well as encryption of the transmitted data for both privacy and integrity.

Application controls are not all "hard" controls. Controls for buying & developing software, policy development, management, communication, education, and change management can all come under the banner of an application control.

An application control is one that it is built into and acts as an element of the business process. Thus, application controls act to ensure completeness, accuracy, business authorisation and validity of processed transactions. It is important to remember that

where controls are implemented in an interconnected environment, the business controls on the processes must also cover the entire range of the operation (being defined as the entire collection of business systems and processes used by this action within the application being assessed).

In assessing application controls, business process definitions need to be analysed to ensure that they are compliant with the business controls. Often these processes are expressed within a notational format (Kramer, 2003). Some example formats include:

- BPEL - Business Process Execution Language
- BPMN - Business Process Modelling Notation
- ebXML Meta-Models
- ERM – Entity relationship models (Inc. CODD Diagrams)
- FDL – Flow Definition Language
- UML – Unified Modelling Language

IT Governance

There are various definitions of IT governance. Weill and Ross focus on "*Specifying the decision rights and accountability framework to encourage desirable behaviour in the use of IT*." (Weill, P. & Ross, J. W., 2004)

We can compare this with the perspective of the IT Governance Institute, which develops the classifications within the keystone system where "*the leadership and organisational structures and processes that ensure that the organisation's IT sustains and extends the organisation's strategies and objectives*." (IT Governance Institute 2003)

Alternatively, the Australian Standard for Corporate Governance of ICT [AS8015] characterises Corporate Governance of ICT through "*The system by which the current and future use of ICT is directed and controlled. It involves evaluating and directing the plans for the use of ICT to support the organisation and monitoring this use to achieve plans. It includes the strategy and policies for using ICT within an organisation.*"

Other Terms

Objectivity

Objectivity is an independent mental attitude that you should maintain in performing any engagement – whether an audit, review or inspection. Objectivity requires you to perform in such a manner that you have an honest belief in your work and that no significant quality compromises are made.

Ethics

When auditing you have an obligation to exercise honesty, objectivity and diligence in the performance of your duties and responsibilities.

You must:

- Exhibit loyalty in all matters pertaining to the affairs of the client or to whomever you may be rendering a service. However, you will not knowingly be a part of any illegal or improper activity.
- Refrain from entering into any activity which may be in conflict with the interest of the client or your firm, or which would prejudice your ability to carry out objectively your duties and responsibilities. Remember, other departments are internal clients.
- Not accept a fee or gift from an employee, a client, a customer or a business associate of the client without the knowledge and consent of your firm's senior management and only when openly announced.
- Be prudent in the use of information acquired in the course of your duties. You shall not use confidential information for any personal gain or in a manner that would be detrimental to the welfare of your firm or their customers.
- When expressing an opinion, use all reasonable care to obtain sufficient factual evidence to warrant such expression. In your reporting, you shall reveal such material facts known to you, which, if not revealed, could either distort the report of the results of operations under review or conceal unlawful practice.

Act professionally at all times.

Planning

Adequate planning should include consideration of:

- Communication with all who need to know about the audit.
- Any personnel to be used on the assignment
- Background information on the customer.
- Work to be done and the general approach.
- The format and general content of the report to be issued.

Planning is important to ensure that results will reflect the objectives of the audit. The planning should be documented and should include:

- Establishing audit objectives and scope of work.
- Obtaining background information about what is to be reviewed.
- Determining the resources necessary to perform the audit.
- Communication with all who need to know about the review.
- Performing, as appropriate, an on-site survey to become familiar with activities and services to be reviewed, to identify areas for emphasis, and to invite client/management comments and suggestions.
- Determine how, when, and to whom results will be communicated.
- Obtaining approval of the work plan from all concerned parties.

Examining and Evaluating Information

You should collect, analyse, interpret, and document information to support your findings. The process of examining and evaluating information is as follows:

- Information should be collected on all matters related to the objective and scope of work.
- Information should be sufficient, competent, relevant, and useful to provide a sound basis for findings and recommendations.

- Sufficient information is factual, adequate, and convincing so that a prudent, informed person would reach the same conclusions as the final report author.
- Information should be reliable and accurate. Ensure that all information is correct through verification. An SRS (Simple Random Sample) or a stratified sample of the information should be verified to source to ensure accuracy.
- The auditor should ensure that all the information supplied is relevant to the particular project and is consistent with the objectives.
- When designing audit procedures and any testing techniques which are to be employed, the procedures should be selected in advance (where practicable), and subsequently expanded or altered where circumstances warrant.

A Preliminary Survey

Sufficient background information must be obtained about the client's activities before an effective program can be prepared. This is usually done through a preliminary survey in which as much information as is practicable and useful is gathered. Most of this information is obtained orally from responsible officials within the organisation. It focuses on the size and scope of activities, operating practices, and internal controls. Some concurrent tests may be made during the survey phase, usually to evaluate assertions regarding operating practices.

The preliminary survey usually identifies matters warranting in-depth attention. These may include areas in which there may be weaknesses in internal controls, inefficient operations, or lack of compliance with internal policies, and legislative requirements.

After preparation the next stage is to write a '*programme'* that will focus on matters that are potentially hazardous to the client (either internal or external), plus any others of special interest. These specific objectives represent the framework around which a fabric of procedures is woven.

The Programme, Criteria for defining Procedures

A '*programme'* should conform to certain criteria if it is to satisfy the overall objectives of the review/audit. When creating the review or audit programme, each work step should

show be documented and justified. The objective of the operation and the controls to be tested must be taken into consideration when designing any test. Further, all stages and processes to be employed in the audit process should include positive instructions with a justification and reasoning for their inclusion. It is not good practice to state these processes in the form of questions without an explanation.

- The audit programme should be flexible and permit the auditor to be able to use his/her judgment in order to deviate from the prescribed procedures. Further, there are instances where it may be necessary to extend the work done in this process. Any time where a major deviation from the original scope is proposed, management must be informed.
- The audit programme should not be cluttered with information or material from sources that are readily available. Where textual or online sources are available, it is preferable to include a reference to the external authority. An example would be a stage of a programme that calls for the use of Microsoft's Baseline Analysis tool (MBSA). Rather than adding a 10-page appendix on how to run the MBSA Scanner, include a link to Microsoft's help site.
- Any unnecessary information should be avoided. Include only what is needed to perform the audit work. Do not include documents just because they are there!

The Programme

Much of the information generated at this point will also serve as the introduction to the final report to the customer and should generally include the following information:

INTRODUCTION AND BACKGROUND

The introduction should include information about the audit client. This would relate to either the external firm or even the internal department being reviewed. Any relevant information to the audit concerning the client's:

- activities,
- function,
- history and objectives
- principal locations or sites

Should be included in this section of the document. This is included such that the personnel conducting the engagement have ready access to all information needed to understand and carry out the programme.

PURPOSE AND SCOPE OF THE REPORT

The purpose and scope of the report should be included early in the process. In particular, the scope should specify the types of services and tests that are in included and in particular, it needs to include any services or systems that are specifically excluded.

OBJECTIVES OF THE PROJECT

The special goals of the review should be clearly stated. In this, it is important to document the reasons why the review is being conducted and any explicit outcomes that have been determined to rely on this process.

DEFINITION OF TERMS

Any unique terms or abbreviations used within the report or the audited entity should be defined or explained. This is particularly important in cases where others will make use of the report (such as a report issued by the Internal Auditors, which is expected to be issued to the external audit team). It should also be remembered that reports are often supplied to parties to whom the report was not initially designed to be distributed. In some cases, company boards may take interest in these reports and it cannot be expected that all the technical jargon and terminology will be known to these recipients.

PROCEDURES

For most audits and reviews, it is necessary to stipulate the procedures that will be followed prior to the start of the engagement. This should be done in a manner that does not restrict your professional judgment. Procedure lists should never be used as a blind checklist in a way that lessens initiative and thoroughness. It is essential to remember that the auditor adds value; otherwise, it would be just like running an automated script. The well-tailored program should not be delayed. The tester should run develop the audit/review programme immediately after he/she has completed a preliminary site or system survey.

Time management is important. Audit programmes prepared too late and hence too close to a deadline are frequently flawed by gaps and inadequacies with the result that they could fail to either determine or give priority to significant issues.

The Assurance of Information Security Risk Management

Risk analysis and risk management are disciplines that have increased in popularity recently (Vaughn et al, 2004) due to a perceived lack of qualified and experienced professionals (Dark, 2004.b).

Fundamentally, IT Risk practitioners need to be able to understand and interpret the fundamental principles of information security (Stallings, 1995).

The practitioner needs to be able to formulate the relationship between objectives policies and procedures. Every objective should have at least one or more policies in place to help reach it (SANS, 2005). They should recognise that most policies would have at least one procedure in place and understand the details and actions required to get the task done.

What is a process?

Processes are the methods that we used to achieve our objectives. We need to ask, "How are processes implemented within an organisation?"

Objectives

An objective is a goal or something that you wish to accomplish. We need to ask, "Who sets objectives and how are these designed to help achieve effective risk management?"

Controls

Controls are the mechanisms through which we reach our goals, but what are controls? How we design and audit controls and thus how to measure the effectiveness of the control is at the heart of the assurance process.

Controls are useless if they are not effective so we must ensure that any control is effective and may be justified in cost terms.

Policies

Policies are themselves controls. Every policy in the organisation should relate to a business or organisational objective. In assessing an organisation, we should always have the following questions at the top of our minds:

- What practices are in effect?
- How does an organisation ensure that the practices are what is in effect?
- Policies and practices should match, how is this checked?
- When a practice does not match policies there is an issue – how do issues get resolved?

System

A system is defined by NIST (in the Risk Assessment Standard Number 800-30) as any collection of processes, and/or devices that accomplishes an objective. The practitioner needs to have a comprehensive understanding of systems design and testing.

Identifying and classify risk.

A risk analysis is a process that consists of numerous stages (Bosworth, 2002; NIST 800-27; Moore, 2001; Dark, 2004.a). The practitioner should become familiar with each of these processes.

- Threat analysis, how is a threat determined?
- Vulnerability analysis, what is a vulnerability?
- Business impact analysis, how will an event have an effect on the organisation's business?
- Likelihood analysis, what is the probability of an event?
- How are these individual components merged in order to deliver the overall risk rating for an organisation and what does that mean?

The Risk analysis process should allow the practitioner to determine the risk for an organisation based on threats and vulnerabilities. From this point, they will be able to classify the severity of the risk and thus assign an overall importance to each risk (Dark, 2004.a; Border, 2006.).

The practitioner should be able to create a risk management plan (SANS, 2005). This should consist of:

- Preparing a risk treatment plan using a variety of control methods.
- Analysing individual risks based on the impact of the threats and vulnerabilities that have been identified from the risks.
- Rate the individual risks from highest to lowest importance.
- Create a risk treatment plan that categorises each of the threats and vulnerabilities in order of its priority to the organisation, together with some possible controls.

An example risk treatment matrix as listed below (as modelled from NIST standard (800-42) and Microsoft (2004)) should be well within the practitioner's capability to create.

No.	Threat/Risk	Priority	Controls						
			Policy	Procedure	Firewall	IDS	Av	Etc	
1	Unauthorised access to application and internal networks	H	*	*	*				
2	Data Integrity	H							
3	Unauthorised transmission of confidential information	H							
4	Data corruption	H							
5	Spoofing	M							

Implementing a risk mitigation strategy

The practitioner must understand what is required for a Gap analysis, and how this allows the identification of controls that have not been implemented (Dodson, 2005). Threat modelling (Myagmar, 2005) and development of attack trees (Moore, 2001) should be taught in order to develop a competence, which will allow the practitioner to decide whether each gap from the gap analysis should be either excepted or mitigated and what type of controls are implemented.

Plan do check act

Originally implemented as a quality control process, ISO 17799 (ISO 17799.2 and subsequently ISO 27001) adopted the PDCA or "plan, do, check, act" methodology. The

practitioner should be aware of this process. This process involves the following stages (Six Sigma, http://www.isixsigma.com):

PLAN

The plan phase consists of an identification of the problem, followed by an analysis of the problem identified. The key components of this phase include threat and vulnerability analysis.

DO

The next phase of the PDCA process requires the development and implementation of ISMS (information security management system) components. This would include controls. The practitioner should understand the categories of controls, and why they have been selected or implemented.

CHECK

The check phase consists of an evaluation of the previously implemented ISMS components for controls. Although audit is a control in itself, it should also be used to measure effectiveness of the overall process and its components.

ACT

Finally, the act phase of a PDCA based process requires that the organisation continuously improve its performance. Using constant incremental improvements, the organisation should be able to improve its security systems consistently in order to minimise risk while remaining cost-effective.

Risk management, security compliance and audit controls

WHAT MAKES UP A RISK PROGRAM?

In order to answer this question the practitioner needs to understand how to identify and quantify the effectiveness and cost of the various risk analysis techniques (Bosworth, 2002). They must understand the risk management process as a whole, and how controls may be implemented to eliminate or mitigate the risk of individual events occurring.

Security compliance has become a major factor in driving risk processes within business and government (Ford, 1997). An understanding of the security controls and measurement techniques, audit controls and processes used to ensure that the controls work within a system is crucial. This should lead to an introduction to the discipline of governance, as it relates to Information Systems.

Risk analysis: techniques and methods

The practitioner needs to be introduced to a variety of risk methods. Any course of risk training should cover some of the key methods defined below.

Overview of Risk Methods

- General types of risk analysis
- FMECA
- CCA
- Risk Dynamics
- Time Based
- Monte Carlo

General risk analysis

Risk analysis is the art (SANS, 2005) and science of determining the real and potential value of an asset, while simultaneously attempting to predict the likelihood of loss based on mitigating security controls [NIST (800-30) and Bosworth, 2002].

Risk analysis models

There are two basic forms of risk analysis:

- Qualitative
- Quantitative

Quantitative analysis will be based on object of data analysing the sufficiency of controls, and uses some numerical method.

Qualitative is designed to analyse the quality of the system from a subjective point of view.

The practitioner must know the differences between these models, the benefits of each end of the downside to selecting either type of risk model.

QUANTITATIVE

The two simple models of quantitative risk that all practitioners must know include:

- Annualised loss.
- Likelihood of loss

In addition, the practitioner should understand that there are other quantitative methods. Some of these methods are detailed later in this paper and should be included as a minimum. Though it is not expected that the practitioner would learn these advanced techniques early in their career they should know of their existence (Dark, 2004.a).

QUALITATIVE

Qualitative analysis is the easiest type analysis, but the results are easily skewed by personal opinion (Bosworth, 2002, Ch 47). These methods are typically focused on measuring or estimating threat and vulnerability. Qualitative analysis is the simplest and cheapest method of analysing risk, but should never be forgotten that perception is not always accurate end of the results are based on guesswork (Dodson, 2005).

FMECA ANALYSIS

MIL-STD-1629 Procedures for Performing a Failure Mode, Effects and Criticality Analysis should be taught in any introductory risk course. Failure mode, effects and criticality analysis helps to identify:

- Risk factors,
- Preventative controls.
- Corrective controls

FMECA couples business continuity planning or disaster recovery into the initial analysis

- identifies potential failures
- identifies the worst case for all failures
- occurrence and effects of failure are reduced through additional controls

The FMECA Process consists of the following stages:

1 Define the system or target

   a. What is the systems mission?

   b. How does the system interface with other systems?

   c. What expectations for example, performance and reliability affect the system

2 Create a block diagrams

   a. FMECA relies on the creation of block diagrams

   b. Diagrams illustrate all functional entities, and how the information flows between them.

3 Identify all possible individual modules system failures and system interface failures:

   a. Every block in every line that connects the block is a potential point of failure.

   b. Identify how each failure would affect the overall mission of the system

4 Analyse each possible failure in a terms of a worst-case scenario.

   a. Determine a severity level for the failure.

   b. Assign this value to the possible outcome.

5 Identify,

   a. Mechanisms for detecting failures.

   b. Compensating controls relating to the failures.

6 Create describe any actions necessary to prevent or eliminate the failure or effects of the failure

   a. the Define additional, setting controls to prevent or detect the failure

7 Analyse and describe any and all effects of the additional controls

a. define the roles and responsibilities to address the compensating controls

8 Document the analysis

a. Explain the problems found in the solutions.

b. Document residual risks -i.e. days without compensating controls.

c. Describe the potential impact of these residual risks.

FMECA SUMMARY

This process involves a detailed analysis based on qualitative methods. It is reasonably objective, helps to identify controls and issues and identifies residual risk. An outcome of the process should include all practitioners being able to complete this process.

CCA - CAUSE CONSEQUENCE ANALYSIS

RISO labs (Riso National Laboratory: 307-312) developed CCA (Cause consequence analysis) which is essentially a fault tree based approach. It is commonly used for analysis of security and safety problems. CCA and fault trees can be easily applied to almost any technology or system (Keong, 2004).

The tree-based approach involves the following steps:

- Identify an event
- Determine the underlying causes of the event.
- For each underlying cause, identify the causes or initiating events.
- Repeat until the underlying cause becomes uncontrollable

The CCA process is repeated until the final underlying cause is beyond the organisation's control (whether through cost or other factors). Thus, the process ends when there is no value in continuing to decompose the problem further.

TWO TREE TYPES

Fault trees

- Identify faults
- Determine underlying causes of the faults

Event trees

- Identify faults.
- Identify consequences

CCA combines both fault trees and event trees. As a result, CCA is good for incident handling analysis, both pre-and post-incident. This helps us to determine how an actual incident may occur. CCA is commonly used as a form of qualitative analysis for determining possible failures

Practitioners should be able to create and analyse fault and event trees in order to diagnose organisational risks.

RISK DYNAMICS

Risk dynamics looks at risk analysis and risk mitigation, as in equilibrium (Rodrigues, 2001). Thus, making a change to any control or other risk factor will affect another term.

Some risk dynamic terms include:

- cost to secure
- level of threat
- severity of the vulnerability
- the impact and consequences of any exposure
- time to detect an incident
- the time to respond to an incident
- recovery time
- the overall risk

Risk dynamics is a qualitative approach to risk that uses the formula:

Threat X Vulnerability = Risk

Practitioners should understand this methodology, its weaknesses and its benefits. They should understand the processes and stages involved with this methodology.

TIME-BASED ANALYSIS (TBA)

Time-based analysis is a quantitative analysis that uses only a small amount of qualitative measures. TBA is extremely effective in measuring the adequacy of a control. This is also useful in terms of fault preparation (Delphi Group, 2005).

TBA involves analysis of the systems to identify:

- The preventative controls (P)
- The detective controls (D)
- And the reactive controls on the system (R)

TBA measures all things in terms of time. As long as the time to detect and react to an incident is less than the amount of time to prevent the fault, risk is maintained at an acceptable level.

Thus, the aim when implementing TBA is to maintain the following situation:

D + R < P

In addition, a measurable loss occurs when:

D + R > P

To analyse controls under a TBA, first assume that preventative controls fail then ask the questions:

- How long does it take detective controls to be enacted?
- How long following detection, does it take a response to be initiated?

The aims of a TBA based risk strategy include reducing both D & R. this can be achieved by improving the detective controls or improving the reactive controls. The TBA model assumes that all preventative controls will eventually fail given enough time (SANS, 2005).

In determining a target, the costs of the preventative, detective and reactive controls are taken into account to create a cost benefit analysis. TBA is one of the simpler quantitative methods of risk analysis and management that is available. All practitioners should be familiar with this methodology.

MONTE CARLO METHOD

A number of stochastic techniques have been developed to aid in the risk management process. These are based on complex mathematical models that use stochastically generated random values to compute likelihood and other ratios for an analysis model.

The Monte Carlo method can also aid in other risk methodologies such as Time-based analysis (Curtis, et al 2001). It further allows the determination of the range of possible outcomes and delivers a normalised distribution of probabilities for likelihood. Combining stochastic techniques with Bayesian probability and complex time series analysis techniques such as Heteroscedastic mapping is mathematically complex, but can aid in situations where accuracy is crucial.

These methods are truly quantitative. They help predict realistic detection, response and thus exposure time. This may be differentiated by the type of attack. This type of statistical method is to have a downside in that they are more expensive than the other methods. The level of knowledge needed to conduct this type of analysis is not readily available and the level of knowledge of the organisation needed by the analyst often excludes using an external consultant in all but the smallest of risk analysis engagements.

SOME EXISTING TOOLS FOR RISK ANALYSIS

Selection of the common tools available should be introduced to the practitioner. Some of the more common tools that may be introduced to the practitioner are included below.

Crystal ball

Crystal ball is a simple Monte Carlo simulation/analysis product. It uses tornado analysis and sampling. Crystal ball is one of the simpler stochastic risk analysis tools available.

Risk +

Risk + is designed for performing schedule risk analysis. It is a simple time based analysis system used to identify potential faults in a fault tree style. Risk + uses Monte Carlo simulations to determine likelihood. This enables the product to demonstrate a possible cost by using the resource allocation values that it has created through cost histograms. This probability histogram is based on stochastically determined outcomes.

Cobra

Cobra is particularly useful for organisations that use ISO 17799 and on as a security model. It is used to measure the ISMS of the organisation against the 10 core controls of ISO 17799.

Cobra uses a cost justification model based on cost benefit analysis. Cobra integrates they risk dynamics based approach to knowledge-based questionnaires.

OCTAVE

As one of the leading risk methodologies, Octave should be included as an attachment to any IT risk course. It would not be expected that a practitioner should understand the process in its entirety, but they should know the fundamentals of how this process works and what its benefits, and downsides are.

Creating a Information Systems Risk Testing Programme

The following is a proposed approach, which is designed to enable the potential risk and security tester/assessor to quickly progress to the level of skills and knowledge to be able to conduct a risk or security assessment or review.

The objectives of any information risk programme should be to introduce the practitioner to arrange of risk assessment models and give them something to use right away. Some of the key skills that should be transferred to the risk practitioner include the following key areas [Dark, 2004.a], which have been defined to be the core components of a risk management service-learning course:

- Being able to competently conduct an information security risk assessment,
- Having a basic understanding and the required knowledge to Perform asset identification and classification for a basic organisation,
- Perform threat identification and understand how to classify threats,
- Perform vulnerability identification and classification based on the organisation's profile,
- Perform a control analysis for a selected organisation,
- Understand how to perform a likelihood determination using both quantitative and qualitative methods,
- Be able to conduct an impact analysis, based on business and management requirements,
- Use the knowledge of processes defined above in order to complete a risk determination for an organisation,
- Identify control recommendations for the organisation and understand the various types of control and implementation programs that are available,
- Developing the skills to enable the practitioner to effectively document the results of the above processes,

- Identify pertinent standards and regulations and their relevance to information security management,
- Describe legal and public relations implications of security and privacy issues.

As such, completion of the course of training should develop the knowledge necessary to allow the practitioner to: [SANS, 2005]

- Identify critical information assets within an organisation that they are familiar with,
- Identify and specify security controls for a variety of systems,
- Specify effective monitoring controls and understand how these may be implemented within an organisation.

Stages to Learning and Developing Risk Assessment Skills

TOPIC 1: INFORMATION SECURITY BASICS

The goal of the first section is to introduce the student practitioner to the basics of IT security and in particular, Information Systems risk management. The practitioner should learn the basics of Information Systems security taxonomy and language, the history of the discipline and where the discipline is moving.

TOPIC 2: NETWORK AND SYSTEM SECURITY

In section two, the practitioner should learn the basic methods used to protect networks and systems from attack. They should learn some of the methods used to protect Information Systems.

TOPIC 3: IS RISK ANALYSIS

In section three, the practitioner would be introduced to the fundamentals of risk analysis. This would include system controls, operational security, auditing and the various standards for security.

At the same time, the practitioner would be exposed to a variety of analysis methodology is through the standards and come to understand the importance of auditing systems.

TOPIC 4: VULNERABILITIES AND VULNERABILITY ASSESSMENT

Topic four should detail the issue of vulnerabilities. The practitioner would be expected to learn what constitutes a vulnerability and the issues associated with vulnerabilities. The practitioner would be exposed to vulnerabilities on different types of systems for example, E-commerce vulnerabilities.

The practitioner would also be exposed to the differences in penetration testing and vulnerability assessments.

TOPIC 5: THREAT ANALYSIS

Knowing how to adequately analyse a threat is a key component of any risk analysis. In section five, the practitioner would be expected to read the NIST standards on threats. The practitioner should be expected to understand how threats develop and how to analyse them for what they are.

TOPIC 6: ATTACKS

Topic six looks at the various types of attacks against systems. The aim of this section is to give the practitioner knowledge of how attacks are defined. At the same time, the practitioner would be expected to be able to create and analyse an attack tree for a selected attack vector.

TOPIC 7: IMPACT ANALYSIS

Topic seven looks at the various types of impact analysis methodologies that are available and how these may be utilised in the risk management process.

TOPIC 8: THE SCOPE OF IT RISK

Topic eight is to look into the scope of an analysis. When conducting a risk program, the practitioner needs to learn how to create and work within a defined scope. This is essential to deliver results, and to stay within budget.

TOPIC 9: CORPORATE GOVERNANCE

One of the major aspects of risk management today has come as a direct result of increased corporate governance requirements. The practitioner should understand how this could influence their organisation's business.

TOPIC 10: EXISTING RISK PROGRAMMES

There are many varieties of risk programmes that are available. The practitioner should understand some of the differences between these programs. It is also important to understand the basic reach of each of the programs including its strengths and weaknesses.

TOPIC 11: CONTROLS

Implementing effective controls is a key component of any risk management program. The practitioner should be able to identify controls, classify controls and to successfully recommend the implementation of selected controls within an organisation.

TOPIC 12: RISK MITIGATION

The overall aim of any risk program is to lower the potential cost to an organisation from an incident. The practitioner should understand some of the effort to implement any controls necessary to protect an organisation. They should understand the cost benefit ratio of risk.

Suggested Readings and Texts on IS Risk

A proposed handbook for the risk assessment practitioner is the "Computer Security Handbook" [Bosworth & Kabay (Ed.) 2002]. Although not specifically a technical handbook, this practitioner text may become an invaluable reference during the assessor's career. This text is a compilation of works from a large number of information security authors. Unlike many books on this topic, it does not focus solely on the technical aspects of security but delivers a detailed knowledge-based approach to Information Systems security learning.

The material in the document, "An Introduction to Computer Security: The NIST Handbook (Special Publication 800-12)" is a comprehensive NIST standard that introduces information systems security and risk.

READINGS

There are various NIST documents referred to in this paper. The NIST standards provide a good grounding for the student or practitioner in a wide variety of risk and general security topics.

Part 2. Audit and Vulnerability Testing Research

Research Abstract

Just as Edsger W. Dijkstra denigrates the concept of "debugging" as being necessitated by sloppy thinking, so to may we relegate external tools based vulnerability tests to the toolbox of the ineffectual security professional.

This report details an experiment that illustrates how “Tools based Ethical Attacks” often do not provide the benefits they purport to hold. The analysis supports the presumption that this type of service may be detrimental to the overall security of an organisation.

Extensive arguments that blind or black box testing may act as a substitute for more in depth internal tests by finding the flaws and allowing the fixing of those flaws before they are exploited are common. This research will show that not only is the premise that external tests are more likely to determine vulnerabilities is inherently flawed, but that this style of testing may actually result in an organisation being more vulnerable to attack.

This leads to the conclusion that “Tools based Ethical Attacks” or a “Pen. Test” may be detrimental to the overall security of an organisation when they replace an effective audit programme as they provide a false sense of security while not adequately disclosing system vulnerabilities.

Introduction

It is a common belief that “Pen.Testing” from an external source for security vulnerabilities is an acceptable method of auditing Internet connected systems. It has been noted that numerous systems vulnerabilities are not reported following external “Pen.Testing” (van Wyk, 2004).

“Tools based Ethical Attacks” have become widely utilised tools in the organisational goal of risk mitigation. The legislative and commercial drivers are a pervasive force behind this push. It should be noted that “Tools based Ethical Attacks” are not in fact penetration tests but rather a limited scanning process. However, perception plays a part

and it is often the case that a “Tools based Ethical Attack” is confused with a full penetration test.

For the purposes of this research, the nomenclature, “Pen.Test” shall be used to define the “Tools based Ethical Attacks” as detailed in the appendix in the section titled “The Methodology – Tools Based External Attacks”. This terminology has been used to represent the marketing derived substitution of a complete “Pen.Test” with the stylised “Pen. Test” methodology used in the experiment.

Pen. Tests (or External vulnerability testing) miss or miss-report a large number of vulnerabilities which exist on a system. As such, it may be further argued that the value of Pen. Tests testing are low compared to a more complete systems audit or threat risk analysis.

It is the hypothesis of this paper that a “Pen. Test” will not discover a large number of system vulnerabilities, which could have been detected using a localised audit technique leaving a system more vulnerable to attack. The review of the results of this experiment provides significant statistical support for this assertion.

The main issue associated with this problem is potentially one of misplaced trust. The results of the experiment reflect that a large amount of money is being spent on information security projects with little benefit to the end user. This is a false sense of security is in fact, dangerous as it reduces the likelihood that systems will be resilient against attack.

The issue at hand is that “Pen. Tests” divert funds from internal infrastructure projects while delivering a lower level of assurance than comparable offerings. As a result, information systems are left at risk of attack unnecessarily. Additionally, the vendor faces an increased risk of litigation, due to the provision of inadequate services.

The primary hypothesis of this paper is that an external test for vulnerabilities on a system will miss several key vulnerabilities, which would be detected from a more thorough audit (using the threat risk methodology listed in the appendix).

This experiment used a test bed of servers set up on two separate networks (internal and DMZ). It involves testing systems configured to each of the SANS and NIST hardening standards.

Systems where tested using both “Pen.Test” methodologies and internal audit/test methodologies to record the numbers of vulnerabilities reported in each of four categories. A control system was in place to account for the control state (i.e. the total of all vulnerabilities on each system). This allowed the mapping of false positives from each test methodology.

This experiment was designed as an exploratory course of study using a replicable set of procedures to disprove the perceived value of external tools based “Pen.Testing” over threat-risk based systems audits.

In order to demonstrate this data was collected from network sensors, vulnerability-testing software, and a known audit source as a control.

This process involved Investigation by Correlation and Hypothesis testing in a Quantitative manner. A part of this experiment involved a descriptive study of a variety of variables to test a number of hypothesises.

This report details the method used in analysing the data obtained during experimental research into the adequacy of “Tools based Ethical Attacks” as an audit methodology. The analysis of in this data will be addressed in five sections represent each of the individual hypotheses.

Hypothesis number one, that external tools based penetration test or a “Pen.Test” is less effective than an IT audit is designed to demonstrate a deficiency in Pen. Test procedures when compared to a more complete audit methodology.

The second hypothesis, that an audit, which takes the same amount of time as an external “Pen.Test” will deliver better results than the “Pen.Test” is demonstrated, if our initial hypothesis is correct. This is a direct result of the testing methodology. The Audit was time limited in this test to ensure that it was completed in a shorter amount of time than the “Pen.Test”.

A complete review based on known attacks was conducted on each of these systems based on current industry best practice. This review delivered a complete snapshot of all the systems vulnerabilities able to be used as a control for this experiment.

Using a control set of all known vulnerabilities it is possible to test the hypothesis that an external "Pen.Test" will not find all the vulnerabilities on a system. This is the third hypothesis being testing.

We are able to test our fourth hypothesis against the control set. This is that an effective threat-risk based audit will find most or all systems vulnerabilities.

A control set of a sample of all known vulnerabilities applicable to the systems was tested. It also enabled a comparison of any false positives that had been detected by either the audit or the "Pen.Test". Any vulnerability found by either the audit or the "Pen.Test" that is not discovered to be a "real" vulnerability against the known control set was listed as a false positive. This allowed the testing of the hypothesis that an audit of the system will discover a lower number of false positives than the "Pen.Test".

Theoretical framework

To determine the level of risk faced by an information system, it is necessary to determine:

- The level susceptibility faced by system. The number of vulnerabilities and the comparative level on those vulnerabilities affecting an information system measures this.
- What are the threats to be system?
- What is the impact of the system because of the threats and vulnerabilities?

For this reason, an accurate level of the vulnerabilities affecting an information system needs to be accurately determined. Without an adequate determination of the vulnerabilities, it is not possible to quantify the level of risk accurately.

The accurate assessment of risk is a key component of good corporate governance covering the management of information systems. In accurate methodologies used in the

early stages of vulnerability and threat detection skew the calculated risk are results, resulting in an ineffectual measure.

Research Question

Are "Tools based Ethical Attacks" an effective method in determining the level of system vulnerability?

Hypotheses

1 A "Pen.Test" is less effective than an interactive systems audit in discovering and reporting known vulnerabilities.

2 Time spent auditing a system will be more productive than the same amount of time spent conducting a "Pen.Test".

3 External "Pen.Tests" will not find all the vulnerabilities affecting a computer system.

4 A well-conducted methodological systems audit will discover most if not all vulnerabilities on the system.

5 A "Pen.Test" will result in the detection of a larger number of false positives (or nonexistent vulnerabilities) than an effective threat-risk based systems audit.

A review of the various methodologies

What passes as an Audit

An “ethical attack” or ““Pen.Testing” is a service designed to find and exploit (albeit legitimately) the vulnerabilities in a system rather than weaknesses in its controls. Conversely, an audit is a test of those controls in a scientific manner. An audit must be designed to be replicable and systematic through the collection and evaluation of empirical evidence.

The goal of an “ethical attack” is to determine and report the largest volume vulnerabilities as may be detected. Conversely, the goal of an audit is to corroborate or rebut the premise that systems controls are functionally correct through the collection of observed proofs.

As Fred Cohen has noted (Penetration Tests?, Cohen 1997) this may result in cases where “*Penetration Tests will succeed at detecting vulnerability even though controls are functioning as they should be. Similarly, it is quite common for Penetration Tests to fail to detect a vulnerability even though controls are not operating at all as they should be*”.

When engaged in the testing of a system, the common flaws will generally be found quickly during testing. As the engagement goes on, less and less (and generally more obscure and difficult to determine) vulnerabilities will be discovered in a generally logarithmic manner. Most “Tools based Ethical Attacks” fail to achieve comparable results to an attacker for this reason. The “ethical attacker” has a timeframe and budgetary limits on what they can test.

On the contrary, an attacker is often willing to leave a process running long after the budget of the auditor has been exhausted. A resulting vulnerability that may be obscure and difficult to determine in the timeframe of an “external attack” is just as likely (if not more so) to be the one that compromises the integrity of your system than the one discovered early on in the testing.

One of the key issues in ensuring the completeness of an audit is that the audit staff are adequately trained both in audit skills as well as in the systems they have to audit. It is all

too common to have auditors involved in router and network evaluations who have never been trained nor have any practical skills in networking or any network devices.

Often it is argued that a good checklist developed by a competent reviewer will make up for the lack of skills held by the work-floor audit member, but this person is less likely to know when they are not being entirely informed by the organisation they are meant to audit. Many “techies” will find great sport in feeding misinformation to an unskilled auditor leading to a compromise of the audit process. This of course has its roots in the near universal mistrust of the auditor in many sections of the community.

It needs to be stressed that the real reason for an audit is not the allocation of blame, but as a requirement in a process of continual improvement. One of the major failings in an audit is the propensity for organisations to seek to hide information from the auditor. This is true of many types of audit, not just IT.

For both of the preceding reasons it is important to ensure that all audit staff have sufficient technical knowledge and skills to both ensure that they have completed the audit correctly and to be able to determine when information is withheld.

Though it is often cast in this manner, an external test using the tools based “*ethical attack*” methodology is in no way a complete systems audit. To again quote Fred Cohen,

- “Pen.Testing” is an effort to penetrate a system in order to demonstrate that protection has weaknesses.
- Protection testing is a way to confirm or refute, through empirical evidence that controls are functioning, as they should be.

Method Section

Data Classifications

Data collected from the vulnerability scanners was collated into the following categories:

- Informational
- Low-level vulnerability
- Medium level vulnerability
- High-level vulnerability

The basis for using these levels is derived from the methodologies developed by SANs, NIST and CIS and is released as S.C.O.R.E. These methodologies provide an explanation on how these levels are determined.

Additionally the data was assigned according to the following categories:

- Exploitable Externally
- Exploitable Internally
- False Positive

Data is defined to be exploitable externally, where the vulnerable condition may be directly accessed through the Internet by an attacker. Exploitable internally, has been defined as, the condition where the "attacker" must reside inside the firewall to be able to successfully complete the attack on the systems. If the attack is one that may not be concluded successfully from inside the network, it is deemed an externally exploitable attack. To reduce bias all of the definitions have been directly replicated from the SCORE methodology.

High and critical level attacks are reported in the data table high for analysis purposes. Both high and critical levels of vulnerability may result in a system compromise. The table field designated as suspicious has been tabulated within the informational data field of the test results. Nessus simplifies the scoring of data. Nessus includes the level of the vulnerability in its scan report output.

It is intended that these results will enable us to summarise the key concerns of this report:

- Total High-level vulnerabilities - Exploitable Externally
- Total System vulnerabilities - Exploitable Externally
- Total High-level vulnerabilities - Exploitable Internally
- Total System vulnerabilities - Exploitable Internally
- Total False Positives

All vulnerabilities, which have been listed in the results, had to be verified manually. Any vulnerability, which has been listed in the report, whether internal or externally exploitable, which in fact cannot be exploited is deemed a false positive

Table 1 – Vulnerability Levels

	Critical	High	Medium	Low	Suspicious
Denial of Service Attack (DOS or DDOS)	Current and continuing loss of service	Possible loss of service if action is not taken	Service could be slightly effected if the attack was to ensue	No loss of service likely to occur	ICMP or large traffic amounts that are unlikely to effect service
Interactive System level compromise	Compromised systems or evidence of such an attempt				
Unauthorised file access/ modification	Compromised systems or evidence of such an attempt	Suspicion of or attempts to access to protected files			
Blocked attacks as noted on the Firewall	Packets that are bypassing the installed firewall policy	Evidence of packet shaping / detailed spoofing in order to bypass firewall rules	Packets targeted at a specific service that may be vulnerable from other sites	General scans	Misc dropped packets
Attacks as noted on the DMZ IDS hosts	System vulnerable to this attack	Targeted attacks on an open service (esp if recently patched)	Detailed probes and continui ng scans against specific services	General Scans	
Virus or Worm attacks	Systems infected	Evidence of a virus or worm passing the Anti -virus system	New virus or worm detected	Virus or worm blocked on external anti -virus server	

Data Collection Process

A network consisting of Microsoft Systems[i] that are configured using Windows 2000 with the latest service packs, Linux, UNIX and Networking equipment was implemented.

This network has been designed to simulate a corporate network. A single test will be done against all non-Microsoft Systems. The tests on the Microsoft Systems will be done three times in the following configurations:

1. Default (Out of the Box) Security Levels
2. Microsoft Secure Server Templates to be applied
3. Microsoft Secure Server – High Security Template to be applied and the host will be configured to S.C.O.R.E. Level 1 (defined as the *Minimum due care security configuration recommendations*)[ii].

All Linux, Sun Solaris and Cisco equipment will be configured to the Sans S.C.O.R.E. level 1 security Benchmarks (available from http://www.sans.org/score/).

All systems were patched to the level determined by an automated download to be completed 1 day prior to the first test. The testing was started on the 8th April 2006. No new patching was done on any system from this time until after the completion of all testing.

The systems were configured on an existing corporate network to simulate a realistic audit for this experiment. Permission was obtained to use these hosts and this network prior to the initiation of the experiment.

Thus our population will be an external test ("Pen.Test") from an external test point of the DMZ (23 systems), as well as the 23 Internal systems (Microsoft systems are tested three times to account for the separate levels). In addition, the Internal testing (Audit) will covered these same systems.

System	Host System	DMZ Systems	Internal Systems
Network Equipment	Cisco IOS Router External	External	External
	Cisco IOS Switch	2xxx Switch	2xxx Switch
Linux Hosts	Redhat Linux	Apache Sendmail Bind	Apache Sendmail Bind IMAP MySQL
Solaris	Sun T1	Netscape	Apache

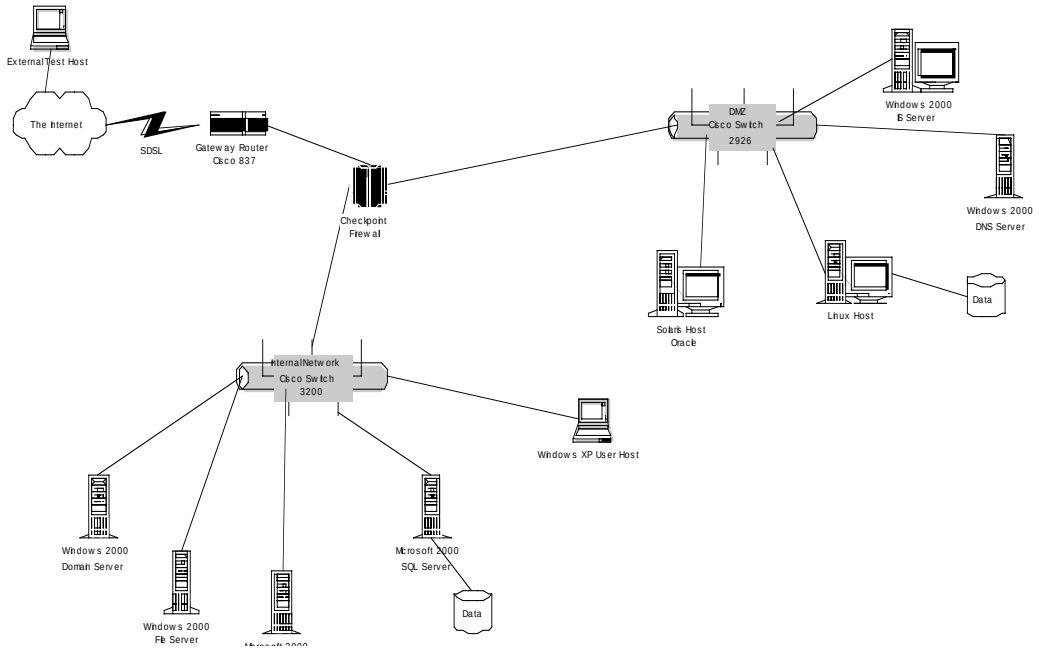
Hosts	Solaris 2.8	Sendmail Bind	Sendmail Bind IMAP Oracle 9i
	Firewall	Checkpoint FW-1 Version VPN-1 2000	
Microsoft	Exchange Server	Microsoft Exchange Server 2000 IIS 6.0	Microsoft Exchange Server 2000 IIS 6.0
	Web Server	IIS 6.0	IIS 6.0 SQL Server 2000
	Domain Server	N/A	Microsoft DNS Active Directory
	External DNS	Microsoft DNS	Microsoft DNS
	File Server	N/A	CIFS, DFS
	Workstation	N/A	Workstation Microsoft Office Tools

Table 2 - A table of systems to be tested

Using a defined methodology all systems were tested using an audit methodology with unlimited time and resources. This audit created the baseline for the data collection process, and thus the control for this experiment.

All systems are on a single segmented network range with no use of Network Address Translation
[NAT].

Figure - 1 – Network Diagram



Vulnerability testing was completed using a network auditing system based on Knoppix (which is booted from a Mini-Linux CD) so that no variation of configuration is possible.[iii] Vulnerability testing will be done from external perspective in order to simulate a "Pen.Test". The same network-testing engine will be configured on internal systems for a comparison. This shall be used to enable an analysis of externally, versus internally accessible vulnerabilities. The test systems used the "Auditor security collection"[iv] CD images

Actual Data Collection

Sonny Susilo from BDO Chartered Accountants was the primary tester engaged to complete the internal and external "Pen.Tests". Mr Susilo ran both the tools as proposed in this experiment, as well as a variety of commercial tools. The tools used by the experiment performed at least as well as the commercial tools, if not better.

The "Pen.Tests" using Nessus and other tools were run on the same systems a second time, independently to verify the results. The same data was collected in both instances. The experiment was not modified to include the results of the commercial tools, as this provided no extra insight.

It was surprising to notice that the freely available toolset provided consistently better results than the commercially supplied product. The testing ran smoothly for the most part. The only issue being a need to reboot the test host on one occasion.

Data Collected

The collected data for the results of the audit, the "Pen.Test" and the control audit are included in the appendix to this document.

In order to determine if any errors and discrepancies exist in the data, a detailed control set of all known vulnerabilities on the systems was completed using the S.C.O.R.E. methodologies for secure systems. The highest levels of methodology were used for testing though this far exceeds the normal requirements of a secured server in most operational systems.

The tables below show a summary of the results obtained from the experiment.

The audit revealed a large number of vulnerabilities from the systems. As may be determined from the table below, the audit process missed a very few vulnerabilities on the systems.

	System	Total High-level vulnerabilities - Exploitable Externally	Total vulnerabilities - Exploitable Externally	Total High-level vulnerabilities - Exploitable Internally	Total System vulnerabilities - Exploitable Internally
1	Cisco Router	0	0	0	3
2	Cisco Switch	0	1	0	9
3	Checkpoint Firewall	0	2	0	2
4	Apache Host	0	6	2	12
5	Redhat Linux Web	4	18	5	25
6	Exchange Server	4	13	5	47
7	IIS Web Server	4	36	5	69
8	Domain Server	2	20	4	47
9	External DNS	4	20	5	27
10	File Server	0	0	5	58
11	Workstation	0	2	8	43
12	Exchange Server	1	8	4	23
13	IIS Web Server	3	6	1	21
14	Domain Server	2	15	2	20
15	External DNS	1	11	3	23
16	File Server	0	0	2	37
17	Workstation	0	0	3	34
18	Exchange Server	1	6	1	15
19	IIS Web Server	2	8	1	10
20	Domain Server	0	2	0	6
21	External DNS	1	12	0	15
22	File Server	0	0	0	13
23	Workstation	0	0	0	15
		29	**186**	**56**	**574**
	% of Control Total	96.67%	99.47%	96.55%	99.14%

Audit Summary

From these results it could be determined the systems audit vulnerabilities is an effective method of determining the majority of vulnerabilities on a system.

The “Pen.Test” appeared to have discovered a significant level of vulnerabilities when looked at in isolation and not relative to the total number of vulnerabilities for the systems as determined by the control.

	System	**Total High-level vulnerabilities - Exploitable Externally**	Total vulnerabilities - Exploitable Externally	Total High-level vulnerabilities - Exploitable Internally	Total System vulnerabilities - Exploitable Internally
1	Cisco Router	0	0	0	1
2	Cisco Switch	0	**0**	0	3
3	Checkpoint Firewall	0	2	0	2
4	Apache Host	0	6	2	12
5	Redhat Linux Web	0	1	5	21
6	Exchange Server	1	5	2	11
7	IIS Web Server	1	6	2	18
8	Domain Server	0	0	2	19
9	External DNS	1	5	2	19
10	File Server	0	0	2	12
11	Workstation	0	0	8	26
12	Exchange Server	0	2	1	7
13	IIS Web Server	0	5	1	12
14	Domain Server	0	0	2	13
15	External DNS	1	4	2	12
16	File Server	0	0	1	7
17	Workstation	0	0	3	18
18	Exchange Server	0	1	1	5
19	IIS Web Server	0	1	1	5
20	Domain Server	0	0	0	0
21	External DNS	0	0	0	0
22	File Server	0	0	0	0
23	Workstation	0	0	0	0
		4	38	37	223
	% of Control Total	13.33%	20.32%	63.79%	38.51%

“Pen.Test” Summary

The issue with this is that this could lead to a false sense of security based on these results. In the comparable in low amount of high-level exploitable vulnerabilities, which were determined by this test, coupled with the total number of vulnerabilities determined would appear to make this a risky proposition.

The control testing determined the total number of known vulnerabilities on the systems. This is our control, which we may use to determine the effectiveness of the other two tests.

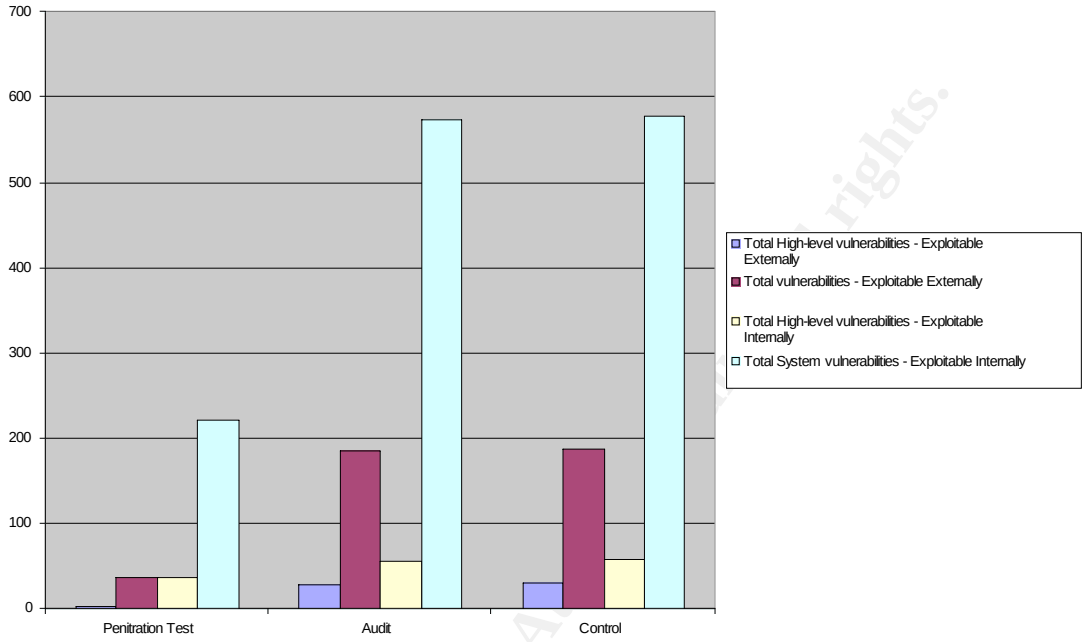
	System	**Total High-level vulnerabilities - Exploitable Externally**	Total vulnerabilities - Exploitable Externally	Total High-level vulnerabilities - Exploitable Internally	Total System vulnerabilities - Exploitable Internally
1	Cisco Router	0	0	0	3
2	Cisco Switch	0	**1**	0	9
3	Checkpoint Firewall	0	2	0	2
4	Apache Host	1	7	3	13
5	Redhat Linux Web	4	18	5	25
6	Exchange Server	4	13	5	50
7	IIS Web Server	4	36	5	69
8	Domain Server	2	20	4	47
9	External DNS	4	20	5	27
10	File Server	0	0	5	58
11	Workstation	0	2	8	43
12	Exchange Server	1	8	4	23
13	IIS Web Server	3	6	1	21
14	Domain Server	2	15	2	20
15	External DNS	1	11	3	23
16	File Server	0	0	2	37
17	Workstation	0	0	3	34
18	Exchange Server	1	6	1	15
19	IIS Web Server	2	8	1	10
20	Domain Server	0	2	0	6
21	External DNS	1	12	0	15
22	File Server	0	0	0	13
23	Workstation	0	0	1	16
		30	187	58	579

Control Data - Complete system review

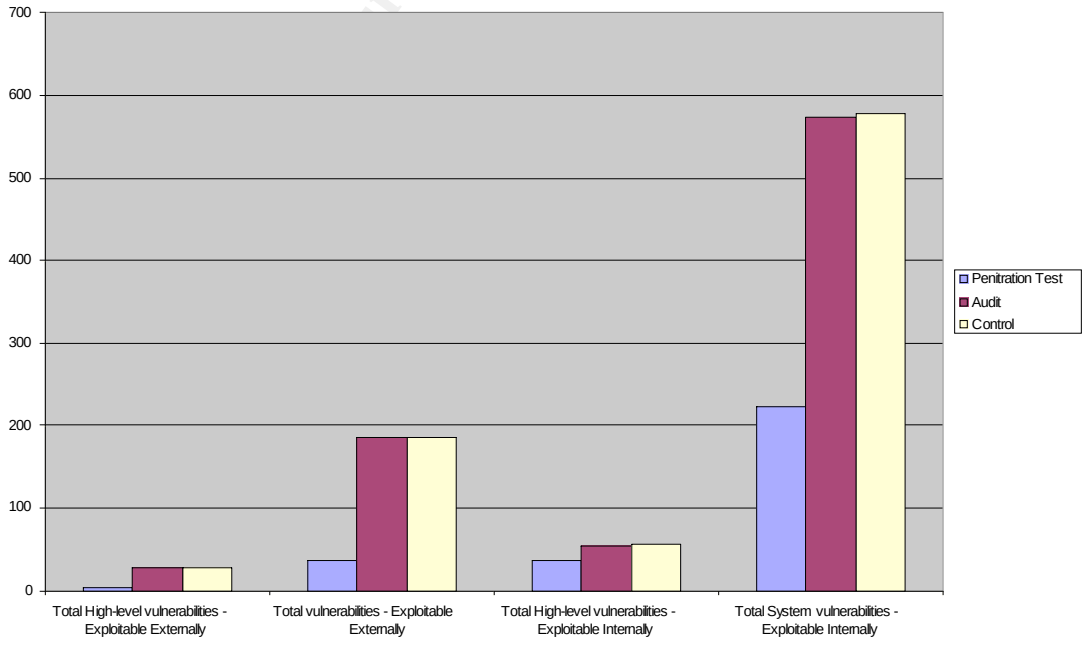
A challenge to the control if mounted would have to be based on the strength of the control used. S.C.O.R.E. is a publicly available peer reviewed system. The Centre for Information Security [CIS] and SANS developed and maintained this methodology for this reason. As the S.C.O.R.E. methodology is generally accepted within the information security community, this is unlikely to be an issue.

Exploratory analysis

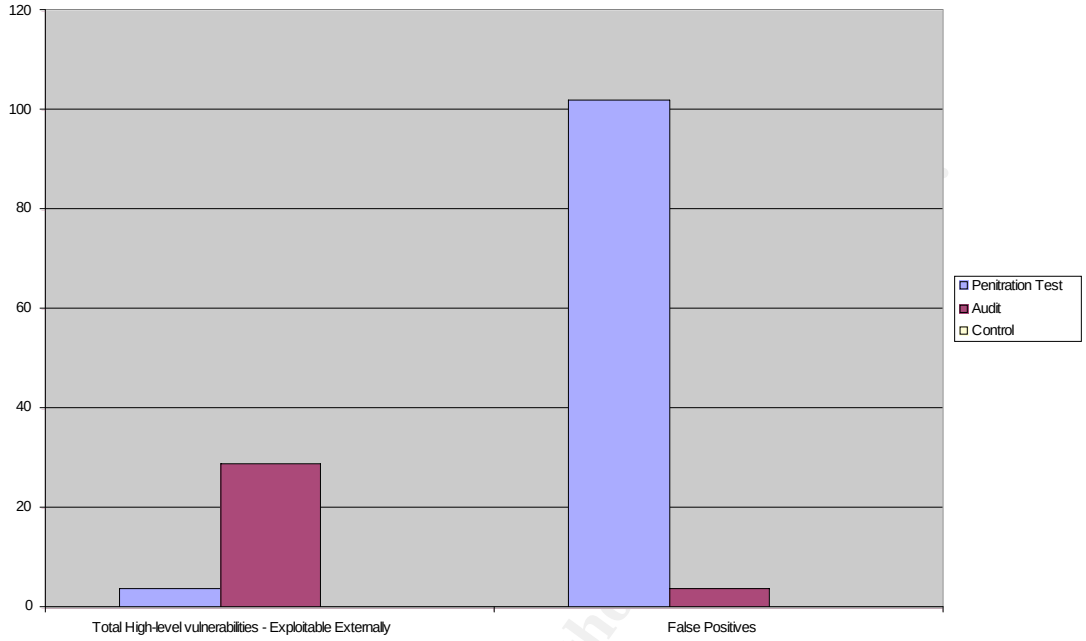
An exploratory analysis of the data would appear to show a relationship between the results obtained from the audit and control sets. The graph below demonstrates this.



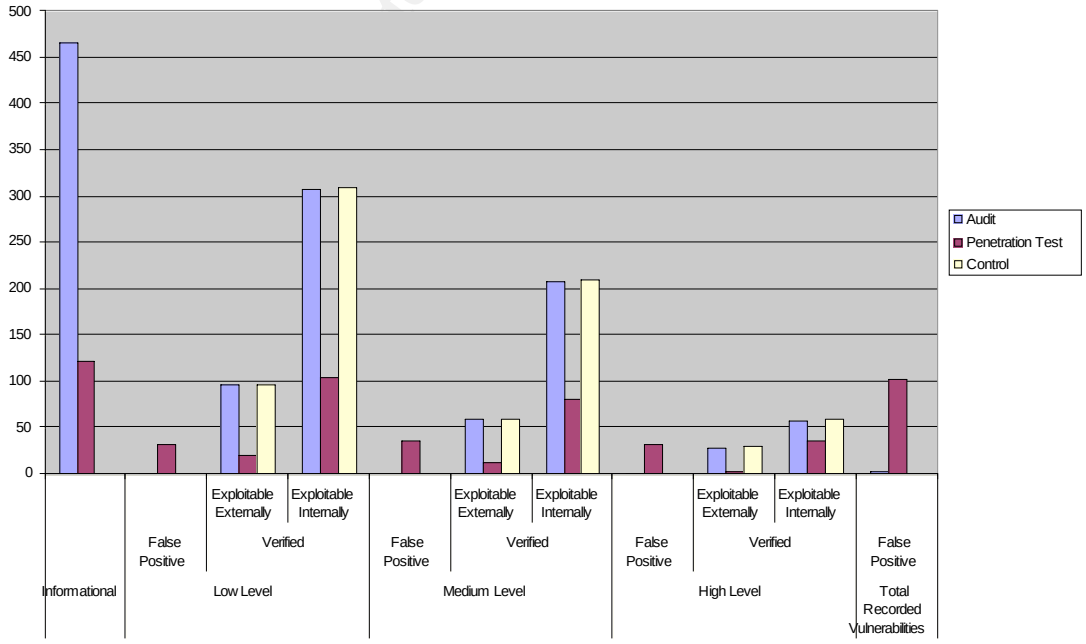
Each of the individual fields correlates closely between the audit and control sets. The "Pen.Test" data appears to vary quite significantly as can be seen in the following graph.



Additionally it would appear that a relationship between the false positives and the penetration and audit tests does not exist.



The data correlates between audit and control in all fields were the control exists. This includes low, medium and high level vulnerabilities. There is little if any apparent relationship between the "Pen.Test" and the control.



Further, the "Pen.Test" clearly gives a smaller quantity of results than the audit in all situations. As is shown in the graph above, the fields, which could not be captured by a control, do not appear to relate between the audit and "Pen.Test". The information between the false positives, and gathered information would appear to have no relationship.

	"Pen.Test"	Audit	Control
Total High-level vulnerabilities - Exploitable Externally	4	29	30
Total vulnerabilities - Exploitable Externally	38	186	187
Total High-level vulnerabilities - Exploitable Internally	37	56	58
Total System vulnerabilities - Exploitable Internally	223	574	579
Total High-level vulnerabilities - Exploitable Externally	4	29	0
False Positives	102	4	0

No results appear to be outliers in the data, and as such, all data will be used without alteration. A preliminary analysis of the data supports the hypothesis in all cases.

From the results, it can be seen that the mean and standard deviations of the audit and control data are closely correlated. The mean and standard deviation for each of the "Pen.Tests" appears to be significantly different to both the audit and control data results.

The preliminary analysis of the data would appear to support the first hypothesis that external tools based "Pen.Testing" is less effective than an IT audit. As the third hypothesis that a "Pen.Test" will not find all vulnerabilities on a system and forth hypothesis, that a threat-risk audit will find most or all system vulnerabilities would appear to be correct, the initial hypothesis is further supported. The time for audit and "Pen.Test" was equal, the hypothesis that an audit of equal cost (i.e. time to complete - hours spent) will deliver more value to the end user is supported if the first and second hypotheses are both correct.

Data Analysis

The results of the "Pen.Test", audit, and control group testing were loaded into an Excel spreadsheet for preliminary analysis. This data was than imported into the JMP statistical package (SAS version 5.1). Finally, this data was also SPSS (for windows version 11.0) for additional statistical analysis.

The primary question as to whether external tools based “Pen.Testing” is able to compare with a systems audit for the detection of vulnerabilities on a system was analysed between each category (high, medium and low) as well as across the totals.

Two sample t tests or ANOVA testing was conducted across the different groups, to test if there were statistical differences between each of the test methodologies. Analysis of variance test will be completed to determine t or F ratio and thus linear strength of the relationships between the groups.

Additional statistical analysis was conducted on the audit and “Pen.Test” to determine if the level of false positives produced in the testing methodology was significant

In each of the tests, the null hypothesis that there are no associations between either of the test types and the control would be rejected if the t or F results at $\alpha = 0.01$.

Finally, the Tukey-Kramer coefficient and between pairs shall be analysed if the ANOVA has rejected the null at the Alpha equals 1% level to investigate paired relationships between the audit, “Pen.Test” and control results.

The results of each of these analyses have been included in the appendix to this document.

Results of the Data Analysis

The results of our test may be summarised in the table below.

	"Pen.Test"		Audit		Control
Total High-level vulnerabilities - Exploitable Externally	4	13.33%	29	96.67%	30
Total vulnerabilities - Exploitable Externally	38	20.32%	186	99.47%	187
Total High-level vulnerabilities - Exploitable Internally	37	63.79%	56	96.55%	58
Total System vulnerabilities - Exploitable Internally	223	38.51%	574	99.14%	579
Total High-level vulnerabilities - Exploitable Externally	4	13.33%	29	96.67%	30
False Positives	102		4		0

Table 3 - Summary of findings

From this table, it is possible to deduce that a report of findings issued from the "Pen.Test" would be taken to be significant when presented to an organisation's management. Without taking reference to either the audit or the control results as to the total number of vulnerabilities on a system, the "Pen.Test" would appear to provide valuable information to an organisation.

However when viewed against the total number of vulnerabilities, which may be exploited on the system, the "Pen.Test" methodology fails to report a significant result. Of primary concern, the "Pen.Test" only reported 13.3% of the total number of high-level vulnerabilities, which may be exploited externally on the test systems. Compared to the system audit, which reported 96.7% of the externally exploitable high-level vulnerabilities on the system, the "Pen.Test" methodology has been unsuccessful.

Experiences from the Design

Sonny Susilo from BDO Chartered Accountants was the primary tester engaged to complete the internal and external "Pen.Test". Sonny ran both the tools as proposed in this experiment, as well as a variety of commercial tools. The tools used by the experiment performed at least as well as the commercial tools, if not better.

The "Pen.Test" using Nessus and the methodology in the appendix was run on the same systems a second time, independently to verify the results. The same data was collected in both instances. The experiment was not modified to include the results of the commercial tools, as this provided no extra insight.

It was surprising to notice that the freely available toolset provided consistently better results than the commercially supplied product. The testing ran smoothly for the most part. The only issue being a need to reboot the test host on one occasion.

One interesting result of the experiment involved the relative times to complete the "Pen.Tests" against the various Windows systems. It was determined that scans of the more highly secured Windows systems took a greater time to complete and the scans against the least secured systems.

The main reason for this result was the comparative lack of responsiveness from the secured hosts. As the secured host did not respond to port scans involving closed TCP ports, the scanning engine quickly ran resources whilst waiting for TCP response packets.

Tools Based External tools based "Pen.Testing" is less effective than an IT audit

To demonstrate that an external "Pen.Test" is less effective than auditing the data it is essential to show that both the level of high-level vulnerabilities detected as well as the total level vulnerabilities discovered by the "Pen.Test" are significantly less than the number discovered during an audit.

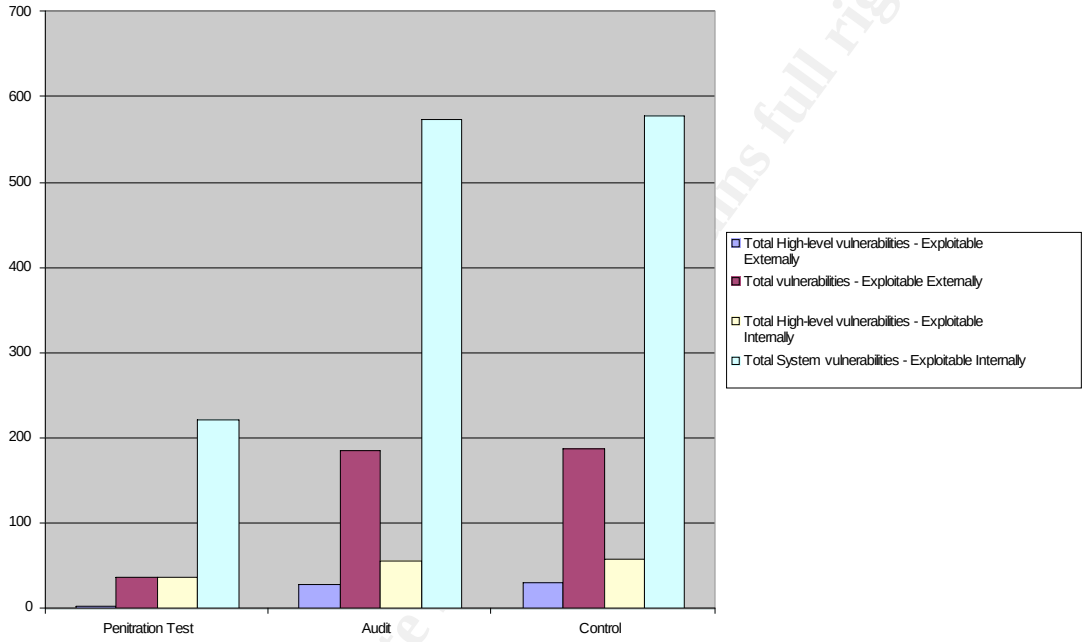


Figure 2 - Graph of Vulnerabilities found by Test type

As may be seen in *Figure 2 - Graph of Vulnerabilities found by Test type* and *Figure 3 - Graph of Vulnerabilities found by exploit type* that the total level of vulnerabilities discovered as well as a the high-level vulnerabilities are appreciably less in the "Pen.Test" results and from the audit results.

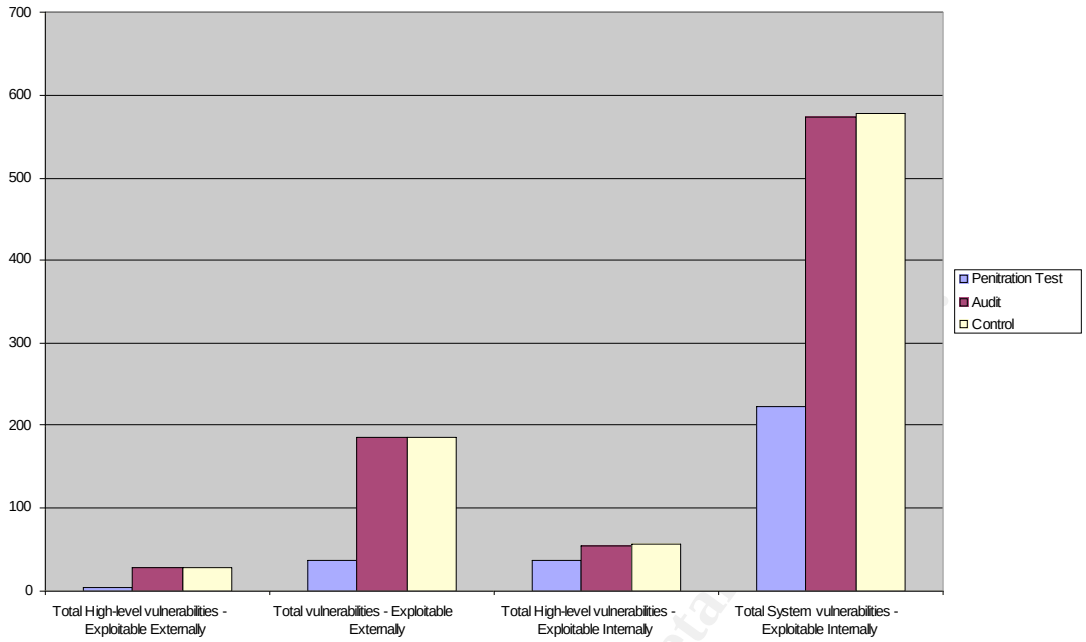


Figure 3 - Graph of Vulnerabilities found by exploit type

The primary indicator of the success of the "Pen.Test" would be both and detection of high-level vulnerabilities and the detection of a large number of vulnerabilities over all.

It is clear from *Figure 4 - Graph of Vulnerabilities* that the "Pen.Test" methodology reported a smaller number of exploitable external vulnerabilities both as a whole and when comparing only the high-level vulnerability results.

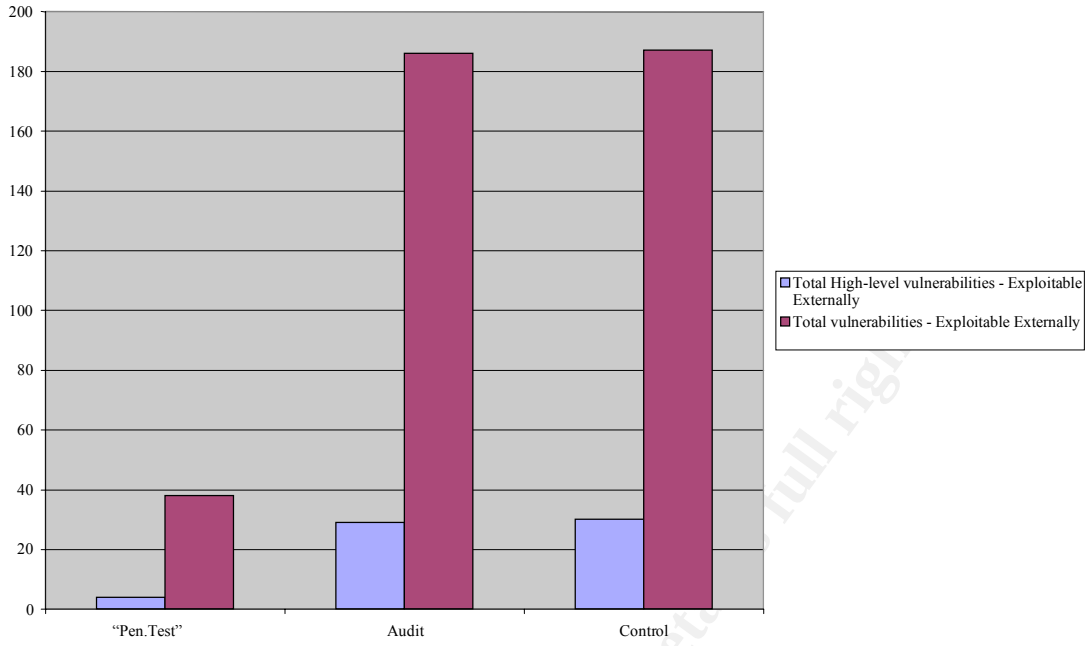


Figure 4 - Graph of Vulnerabilities

Each hypothesis was then tested. The first hypothesis, that external tools based "Pen.Testing" is less effective than an IT systems audit comes primarily because of proving each of the other hypotheses.

External tools based "Pen.Testing" is less effective than an interactive systems audit in discovering and reporting known vulnerabilities.

Null: H_{1o} $\mu_A = \mu_P$ or the alternate hypotheses,

Alternative: H_{1a} $\mu_A > \mu_P$ where ($\acute{\alpha} = 0.01$)

From Appendix 5, *2 sample t test results (P 74)*, the results ($t = 3.275$; $p=0.0010$) substantiates *hypothesis 1* as the p value is significantly lower than the acceptance level of $p < 0.01$.

Further, *One-way Analysis of Total vulnerabilities - Exploitable Externally By Type of Test*, supports the previous results ($t = 3.2905$; $p=0.0001$) and substantiates the alternate for *hypothesis 1* as the p value is significantly lower than the acceptance level of $\acute{\alpha} < 0.01$.

There is strong evidence that IT Systems Audit will report a greater number of systems vulnerabilities than External "Pen.Testing".

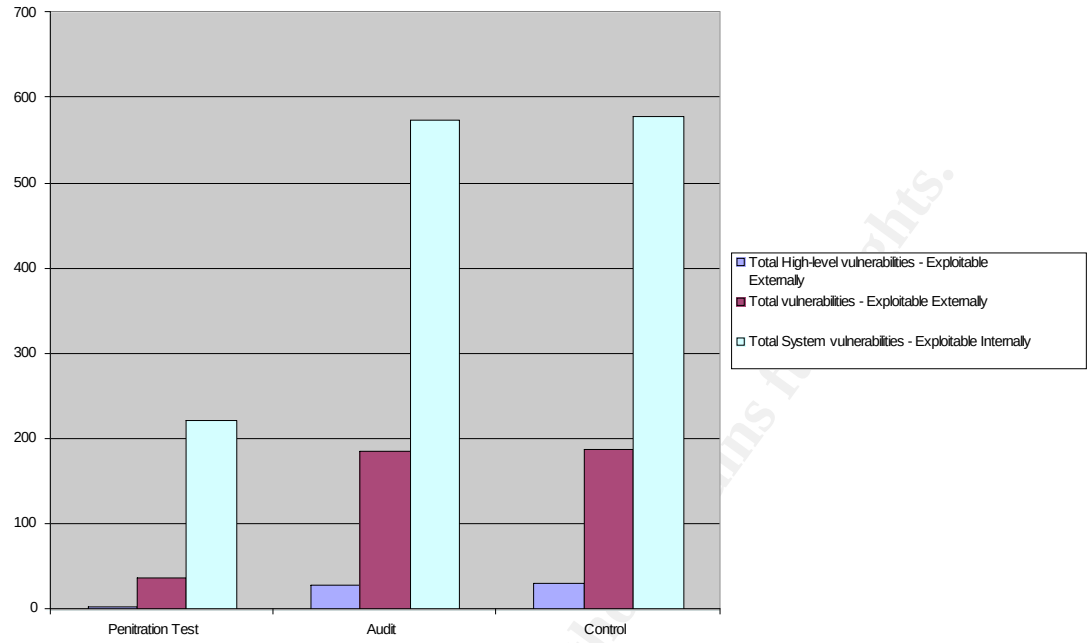


Figure 5 - Comparison of Test results

More appreciably, external testing fails to detect vulnerabilities, which may be exploited inside the network. As *Figure 5 - Comparison of Test results* demonstrates that "Pen.Test" methodologies fair equally poorly when testing internal vulnerabilities.

Testing of Hypothesis three

The hypotheses that:

External "Pen.Tests" will not find all the vulnerabilities affecting a computer system.

May be expressed as follows:

Null: H_{3o} $\mu_C = \mu_P$ or the alternate hypotheses,

Alternative: H_{3a} $\mu_C < \mu_P$ where ($\acute{\alpha} = 0.01$)

The hypothesis that an external "Pen.Test" will not find all vulnerabilities on a system was tested using 2-sample t tests. The Results of these tests are included in Appendix 5,

subsection - *Hypotheses Three - An external "Pen.Test" will not find all vulnerabilities on a system*.

Testing the total vulnerabilities exploitable externally for the "Pen.Test" results against the control (*2 sample t tests, P*) showed significant difference ($t = 3.32$; $p=0.0009$) between these results. Likewise, the results from a comparison of the total systems vulnerability (*One-way Analysis of Total System vulnerabilities - Exploitable Internally By Type of Test*), which were exploitable internally of the "Pen.Test" results compared to the control results also showed significant difference ($t = 3.77$; $p=0.0002$). Thus, we reject the Null and *Hypothesis 3* is substantiated.

As it may be shown that the "Pen.Test" results are significantly different to the control set for all tests, both external and internal, there is overwhelming support for the hypothesis that an external "Pen.Test" will not find all the vulnerabilities on a system.

In fact, we can be confident that are "Pen.Test" will only find a small percentage of the total number of vulnerabilities affecting a system.

Hypothesis four

Moreover, the hypothesis that an audit will find most or all the vulnerabilities on a system is supported by the findings. *Figure 6 - Vulnerability comparison* demonstrates that unlike the "Pen.Test" results, a system audit will relate strongly to the control set.

These results are reported in Appendix 5.4 - *Hypotheses Four - An audit will find most or all system vulnerabilities.*

This hypothesis may be expressed as follows: *A well-conducted methodological systems audit will discover most if not all vulnerabilities on the system*.

Null: H3o $\mu_A = \mu_C$ or the alternate hypotheses,

Alternative: H3a $\mu_A < \mu_C$ where ($\alpha = 0.05$)

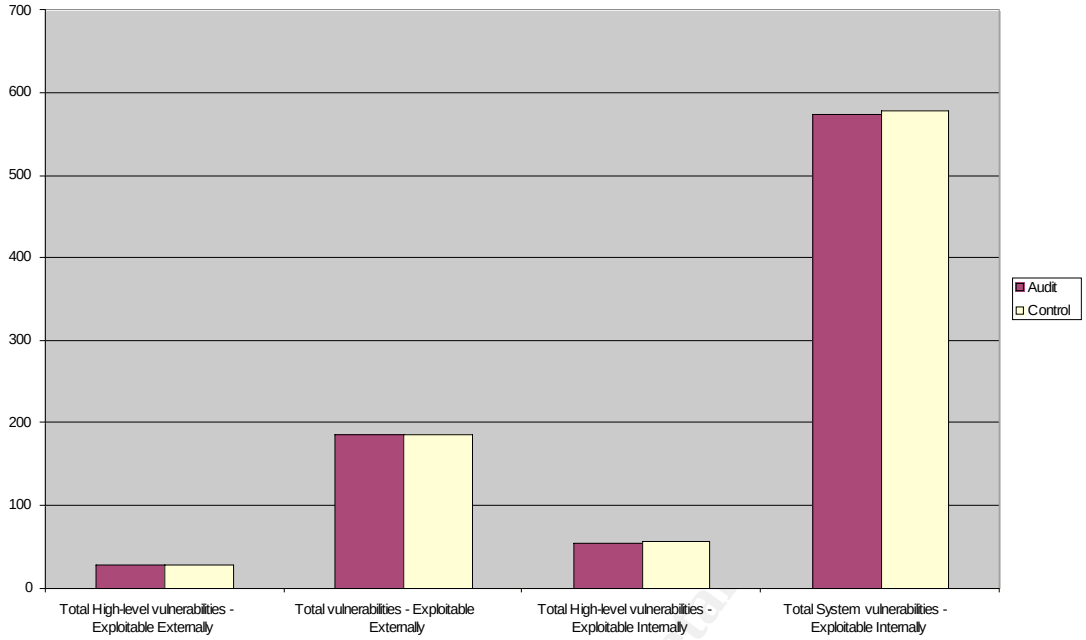


Figure 6 - Vulnerability comparison

The results of the 2-sample t test for the total system vulnerabilities when comparing the audit to the controls set ($t = -0.041$; $p=0.4837$) and the ANOVA test the high-level vulnerabilities and ($t = -0.1286$; $p = 0.4491$) indicate no significant differences.

Thus, Null hypothesis four (H_{4o}) is not rejected. There is no evidence to reject the assertion that an audit will find an amount significantly close to all of the systems vulnerabilities.

An audit of the systems will discover a lower number of false positives than a "Pen.Test"

Next, the Hypothesis that *a "Pen.Test" will result in the detection of a larger number of false positives (or nonexistent vulnerabilities) than a systems audit* is clearly demonstrated in **Figure 7 - Graph of False Positives** below. This graph demonstrates that and "Pen.Test" will result in a significantly larger number of false positives an audit

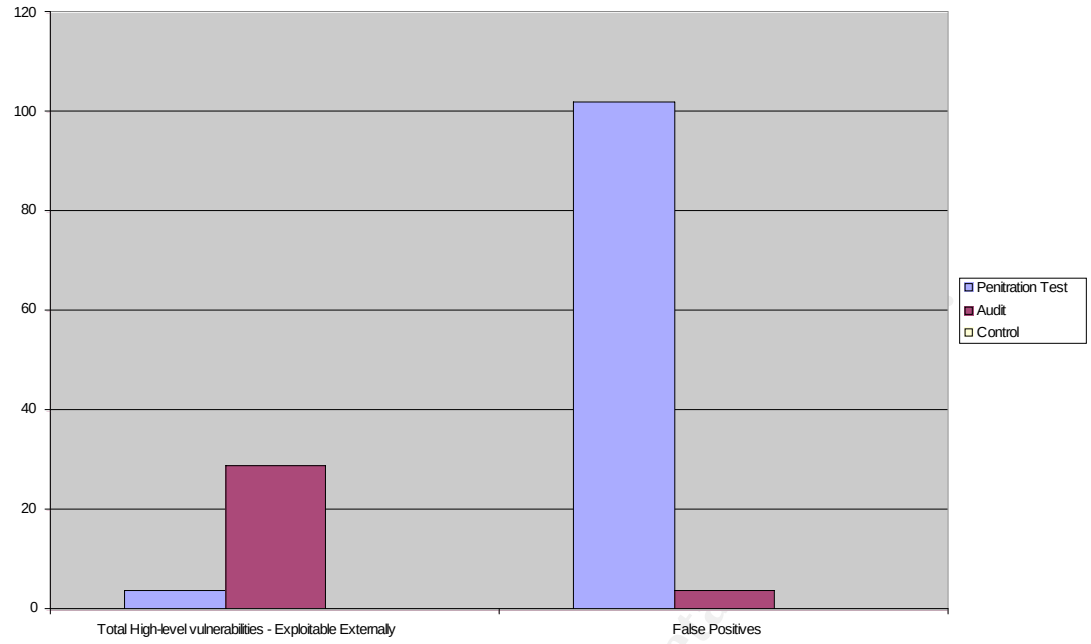


Figure 7 - Graph of False Positives

The results from Appendix are used in the next hypotheses:

Null: $H_{5.1o}$ $\mu_P = \mu_A$ and

Alternative: $H_{5.1a}$ $\mu_P < \mu_A$

Again, two sample t test results (as reported in *Appendix 5.5 - Hypotheses Five - An audit of the systems will discover a lower number of false positives than a "Pen.Test".*) support the hypothesis. Two sample t test of the both the false positives from the High-level results in the section "*Analysis of False Positives - High by Type of Test" using t test* ($t = -2.3097$; $p=0.0128$, $\alpha = 0.025$) and those from the entire results base" *One-way Analysis of False Positives - Total by Type of Test*" ($t = -3.4763$; $p=0.0006$) demonstrate significant difference.

Next, a Comparison with the Control was conducted as is shown is *Appendix 5.5.2 - Comparison with the Control*.

Table 4 - All Pairs Tukey-Kramer at Alpha = 0.01 below indicate that:

$\mu_P \neq \mu_A$ and

$$\mu_P \neq \mu_C$$

but that,

$$\mu_A = \mu_C = 0$$ (by definition the control has 0 false positives).

As $\mu_P > 0$ from the results above and an audit will give a low amount of false positives ($\mu_A = \mu_C = 0$), we thus accept the hypothesis that an audit of the systems will discover a lower number of false positives than a "Pen.Test".

Level		Mean
Pen Test	A	4.0869565
Audit	B	0.1739130
Control	B	0.0000000

Table 4 - All Pairs Tukey-Kramer at Alpha = 0.01

Equally, it may be also shown by the same means that a well-structured audit using a structured methodology will result in a low number of false positives being detected.

This result is significant in that the large number of "Pen.Test" results that have been shown to be false positive results. When compared to the audit methodology the number of false positives detected by the "Pen.Test" that needed to be verified for accuracy resulted in an extension of the times at required to complete the tests.

The Primary result

External tools based "Pen.Testing" is less effective than an interactive systems audit in discovering and reporting known vulnerabilities.

The methodology used in this experiment limited the time available to complete the audit to the time used to complete the "Pen.Test". In this manner, the experimental results obtained from the audit were delivered in an equal or lesser time than that required to complete the "Pen.Test". For this reason if the results of the audit are superior to the results of the "Pen.Test", we can say that the results support the second hypothesis. This is that an audit of equal cost will deliver greater value than a "Pen.Test".

It has already been shown that there is a significant difference between the results obtained from the "Pen.Test" to our control set. In addition, there is no evidence to demonstrate that an audit is not effective when compared to the control. Further, it has been demonstrated that there is no relationship between the "Pen.Test" methodologies, and the control set of all system vulnerabilities in our experimental population.

Next, it was shown that the methodologies used to conduct a "Pen.Test" deliver a significantly larger volume of false positives than does an audit.

Table 5 - All Pairs Tukey-Kramer at Alpha = 0.01 indicates that:

$$\mu_P \neq \mu_A \quad \text{and}$$

$$\mu_P \neq \mu_C$$

but that,

$$\mu_A = \mu_C = 0$$ (by definition the control has 0 false positives).

Level		Mean
Pen Test	B	1.6521739
Control	A	8.1304348
Audit	A	8.0869565

Table 5 - All Pairs Tukey-Kramer at Alpha = 0.01

A comparison of the means from the Analysis of "Total vulnerabilities - Exploitable Externally" as displayed in Appendix 5.2 "*Hypotheses Two - An audit of equal cost will deliver more value to the end user*" using a 2-Sample Test with a Normal Approximation (S = 411.5; Z = 2.91673; p=0.0036) and an two sample t test of the "*Total High-level vulnerabilities - Exploitable Externally*" shows (F = 10.7254, p=0.0021) that there is no significant correlation between the results and "Pen.Test" and the results of an audit.

Thus there is significant evidence to support both hypothesis one and hypothesis two. For this reason, we can say that external tools based "Pen.Testing" is less effective than a systems audit methodology in the detection and reporting of vulnerabilities.

Further a systems audit is not only more effective but more efficient than “Pen.Testing”.

Discussion and implications

The results obtained in this study support the assertion that external tools based “Pen.Testing” or "Tools based Ethical Attacks" are not an effective means of reporting vulnerabilities on an information system. As has been noted, the assessment of the vulnerabilities on the system is a critical component in conducting a risk assessment. The use of ineffectual test methodologies for the determination of vulnerabilities affecting a system will result in unrealistic results on the risk analysis.

The answer to the research question, *“Are question “Tools based Ethical Attacks” or external “Pen.Tests” an effective method in determining the level of system vulnerability when a more effective audit methodology is available?”* may be answered in the negative.

It could be argued that an "ethical attack" requires a level of skill. A lower skilled auditor would result in a lower cost per unit time. There are two arguments against these points:

1 A large number at of training organisations have started in-depth training courses specialising in “Pen.Testing” training and certification. This includes the “Certified Ethical Hacker” designation.

2 Many firms specialising in external “Pen.Tests” charge in equal or greater amount for these services than they do for audit services.

Research into the reasoning for the widespread use of "Tools based Ethical Attacks" instead of vulnerability testing using an audit methodology is warranted. There are number of possible reasons for this, one being greater potential profitability for the firm doing the testing.

Often this argument has been justified by the principle that the auditor has the same resources as the attacker. For simple commercial reasons this is never the case. All audit work is done to a budget, whether internal or externally sourced. When internal audit tests a control, they are assigned costs on internal budgets based on time and the effectiveness of the results.

Externally sourced auditors are charged at an agreed rate for the time expended. Both internal and external testing works to a fixed cost.

An external attacker (often referred to as a “hacker”) on the other hand has no such constraints upon them. They are not faced with budgetary shortfalls or time constraints. It is often the case that the more skilled and pervasive attacker will spend months (or longer) in the planning and research of an attack before embarking on the execution.

Further, audit staff are limited in number compared to the attackers waiting to gain entry through the proverbial back door. It is a simple fact the pervasiveness of the Internet has led to the opening of organisations to a previously unprecedented level of attack and risk. Where vulnerabilities could be open for years in the past without undue risk, systems are unlikely to last a week un-patched today.

The critical issue, however, is that systems are being left in a vulnerable state and the experts, who should know better are supporting the status quo rather than pushing a more effective audit.

The foundation of the argument that an auditor has the same resources must be determined to be false. There are numerous attackers all “seeking the keys to the kingdom” for each defender. There are the commercial aspects of security control testing and there are the realities of commerce to be faced.

It seems easier to give the customer what they perceive they want rather than to sell the benefits of what they need. It is the role of security professionals to ensure that we do what is right and not what is just uncomplicated.

Fred Cohen has noted in several of his works that when engaged in the testing of a system, the common flaws will generally be found quickly during testing. As the engagement goes on, less and less (and generally more obscure and difficult to determine) vulnerabilities will be discovered in a generally logarithmic manner. Most “Tools based Ethical Attacks” fail to achieve comparable results to an attacker for this reason. The “ethical attacker” has a timeframe and budgetary limits on what they can test.

On the contrary, an attacker is often willing to leave a process running long after the budget of the auditor has been exhausted. A resulting vulnerability that may be obscure

and difficult to determine in the timeframe of an "external attack" is just as likely (if not more so) to be the one that compromises the integrity of your system than the one discovered early on in the testing.

Dijkstra in many of his works has promoted the idea that, especially these days, as the size of the systems we work on so hard is so big, we need to know our limits and act accordingly. The same issues apply to auditing code as to auditing networks, "*biting off more that you could chew mentally can lead to some disastrous results*". Test driven development is all about taking very small bites and processing them as you go. This way, even the most complex task begins with one small test.

Dijkstra advocated formal methods, the agile methods most advocates of test-driven development believe in, mirror Tools based Ethical Attacks.

This research supports this stance and shows that the Internet is nothing new in computing but just an extension of past issues as far as audit is concerned.

Irvine states "*Computer security addresses the problem of enforcement of security policies in the presence of malicious users and software. Systems enforcing mandatory policies can create confinement domains that limit the damage incurred by malicious software executing in applications. To achieve assurance that the confinement domains cannot be breached, the underlying enforcement mechanism must be constructed to ensure that it is resistant to penetration by malicious software and is free of malicious artefacts*".

This is one of the key failings of external "hacker testing" is that certain aspects of security policy can be described in completely non-subjective terms. "*For example, the policy may state that unauthorized individuals are not permitted to read classified material. Can testing ensure that policy will not be violated?"* (Irvine, Stemp, & Warren, 1997)

Irvine similarly point out that, "*there are 10^{19} possible eight character passwords. This means that to ensure that a particular combination of eight-character name and eight-character password will not yield total control of the system, 10^{38} tests would be required. Suppose that tests could be run at a rate of 10 million per second. A total of*

10^{15} tests could be performed per year on a fast multiprocessing machine. Even with a billion such machines the testing would take 10^{14} years to complete--10,000 times longer than the current age of the universe."

By this reasoning, an external testing is a gamble at best. No organisation has an unlimited budget and thus all testing is limited. This research has shown that audits are more effective than external tests.

*The assurance of correct protection policy enforcement gained in penetration testing … is directly proportional to the expertise of the team conducting those tests. Members must have expertise using the target system and experience in security testing (*Weissman, *1995*). As such, that tester needs to have detailed knowledge of a system being tested to have any hope in finding its vulnerabilities. External "Pen.Tests" are solely a scare mongering technique.

Ethical Hacking" has been widely marketed as an essential tool in information security but there are obvious conflicts of interest. Security firms have an incentive to exaggerate threats and invent threats.

It unlikely that teams conducting external tools based "Pen.Testing" will be able to keep up with the emergence of new vulnerabilities. The dependence upon traditional vulnerability or "Pen.Testing" techniques is a less effective methodology to supply the vulnerability verification and extrapolation element of a risk assessment. This research supports this assertion.

Irvine's works and the results of this experiment show that simply applying the "*bigger hammer" of more vulnerability signatures for brute-force testing is not a rigorous approach to a meaningful understanding of vulnerabilities within an enterprise.* (Stephenson, 2004)

With external test methodologies a *"tester does not find all the system's vulnerabilities, and does not even confirm the existence of all the vulnerabilities that the test may have detected. All a penetration test proves is that a system can be compromised".* (Winkler, 1999).

It is well known that all systems can be compromised. The real issue is what the risk is. This does not involve paying testers for statements of know fact.

In “What Makes A Good “Pen.Test”?” Weissman (1995.a, p3) states that:

A consistent test philosophy is basic to good penetration testing. A philosophy that focuses efforts on finding flaws and not on finding booty or other hidden targets adds professionalism to the tests by placing a premium on thinking instead of scavenger-hunt searches.

The issue with “Pen.Tests” today is in the issuing of “Tools based Ethical Attacks” is that they do not even cover the requirements of a penetration test or vulnerability assessment.

Future Research

This experiment has demonstrated that the mistaken belief in using external attacks as a vulnerability scanning methodology is flawed. Research into better methods needs to be conducted.

Some areas for research include:

1. The development of the most effective (both in cost and results) computer systems audit techniques,
2. Explore the most effective methods of training auditors,
3. Why people believe external testing alone is effective.

Summary

Just as Edsger W. Dijkstra in his paper “A Discipline of Programming” denigrates the concept of "debugging" as being necessitated by sloppy thinking, so to may we relegate external vulnerability tests to the toolbox of the ineffectual security professional.

In his lecture, "The Humble Programmer", Edsger W Dijkstra is promoting –

"Today a usual technique is to make a program and then to test it. But: program testing can be a very effective way to show the presence of bugs, but it is hopelessly inadequate for showing their absence. The only effective way to raise the confidence level of a program significantly is to give proof for its correctness. But one should not first make the program and then prove its correctness, because then the requirement of providing the proof would only increase the poor programmers burden. On the contrary: the programmer should let correctness proof and program to go hand in hand..."

Just as in programme development where the best way of avoiding bugs is to formally structure development, systems design and audit needs to be structured into the development phase rather that testing for vulnerabilities later.

It is necessary that the computer industry learn from the past. Similar to the assertion (Dijkstra, 1972) that "*the competent programmer is fully aware of the strictly limited size of his own skull; therefore he approaches the programming task in full humility, and among other things he avoids clever tricks like the plague..",* security professionals, including testers and auditors need to be aware of their limitations. Clever tricks and skills in the creation of popular “hacker styled” testing are not effective.

As the market potential has grown, unscrupulous vendors have been quoted overemphasising dangers to expand customer base and in some cases selling products that may actually introduce more vulnerabilities than they guard.

External testing is an immense industry. This needs to change. It is about time we started securing systems and not just reaping money in from them using ineffectual testing methodologies.

Conclusions/ Recommendations

An audit is not designed to distribute the allocation of blame. It is necessary that as many vulnerabilities affecting a system as is possible are diagnosed and reported. The evidence clearly support to the assertion that external tools based “Pen.Testing” is an ineffective method of assessing system vulnerabilities compared against a systems audit.

The key is sufficient planning. When an audit has been developed sufficiently, it becomes both a tool to ensure the smooth operations of an organisation and a method to understand the infrastructure more completely. Done correctly an audit may be a tool to not just point out vulnerabilities from external “hackers”. It may be used within an organisation to simultaneously gain an understanding or the current infrastructure and associated risks and to produce a roadmap towards where an organisation needs to be.

In some instances, it will not be possible or feasible to implement mitigating controls for all (even high-level) vulnerabilities. It is crucial however that all vulnerabilities are known and reported in order that compensating controls may be implemented.

The results of the experiment categorically show the ineffectiveness of vulnerability testing by "Tools based Ethical Attacks". This ineffectiveness makes the implementation of affected controls and countermeasures ineffective.

A complete audit will give more results and more importantly is more accurate than any external testing alone. The excess data needs to be viewed critically at this point, as not all findings will be ranked to the same level of import. This is where external testing can be helpful.

After the completion for the audit and verification of the results, an externally run test may be conducted to help prioritise the vulnerable parts of a system. This is the primary areas where external testing has merit.

“Blind testing” by smashing away randomly does not help this process. The more details an auditor has, the better they may do their role and the lower the risk.

External blind testing alone results in an organisation's systems being susceptible and thus vulnerable to attack. The results of this experiment strongly support not using "Tools based Ethical Attacks" alone as a vulnerability reporting methodology.

The deployment of a secure system should be one of the goals in developing networks and information systems in the same way that meeting system performance objectives or business goals is essential in meeting an organisation's functional goals.

Additional Information and Bibliography

1. Abbott, R.P., et. al, "Security Analysis and Enhancement of Computer Operating Systems," NBSIR 76-1041, National Bureau of Standards, ICST, Gaithersburg, MD., April 1976.

2. Aldrich, John (1997), “R.A. Fisher and the making of maximum likelihood, 1912-1922”, Statist. Sci. 12, no. 3 (1997), 162–176

3. Anderson, R. J. (2001) “Security Engineering – A guide to building dependable distributed systems”. John Wiley & Sons

4. Anderson, Alison, and Michael Shain. "Risk Management." Information Security Handbook. Ed. William Caelli, Dennis Longley and Michael Shain. 1 ed. New York City: Stockton Press, 1991. 75-127.

5. Anderson, James P. (1972). Computer Security Technology Planning Study. Technical Report ESD-TR-73-51, Air Force Electronic Systems Division, Hanscom AFB, Bedford, MA. (Also available as Vol. I, DITCAD-758206. Vol II, DITCAD-772806).

6. AS8015 Australian Standard for Corporate Governance of ICT

7. Azadegan, S. Lavine, M. O'Leary, M. Wijesinha, A. Zimand, M. “An Undergraduate Track in Computer Security”. ACM SIGCSE Bulletin, Proceedings of the annual conference on Innovation and technology in computer science education, Volume 35 Issue 3. June 2003. Available on December 14, 2006 - http://portal.acm.org/

8. Attanasio, C.R., P.W. Markstein and R.J. Phillips, "Penetrating an Operating System: A Study of VM/370 Integrity," IBM Systems Journal, Vol. 15, No. 1, 1976, pp. 102-106.

9. Bauer, D.S. and M.E. Koblentz, "NIDX - A Real-Time Intrusion Detection Expert System," Proceedings of the Summer 1988 USENIX Conference, June 1988

10. Bishop, M., "Security Problems with the UNIX Operating System," Computer Science Department, Purdue University, West Lafayette, Indiana, March 1982.

11. Bogolea, Bradley & Wijekumar, Kay (2004) “Information Security Curriculum Creation: A Case Study” ACM InfoSecCD Conference’04, October 8, 2004, Kennesaw, GA, USA.

12. Border, Charles. Holden, Ed. “Security Education within the IT Curriculum”. Proceeding of the 4th conference on information technology curriculum on Information technology education. Oct 2003. Available on December 14, 2006 - http://portal.acm.org/

13. Bosworth, Seymour & Kabay, M. E. (Ed.) (2002) “Computer security Handbook” Fourth Edition, John Wiley & Sons Inc. USA

14. Boud, D., Dunn, J. and Hegarty-Hazel, E. (1986) “Teaching in Laboratories”. SRHE & NFER-Nelson, Surrey, UK

15. Boyd, C. and Mathuria, A. (2003) “Protocols for Authentication and Key Establishment”. Springer-Verlag, Berlin, Germany

16. Bull, A., C.E. Landwehr, J.P. McDermott and W.S. Choi, "A Taxonomy of Computer Program Security Flaws," Center for Secure Information Technology, Naval Research Laboratory, draft in preparation, 1991.

17. Chan, Sally, and Stan Lepeak. "IT and Sarbanes-Oxley." CMA Management 78.4 (2004): 33(4).

18. Coe, Martin J. "Trust services: a better way to evaluate I.T. controls: fulfilling the requirements of section 404." Journal of Accountancy 199.3 (2005): 69(7).

19. Cohen, Fred, “Protection Testing”, http://www.sdmagazine.com/documents/s=818/sdm9809c/, September 1998

20. Cohen, Fred (1997) ""Pen.Testing"?" http://all.net/journal/netsec/1997-08.html

21. Cohen, Fred (1998-2) "Red Teaming and Other Agressive Auditing Techniques" http://all.net/journal/netsec/1998-03.html

22. Cox, R. and Light, G. (2001) "Learning & Teaching in Higher Education: The reflective professional". Sage Publications, London, UK

23. Crowley, Ed. "Information System Security Curricula Development." Proceeding of the 4th conference on information technology curriculum on Information technology education. Oct 2003. Available on December 11, 2006 at http://portal.acm.org/

24. Curtis L. Smith, John A. Schroeder, Scott T. Beck, and James K. Knudsen (2001) "MODELING POWER NON-RECOVERY USING THE SAPHIRE RISK ASSESSMENT SOFTWARE", Bechtel BWXT Idaho, LLC. Viewed 20th December 2006 (http://nuclear.inl.gov/docs/papers-presentations/psam_6_nonrecovery.pdf)

25. Dark, Melissa J. (2004.a) "Assessing Student Performance Outcomes in an Information Security Risk Assessment, Service Learning Course" Purdue University

26. Dark, Melissa. J. (2004.b). "Civic Responsibility and Information Security: An Information Security Management, Service Learning Course". Proceedings of the Information Security Curriculum Development Conference, 2004.

27. Delphi Group (2005) "Time-Based Analysis: Process De-engineering (TBA)" White Paper.

28. Dias, G.V., et. al., "DIDS (Distributed Intrusion Detection System) - Motivation, Architecture, and An Early Prototype," Proceedings of the 14th National Computer Conference, Washington, D.C., October 1991, pp. 167-176.

29. Edsger W Dijkstra, (1972) "EWD 340: The humble programmer" published in Commun. ACM 15 (1972), 10: 859–866.

30. Dijstra, Edsger W. (1976). A Discipline of Programming. Englewood Cliffs, NJ: Prentice Hall

31. Dodson, Bryan & Nolan, Dennis (2005 Ed) “The Reliability Engineering Handbook” Quality Publishing.

32. Farmer, D. and E.H. Spafford, "The COPS Security Checker System," CSD-TR-993, Department of Computer Sciences, Purdue University, West Lafayette, Indiana, 1990. (Software available by anonymous ftp from cert@sei.cmu.edu)

33. Farmer, Dan, and Weitse Venema. "Improving the Security of Your Site by Breaking into It." 1992

34. Ford, W. and Baum, M. S. (1997) “Secure Electronic Commerce”. Prentice Hall

35. Garfinkel, S. and Spafford, G. (2001) “Web Security, Privacy & Commerce”. 2nd Edition. Cambridge, Mass: O'Reilly

36. Garfinkel, S., G. Spafford, Practical Unix Security, O'Reilly & Associates, Inc., Sebastopol, CA., 1991.

37. Gasser, M., Building a Secure Computer System, Van Nostrand Reinhold, New York, N.Y., 1988. [GUPTA91] Gupta, S. and V.D. Gligor, "Towards a Theory of Penetration- Resistant Systems and its Application," Proceedings of the 4th IEEE Workshop on Computer Security Foundations, Franconia, N.H., June 1991, pp. 62-78.

38. Ghosh, A. K. (1998) “E-Commerce Security”. Wiley

39. Goodwin, Bill. "IT should lead on Sarbanes-Oxley." Computer Weekly 27 April 2004: p5.

40. Gomolski, Barbara. "The top five issues for CIOs." Computerworld January 2004: 42(1).

41. Gupta, S. and V.D. Gligor, "Experience With A Penetration Analysis Method And Tool," Proceedings of the 15th National Computer Security Conference, Baltimore, MD.,October 1992, pp. 165-183.

42. Hagerty, John. "Sarbanes-Oxley Is Now a Fact of Business Life-Survey indicates SOX IT-compliance spending to rise through 2005." VARbusiness Nov. 15 2004: 88.

43. Hamlet, Richard (1989). Testing for Trustworthiness. In J. Jackey and D. Schuler (Eds.), Directions and Implications of Advanced Computing, pp. 97-104. Norwood, NJ: Ablex Publishing Co. Myers, Phillip (1980). Subversion: The Neglected Aspect of Computer Security. MS thesis, Naval Postgraduate School, Monterey, CA.

44. Harris, Shon; Harper, Allen; Eagle, Chris; Ness, Jonathan; Leste, Michael, 2004, “Gray Hat Hacking: The Ethical Hacker's Handbook”, McGraw-Hill Osborne Media; 1st edition (November 9, 2004).

45. Hollingworth, D., S. Glaseman and M. Hopwood, "Security Test and Evaluation Tools: an Approach to Operating System Security Analysis," P-5298, The Rand Corporation, Santa Monica, CA., September 1974.

46. Humphrey, Christopher; Jones, Julian; Khalifa, Rihab; Robson, Keith; “Business Risk Auditing And The Auditing Profession: Status, Identity And Fragmentation”, Manchester School of Accounting and Finance), CMS 3 Conference Paper 2003.

47. Kalakota, R. and Whinston, A. B. (1996) “Frontiers of Electronic Commerce”. Addison-Wesley

48. Keong, Tan Hiap (2004) “Risk Analysis Methodologies” http://pachome1.pacific.net.sg/~thk/risk.html (Last Viewed 27th December 2005)

49. Infosec Graduate Program. Purdue University. Available on December 12, 2006 at http://www.cerias.purdue.edu/education/graduate_program/

50. IT Governance Institute 2003, "Board Briefing on IT Governance, 2nd Edition". Retrieved January 18, 2006 from Http://www.isaca.org/Content/ContentGroups/ITGI3/Resources1/Board_Briefing_on_IT_Governance/26904_Board_Briefing_final.pdf

51. Intal, Tiina and Do, Linh Thuy; “Financial Statement Fraud - Recognition of Revenue and the Auditor’s Responsibility for Detecting Financial Statement Fraud”, Göteborg University, Masters Thesis, 2002, ISSN 1403-851X

52. Irvine, Cynthia E., Stemp, Roger & Warren, Daniel F. (1997) “Teaching Introductory Computer Security at a Department of Defense University” Department of Computer Science, Naval Postgraduate School

53. Jacobson, Robert V. "Risk Assessment and Risk Management." Computer Security Handbook. Ed. Seymour Bosworth and M. E. Kabay. 4 ed. New York: Wiley, 2002. 47.1-47.16.

54. Jacques, D., Gibbs, G. and Rust, C. (1991) “Designing and Evaluating Courses”. Oxford Brookes University, Oxford, UK

55. Johnston, Michelle. "Executing an IT Audit for Sarbanes-Oxley Compliance." http://www.informit.com/articles/article.asp?p=337041&rl=1. 17 Sept. 2004 <www.informit.com>.

56. Jones, Andrew. "Identification of a Method for the Calculation of Threat in an Information Environment.

57. Kramer, John B. “The CISA Prep Guide: Mastering the Certified Information Systems Auditor Exam”, John Wiley and Sons, USA 2003. ISBN:0471250325

58. Lawrence, E., Corbitt, B., Fisher, J., Lawrence, J. and Tidwell, A. (1999) “Internet Commerce” 2nd Edition, Wiley

59. Linde, R.R., "Operating System Penetration," Proceedings of the National Computer Conference, Vol. 44, AFIPS Press, Montvale, N.J., 1975

60. Lurie, Barry N. "Information technology and Sarbanes-Oxley compliance: what the CFO must understand." Bank Accounting and Finance 17.6 (2004): 9 (5).

61. McCollum, Tim. "IIA Seminar Explores Sarbanes-Oxley IT Impact." IT Audit 6 (2003).

62. Master of Science degree program in Information Security and Assurance. George Mason University. Available on December 12, 2006 at http://www.isse.gmu.edu/ms-isa/

63. Master of Science in Security Informatics. Johns Hopkins University. Available on November 12, 2006 at http://www.jhuisi.jhu.edu/education/index.html

64. Mauw, Sjouke & Oostdijk, Martijn (2004) “Foundations of Attack Trees” Eindhoven University of Technology, Emerald

65. McConnell Jr., Donald K, and George Y. Banks. "How Sarbanes-Oxley Will Change the Audit Process." http://www.aicpa.org/pubs/jofa/sep2003/mcconn.htm (2003).

66. McGill, Tanya, Ed. (2002) “Current issues in IT education”, Murdock University, IRM Press, Melbourne, Australia.

67. Microsoft (2004) “The Security Risk Management Guide” v1.1, Microsoft Corporation, USA

68. MIL-STD-1629 “Procedures for Performing a Failure Mode, Effects and Criticality Analysis”

69. Moore, Andrew P., Ellison, Robert J. & Linger Richard C. (2001) “Attack Modeling for Information Security and Survivability”, Carnegie Mellon University. The Software Engineering Institute US

70. Munter, Paul. "Evaluating Internal Controls and Auditor Independence under Sarbanes-Oxley." Financial Executive 19.7 (2003): 26 (2).

71. Myagmar, Suvda, Lee Adam J. & Yurcik, William (2005) “Threat Modeling as a Basis for Security Requirements”, National Center for Supercomputing Applications (NCSA)

72. NIST (800-42) “Guideline on Network Security Testing” NIST Special Publication 800-42

73. NIST (800-12) “An Introduction to Computer Security: The NIST Handbook” (Special Publication 800-12)

74. NIST (800-41) “Guidelines on Firewalls and Firewall Policy” (Special Publication 800-41)

75. NIST (800-27) “Computer Security” (Special Publication 800-27)

76. NIST (800-30) “Risk Management Guide for Information Technology Systems” (Special Publication 800-30), 2002

77. Parker, D.B., "Computer Abuse Perpetrators and Vulnerabilities of Computer Systems," Stanford Research Institute, Menlo Park, Ca., December 1975.

78. Phillips, R. , "VM/370 Penetration Study Final Report," TM(L)-5196/006/00, System Development Corporation, Santa Monica, CA., October 1973.

79. Piazza, Peter. "IT security requirements of Sarbanes-Oxley." Security Management June 2004: 40(1).

80. “Perspectives on Internal Control Reporting: A Resource for Financial Market Participants." Deloitte & Touche LLP, Ernst & Young LLP, KPMG LLP, PricewaterhouseCoopers LLP. December 2004.

81. Rodrigues, Alexandre G. (2001) “Managing and Modelling Project Risk Dynamics A System Dynamics-based Framework”, Presented at the Fourth European Project Management Conference, PMI Europe 2001, London

82. Rub, J.W., "Penetration Handbook," The Aerospace Corporation, El Segundo, CA., January 1986.

83. Ryan, P. and Schneider, S. (2001) “Modelling and Analysis of Security Protocols”. Addison-Wesley London, UK

84. Salamasick, Mark (2006) “Information Technology Risk Management”, UNIVERSITY OF TEXAS AT DALLAS Course Syllabus – AIM6336 Spring 2006

85. Sanchez, Luis, et al. "Requirements for the Multidimensional Management and Enforcement (MSME) System."

86. SANS (2005) “GIAC ISO 17799 Training Notes”, SANS GIAC 2005, Sydney AU

87. Security Concepts for Distributed Component Systems. 1. Ed. Walt Smith. 16 June 1998. NIST. 14 Nov 2003 <http://csrc.nist.gov/nissc/1998/proceedings/tutorB2.pdf>. (page 53)

88. Security Tracker Statistics. 2002. SecurityGlobal.net LLC. 23 October 2003 <http://securitytracker.com/learn/securitytracker-stats-2002.pdf>.

89. Shaikh, Siraj A. (2004) “Information Security Education in the UK: a proposed course in Secure E-Commerce Systems” ACM InfoSecCD Conference’04, October 8, 2004, Kennesaw, GA, USA.

90. Sherif, M. H. (2000) “Protocols for Secure Electronic Commerce”. CRC Press

91. Siebenlist, Frank. CORBA-SECURITY-Glossary. 14 Nov 2003 <http://wwwunix. mcs.anl.gov/~franks/CORBA-SECURITY-Glossary.htm>.

92. Stallings, William. (2002) “Cryptography and Network Security”, Third Edition, Prentice Hall,

93. Stallings, W. (1995) “Network and Internetwork Security: principles and practice.” Englewood Cliffs, N.J: Prentice Hall New York: IEEE

94. Stanger, James; Lane, Patrick T. & Crothers, Tim (2002) “CIW: Security Professional Study Guide”, Sybex USA

95. Stein, L. D. (1998) “Web Security”, Addison-Wesley

96. Stephenson, Peter. (2005) “Modeling of Post-Incident Root Cause Analysis”. International Journal of Digital Evidence Volume 2, Issue 2, http://www.ijde.org

97. Stephenson, Peter (2004) “Forensic Analysis of Risks in Enterprise Systems”, Center for Regional and National Security, Eastern Michigan University

98. Sterne, Daniel F. (1991). On the Buzzword "Security Policy". In Proceedings 1991 IEEE Symposium on Research in Security and Privacy, Oakland, pp. 219-230. Los Alamitos, CA: IEEE Computer Society Press.

99. Thompson, Kenneth (1984). Reflections on Trusting Trust. Communications of the A.C.M. 27(8), 761-763.

100. Vaughn Jr. Rayford B., Dampier, David A. & Warkentin, Merrill B (2004) “Building an Information Security Education Program” ACM InfoSecCD Conference’04, October 8, 2004, Kennesaw, US

101. Viega and McGraw. (2002) "Risk Analysis: Attack Trees and Other Tricks", Software Development, Vol. 10(8), pp. 30-36.

102. Weill, P. & Ross, J. W., 2004, IT Governance: How Top Performers Manage IT Decision Rights for Superior Results', Harvard Business School Press, Boston.

103. Weissman, Clark (1995). ““Pen.Testing””. In M. Abrams, S. Jajodia, and H. Podell (Eds.),Information Security: An Integrated Collection of Essays, pp. 269-296. Los Alamitos, CA: IEEE Computer Society Press.

104. Weissman, Clark (1995.a) “Handbook for the Computer Security Certification of Trusted Systems”, US Navy Press

105. Winkler, Ira, (1999) “AUDITS, ASSESSMENTS & TESTS (OH, MY)”, Corporate Espionage (Prima, 2nd ed.).

106. Winfield Treese, G. and Stewart, L. C. (2002) “Designing Systems for Internet Commerce”. 2nd Edition, Addison-Wesley

107. van Wyk, Kenneth, “Finding the Elusive Value in “Pen.Testing””, http://www.cioupdate.com/trends/article.php/3393591, August 11th 2004

108. Yang, Andrew. “Computer Security and Impact on Computer Science Education”. The Journal of Computing Small Colleges , Proceedings of the sixth annual CCSC north-eastern conference on The journal of computing in small colleges, Volume 16 Issue 4. April 2001. Available on December 21, 2006 - http://portal.acm.org/

109. Zwicky, E. D., Cooper, S., Chapman, D. B. and Russell, D. (2000) “Building Internet Firewalls”. 2nd Edition, O'Reilly, UK

Web Sites References

1. S.C.O.R.E. – a standard for information security testing - http://www.sans.org/score/

2. The Auditor security collection is a Live-System based on KNOPPIX http://remote-exploit.org/

3. Nessus is an Open Source Security Testing toolset http://www.nessus.org

Definitions

The following table defines abbreviations used in this document:

GIAC	Global Information Assurance Certification
MAC	Modified, Accessed, Created times
SEF	Security Enforcing Functions
SM	Security Mechanisms
SOE	Standard Operating Environment
SANS	SysAdmin, Audit, Network, Security
USB	Universal Serial Bus

Appendix

The Methodology – Tools Based External Attacks

Phase 1 – Gain an Understanding of your System

In the fist phase of the examination, you should:

- Examine your Concept of Operations documentation to gain an understanding of what your system is intended to do, and how it is intended to do it.
- Analyse the network topology and systems configuration documentation, to identify all network devices including servers, workstations, routers and security enforcing devices.
- Examine you Access Requirements (Access Policy) to gain an understanding of what you intend to permit and what you wish to have denied by your system security. This is a very important aspect of the assessment

WHAT A CRACKER DOES

To be able to attack a system systematically, a hacker has to know as much as possible about the target. Reconnaissance is the first stage. A Hacker will want to get an overview of the network and host systems. Consulting the whois, ripe and arin databases is a good method of gaining information without leaving a trail. Information such as DNS servers used by the domain, administrator contacts and IP ranges routed to the Internet can be obtained. Searching the Usenet for old postings of an administrator may reveal problems, products and occasionally configuration details.

An initial scan of the hosts may show up some interesting services where some in depth researching may lead to interesting attack possibilities. Another issue is looking up possible numbers for the company and trying to connect to a modem. Scanning telephone networks for answering devices and collecting these numbers for a later access attempt may lead to a first entry into the network. Such scans of telephone networks are usually referred to as "war dialling" and were heavily before the Internet existed.

The reconnaissance phase may even consider going through trash bins which is known as "dumpster diving" or visiting loading docks of the target to collect additional intelligence. During the reconnaissance phase, different kind of tools can be used such as network mapping tools, and vulnerability scanning tools. It is a great help during the attack phase to have an overview about the network.[1]

Network mapping tools are especially important when doing an internal network assessment as more information is provided than an external scan. For getting a fast report on possible vulnerabilities and security weaknesses, a freeware or commercial vulnerability scanner is useful. These tools scan specified hosts or IP ranges for services and known vulnerabilities. These have to be checked as a large number of false positives are often reported.

Phase 2 –Vulnerability Assessment

The vulnerability assessment is conducted to speculate on induced vulnerabilities, which may have been generated by the network's use (or lack) of a certain product, component, or any topology design errors.

- Some design and configuration problems we may find within your system are:
- Network topology design not as effective as current industry best practices
- Network management not as effective as current industry best practices
- Configurations not as effective as current industry best practices
- Well-known weaknesses in applications software
- A certain software package or configuration, which has known, exploitable weaknesses, is in use throughout the network;
- Well-known weaknesses in operating systems

[1] For the purpose of this test all information will be considered available and thus make the ""Pen.Test"" phase easier in comparison.

- A certain type or family of devices, which has known, exploitable weaknesses, is in use throughout the network;
- Operating Systems configurations not as effective as with current industry best practices

While Phase 2 focuses on identifying weaknesses in the configuration of the networks and systems, an examination of management and administrative approaches is also undertaken.

For example, the vulnerability examination may point out the following types of weaknesses:

- Sensitive data being transmitted across the network in the clear;
- Passwords are not changed on a regular basis;
- Audit trail information is not being collected, or if it is collected, is not being reviewed to identify possible irregularities in system access or usage;
- There are no Security Practices and Procedures document which specifically states the user and administrator security features and responsibilities;
- All weaknesses discovered need be prioritized in readiness for the next Phase.

Phase 3 – Penetration Planning

The penetration-planning phase is where we prepare to conduct the exploits required to compromise the potential vulnerabilities. We identify what vulnerabilities we are going to attempt to exploit and put together a suite of tools in preparation for the next phase, the Penetration Attack.

The tools, which you will use, will consist of:

- Commercially available security tools,
- Publicly available hacker tools

Once you have allocated all of the required tools functionality to the penetration plan, you can proceed to Phase 4.

Phase 4 - Penetration Attack

The penetration attack attempts to confirm or discount the presence of actual vulnerabilities from the list of potential vulnerabilities discovered in Phase 2.

In-depth testing will be conducted on the customer's network components, using industry best practice tools and techniques, to identify:

Confirm the security enforcing functions support any Access Requirements by identifying what is accessible from:

- Externally, normal public user;
- Internal Restricted Management Segment (if access can be obtained externally);
- Internal Network (if access can be obtained externally)

Using specialist tools attempt to locate an exploit:

- well-known weaknesses in applications software,
- well-known weaknesses in operating systems,
- well-known weaknesses in security enforcing devices,

Additionally, testing will measure the ability of:

- audit capabilities
- system administration practices and procedures
- intrusion detection capabilities
- reaction to intrusions when discovered by audit or intrusion detection mechanisms:
    - incident response plan,
    - contingency plans,

Each confirmed vulnerability should be analysed to:

- Determine the likelihood of someone exploiting the vulnerability, and
- The potential gain by the adversary or loss to your organisation.

Appendix – Threat Risk Assessment Methodology

{As used for the Vulnerability Assessment / Audit in this research}

In simple terms, a risk is realised when a threat takes advantage of a vulnerability to cause harm to your system. Security policy provides the basis for implementing security controls to reduce vulnerabilities thereby reducing risk. In order to develop cost effective security policy for protecting Internet connections some level of risk assessment must be performed to determine the required rigour of the policy, which will drive the cost of the security controls deployed to meet the requirements of the security policy. How rigorous this effort must be is a factor of:

- The level of threat an organization faces and the visibility of the organization to the outside world
- The sensitivity of the organization to the consequences of potential security incidents
- Legal and regulatory issues that may dictate formal levels of risk analysis and may mandate security controls for specific systems, applications or data.

Note that this does not address the value of information or the cost of security incidents. In the past, such cost estimation has been required as a part of formal risk analyses in an attempt to support measurements of the Return on Investment (ROI) of security expenditures. As dependence on public networks by businesses and government agencies has become more widespread, the intangible costs of security incidents equal or outweigh the measurable costs. Information security management time can be more effectively spent assuring the deployment of “good enough security” rather than attempting to calculate the cost of anything less than perfect security.

For organisations that are subject to regulatory oversight, or that handle life-critical information, more formal methods of risk assessment may be appropriate. The following sections provide a methodology for rapidly developing a risk profile.

It can be prohibitively expensive and probably impossible to safeguard information against all threats. Therefore, modern Information Security practice is based on assessing

threats and vulnerabilities and selecting appropriate, cost-effective safeguards. A realistic approach is to manage the risk that these threats pose to information and assets.

It is recognized industry best practice for all organizations to identify their information assets and apply the appropriate security measures based on a Threat and Risk Assessment.

To help organizations meet this requirement, many organizations use an industry standard methodology (similar to the one below) which has been developed to assess the value of the information that the organization is processing and allows greater flexibility for providing recommended safeguards.

The following diagram illustrates the four-phased approach to performing a Threat and Risk Assessment.

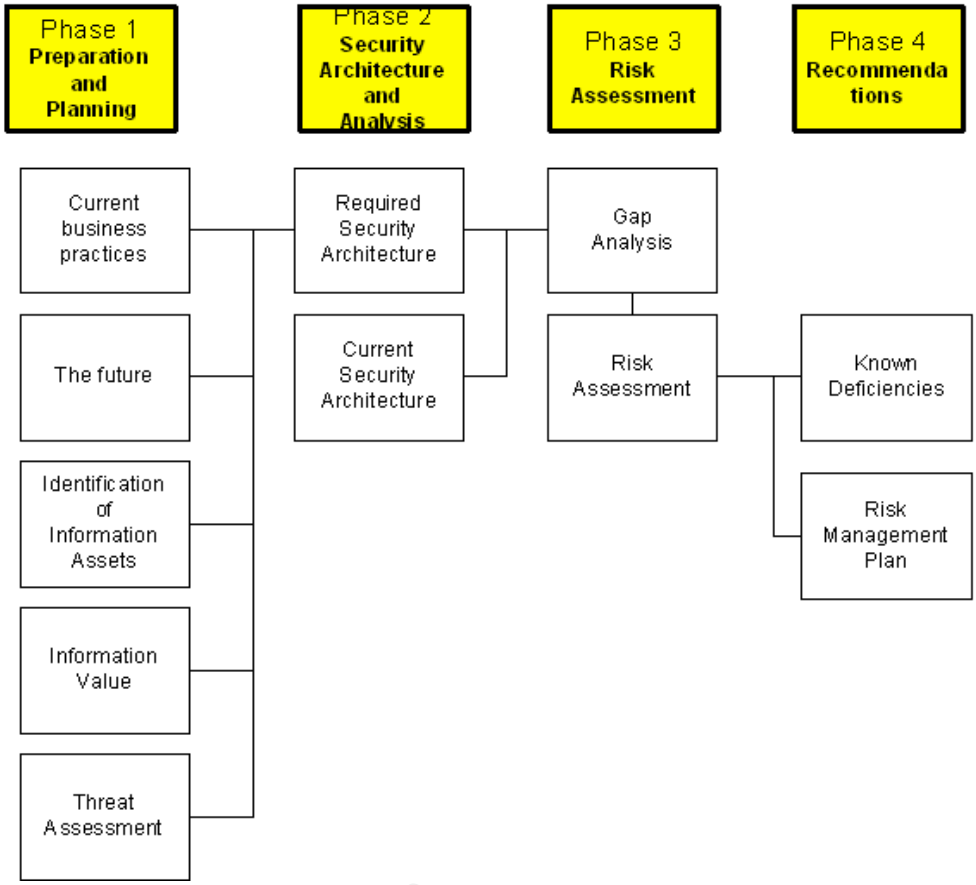


Figure 8 - Risk Assessment Methodology

Phase 1 - Preparation and Identification

CURRENT BUSINESS PRACTICES

The first step in performing a Threat and Risk Assessment is to define the business practices that are required by the organization to accomplish corporate goals. The Current Business Practices of the organization are documented by analysing the organization's mission statement, corporate plan, type of clients and the services that it provides.

THE FUTURE

It is critical that the organization's future business practices and corporate goals are considered throughout the Threat and Risk Assessment process. The plans of the organization must be documented at the start to avoid any possible oversight, preventing the assessment being dated within a short period.

IDENTIFICATION OF INFORMATION ASSETS

The organization's information assets are identified to determine what has to be protected. This requires producing an inventory that lists all information systems and their assets. Each list typically includes the following information:

- the system owner,
- the system's location,
- the nature of business,
- the type of information processed,
- the purpose or application of the system,
- the system configuration,
- the user community, and
- any known inherent strengths or weaknesses of the system.

INFORMATION VALUE

After an inventory of the information assets has been produced, a Statement of Sensitivity is documented for each asset. This documents the asset's importance and value to the organization and should reflect its criticality. The statement is produced by analysing the system and the data it processes with regard to integrity, confidentially and availability requirements.

THREAT ASSESSMENT

The next step is to identify all threats and threat sources to the organization's information assets and assign a classification that reflects the probability of it occurring. The five levels of threat classification are defined as follows:

- Low: There is no history and the threat is unlikely to occur.
- Low Plus: There is no history and the threat could occur.
- Medium: There is some history and the threat could occur.
- Medium Plus: There is some history and the threat is likely to occur.

- High: There is significant past history and the threat is likely to occur.

Phase 2 - Security Architecture Analysis

Required Security Architecture

The information gathered in phase I is used to document the business requirements for security within the organization. The key security strategies are identified that will enable the organization to effectively protect its information assets.

Each pre-determined threat to the information assets is matched with an effective safeguard or safeguards. A safeguard is described as a number of Security Enforcing Functions (SEFs) and associated mechanisms that perform that function are the Security Mechanisms (SM). The process of identifying the required SEFs and the associated mechanisms gives the Organization a security architecture baseline to work towards implementing.

Identification of Current Security Architecture

The organization's current security architecture is documented to identify existing Security Enforcing Functions (SEF) and Security Mechanisms (SM). These safeguards and any existing policy or doctrine is identified to produce the current security baseline. This enables identification of differences between the current and required security baselines.

Phase 3 - Risk Assessment

GAP ANALYSIS

A gap analysis is performed to highlight any differences between the organization's current security architecture and the required security architecture, determined in phase II of the assessment. The output from this analysis will give the reviewer an indication of the residual risk.

RISK ASSESSMENT

After the gap analysis has been performed, the determined residual risk has to be assessed. This assessment produces a level of risk that is measured by the probability of compromise to the confidentiality, integrity or availability of the designated information system and the data processed on it. Determining the level of risk is completed by comparing the relationship between the threats associated to the residual risks and known vulnerabilities or weaknesses.

Phase 4 - Recommendations

KNOWN DEFICIENCIES

Where the assessment of the systems safeguards indicates that they are not able to counter known threats effectively, additional safeguards will be recommended to reduce the risk to an acceptable level. The reviewer will also recommend the type of safeguard required its priority and suggested schedule of implementation.

RISK MANAGEMENT PLAN

The Threat and Risk Assessment process provides the system manager with an appreciation of the status of the safeguards protecting information assets within his/her organization. An assessment of the adequacy of existing safeguards is performed to provide recommendations to assist the system manager in making an informed decision as to which risks the organization should manage or accept.

The level of acceptable risk is a managerial decision that should be based on the information and recommendations provided in the Threat and Risk Assessment.

Assessment and Conclusion

This methodology has been successful in providing assessments for organizations by producing relevant results. This is achieved by considering the business value of information and the business practices of the organization.

The four-phased approach provides a logical progression, which enables the client to trace through the results from each phase to see how the recommendations were obtained.

Appendix – Data Analysis

This Appendix is a compilation of the results of the data analysis process listed against each of the hypotheses being tested.

Hypotheses One - external tools based "Pen.Testing" is less effective than an IT audit

---

External Test Results

Analysis of Total High-level vulnerabilities - Exploitable Externally By Type of Test

t Test
Audit-Pen Test
Assuming equal variances

Difference	1.08696	t Ratio	3.27497
Std Err Dif	0.33190	DF	44
Upper CL Dif	1.98052		
Lower CL Dif	0.19339	Prob > t	0.0010
Confidence	0.99		

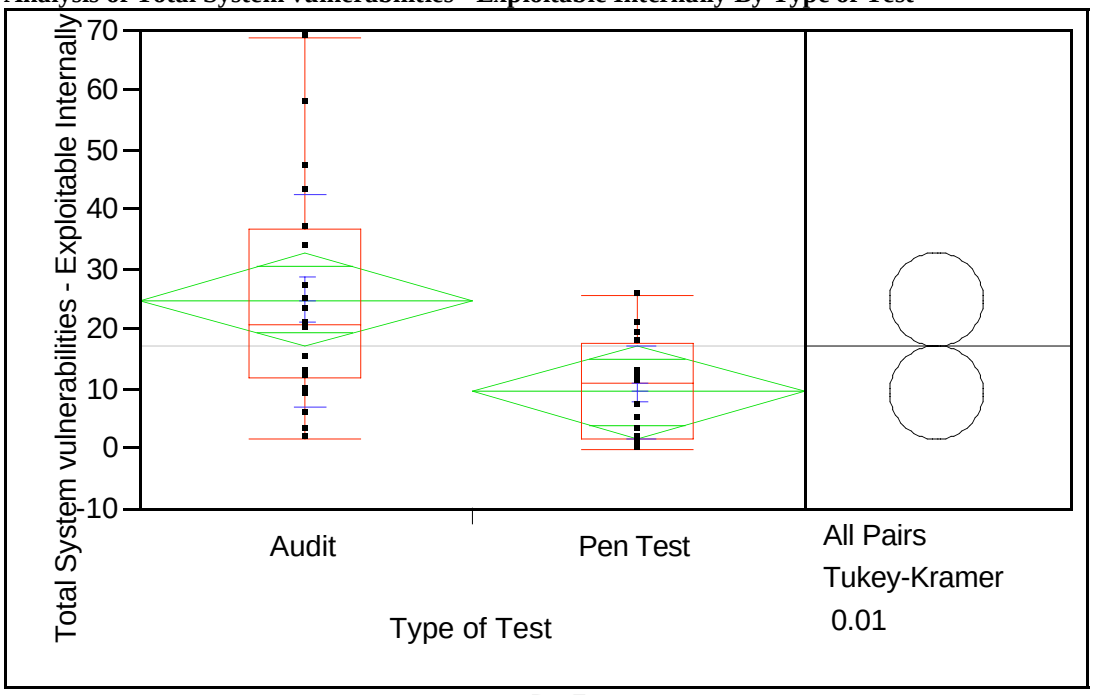
Analysis of Total vulnerabilities - Exploitable Externally By Type of Test

t Test
Audit-Pen Test
Assuming equal variances

Difference	6.4348	t Ratio	3.290514
Std Err Dif	1.9556	DF	44
Upper CL Dif	11.6997		
Lower CL Dif	1.1699	Prob > t	0.0010
Confidence	0.99		

Internal Test Data

Analysis of Total System vulnerabilities - Exploitable Internally By Type of Test

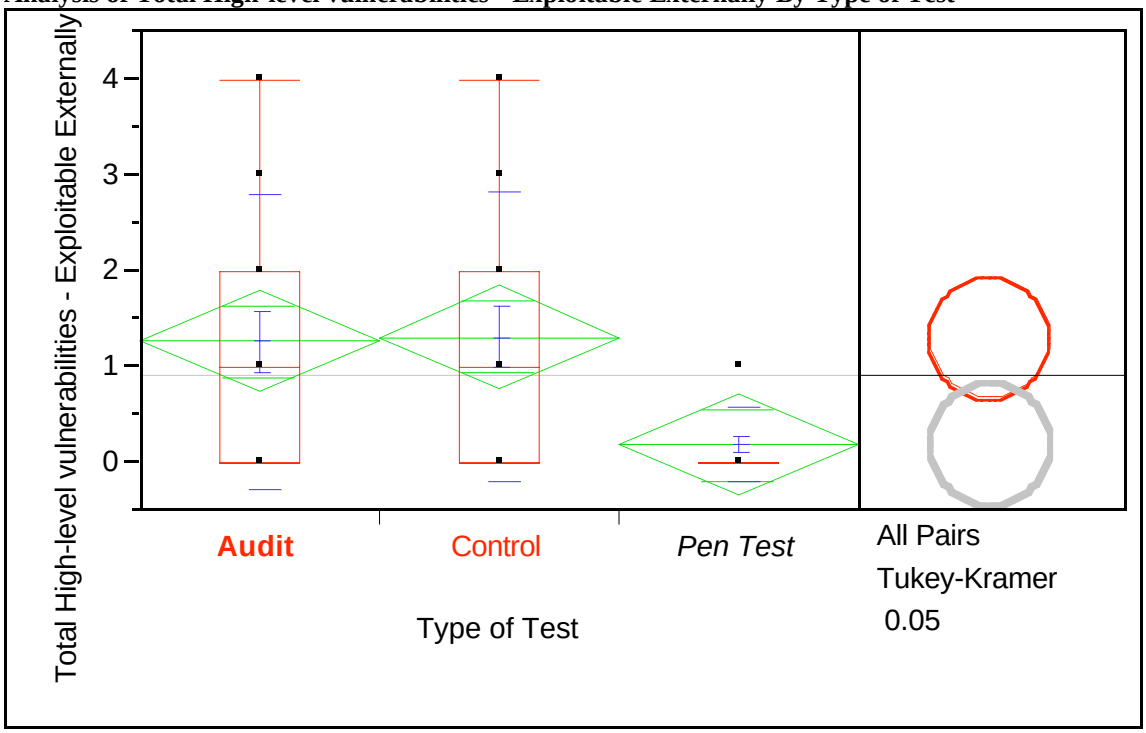


t Test
Audit-Pen Test
Assuming equal variances

Difference	15.2609	t Ratio	3.742021
Std Err Dif	4.0782	DF	44
Upper CL Dif	26.2406		
Lower CL Dif	4.2811	Prob > t	0.0003
Confidence	0.99		

Hypotheses Two - An audit of equal cost will deliver more value to the end user

Analysis of Total High-level vulnerabilities - Exploitable Externally By Type of Test



One-way ANOVA
Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Type of Test	2	18.86957	9.43478	5.8409	0.0046
Error	66	106.60870	1.61528		
C. Total	68	125.47826			

Means Comparisons
Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.39771	0.05

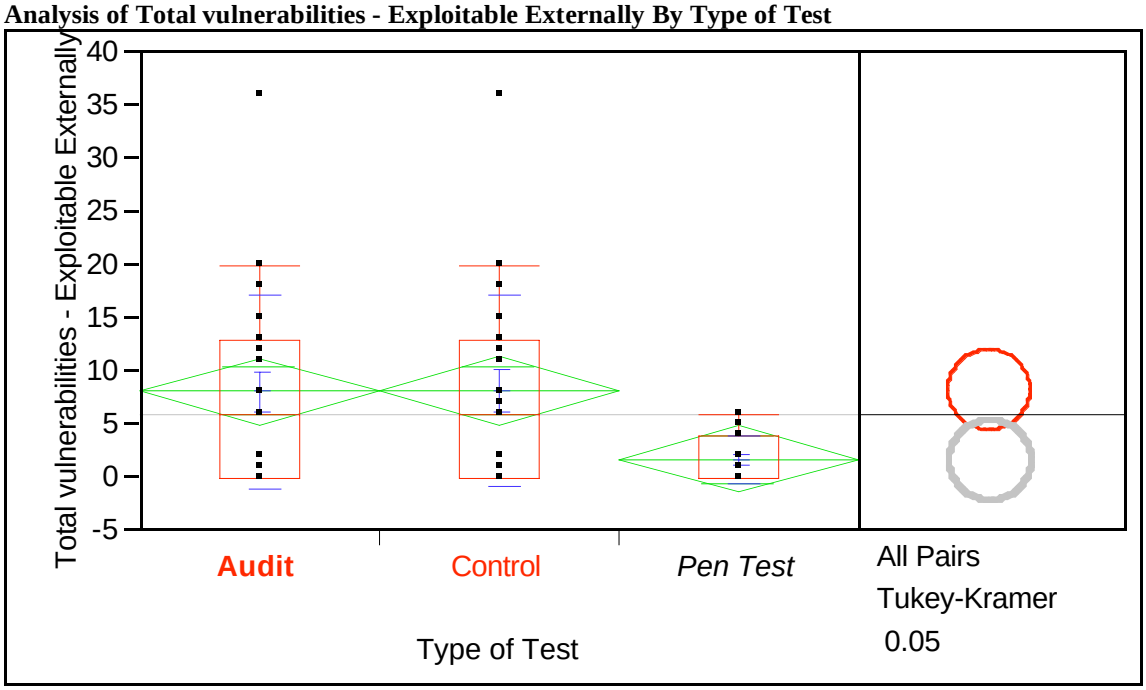
Abs(Dif)-LSD	Control	Audit	Pen Test
Control	-0.8986	-0.8551	0.2318
Audit	-0.8551	-0.8986	0.1883
Pen Test	0.2318	0.1883	-0.8986

Level		Mean
Control	A	1.3043478
Audit	A	1.2608696
Pen Test	B	0.1739130

Levels not connected by same letter are significantly different

Level	- Level	Difference	Lower CL	Upper CL	Difference
Control	Pen Test	1.130435	0.231823	2.029047	
Audit	Pen Test	1.086957	0.188344	1.985569	
Control	Audit	0.043478	-0.855134	0.942090	

Analysis of Total vulnerabilities - Exploitable Externally By Type of Test



One-way ANOVA

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Type of Test	2	639.2174	319.609	5.6166	0.0056
Error	66	3755.6522	56.904		
C. Total	68	4394.8696			

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
2.39771	0.05

Abs(Dif)-LSD	Control	Audit	Pen Test
Control	-5.3336	-5.2901	1.1447
Audit	-5.2901	-5.3336	1.1012
Pen Test	1.1447	1.1012	-5.3336

Level		Mean
Control	A	8.1304348
Audit	A	8.0869565
Pen Test	B	1.6521739

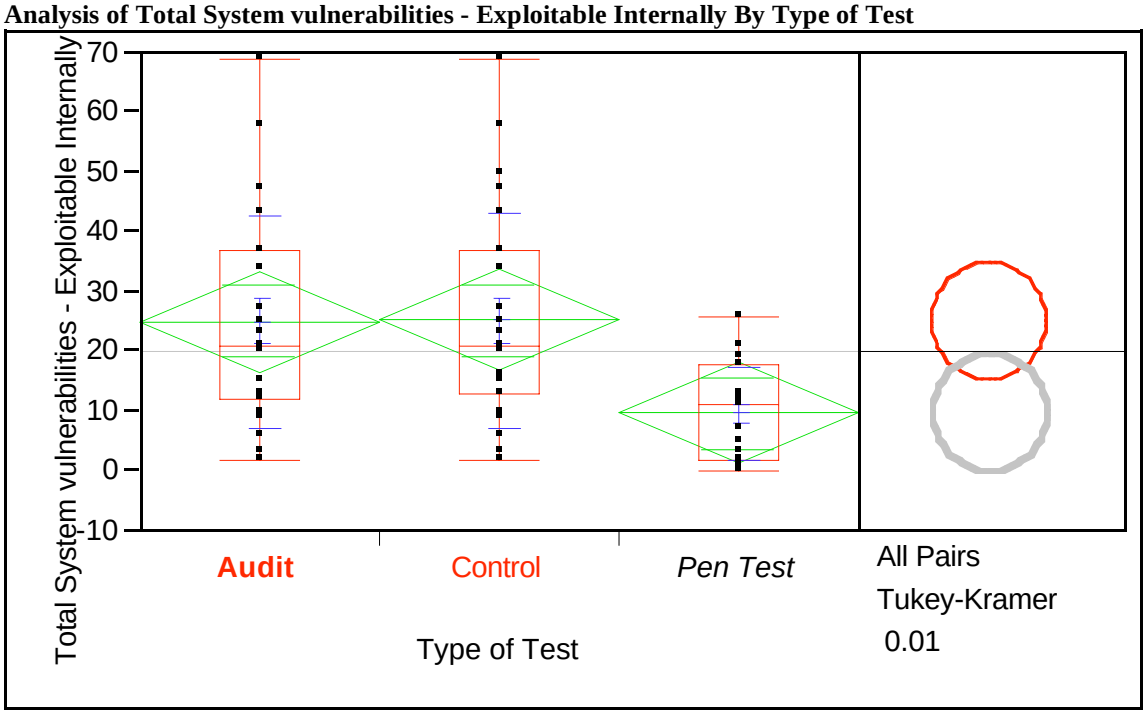
Levels not connected by same letter are significantly different

Level	- Level	Difference	Lower CL	Upper CL	Difference
Control	Pen Test	6.478261	1.14468	11.81184	
Audit	Pen Test	6.434783	1.10120	11.76836	
Control	Audit	0.043478	-5.29010	5.37706	

Wilcoxon / Kruskal-Wallis Tests (Rank Sums)

Level	Count	Score Sum	Score Mean	(Mean-Mean0)/Std0
Audit	23	932.5	40.5435	1.655
Control	23	936.5	40.7174	1.707
Pen Test	23	546	23.7391	-3.368

Analysis of Total System vulnerabilities - Exploitable Internally By Type of Test



One-way ANOVA

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Type of Test	2	3622.638	1811.32	7.6755	0.0010
Error	66	15575.130	235.99		
C. Total	68	19197.768			

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
3.01726	0.01

Abs(Dif)-LSD	Control	Audit	Pen Test
Control	-13.668	-13.451	1.810
Audit	-13.451	-13.668	1.593
Pen Test	1.810	1.593	-13.668

Level		Mean
Control	A	25.173913
Audit	A	24.956522
Pen Test	B	9.695652

Levels not connected by same letter are significantly different

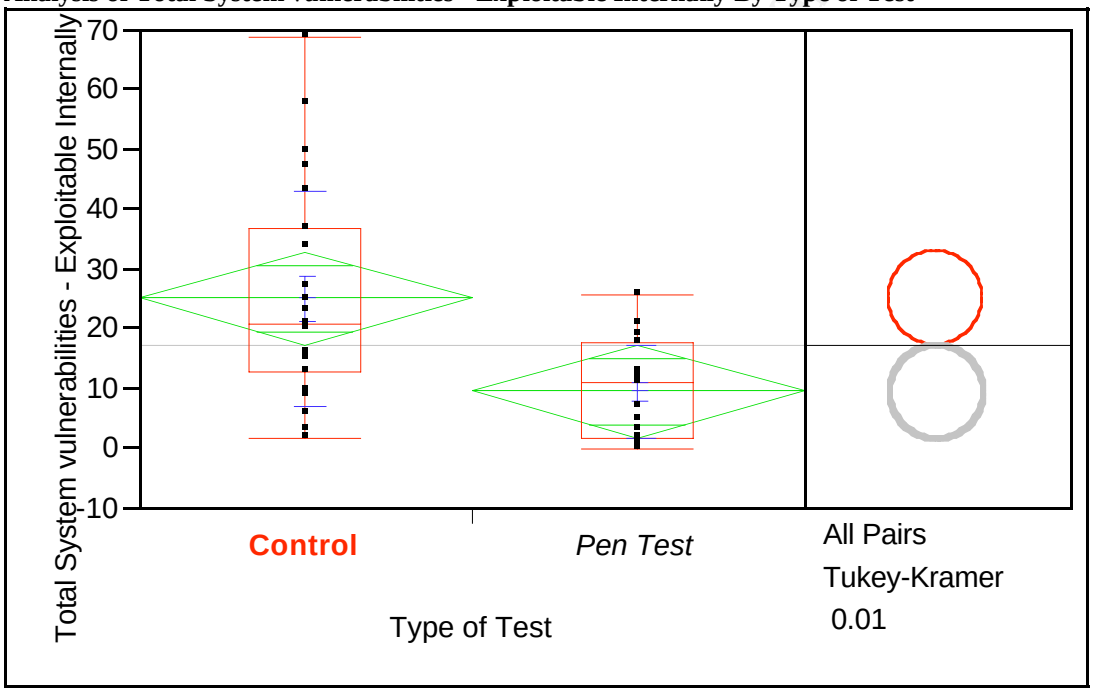
Level	- Level	Difference	Lower CL	Upper CL	Difference
Control	Pen Test	15.47826	1.8102	29.14634	
Audit	Pen Test	15.26087	1.5928	28.92895	
Control	Audit	0.21739	-13.4507	13.88547	

Hypotheses Three - An external "Pen.Test" will not find all vulnerabilities on a system

Analysis of Total vulnerabilities - Exploitable Externally By Type of Test
t Test
Control-Pen Test
Assuming equal variances

Difference	6.4783	t Ratio	3.315505
Std Err Dif	1.9539	DF	44
Upper CL Dif	11.7388		
Lower CL Dif	1.2177	Prob > t	0.0009
Confidence	0.99		

Analysis of Total System vulnerabilities - Exploitable Internally By Type of Test

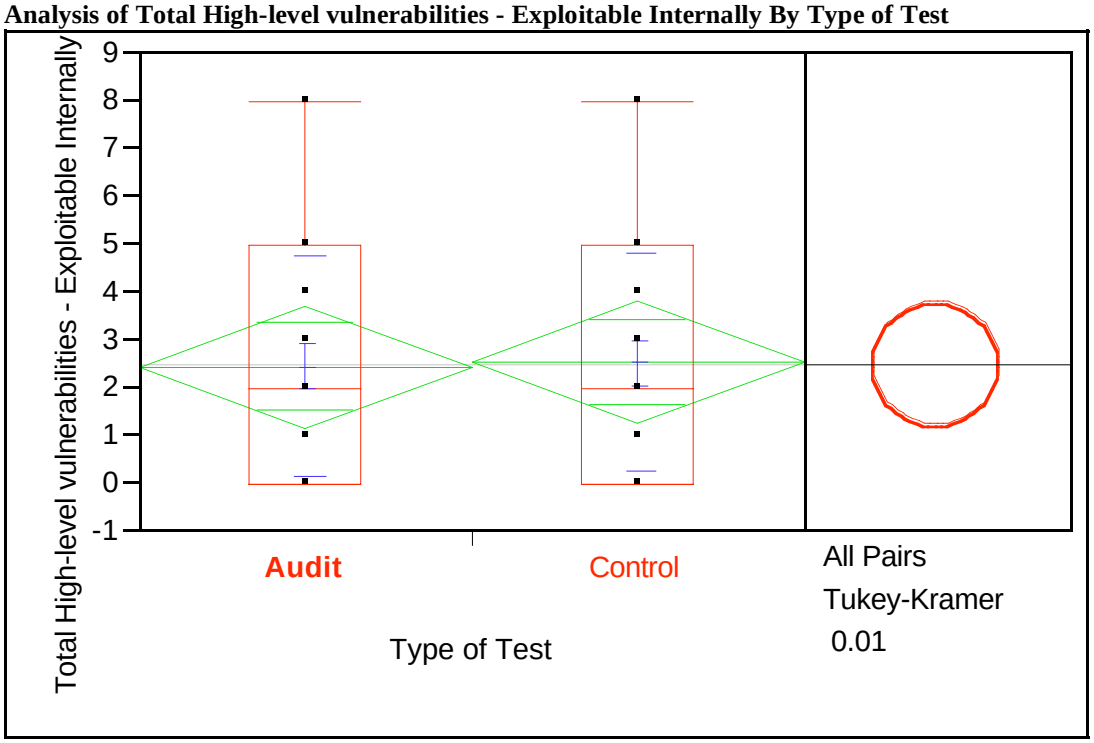


t Test
Control-Pen Test
Assuming equal variances

Difference	15.4783	t Ratio	3.773785
Std Err Dif	4.1015	DF	44
Upper CL Dif	26.5207		
Lower CL Dif	4.4358	Prob > t	0.0002
Confidence	0.99		

Hypotheses Four - An audit will find most or all system vulnerabilities

Analysis of Total High-level vulnerabilities - Exploitable Internally By Type of Test



t Test

Audit-Control
Assuming equal variances

Difference	-0.0870	t Ratio	-0.12859
Std Err Dif	0.6762	DF	44
Upper CL Dif	1.7337	Prob < t	0.4491
Lower CL Dif	-1.9076		
Confidence	0.99		

Analysis of Total System vulnerabilities - Exploitable Internally By Type of Test

t Test

Audit-Control
Assuming equal variances

Difference	-0.217	t Ratio	-0.041
Std Err Dif	5.302	DF	44
Upper CL Dif	10.467	Prob < t	0.4837
Lower CL Dif	-10.902		
Confidence	0.95		

Hypotheses Five - An audit of the systems will discover a lower number of false positives than a "Pen.Test".

Direct Comparison between Audit and "Pen.Test"s

Analysis of False Positives - High By Type of Test
t Test
Audit-Pen Test
Assuming equal variances

Difference	-0.9565	t Ratio	-2.30969
Std Err Dif	0.4141	DF	44
Upper CL Dif	0.1584	Prob > |t|	0.0257
Lower CL Dif	-2.0715	Prob > t	0.9872
Confidence	0.99	Prob < t	0.0128

Analysis of False Positives - Total By Type of Test
t Test
Audit-Pen Test
Assuming equal variances

Difference	-3.9130	t Ratio	-3.4763
Std Err Dif	1.1256	DF	44
Upper CL Dif	-0.8825	Prob > |t|	0.0012
Lower CL Dif	-6.9436		
Confidence	0.99		

COMPARISON WITH THE CONTROL

Analysis of False Positives - High By Type of Test
One-way ANOVA
Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Type of Test	2	15.42029	7.71014	5.8637	0.0045
Error	66	86.78261	1.31489		
C. Total	68	102.20290			

Means Comparisons
Comparisons for all pairs using Tukey-Kramer HSD

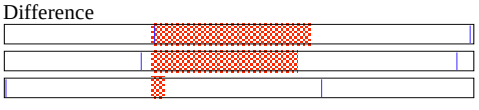
q*	Alpha
3.01726	0.01

Abs(Dif)-LSD	Pen Test	Audit	Control
Pen Test	-1.02025	-0.06373	0.02322
Audit	-0.06373	-1.02025	-0.93330
Control	0.02322	-0.93330	-1.02025

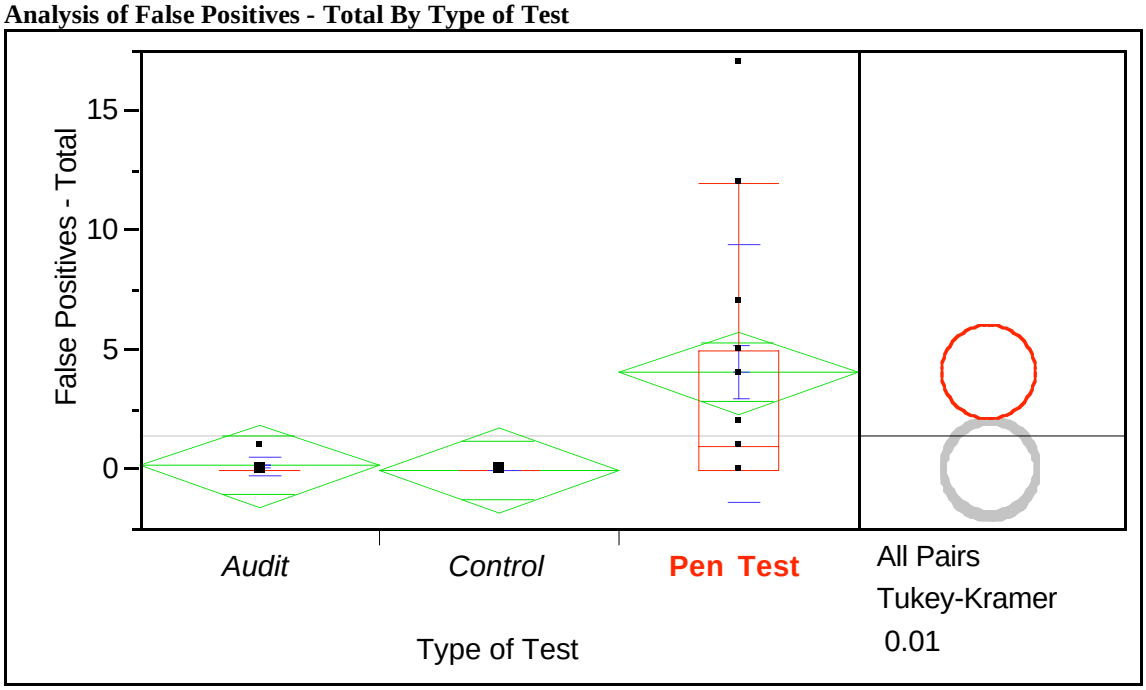
Level			Mean
Pen Test	A		1.0434783
Audit	A	B	0.0869565
Control		B	0.0000000

Levels not connected by same letter are significantly different

Level	- Level	Difference	Lower CL	Upper CL	Difference
Pen Test	Control	1.043478	0.023225	2.063732	
Pen Test	Audit	0.956522	-0.063732	1.976775	
Audit	Control	0.086957	-0.933297	1.107210	



Analysis of False Positives - Total By Type of Test



One-way ANOVA

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio	Prob > F
Type of Test	2	245.68116	122.841	12.6456	<.0001
Error	66	641.13043	9.714		
C. Total	68	886.81159			

Means Comparisons

Comparisons for all pairs using Tukey-Kramer HSD

q*	Alpha
3.01726	0.01

Abs(Dif)-LSD	Pen Test	Audit	Control
Pen Test	-2.7731	1.1399	1.3139
Audit	1.1399	-2.7731	-2.5992
Control	1.3139	-2.5992	-2.7731

Level		Mean
Pen Test	A	4.0869565
Audit	B	0.1739130
Control	B	0.0000000

Levels not connected by same letter are significantly different

Level	- Level	Difference	Lower CL	Upper CL	Difference
Pen Test	Control	4.086957	1.31386	6.860052	
Pen Test	Audit	3.913043	1.13995	6.686139	
Audit	Control	0.173913	-2.59918	2.947009	

Appendix – Related Organisations

AICPA	American Institute of Certified Public Accountants
ANSI	American National Standards Institute
ASBDC-US	Association of Small Business Development Centres
BSA	Business Software Alliance
BSI	British Standards Institute
BSI	Bundesamt mfr Sicherheit in der Informationstechnik Federal Office for Information Security (BSI) Germany
CERT	Computer Emergency Response Team
CIAO	Critical Infrastructure Assurance Office
CICA	Canadian Institute of Chartered Accountants
CIS	The Center for Internet Security
CMU SEI	Carnegie Mellon University, Software Engineering Institute
COSO	Committee of Sponsoring Organizations for the Commission on Fraudulent Financial Reporting (Treadway Commission)
DHS	Department of Homeland Security
DISA	Defense Information Systems Agency
FFIEC	Federal Financial Institutions Examination Council (USA)
FSR	Financial Services Roundtable
FTC	Federal Trade Commission (USA)
GAISPC	Generally Accepted Information Security Principles Committee
IAIP	Information Assurance and Infrastructure Protection Directorate of the U.S. Department of Homeland Security (DHS)
IATF	Information Assurance Task Force, National Security Agency Outreach

ICAEW	Institute of Chartered Accountants in England & Wales
ICC	International Chamber of Commerce
IFAC	International Federation of Accountants
IIA	The Institute of Internal Auditors, Inc.
ISECOM	The Institute for Security and Open Methodologies
ISA	Internet Security Alliance
ISACA	The Information Systems Audit and Control Association
ISF	Information Security Forum
ISO	International Organization for Standardization
ISSA	Information Systems Security Association
NACD	National Association of Corporate Directors
NCSA	National Cyber Security Alliance
NCSP	National Cyber Security Partnership
NERC	North American Electric Reliability Council
NIST	National Institute for Standards and Technology
NSA	National Security Agency
NVD	National Vulnerability Database, NIST
OCEG	Open Compliance and Ethics Group
OWASP	Open Web Application Security Project
OECD	Organization for Economic Cooperation and Development
PCAOB	Public Company Accounting Oversight Board
SANS	Systems Administration, Audit, and Network Security Institute
SEC	Securities & Exchange Commission
SEI	Carnegie Mellon University Software Engineering Institute

SNAC	Systems and Network Attack Center (NSA)
US-CERT	U.S. Computer Emergency Readiness Team
WB	World Bank

Appendix – Standards

The following is a non-exclusive collection of legislation, standards and testing criteria that may be used as a baseline against which to audit or review a system.

- Basel II - Revised international capital framework – Basel Committee on Banking Supervision, Bank for International Settlements
- BS 7799 - Parts 1 & 2, Code of Practice for Information Security Management (British Standards Institute)
- COBIT - Control Objectives for Information and Related Technologies (ISACA)
- Common Criteria
- Consensus Benchmark Scoring Tools - http://www.cisecurity.org
- The Corporate and Auditing Accountability, Responsibility, and Transparency Act of 2002, Public Law 107-204 – 107th Congress, the “Sarbanes-Oxley Act of 2002”.
- EU Data Protection Directive - Part 1 & Part 2
- Federal Information Security Management Act of 2002 (FISMA) U.S. Congress, 2002
- GAISP - Generally Accepted Information Security Principles
- GAPP - "Generally Accepted Principles and Practices" NIST SP 800-18, "Guide for Developing Security Plans for Information Technology Systems"
- A Guide to Building Secure Web Applications, The Open Web Application Security Project (OWASP)
- Gramm, Leach, Bliley Act (GLBA) The Financial Modernization Act of 1999
- HIPPA - Health Information Portability and Accountability Act
- ICAT Metabase of Common Vulnerabilities and Exposures – National Institute of Standards and Technology (NIST)

- Information Assurance Technical Framework, Information Assurance Task Force (IATF) National Security Agency Outreach
- The Information Technology Baseline Protection Manual, Federal Office for Information Security (BSI) Germany
- Information Technology Controls, Global Technology Audit Guide, The Institute of Internal Auditors
- Information Technology Security Evaluation Criteria (ITSEC) – Harmonised Criteria of France, Germany, the Netherlands, the United Kingdom
- IFAC - - International Guidelines on Information Technology Management
- International Standards for the Professional Practice of Internal Auditing, The Institute of Internal Auditors, Inc
- ISO 17799 / ISO 27001 - IT Code of Practice for Information Security Management
- NIST 800-14 - Generally Accepted Principles and Practices for Securing IT Systems
- NIST 800-27 - Engineering Principles for IT Security
- NIST 800-53 - Recommended Security Controls for Federal Info Systems
- Open Web Application Security Project (OWASP)
- The Organization for Economic Cooperation and Development, OECD Guidelines for the Security of Information Systems and Networks
- Personal Information Protection and Electronic Documents Act (PIPEDA), Canada
- Policy statement regarding implementation of auditing standard No. 2, an audit of internal control Over financial reporting performed in Conjunction with an audit of financial Statements, PCAOB Release No. 2005-009
- Standard of Good Practice for Information Security (Information Security Forum),

- Trusted Computer System Evaluation Criteria (TCSEC), U.S. Department of Defense, Trust Services Criteria; including SysTrust/WebTrust (AICPA)

---

[i] No effort to test the relative security of Unix/Linux vs. Microsoft or the relative security of the various releases (of these systems) is being made within this project. The tests are purely aimed at audit vs. "Pen.Testing" and arguments for Windows 2003 or NT 4.0 systems or other Linux variants are not relevant to the results.
[ii] S.C.O.R.E. - http://www.sans.org/score/
[iii] The Auditor security collection is a Live-System based on KNOPPIX
[iv] http://remote-exploit.org/

Last Updated: February 22nd, 2018

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

CyberThreat Summit 2018	London, GB	Feb 27, 2018 - Feb 28, 2018	Live Event
SANS London March 2018	London, GB	Mar 05, 2018 - Mar 10, 2018	Live Event
SANS Secure Osaka 2018	Osaka, JP	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS San Francisco Spring 2018	San Francisco, CAUS	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS Paris March 2018	Paris, FR	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS Secure Singapore 2018	Singapore, SG	Mar 12, 2018 - Mar 24, 2018	Live Event
SANS Northern VA Spring - Tysons 2018	McLean, VAUS	Mar 17, 2018 - Mar 24, 2018	Live Event
ICS Security Summit & Training 2018	Orlando, FLUS	Mar 18, 2018 - Mar 26, 2018	Live Event
SANS Munich March 2018	Munich, DE	Mar 19, 2018 - Mar 24, 2018	Live Event
SEC487: Open-Source Intel Beta One	McLean, VAUS	Mar 19, 2018 - Mar 24, 2018	Live Event
SANS Pen Test Austin 2018	Austin, TXUS	Mar 19, 2018 - Mar 24, 2018	Live Event
SANS Secure Canberra 2018	Canberra, AU	Mar 19, 2018 - Mar 24, 2018	Live Event
SANS Boston Spring 2018	Boston, MAUS	Mar 25, 2018 - Mar 30, 2018	Live Event
SANS 2018	Orlando, FLUS	Apr 03, 2018 - Apr 10, 2018	Live Event
SANS Abu Dhabi 2018	Abu Dhabi, AE	Apr 07, 2018 - Apr 12, 2018	Live Event
Pre-RSA® Conference Training	San Francisco, CAUS	Apr 11, 2018 - Apr 16, 2018	Live Event
SANS Zurich 2018	Zurich, CH	Apr 16, 2018 - Apr 21, 2018	Live Event
SANS London April 2018	London, GB	Apr 16, 2018 - Apr 21, 2018	Live Event
SANS Baltimore Spring 2018	Baltimore, MDUS	Apr 21, 2018 - Apr 28, 2018	Live Event
SANS Seattle Spring 2018	Seattle, WAUS	Apr 23, 2018 - Apr 28, 2018	Live Event
Blue Team Summit & Training 2018	Louisville, KYUS	Apr 23, 2018 - Apr 30, 2018	Live Event
SANS Riyadh April 2018	Riyadh, SA	Apr 28, 2018 - May 03, 2018	Live Event
SANS Doha 2018	Doha, QA	Apr 28, 2018 - May 03, 2018	Live Event
SANS SEC460: Enterprise Threat Beta Two	Crystal City, VAUS	Apr 30, 2018 - May 05, 2018	Live Event
Automotive Cybersecurity Summit & Training 2018	Chicago, ILUS	May 01, 2018 - May 08, 2018	Live Event
SANS SEC504 in Thai 2018	Bangkok, TH	May 07, 2018 - May 12, 2018	Live Event
SANS Security West 2018	San Diego, CAUS	May 11, 2018 - May 18, 2018	Live Event
SANS Melbourne 2018	Melbourne, AU	May 14, 2018 - May 26, 2018	Live Event
SANS Northern VA Reston Spring 2018	Reston, VAUS	May 20, 2018 - May 25, 2018	Live Event
SANS New York City Winter 2018	OnlineNYUS	Feb 26, 2018 - Mar 03, 2018	Live Event
SANS OnDemand	Books & MP3s OnlyUS	Anytime	Self Paced