Global Information Assurance Certification Paper

Auditing the Perimeter: Conducting an External 'Zero Knowledge' Vulnerability Assessment

An Auditor's Perspective

SANS GIAC Systems and Network Auditing Practical Version 2.1

Option 2

Wipul Jayawickrama

13 August 2003

(This page is intentionally left blank)

Table of Contents

Auditing the Perimeter: Conducting an External 'Zero Knowledge' Vulnerability Assessment

1 Abstract/Summary

This document constitutes the practical component of the SANS GIAC Systems and Network Auditor Certification and is based on an External vulnerability assessment conducted on an accounting firm.

Due to the nature of the audit, which is auditing a range of IP addresses, Option 2 of the practical assignment has been the format agreed upon for the submission.

Four assignments make up the practical and a description of these assignments is as follows.

Assignment1 – Methodology

- Description of the Methodology
- Current State of Practice
- Need for a new Methodology

Assignment 2 – Methodology Check List

- Reference
- Vulnerabilities and Assets Detected
- Tools, Techniques and Testing Procedure
- Evaluation of the Test Value to Methodology

Assignment 3 – Sample Audit

- Conduct the Audit
- How to determine the Risk

Assignment 4 – Follow Up Report

- Executive Summary
- Audit Findings
  - Risk
  - Audit Recommendations
  - Cost
- Risk Analysis

2 Assignment 1 – Methodology

2.1 Description

External vulnerability assessments are becoming a commonplace in the security auditing practices. As evidenced by a search on the world wide web for 'Internet "vulnerability assessment"' or "vulnerability and penetration testing", there are a large number of organisations providing vulnerability assessments, and there are also several training programmes available on the subject.

As much as there is discussion on he subject, there does not seem to be a standard practice related to external vulnerability assessment among the many practitioners of this emerging branch of auditing. While there are some emerging standards, which will be discussed later on in this paper, the practice of vulnerability assessment appears to be driven by the providers of this service, and the needs of their clientele.

There are three different forms of external vulnerability assessment: full knowledge, partial knowledge and zero knowledge. With a full knowledge assessment the assessor knows the systems s/he is about to assess. S/he knows the IP addresses of the systems, operating system versions, software applications running on these systems, the roles these systems play within the organisation, and the consequences of these systems being unavailable. A partial knowledge assessment is where some of this information is made available to the assessor, typically the IP address range to be audited and the operating systems and software running on these systems.

Zero Knowledge vulnerability assessments are different to the typical systems audit in many ways. In most cases the auditor does not have any information about the organisation s/he is about to assess apart from the name of the company. Armed with this information, the auditor begins the investigations and builds up a picture of the organisation's information assets. S/he then probes these assets for any vulnerability. These vulnerabilities could be either related to the technology used, or related to bad practices. The auditor then has to assess the threat and risks to the organisation posed by these vulnerabilities. This is not an easy task, as the auditor has no information as to the value of these information assets from the organisation's perspective.

In this paper, a methodology for conducting such zero knowledge vulnerability assessments is proposed. This methodology takes into consideration the following characteristics of a zero knowledge vulnerability assessments.

- The auditor has no information about the organisation s/he is about to audit other than the name of the organisation.

- The auditor has minimal contact with the organisation during the audit. This contact may be limited to a technical contact to ensure that the system being audited does belong to the organisation being audited.
- Any business intelligence provided by the auditor, as part of the audit could be subject to a different interpretation by the organisation being audited.

2.2 Proposed methodology for conducting an external Zero Knowledge Vulnerability Assessment:

The methodology proposed consists of three main stages: engagement, audit and Reporting. Each stage has sub-components defined, and also has several break points in which the audit findings are confirmed before continuing on to the next stage.

2.2.1 Methodology

1. Engagement
    a. Define the scope of the engagement
    b. Define the rules of engagement
2. Audit
    a. Planning the audit
    b. Reconnaissance
        i. Identify the organisation's Internet presence
        ii. Enumerate Information Assets
        iii. Map the organisation's network systems and topology as visible to the Internet
    c. Identify potential vulnerabilities
        i. Identify leaked information that provides information used for potential attackers
        ii. Scan the systems to identify any vulnerabilities in perimeter network configuration such as poorly constructed firewall rule sets, services accessible from the Internet, and identify operating systems and software used within the systems.
        iii. Probe the identified services for any known vulnerabilities
        iv. Identify vulnerabilities in any publicly available web based applications
    d. Verify and rate the identified vulnerabilities

e. Exploit identified vulnerabilities if within scope

3. Reporting and follow up
    a. Executive summary
    b. Findings
    c. Risk Assessment
    d. Recommendations
    e. Cost to implement Audit Recommendations

2.2.2 Check points

Two mandatory check points are proposed to ensure that the audit remains within scope and inadvertent breaches of law does not occur during the audit. These check points require that the auditor contact the organisation to verify information discovered during the assessment.

1. Checkpoint 1 – verify that the identified information assets are the ones to be audited
2. Checkpoint 2 – reconfirm that systems identified with vulnerabilities do belong to the organisation, and that the exploitation of the vulnerabilities at this stage is acceptable to the organisation.

For assessments with extensive scopes, it may be necessary to have more checkpoints during the audit.

2.2.3 Vulnerability Ratings

Identified vulnerabilities would be assigned a rating using the following scale:

- **Critical –** Critical vulnerability identified. Needs immediate attention and action. The vulnerability has known exploits, and they are easy to deploy. The use of such an exploits is a total compromise of the system that may result in the loss, modification or theft of data, or a denial of service that may render the system unavailable. The system is exposed to external attackers.
- **High –** High-risk vulnerability identified. Needs to be addressed as soon as possible. The vulnerability has been known publicly and there may be unpublicised exploits. The result of the exploit could be a total compromise of the system, or a denial of service.
- **Moderate** – Needs to be addressed once vulnerabilities rated critical and high have been addressed.
- **Low** – Could be addressed during scheduled maintenance.
- **Informational** – Possible vulnerability identified. Further tests/research may be necessary to determine the actual impact on the organisation.

2.3 Current State of Practice

As mentioned earlier in this paper, currently there is no established standard of practice for external vulnerability testing. There are a few emerging frameworks that attempt at providing the vulnerability assessor with some guidance. Vulnerability Analysis and Penetrations Testing supplement to the Common Criteria's Common Methodology for Information Technology Security Evaluation (CEM Supplement) is one such emerging framework. The National Institute for Standards in Technology publication Draft Guideline on Network Security Testing (NIST 800-42) discusses vulnerability analysis in some detail, and includes descriptions of some tools available to conduct such an evaluation. The Open Source Security Testing Methodology Manual (OSSTMM) is another document referred to in the vulnerability analysis literature. There are few books on the subject, and most of the publicly available methodology seems to be developed by organisations offering vulnerability analysis services.

The following sub-sections of the paper summarise the available literature on the subject.

2.3.1 Common Methodology for Information Security Evaluation, Supplement: Vulnerability Analysis and Penetration Testing

One of the guidelines available for vulnerability and penetration testers is the Vulnerability Assessment and Penetration Testing supplement to the Common Criteria (CEM Supplement). Although still in draft status, it builds upon the established Common Criteria for Information Technology Security Evaluation (Common Criteria) and contains several definitions and guidelines useful to the vulnerability analyst.

According to the CEM Supplement, there are two main factors to consider when performing a vulnerability analysis. They are:

a) Identifying the potential vulnerabilities

b) Establishing that these identified vulnerabilities are exploitable

Exploitability of the vulnerabilities can be determined either through a theoretical assessment of the vulnerability and available attack methods, or by actually exploiting the identified vulnerabilities by running the attacks against these vulnerabilities. The exploitation of these vulnerabilities is known as penetration testing. (p 12, 17)

Four types of potential attacks have been defined in the CEM Supplement document.

a) Bypassing, where the attacker avoids any security enforced on the system to gain access to the system.

b) Tampering, where the attacker influences the behaviour of a security function or a mechanism by either corrupting or deactivating the enforced security

c) Direct Attack against the security mechanism or the system

d) Misuse of the system (pages 62-68)

The assessor could use different approaches to identify potential vulnerabilities that could lead to an attack on the system. S/he may examine the development and guidance evidence that is made available to the assessor. Or s/he may use his/her knowledge and understanding of the system assessed, identify potential vulnerabilities in this particular implementation. Any publicly available information on the system and its vulnerabilities may constitute the body of knowledge that the assessor uses to identify the vulnerabilities (p 30). Currently in vulnerability assessments this publicly available knowledge comprises of various tools and knowledge bases made available to assessors.

Once the potential vulnerabilities of the systems have been identified, the assessor needs to consider the vulnerability against the operating environment of the system. The assessor needs to determine if the vulnerability is actually exploitable by an attacker in the system's operating environment. This is because, a vulnerability with a high attack potential may still only pose minimal risk to a system. The following example can be used to illustrate this point.

In a recent internal and external audit, the author of this paper identified several systems with the recently identified Microsoft DCOM vulnerability. Exploiting this vulnerability, an attacker could spawn a remote shell with full NT System Authority. While this was generally considered a high severity vulnerability, the organisation was not exposed to any risk of an external attack at the time of the audit. The organisation did not host any services within the internal network, hence the firewall was configured to reject all incoming packets. Therefore the attack potential assigned to remote exploitation of this vulnerability by the auditor was low. The situation changed slightly a few weeks later with the spread of the Blaster worm, which exploited the vulnerability. However, this time the attack was possible via email, which the company downloaded from their mail server hosted at an ISP. The risk of the remote exploit was higher and a new control mechanism was required. A vendor provided hot-fix had to be applied, and virus signatures needed to be upgraded, to mitigate the risk posed by the same vulnerability.

Another aspect to be considered when mapping the attack potential to vulnerability is the potential attacker. The CEM Supplement recommends that at least three attack scenarios be considered when assessing a system and its operating environment. These three scenarios are:

a) Does the system, in its operational environment, have vulnerabilities exploitable by an attacker possessing basic attack potential?

b) Does the system, in its operational environment, have vulnerabilities exploitable by an attacker possessing moderate attack potential?

c) Does the system, in its operational environment, have vulnerabilities exploitable by an attacker with possessing high attack potential? (pp 42-54)

There are three key factors to be considered in determining the attack potential against an identified vulnerability. The CEM Supplement is somewhat vague in the definitions of these factors. What follows here is the author's interpretation of these terms.

a) Cost – the cost to identify the vulnerability, develop an attack and exploit the vulnerability

b) Risks – the risks that the attacker is willing to take to exploit the vulnerability

c) Objectives – objective of the attacker.

Some terminology related to vulnerability analysis and penetration testing has been defined in the CEM. Some of these definitions that will be useful in the proposed methodology have been included herewith (pp 9-10)

a) Vulnerability – a weakness in the system assessed that can be exploited to violate a security policy applied to that system

b) Vulnerability analysis – systematic search for vulnerabilities and an assessment and/or testing of those vulnerabilities to identify their relevance for their operation environment

c) Potential vulnerability – suspected, but not confirmed vulnerability

d) Exploitable vulnerability – a vulnerability that can be exploited in the operational environment of the system being assessed

e) Non-exploitable vulnerability – a vulnerability that can not be exploited in the operational environment of the assessed system

f) Residual vulnerability – a non-exploitable vulnerability that could be exploited by an attacker with greater attack potential than normally expected in that operating environment.

g) Area of concern – potential vulnerability that were identified, although not directly related to the class of vulnerabilities assessed. Further investigation of these areas of concern is recommended.

Guidelines for evaluating vulnerabilities are discussed to some extent in the CEM supplement. However, this is limited to a high level view of the evaluation methodology. The entire guideline is an abstraction of the process of vulnerability assessment. The following extract from the CEM Supplement CEM Supplement can be used to illustrate this point. This extract is on identifying vulnerabilities caused by leaked information on the Internet, which maps to the reconnaissance section on the methodology proposed in this paper.

> The evaluator will use information from sources of information publicly available to support the identification of possible security

vulnerabilities in the TOE. There are many sources of publicly available information which the evaluator should consider using:

a) world wide web;
b) specialist publications (magazines, books);
c) research papers;
d) conference proceedings. (p 80)

It then states that the evaluator should not constrain the search to the above list, and should look at "any other information available" (p 80). There aren't any examples provided on what the evaluator should look for, or what type of information can be causing vulnerabilities to the systems.

As seen from the above discussion of some of the content in the CEM supplement, one can see that the aim of the document is to provide a high level document providing definitions, interpretations and some guidelines to evaluators. This is acceptable, as the aim of the document is just that, to provide a high level document to provide guidance to the process of vulnerability assessment and penetration testing, and allow assessors to develop methodologies to suit specific needs of individual engagements.

2.3.2 National Institute for Standards in Technology – Draft Guideline on Network Security Testing (NIST 800-42)

The NIST 800-42 document goes one step further that the CEM Supplement by providing a more specific discussion of threats and vulnerabilities.

This document considers the threats in cyberspace to be the same as in the physical world, listing fraud, theft and terrorism as examples. However, three important developments makes it necessary to carry out periodical security testing. These three developments are:

- increased profitability due to automation of attacks,
- the ability to action these attacks from a distance
- ability to propagate these techniques rapidly before counter measures are developed.

Security testing is defined as a one of the most conclusive ways to determine if "existing security measures and procedures are working as intended" as well as to identify unknown weaknesses and vulnerabilities. (NIST 800-42:3). The document, according to the authors, aims to provide basic techniques and tools to develop a security testing methodology for firewalls, routers and switches, perimeter security systems such as intrusion detection systems, web servers and email servers and other servers such as domain name servers and directory servers.

The primary reason for security testing a system is to identify potential vulnerabilities and subsequently repair them. Hackers (sic) repeatedly exploit these vulnerabilities, making security testing a fundamental security activity

that organisations need to undertake in order to ensure a secure operating environment for their systems (p.7).

The guideline lists several types of security testing techniques, types of tools to be used for security testing and some examples of tools, their usage and limitations.

- **Network mapping** – identifying all active hosts connected to an organisation's network, network services running on those hosts, and the specific applications running on those hosts. Value of network mapping is to identify unauthorised hosts connected to the organisation's network, identify vulnerable services, and identify deviations from the allowed services defined in the organisation's security policy.

  Use of results – disconnect unauthorised hosts, disable or remove unnecessary and vulnerable services, modify vulnerable hosts to restrict access to vulnerable services to limited number of required hosts and modify firewalls to restrict outside access to known vulnerable services.

- **Vulnerability Scanning** – identifies hosts, open ports and vulnerabilities associated with those ports. Some scanners attempt to provide information to mitigate discovered vulnerabilities.

  Use of results – upgrade or patch systems to mitigate vulnerabilities, deploy technical or procedural measures to minimise vulnerability, tighten configuration management program and procedures, monitor vulnerability alerts and mailing lists and initiate appropriate systems changes if applicable, modify the policies, architecture and documentation to ensure timely system updates and upgrades are carried out.

- **Penetration Testing** – evaluators attempt to circumvent the security features of a systems based on their understanding of the systems design and implementation, using common tools and techniques developed by hackers (sic). Penetration testing simulates an attack, and written permission is necessary to conduct such a test.

  Use of results – determine how vulnerable an organisation's network is and the level of damage that can occur if compromised. Apply corrective measures to mitigate vulnerabilities, and improve security practices.

- **Security test and evaluation** – analysis of the protective measures placed on an information system and includes communications security, emanations security, physical security, personnel security, administrative security and operational security components.

  Use of results – to uncover design, implementation and operational flaws, determine the adequacy of security mechanisms, assurances

and other properties to enforce security policy, and assess the degree of consistency between system documentation and implementation.

- **Password Cracking** – to identify weak passwords

  Use of results – to modify policy if policy allowed weak passwords, to educate users on possible impact of using weak password if policy disallowed use of weak passwords. In the latter case ensure enforcement of policy

- **Log review** – identify deviations from the security policy, review ongoing systems activities compared to policy, validate system is operating according to policy.

  Use of results – reconfigure systems to reduce chance of compromise, change firewall policy to limit access to vulnerable system or service, change firewall policy to limit access from the IP subnet that is the source of the compromise.

- **File Integrity Checkers** – compute and establish a database of file check sums, identify authorised and unauthorised changes to those files by periodically computing checksums and comparing with initial database.

  Use of results – identification of a security breach, investigate and respond to incident.

- **Virus Detection** – mitigate the risk of contracting computer viruses, Trojans and worms.

- **War dialling** – identify unauthorised modems on the network that bypass perimeter security.

Appendix C of the guidelines lists both commercial and non-commercial tools that may be used in the above security tests.

While the information in the NIST 800-42 document is less abstract than in the CEM Supplement, the information in the NIST document does not cover several aspects of the vulnerability assessment. For example, the approach is limited to identifying technical vulnerabilities of the systems. There are no methods discussed on the vulnerabilities caused by human practices. The entire approach is based on the use of certain tools, and the range of tools discussed is minimal. The most limiting factor of this document is that it does not provide sufficient information to the assessor to map the identified vulnerabilities to specific risks related to the organisation being assessed.

2.3.3 Open Source Security Testing Methodology Manual v2.0

A commendable attempt to standardize security testing methodology is the Open Source Security Testing Methodology Manual (OSSTMM) developed by Pete Herzog (Herzog, 2002). Assisted by several volunteer contributors, Herzog has developed a methodology that "… focuses on security testing from the outside to the inside... " As indicated on the front page of the manual.

Herzog defines a security test as an attack, and identifies two types of such attacks. Data collection, which does not directly influence or trespass upon the target, is called a passive attack. Security tests that intrude on the system, and that can be monitored and logged, and could generate alerts are called intrusive attacks. Using these two types of attack an auditor could test the security map of an organization. Six areas comprise the security map: Internet Security, Information Security, Physical Security, Communications Security, Wireless Security, and Social Engineering. The OSSTMM has modules that address security issues within each of these areas.

The Internet Security module of the OSSTMM (p18-34) describes the external vulnerability assessment/ penetration test. This module suggests a methodology for testing the components of a network/system as seen from the Internet. The components tested under this module are:

a) Network surveying
b) Port scanning
c) System identification
d) Services identification
e) Vulnerability research and verification
f) Internet application testing
g) Router testing
h) Firewall testing
i) IDS systems testing
j) Trusted systems testing
k) Password cracking
l) Denial of service testing
m) Containment measures testing.

Each of these modules is supported by test templates, and a list of tasks that needs to be carried out to complete the test. In addition to the technical assessment, Herzog also includes the groundwork for risk assessment using competitive intelligence gathering techniques. Another excellent feature of this document is that it defines a list of expected outcomes with each test template.

Of the public guidelines and standards, the author found the OSSTMM to cover more aspects of the vulnerability assessment than any other document, However, this is still a work in progress, and in the future there is the likelihood that each test template will have tools that may be used to complete that task.

2.3.4 Books and other Literature on Vulnerability Assessment

While the above mentioned frameworks provide the guidelines for conducting an external vulnerability assessment, they do not provide the technical know-how nor recommend tools for actually carrying out the assessment tasks. Several books published recently provide this missing link, providing information on the tools are used in systems attacks, and how those tools may be used in assessing vulnerabilities.

The Hack-Counter Hack Training Course (Skoudis, 2002) identifies a 5 phase process that attackers use when compromising systems. These 5 phases are reconnaissance, scanning, gaining access, maintaining access and covering the tracks. Skoudis discusses several tools and utilities used by attackers to gather information and gain access to systems.

An external vulnerability analysis uses a somewhat similar methodology to gather information, identify vulnerabilities, and exploit the identified vulnerabilities. However, there are several key differences in the attacker approach to that of the vulnerability assessors approach. One of these key differences is the objective for the identification of the vulnerability. While the attackers objective may be to identify and exploit the vulnerability for personal gain or gratification, the assessor's objective is to assist the organisation address the vulnerability and fix it. Also the attacker may be content with identifying and exploiting 'A' vulnerability as opposed to an assessor's task involving the identification of all existing vulnerabilities.

There are several other publications that discuss systems vulnerabilities. They mainly focus on individual attacker methodology and toolsets used by attackers. Some of the books that were consulted during the authors research McLure et al (2001), Klevinski et al (2002) and various articles at the SANS reading room (http://rr.sans.org) also provided some interesting reading on external security audits and vulnerability assessment techniques.

2.4 Need for a new methodology for conducting an external 'Zero Knowledge' vulnerability assessment.

While there are draft guidelines/standards publicly available, and several books on the use of tools for external assessment of vulnerabilities, the methodology suggested in these publications are incomplete. As mentioned earlier in this paper, the NIST 800-42 guideline and the CME Supplement are still in draft stage, and need much more work to be comprehensive and authoritative. The author's research of these books subject indicates that these books contain useful information on the use of tools for testing vulnerabilities. However, they lacked depth in providing a cohesive methodological approach to a corporate vulnerability assessment. The OSSTMM discussed above, provided a methodology. Although the templates

provided in the guide provide a good starting point for the external vulnerability assessor, it did not provide sufficient detail and depth in conducting an external vulnerability assessment.

One of the major limitations of all methodologies and books discussed in the current state of practice section is that they lacked information on mapping identified vulnerabilities to the assessed organisation's operations. There was also very little information on how to rate the vulnerabilities that were identified, and assess the risks posed to the organisation by the vulnerabilities that were identified. Mapping the identified vulnerabilities to risks should be a key component of the vulnerability assessment process.

Another major limitation that the author found in his research in the existing methodologies is that most of the available literature does not include another very important component of the vulnerability assessment, the report presented to the organisation. There was very little information on how to present the findings of an external vulnerability assessment to an organisation in a manner that management and staff of varying technical capabilities may understand the meaning of the findings. Most of the sample external vulnerability reports the author examined followed two major trends. The report either consisted of,

a) direct output from the tools used for the assessment, with no analysis from the assessor, or

b) technical jargon that probably did not make much sense to the non-technical and decision making persons within the organisation.

The biggest challenge the author faced in developing this methodology and the checklist is that there are no benchmarks or baselines in perimeter vulnerability assessments. There are many available guidelines, baselines and checklists to assist systems auditors in evaluating systems. By definition a systems audit, measures a system for compliance against a policy or a baseline, and evaluates the compliance of that system to the policy or the baseline. In a typical systems audit operating system, versions, and applications running on the system, its role within the organisation, and how critical the system is for the functions of the organisation. The findings are then compared with the baselines and policies related to the systems and are then flagged as whether they were compliant with the control objectives governing that system. There are many baselines available to define the secure state of a system that the systems auditors may use. Unfortunately, for the perimeter vulnerability assessor, there are no such baselines available. The author could not find a common definition for a secure perimeter.

Since the task is to identify vulnerabilities, regardless of the threat level they pose to the organisation, it is not possible to develop a comprehensive and exhaustive checklist. What a perimeter vulnerability assessor has to do is to start the assessment with a common checklist, and develop a check list suitable for the particular environment being tested as the assessment progresses.

Therefore, it is evident that there is a need for a methodology that provides external vulnerability assessors a guideline to conducting perimeter assessment. This methodology needs to consider the following:

a) a cohesive process that provides a guideline for a complete vulnerability assessment from engagement to the submission of the findings to the organisation's management.

b) a well defined structured approach, with different stages of the assessment clearly demarcated.

c) a baseline checklist that identifies different components of the perimeter, and allowed adaptation based on the needs of individual organisations.

d) sufficient information on available tools and how to interpret their results

e) a mechanism to identify risks posed by the identified vulnerabilities, and

f) a follow up process that included a comprehendible report format.

The proposed methodology attempts to bridge some of the gaps between the above discussed guidelines, standards and methodologies and provide a vulnerability assessment model incorporating the positive aspects of those guidelines and the real world experience of the author. The author acknowledges that there needs to be extended collaborative research with input from practitioners of perimeter/external vulnerability assessments to ensure the maturity and standardisation of vulnerability assessments.

3 Methodology Checklist

The following checklist is structured to follow the methodology, and by no means an exhaustive list. It is not possible to compile a prescriptive list as the technologies, scopes, and even reporting requirements surrounding external vulnerability assessments vary from assignment to assignment. The checklist below has been developed as an indicative guideline or a staging platform with the flexibility to adapt, modify or add according to the nature of the engagement. A tool or a selection of tools that could be used to achieve the aim of the checklist accompanies each item in the checklist.

An important observation and a word of caution is appropriate at this stage. The checklist and the tools alone do not comprise the methodology. As the CME states, "Vulnerability Analysis is, by nature, a subjective evaluator activity based on evaluator experience, knowledge and creativity" (p17). It is possible that the use of some of the tools may create Denial of Services and other undesirable effects on the systems being tested. It is recommended that inexperienced assessors practice the application of these tools on non-production systems.

3.1 Administrative Matters

An external vulnerability assessment is an intrusive audit. The auditor will be using various tools for scanning and probing from the outside of the organisation's perimeter. Both commercial and open source tools will be used, and it is possible that the systems may respond with unexpected behaviour, resulting in non-responsive (hung) or failed (crashed) systems. It is essential that both the auditor and the organisation being audited understand the impact of the tests to be performed.

Checklist Item 1: Ensure that the scope of the engagement is defined, exclusions identified, and the agreement is documented and signed by both parties, i.e. the organisation and the auditor.

It is important that the scope be clearly defined at the outset of the assignment. Typically the scope of an assessment defines the following aspects of the Assessment.

- What the assessment covers
- Who will be involved, both from the assessors company and the organisation audited
- The depth of the assessment

Checklist Item 2: Ensure that the rules of engagement have been agreed upon and is documented and signed by both parties.

The rules of engagement typically defines the following:

- Time frame for the scans and probes
- Any periods of times that these scans and probes should not be conducted.
- Any particular type of scans and probes that should not be run.
- If any person should be informed before or during a scan, and the name of the person/s and contact number/s.
- That the organisation understands the risks related to the scans and probes. E.g. that some of the scans and probes may result in denial of services or even require the rebuilding of a server
- Professional indemnity clauses

Checklist Item 3. Sign any non-disclosure agreements if required by the organisation.

The assessment may uncover sensitive information about the organisation and its systems. Therefore the organisation may require the assessor sign a non-disclosure agreement. Some organisations may specifically request not to be listed as a reference site for the assessors work.

3.2 Reconnaissance

This stage of the audit is non-intrusive. Information gathering is usually carried out using publicly available resources, including the organisation's web server and Internet search engines.

Checklist Item 4: Identify the Internet Domain Name

Vulnerability Tested:

None

Reference:

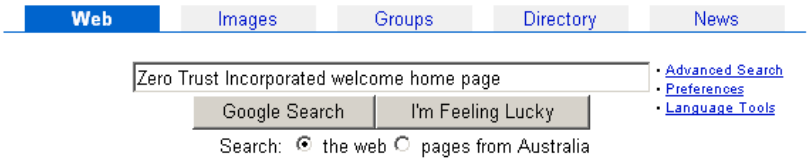
Common Knowledge, "Google, Hackers Best Friend" by Paris 2k Labs
http://www.astalavista.com/library/basics/guides/GoogleHTool.pdf

Tools, techniques and testing procedure:

Tools used: Internet Search Engines such as Google, AltaVista, and Yahoo.

Execute a search for the company name followed by strings typically used in web homepages

Example search using Google:



Once the Internet domain is identified, it would be useful to connect to the organisation's web server to gain some background information on the organisation.

The Google Search Engine can be accessed at http://www.google.com

Evaluation of Test Value to Methodology:

While the information discovered at this stage does not pose a vulnerability to the organisation, identifying the domain name is the crucial first step in determining the organisation's publicly available information assets.

The next few steps, i.e. tests identifying the information assets that follow rely the accuracy of the results of this test.

Checklist Item 5: Identify information related to Internet domain registration including

- Domain name
- IP Address Block Assigned to the Organisation,
- name service providers
- name servers, mail servers
- Technical contact information for the organisation.

Vulnerability Tested:

None

Reference:

Common Knowledge, experience

Tools, techniques and testing procedure:

- Using a *nix based system it is possible to issue a command from the shell to query the registrar for the same information.

  A simple query uses the following syntax

  whois –h whois_server domain_name

  Example:

  ```
  whois -h whois.register.com sans.org
  ```

- There are online tools provided by Domain Registrars to look for this information.
- Sam Spade for Windows – a tool with a graphical user interface for Microsoft Windows is available for free download at http://www.samspade.org

Please refer Appendix 1 for examples and sample screenshots for these tools.

Evaluation of Test Value to Methodology:

Correctly identifying the domain name is important for several reasons.

By verifying the domain name against the physical address and other information provided the auditor can confirm that the information s/he is gathering belongs to the correct organisation.

The organisation's web, mail, and other publicly accessible services, the IP addresses assigned to the organisation etc. are researched at this stage.

Checklist Item 6: Conduct a general search for information on the organisation.

Vulnerability Tested:

None

Reference:

None

Tools, techniques and testing procedure:

Internet searches on news archives, organisation's web pages for information about the organisation. While it is not possible to create a list of search items some general types of information that may be of interest are:

Business Information

- Any profit or loss postings
- Innovations
- Upcoming mergers and joint ventures
- Any bids for contract or tenders
- Who the Competition is (other companies that trade in the same fields of interests)

Other general information

- The political outlook of the organisation and its key leadership
- Individuals within the organisation that receive public exposure

While not directly considered as a vulnerability, this information may assist the auditor to establish the risk of not addressing the vulnerabilities discovered during the audit.

Evaluation of Test Value to Methodology:

As part of the assessment, the auditor's task would be to analyse the risks associated with the identified vulnerabilities.

For the risk analysis to be accurate, the auditor needs to understand the organisation and its background as much as possible. The information gathered through this step, will assist the auditor postulate why an attacker would be attracted to attack this organisation, apart from their systems being easy target to an attack.

Therefore this step constitutes an important stage of the assessment.

3.3 Enumerate Information Assets

In this stage of the audit, the auditor will attempt to identify the technologies, operating systems and software used within the organisation's network, and try to map the network topology of the network. Information gathering tools such as Internet search engines may be used, for non-intrusive assessment. Some social engineering may be used to identify how much information is provided by unsuspecting IT and non-IT staff to outsiders whose identities have not been verified.

At this stage there will also be some intrusive assessment of the systems when certain network mapping and port scanning tools are used. Therefore it is important to verify that the systems the auditor has identified, do belong to the organisation being audited, and are the ones that have been defined in the scope. Failure to do so may result in the auditor intruding upon a system that does not belong to the organisation s/he is auditing. The auditor may legally be liable for any damages caused in either situation.

It is also necessary at this stage to ensure that the services to be tested are not hosted at a third party site such as an Internet Service Provider. If they are, and the audit is to be carried out, it is necessary to have the permission of the third party in writing before any intrusive audits are carried out. It is also necessary to ensure that the servers and network devices are not shared with other organisations that have services hosted at the same third party.

Check Point 1: Verify that the domain name, IP address block, and name servers that are about to be scanned belong to the organisation being audited, and are within the agreed scope, and if the audited devices belong to a third party such as an ISP, explicit permission has been granted by the third party before any intrusive audits are conducted.

Checklist Item 7: Identify critical systems such as mail, web and other servers.

Vulnerability Tested:

None

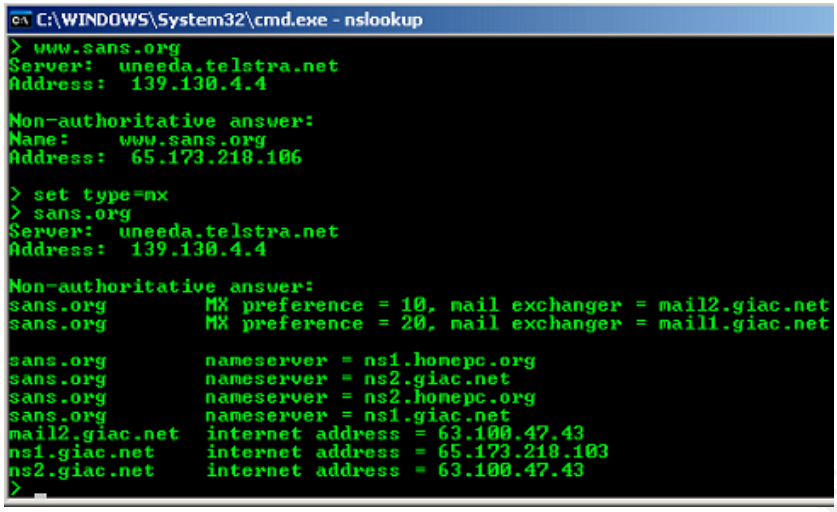
Reference:

Experience, common knowledge, Chirillo (2001), Klevinsky (2002), McLure (2001)

Tools, techniques and testing procedure:

Standard name service query tools built into operating systems. These tools include, nslookup (both Windows and Linux), dig and host (*nix).

Example use of nslookup:



Some Useful commands:

Invoke interactive command shell – **nslookup**

Set query type for mail servers – **set type=mx**

To identify authoritative name server – **set type=soa**

The query is issued as a domain name, fully qualified domain name (fqdn) or in the case of a reverse lookup (matching an IP address to a fqdn) as an IP address.

Example of valid queries: sans.org, ns2.giac.net 65.173.218.103

Evaluation of Test Value to Methodology:

At this point the auditor is identifying some key information assets that are / may be critical to the functioning of the organisation. Some IP addresses are mapped to host names at this stage, and other key information such as the domain mail suffix, start of authority for the Internet domain is established.

Several other steps following this step are reliant on the information found in this step.

Checklist Item 8: Identify system information inadvertently posted on the Internet by IT and Systems Support Staff

Vulnerability Tested:

Systems Information Leakage

Reference:

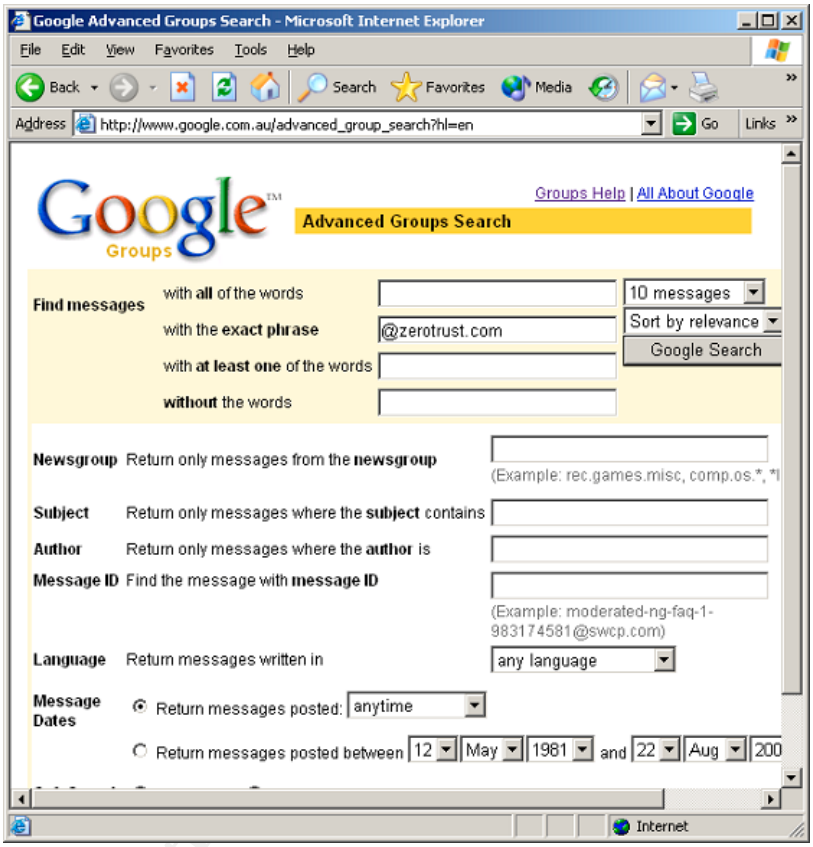
Chirillo (2001), Klevinsky (2002), McLure (2001)

Tools, techniques and testing procedure:

Using Internet search engines, search news groups and mailing lists, for any systems information posted by IT and Systems support staff. Domain mail

suffix identified during the preceding step is used for building the search query.

It is possible to use search engines such as google, or altavista to search the entire web, or if the technologies used within the organisation is used is identified, to search publicly available vendor or independent support forums for such postings. An example of a search engine used to search news groups:



Evaluation of Test Value to Methodology:

Posting system information in public forums exposes technologies and the infrastructure deployed within the organisation. Attackers search for this information. Making this information available eases the attackers task, as based on this information s/he can identify what exploits and scripts to use in the attack. It may even attract attackers looking for easy targets to attack.

An auditor needs to identify such exposures that make an organisation and its information assets vulnerable.

Checklist Item 9: Identify system information inadvertently posted by non-IT personnel and processes.

Vulnerability Tested: System information leakage through position descriptions, news releases, partnership announcements etc.

Reference: Experience, Common knowledge
Tools, techniques and testing procedure: As above checklist item 8. Search engines and employment and recruitment web sites search facilities are the mail tools used.
Evaluation of Test Value to Methodology: As Above.

Checklist Item 10: Check if name servers allow dns zone transfers

Vulnerabilities Tested: System information leakage. Exposure to several known Denial of Service (DoS) vulnerabilities

Reference: Common vulnerabilities and exposures database located at **http://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=zone+transfer**

Tools, techniques and testing procedure:

Standard nslookup tool available with the operating system.

To execute a domain transfer for zerotrust.com, at the command prompt on a Microsoft Windows computer:

Enter interactive nslookup shell – **c:\nslookup**

Execute domain transfer query – **ls –d zerotrust.com**

To redirect the output from the query to a file named zerotrust_domain.txt – **ls –d zerotrust.com > zerotrust_domain.txt**

If zone transfer is allowed, a full listing of the hosts in

Previously discussed Windows based tool Sam Spade has the capability to execute a domain name transfer.

Evaluation of Test Value to Methodology

The test is for several known, common vulnerabilities.

Identifying the hosts within the domain enables the auditor (as well as any malicious attacker) to develop a map of the domain.

The dns zone, may also reveal information about hosts otherwise are not visible to the Internet.

Checklist Item 11: Identify live systems on the network

Vulnerability Tested: Incorrectly configured firewall rule set allowing various types of ICMP packets through to the internal network.

Reference: Common Knowledge, best practices, personal experience

Tools, techniques and testing procedure:

1. The ping command built in to operating systems with various options set. Microsoft Windows command ping.exe utility has fewer user configurable options compared to *nix based ping.

2. The Open source tool nmap with various options set. Nmap can be used on both *nix and Microsoft platforms, and has many more configurable options

than the built in ping command. These options include using TCP pings instead of standard ICMP pings as well as more flexibility in defining host ranges.

3. SING (Send ICMP Nasty Garbage) by Alfredo Andres. Sing is useful in testing various ICMP types and can be used to send ICMP types that are not typically blocked by firewalls.

Some firewalls block incoming traffic when a ping sweep is detected therefore it may be necessary to specify a longer interval of time between the packets.

It may be necessary run a TCP ping with different options in case ICMP pings are filtered out at the firewall.

Examples

A ping scan from a Linux computer:

ping –i 30 –T hostname

In this example ping requests are sent to the hosts specified by the host list at an interval of 30 seconds apart with the ICMP time stamp option set.

A ping scan using nmap:

nmap –PM –T sneaky host-list

In this example the option for ICMP address mask has been set with the flag –PM. The host list can be specified as a network address and mask such as 192.168.1.0/24, or a range of IP addresses such as 192.168.1.1-254. Even wild cards are allowed; the destination 192.168.1.* would ping sweep the 192.168.1.0/24 subnet.

The option –T sneaky tells nmap to wait for an interval of 15 seconds between packets

For the full list of options, please refer to the nmap manual pages.

Standard nmap options are listed in Appendix 1.

Using SING to get the ICMP address mask

sing –mask host-address

Output of **sing –h** (help)

```
Usage: SING [-RnvqQOGBU] [-c count] [-T wait] [-p pattern] [-s
garbagesize]
       [-t ttl] [-TOS tos] [-F bytes] [-i interface] [-S spoof addr]
[-L file]
       [-MAC hw_addr] [type] host
Type:
  -echo    Echo Request (default).   -reply  Echo Reply
  -du      Destination Unreach.      -info   Information Request
  -mask    Address Mask Request.     -param  Parameter Problem
  -rta     Router Advertisement      -rts    Router Solicitation
```

```
  -red     Redirect                    -sq     Source Quench
  -tstamp  Timestamp                   -tx     Time Exceeded
  -h       This help screen            -V      Program version
  -v       Verbose mode on

Host:
   host                           Sending to a host.
   router1%router2%router3%host   Sending with Strict Source Routing.
   router1@router2@router3@host   Sending with Loose Source Routing.
```

Evaluation of Test Value to Methodology:

When deploying a firewall, it is important to filter out ICMP packets as ICMP can be used to identify certain types of systems information, such as system time, IP address mask. ICMP replies also indicate the systems that are alive within the network.

Making this information available to unauthorised persons outside the network is exposing the network to further attacks, and therefore is a vulnerability. Therefore This test is an integral part of any external vulnerability assessment.

Identifying the live and visible systems also narrows down the scans and probes that are conducted in the following steps. With the live hosted identified, the auditor (or an attacker) needs to scan for applications and open ports only on those hosts identified during this stage of the test.

Checklist Item 12: Identify applications and listening ports.

Vulnerability Tested: Services not required for business functionality are not open and exposed to the Internet.

The more services exposed to the Internet, the more the threat of a vulnerability been discovered and exploited in one of those services. Therefore it is a good practice to ensure all services not required to be open to the Internet either be shutdown, or filtered out at the firewall.

Reference: Common Knowledge, Best practices, Experience, Manual pages fro nmap, Chirillo (2001), Klevinsky (2002), McLure (2001)

Tools, techniques and testing procedure:

Key tool used in this step is nmap

It may be possible that all or some ICMP packets are filtered out at the firewall. Therefore it may be necessary to scan the entire IP address block.

Using nmap to identify open services

In its simplest form an nmap scan can be executed using the command – **nmap host-list**. This command pings the host-list and carries out a TCP syn scan.

However, it may be necessary to set various scanning and timing options to ensure that the packets do not get rejected at firewalls, or the assessor's IP address does not get blocked out.

Here's an example of a more specific nmap scan.

nmap –sF –T paranoid –P0 -O –p 1-10000 host-list

nmap executes a TCP stealth FIN scan (-sF) using an interval of 5 minutes between packets (-T paranoid) on port s 1-10000 (-p 1-10000) against the host-list. –P0 tells nmap not to ping the hosts, and –O indicates an operating system fingerprinting should be attempted.

Evaluation of Test Value to Methodology:

This test Identifies applications and services open to the Internet.

The result of this test is integral to identifying any vulnerabilities that are exposed to the Internet.

Checklist Item 13: Identify software and versions used on the systems through application banners and headers.

Vulnerability Tested:

Information leakage through application banners and headers.

Reference:

Chirillo (2001), Klevinsky (2002), McLure (2001)

Tools, techniques and testing procedure:

The tools used in this step are telnet, netcat and the scanline (sl.exe) utility from Foundstone inc.

Manual checking for banners can be done by connecting to ports identified as in listening mode in the previous step. Both netcat and telnet can be used to connect to a listening port and display the remote application's banner and headers.

Using netcat to capture the banner of a web application running on port 80:

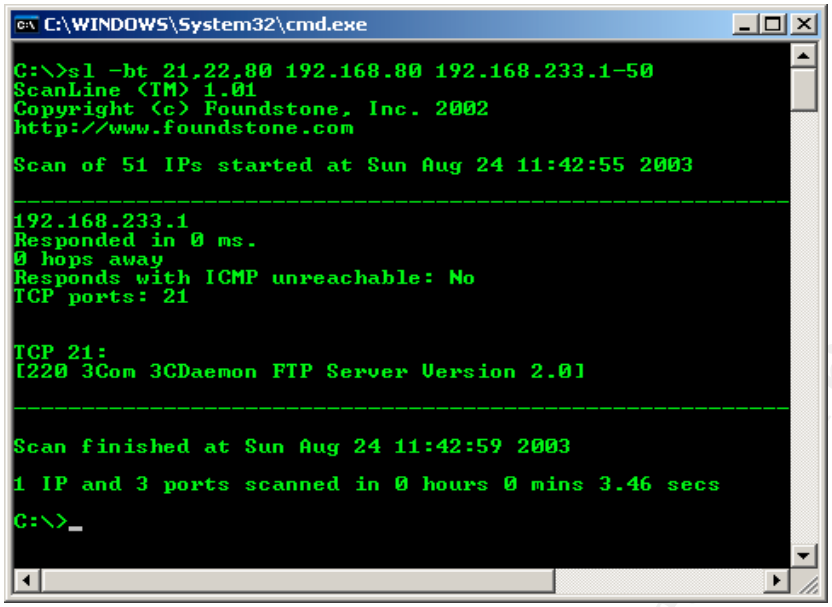
```
[root@cobra root]# nc 192.168.233.132 80
get
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<HTML><HEAD>
<TITLE>501 Method Not Implemented</TITLE>
</HEAD><BODY>
<H1>Method Not Implemented</H1>
get to /index.html not supported.<P>
Invalid method in request get<P>
<HR>
<ADDRESS>Apache/1.3.28 Server at w2ksrv.zerotrust.com Port 80
</BODY></HTML>
[root@cobra root]# _
```

Scanline

Scanline is a utility that runs on windows. IT automates the capturing of banners, and also provides many other scanning options making it a versatile utility in testing open applications and servers.

Following is an example of using scanline for capturing banners.

The option –b captures banners, -t followed by 21,22,80 tells it to test TCP ports 21, 22 and 80, and the 192.168.233.1-50 indicates the range of hosts to scan.



Please refer Appendix 1 for more options that can be used with scanline.

Evaluation of Test Value to Methodology:

Identifying the applications and their versions allows an attacker to search for any known vulnerabilities and exploits on those applications. Therefore banners displayed expose the applications and the servers on which they run to potential attackers, poses a serious security risk.

The assessor needs to identify any such exposure and provide advise to the organisation on eliminating these exposures.

Checklist Item 14: Identify software and versions used on the systems through application logon screens.

Vulnerability Tested:

Information leakage through application logon screens.

Reference:

Chirillo (2001), Klevinsky (2002), McLure (2001)

Tools, techniques and testing procedure:

Connect to application logon screens using appropriate logon applications. Telnet, SSH clients and web browsers are some of the more commonly used logon mechanisms. The auditor may also have to use proprietary client tools to connect to these applications.

Evaluation of Test Value to Methodology:

As above.

The assessor needs to identify any such exposure and provide advise to the organisation on eliminating these exposures.

3.4 Identify system vulnerabilities

Checklist Item 15: Identify any insecure logon mechanisms used in applications exposed to the Internet.

Vulnerability Tested:

Use of clear text passwords and other insecure communications channels.

Reference:

Common Knowledge, experience, best practices

Tools, techniques and testing procedure

Standard application logon mechanisms, such as telnet, web browsers, rsh clients, Virtual Network Computing (VNC) clients etc.

Some of these services can be identified using the results of previous checklist items 8, 9 and 10.

For example nmap reporting port 23 as open on a particular host indicates that the host may be running a telnet server on that port.

However, the assessor needs to verify that this is indeed a telnet server by connecting to the port. Sometimes the service may be required for business functionality, for example legacy applications that do not have secure communication mechanisms built in.

Evaluation of Test Value to Methodology:

Passwords and other sensitive information sent in unencrypted clear text are susceptible to sniffing attacks. This is a system vulnerability that has to be identified.

Even if the application is required for business functionality, the auditor has to flag this as a vulnerability and bring the ramifications of continuing to use this application to the attention of the organisation. It will be up to the organisation to accept or mitigate the risk.

Therefore this is an important test to conduct in an external vulnerability assessment.

Checklist item 16: Check web, mail and other servers exposed to the Internet to ensure correct patch levels are maintained.

Vulnerability Tested: Servers and applications that are exposed to the Internet do not run versions of software and operating systems with known vulnerabilities

Reference:

The Open Web Application Security Project, CERT 2001.

Tools, techniques and testing procedure:

Examine banners collected in step 9 for software and operating system versions if indicated.

This may produce false positives, especially in the case of some software that may be patched, but the original banner is not changed in the process.

At this stage it may also be necessary to run an automated vulnerability scanning tool to identify systems vulnerabilities on the systems exposed to the Internet.

There are many automated scanners available, both from commercial and open source segments of the market.

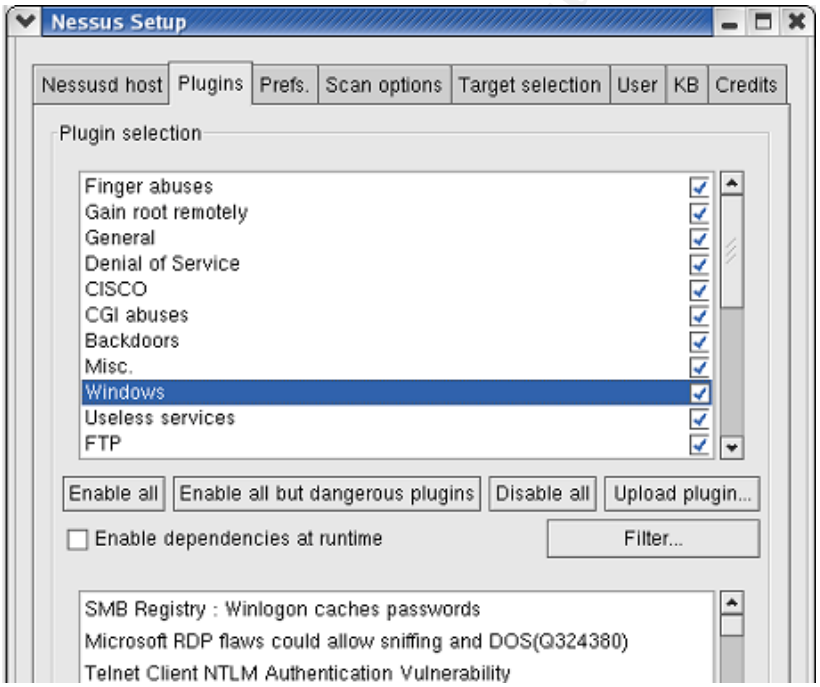
Nessus vulnerability scanner is an open source automated scanner available for the *nix platforms. Nessus uses a client server architecture, and a GUI front end to the client provides easy configuration.

It has its own scripting language to allow auditors to write their own tests if required.

To run nessus server execute **nessusd –D** at the shell.

To start the Nessus client execute **nessus &** at the shell.

A screen shot of the nessus GUI is displayed here.



Evaluation of Test Value to Methodology:

If the critical servers are not up to date with the patches and bug fixes, they are exposed to the threat of attackers exploiting those vulnerabilities.

Identifying systems not patched with known security and bug fixes is an important component of any vulnerability assessment.

Consider the following output from a nessus scan:

Vulnerability domain (53/tcp)

The remote BIND 9 server, according to its version number, is vulnerable to a buffer overflow which may allow an attacker to gain a shell on this host or to disable this server.

Solution : upgrade to bind 9.2.2 or downgrade to the 8.x series
See also : http://www.isc.org/products/BIND/bind9.html
Risk factor : High
Nessus ID : 11318

According to this result it is possible for an attacker to gain a shell using a known buffer overflow exploit. The scan result therefore warrants further investigation of the installed version of BIND, and to carry out remedial action if required.

What the assessor sees through the nessus scan is also what an attacker sees. This type of information is extremely valuable to the attacker as then s/he can focus on finding or creating an exploit to gain a shell on the system, and fully compromising it.

The test conducted under this checklist item, and the result is critical to the proposed methodology as known and potential vulnerabilities on the systems are discovered in this stage of testing.

Checklist Item 17: Identify unsecured web pages containing sensitive information.

Vulnerability Tested:

Sensitive corporate information is transmitted over the Internet encrypted.

Reference:

The Open Web Application Security Project

Tools, techniques and testing procedure:

Using a web browser navigate the web site. Check for logon screens that do not use Secure Socket Layer (SSL) or Transport Layer Security (TLS) encryption. Check for authenticated user areas that may not use any form of encryption.

Evaluation of Test Value to Methodology:

Not encrypting sensitive information transmitted over the web may result in potential attackers sniffing, and possibly altering sensitive information. The information transmitted could include passwords, as well as other corporate information.

This test identifies the potential vulnerability of sensitive corporate data being stolen, or modified in transit.

Checklist Item 18: Identify backend databases that are visible to the Internet

Vulnerability Tested:

Backend databases exposed to the Internet.

Back end databases should not be exposed to the Internet as they run the risk of being attacked and the data contained within being modified or stolen.

Reference:

Best practices, Common Knowledge

Tools, techniques and testing procedure:

The results obtained in Checklist Item 11, port scanning with nmap, could be the basis for this test.

Some of the commonly used database servers can be identified by checking for the following open ports.

TCP port 1433 – Microsoft SQL server
TCP port 3306 – mySQL
TCP port 1521 – Oracle

Evaluation of Test Value to Methodology:

SQL databases are used to store corporate information, and sometimes are used as backend to web based applications. These databases may contain sensitive information such as client information, credit card numbers, usernames and passwords etc.

Such databases should not be visible or accessible from the Internet.

This test identifies any such database is not exposed to the Internet and therefore is integral to an external vulnerability assessment.

Checklist Item 19: Test Applications exposed to the web to ensure that they are not susceptible to SQL injection and other input validation attacks.

Vulnerability Tested:

Poor input validation mechanisms open up applications to SQL injection and other input validation attacks.

By using carefully crafted SQL statements, attackers could gain information about the databases at the back end of the application as well as steal the data or modify the data contained within the database.

Reference:

Open Web application Security Project, SPI Labs SQL Injection Whitepaper

Tools, techniques and testing procedure:

The key testing process relies on Standard SQL statements.

The methodology includes suppling various SQL statements to form fields on the web applications, as well as trying out various inputs such as large number of special characters, which the application may not expect as input.

The out put error messages are then used to construct information about the database. The database may also output data contained within as part of the error message.

Some of the information revealed through error messages may contain information about the application flow, operating system and database server software information including version numbers and patch levels, physical file locations and IP addresses of other trusted servers.

Evaluation of Test Value to Methodology:

This test identifies weaknesses in applications that can be exploited to extract data without proper authorisation.

Error messages that expose information provide an attacker sufficient information to launch an attack against the application and back end servers.

Therefore it is important that such vulnerabilities be identified and addressed.

Checklist Item 20: Check web application source code accessible from the Internet for any hidden fields and comments that reveal internal network, application or server information.

Vulnerability Tested:

Weak programming practices that reveal important information about the systems, databases, applications and the network exposed.

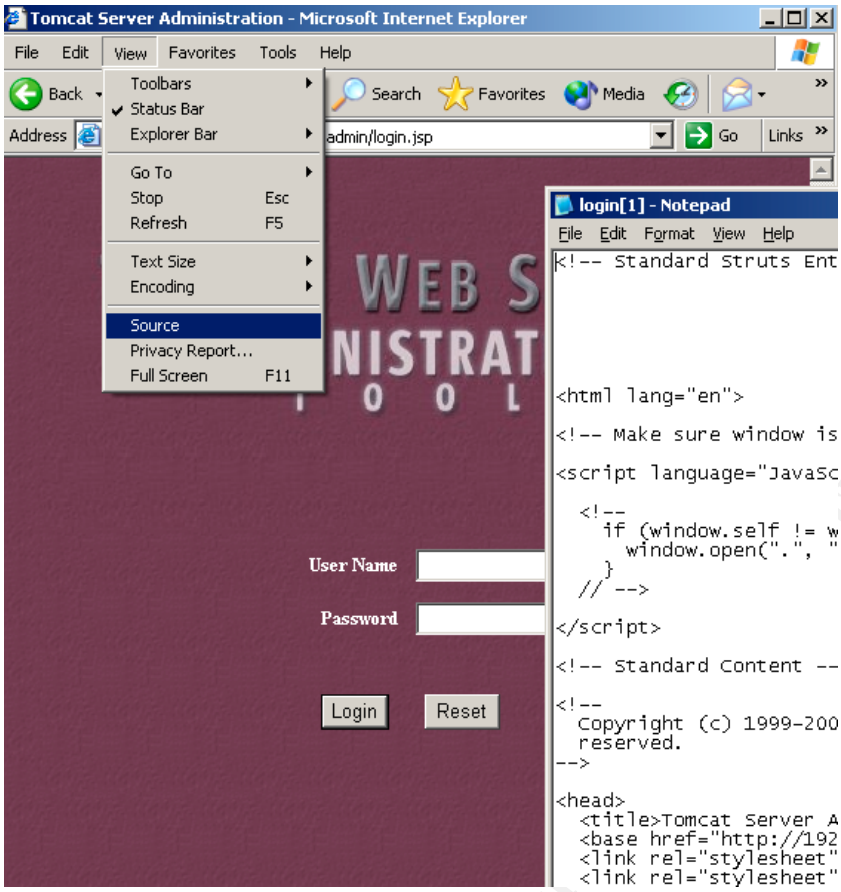
Reference:

Open web application security project, best practices.

Tools, techniques and testing procedure:

Most web browsers come with a menu option to view the source code of the web page being accessed.

Example - viewing source code of a web page using Microsoft Internet Explorer.



Another area to test for this vulnerability is through help facilities provided for the application.

Evaluation of Test Value to Methodology:

Comments and hidden fields may reveal information that an attacker could use to map the network, or use in other attacks such as denial of service. It is important that this leaking of information be identifies and corrected.

3.5 Test Identified Vulnerabilities (if applicable)

The above checklist, although not exhaustive, provides an auditor with a fairly descriptive picture of the system s/he is auditing. It is possible that a need to expand the checklist arises based on the findings of the audits conducted. It is also possible that the security of the system s/he is auditing is governed and managed by security policies and secure computing practices. In which case the findings may indicate that the system has minimal vulnerabilities.

However, if the audit uncovers vulnerabilities that expose the system to exploitation and compromise, the next step would be to test their 'exploitability'. This stage of an external vulnerability assessment is commonly known as penetration testing and involves running known exploits, the end result being denial of services or system compromises on some systems. Some organisations do wish to have these identified vulnerabilities tested,

while others are satisfied with discovering and addressing the vulnerabilities that were discovered in the previous stage.

If further testing of vulnerabilities is to be conducted, it is important that

a.) the extent of the testing is defined within the scope of the assessment

b.) both parties, i.e. The organisation and the auditor are aware that the testing may cause disruption or loss of services,

c.) both parties agree upon timing of the tests.

Checkpoint 2

Reconfirm that systems identified with vulnerabilities do belong to the organisation, and that the exploitation of the vulnerabilities at this stage is acceptable to the organisation.

It is not possible to compile a prescriptive checklist for this component of the audit as the basis for the tests that would be conducted would be the results of the tests conducted in the previous steps. Most tests at this stage would replicate activities of malicious attackers, where known or crafted attacks against the identified vulnerabilities.

Checklist Item 21: Identify systems with passwords that have been left unchanged after installation.

Vulnerability Tested:

Systems operating with default passwords set by the vendor or the manufacturer leave themselves open to password guessing attacks that may result in total compromise of the system.

Reference:

Best Practices

Tools, techniques and testing procedure:

Manual attempts to login, use of Brute force login tools such as Brutus (discussed under Checklist Item 23)

Evaluation of Test Value to Methodology:

Since vendor default passwords are common knowledge, it is important to change these passwords immediately upon install. Leaving these passwords unchanged leaves the systems open to password guessing attacks. Therefore it is important to test for such unchanged passwords on systems during a vulnerability assessment.

Checklist Item 22: Identify systems and devices with operating with default manufacturer set community strings.

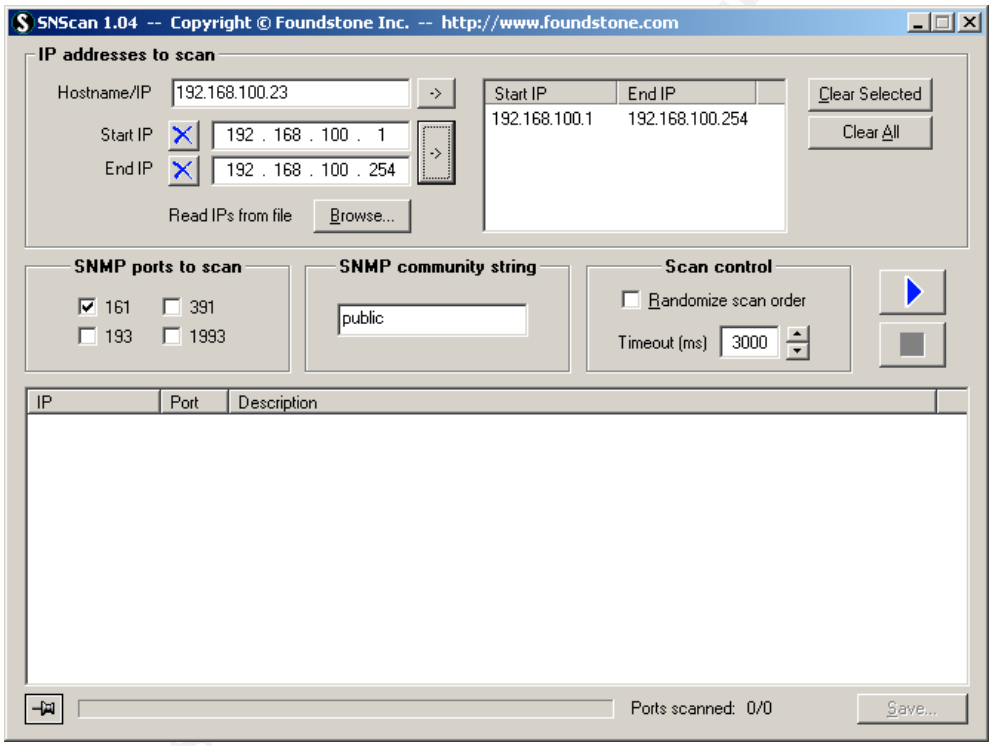
Vulnerability Tested:

Systems and devices operating with default passwords set by the vendor or the manufacturer leave themselves open to SNMP community string guessing attacks that result in a total compromise of the system.

Tools, techniques and testing procedure:

Tool used in identifying default SNMP Community strings is SNScan from Foundstone Institute. SNScan is a Win32 GUI based utility that is Intuitive to use. The user can specify an IP address range to scan, and also specify community strings to use against the tested device. Vendor documentation usually contains information default community strings that are used in their products, and there are may lists on the Internet that contain a compilation of default SNMP community strings used by various vendors.

Below is an example of an SNScan session set up and ready to run.



Evaluation of Test Value to Methodology:

As with vendor default passwords, vendor default SNMP community strings are publicly available information. Therefore it is possible for an attacker to identify the device and then attempt community string guessing attacks. Once an attacker identifies read community string s/he has access to the device configuration, routing tables and many other pieces of useful information including network setup parameters. If the attacker identifies the read/write community string s/he can modify configuration and cause denial of service attacks.

Therefore it is important to ensure that SNMP enabled devices are not vulnerable to community string guessing attacks.

Checklist Item 23: Identify systems and applications with weak or null passwords.

Vulnerability Tested:

Systems with weak or null passwords.

Systems with weak or null passwords leave themselves open to compromise by malicious attackers. Attackers may manually guess passwords or use a password-guessing program to guess passwords by automatic and repetitious logon attempts using password lists provided by the attacker.

Reference:

Best Practices

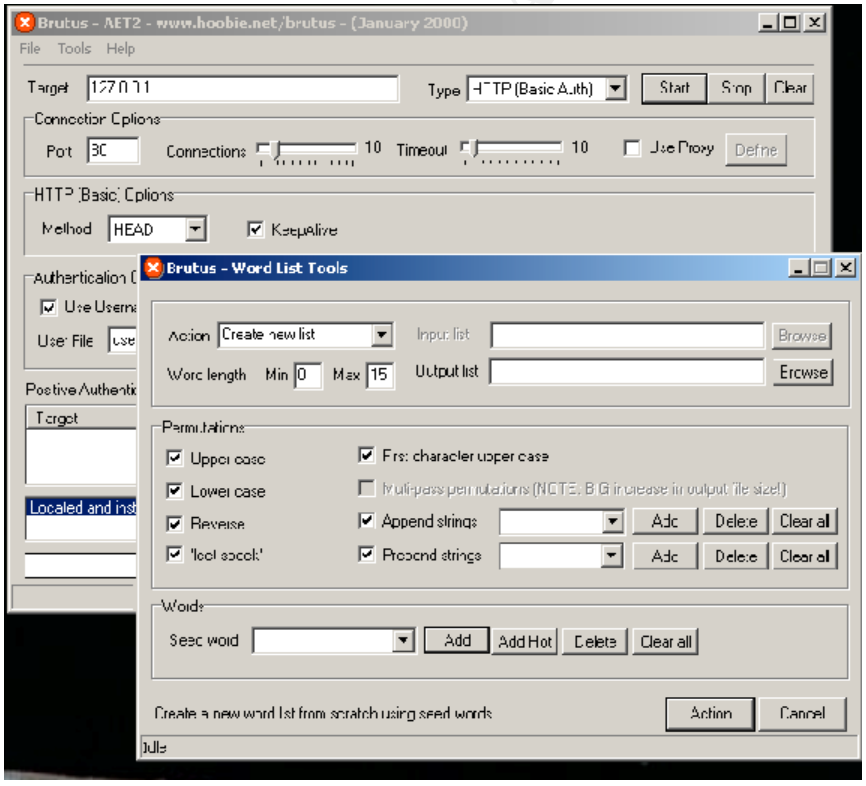
Tools, techniques and testing procedure:

Repetitious, manual logon attempts at guessing the password.

Sniffing for passwords if the network is accessible.

Using automated password crackers such as LC4 or "John the Ripper."

Using an automated password-guessing program such as Brutus to generate password lists and attempt logons.



Apart from above mentioned password guessing attacks actual attacker exploits could be used to test the systems ability to withstand such attacks. As

mentioned before the auditor needs to make the organisation aware of the possible results of the attack and reconfirm that the use of these attacks is agreeable to the organisation.

Evaluation of Test Value to Methodology:

3.6 Operational Vulnerabilities

Checklist Item 24: Intrusion Identification capability of the organisation.

Vulnerability Tested:

Identify the limitations of the intrusion detection system, or the lack of such intrusion detection mechanisms, and if present they are monitored.

Reference:

Best practices.

Tools Techniques and testing procedure:

The above tests apart from those conducted during the reconnaissance stage should have triggered many audit events and alerts on the firewalls and any intrusion detections systems. If the organisation does not identify these events and alerts, attacks and systems compromises may go unnoticed for a long period of time, or at worst may never be noticed.

Evaluation of Test Value to Methodology.

An intrusion detection system detects any attempt at port scans, probes and other forms of intrusions and anomalous behaviour of the system. The intrusion detection system does not necessarily be an expensive appliance, or software. Most operating systems and devices come with systems logging and alerting capabilities, and it is important that these features be turned on and monitored. There are also several open source intrusion detection software that can run on inexpensive hardware.

Without any intrusion detection mechanism the organisation does not have any means to identify any attempted or successful intrusions and security breaches.

Some organisations do have the intrusion detection systems installed, but their staff fail to monitor the alerts generated by these systems. Not monitoring the intrusion detection systems leaves the systems vulnerable to undetected attacks as much as not having an intrusion detection system.

Therefore it is important in any perimeter audit to ensure that these systems are functional and are monitored.

Checklist Item 25: Incident response readiness of the organisation

Vulnerability:

Not having an incident response policy and guidelines exposes the organisation to not being able to identify and contain to a security incident.

Reference:

Schweitzer (2003)

Tools Techniques and testing procedure:

Any or all of the intrusive tests conducted above should have generated sufficient log entries to alert the IT staff that some attempts at intrusion to their systems are being made. Tools such as nessus, and nmap run with default settings generate a lot of network traffic that simulate network intrusions.

It is also possible to run tools such as nmap in a more aggressive mode, where an excessive amount of traffic is generated.

Evaluation of test value to methodology:

An incident response policy outlines the action to carry out in the event of a security incident. The lack of incident response policy and procedures leaves the organisation unprepared for security breaches. In the Introduction to his book, Incident Response Schweitzer states:

The protection of critical IT resources requires not only adopting reasonable precautions for securing these systems and networks, but also the ability to respond quickly and efficiently when systems and network security defences have been breached (Schweitzer, 2003:xx)

If the organisation is unprepared for a systems security breach, and the staff are not trained and informed of what to do in such a situation to contain damage, it is quite likely that organisation suffers substantial loss. For example, if the organisation's database server has been identified as compromised, but the staff are unaware as to what to do to minimise damages, the intruder may have time to cause the intended damage as well as to cover his/her tracks. And an untrained Systems Administrator, in an attempt to minimise damage, may trigger off further incidents, or destroy the evidence of the intrusion, thereby destroying all chances of identifying the source of the intrusion and the vulnerability that allowed it.

In this context, the lack of incident policy leaves the organisation vulnerable to extended damage, and therefore there is a need to test the organisation's incident response readiness in the event of a security breach.

4 Assignment 3 – Sample Audit

4.1 System to be Assessed

Zero Trust Inc. (ZTI) is an accounting firm that has seen rapid growth in the past few years. ZTI has maintained a web presence for several years, and relies heavily on electronic transactions in its day-to-day operations. ZTI's customers enjoy the convenience of an Extranet provided by ZTI to access their transaction and to submit documents and other information. This on line infrastructure is hosted in-house at ZTI.

With the increasing market share, and their elevated profile, ZTI's Management needed assurance that their information assets were secure from malicious activity that could cost them financially and otherwise. The IT department was entrusted with the task of securing the systems from such malicious activity and intrusions. However, the IT Department did not have the necessary skills to conduct a complete system and network Audit. Therefore they employed the services of Hackville Security Consultants (HSC) to audit their systems for them. This assignment contains the external perimeter audit component of the systems and network audit conducted by HSC.

The perimeter and external auditing of ZTI was conducted as a zero knowledge vulnerability assessment. The consulting auditor was provided with the minimum information about the systems within ZTI. The information provided to the auditor was limited to the name of the company, and that the company had a web presence. It was expected that the auditor attempt to identify the systems as part of the audit, similar to how an unauthorized individual with malicious intent identifies the vulnerabilities of a system he/she intends to compromise.

In summary, the auditor's tasks included:

- Identifying the systems within ZTI network
- Mapping the topology of the network as it is visible from the Internet.
- Conducting an assessment of the system, identifying vulnerabilities exposed to the Internet.
- Recommending processes and technologies to mitigate those identified vulnerabilities.

4.2 Conduct the Audit

Following is a sample of the tests conducted. Several tools mentioned in the checklist have been used in conducting the audit. However, it is important to note that the use of tools alone and the output they produce do not comprise the audit. The critical component is the analysis of the result. The evaluator needs to test the output of the tools for false positives and negatives, and make many deductions and assumptions based on the output. An experienced evaluator can with a simple test result deduce a substantial amount of information, and thereby decide if further tests should be conducted or not.

The following samples of the audit have revealed a substantial amount of information to the assessor, and have been used in developing the sample report, which comprises the final component of the practical. Therefore it is recommended that the audit sample be read in conjunction with the follow up report to gain an understanding of the assesors thought process in identifying vulnerabilities based on the sample output of the test results.

4.3 Audit Samples

whois Query

Check list Item 5 – Identify information related to Internet Domain Registration

A whois query returns some information about the organisation's systems. This information, although publicly available is not a vulnerability. Information included here is typically a contact for the organisation, domain delegation information and some IP address information. A whois query can be run on Domain Name Registration authority websites or using software such as SamSpade.

```
04/21/03 23:42:18 whois www.zerotrust.com.au
.au is a domain of Australia
(international dialing code 61)
Searches for .au can be run at http://www.aunic.net/cgi-
bin/whois.aunic

whois -h whois.aunic.net zerotrust.com.au ...

% Copyright 2001 auDA.  Terms of Use at
http://www.aunic.net/copyright.html

The object shown below is NOT in the AUNIC database.
It has been obtained by querying a remote server:
(whois.ausregistry.net.au) at port 43.
```

```
To see the object stored in the AUNIC database
use the -R flag in your query.

Domain ROID:              xxxxxx-AR
Domain Name:              zerotrust.com.au
Last Modified:            20-Nov-2002 02:52:16 UTC
Registrar ID:             xxxxxx-AR
Registrar Name:                xxxxxxx IT
Status:                   ok

Registrant:               Zero Trust Pty Ltd
Registrant ID:            OTHER xxx xxx xxxx

Registrant ROID:          C0232081-AR
Registrant Contact Name: THE MANAGER
Registrant Email:         manager@zerotrust.com.au

Tech ID:                  C0232083-AR
Tech Name:                Joe Bloke
Tech Email:               jbloke@zerotrust.com.au

Name Server:              ns1.zerotrust.com.au
Name Server IP:           192.168.100.1
Name Server:              ns2.zerotrust.com.au
Name Server IP:           192.168.101.2

%%% End of referred query result
```

DNS Zone Transfer

Checklist Item 7 – Identify critical systems such as mail, web and other servers.

Checklist Item 10: Check if name servers allow DNS Zone transfers.

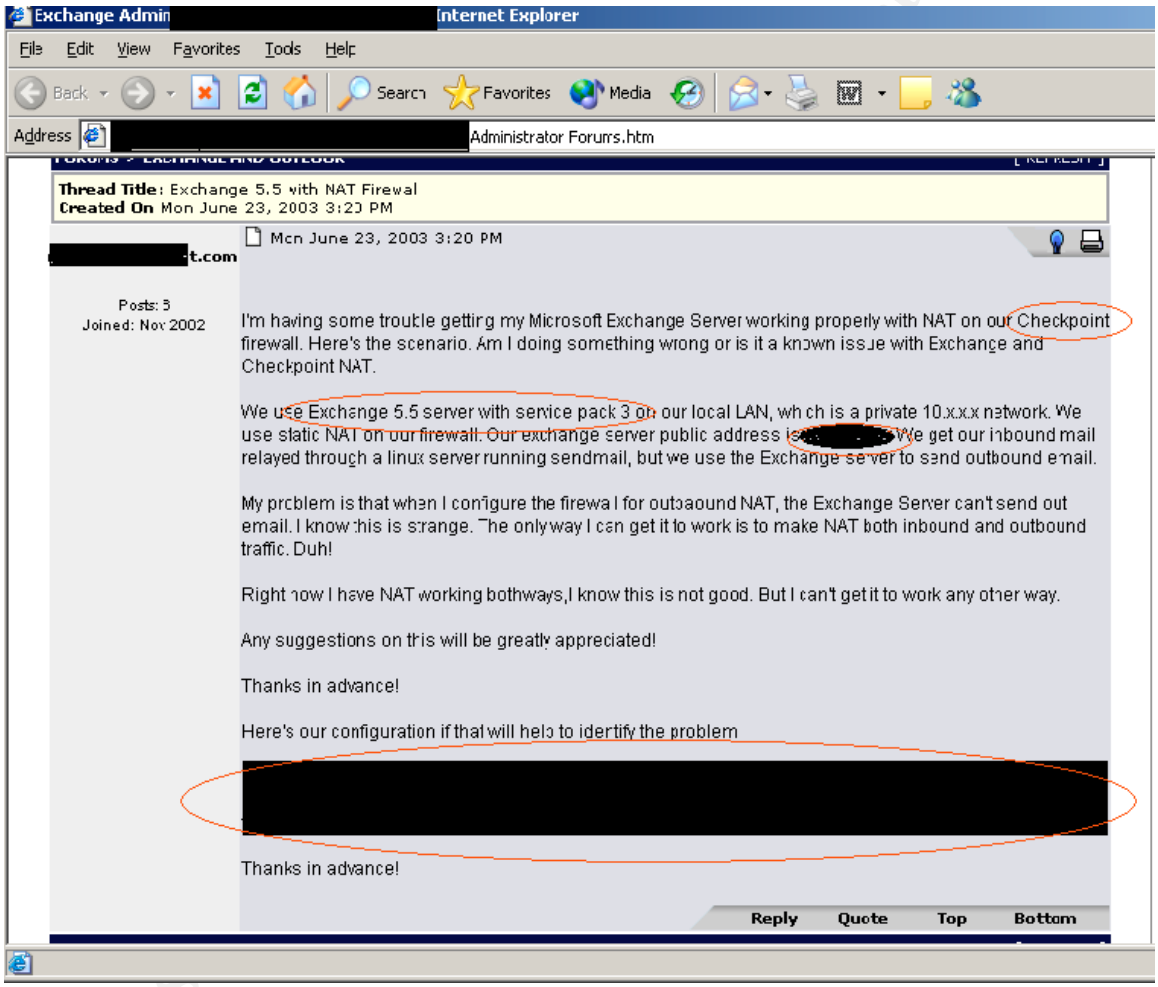
A zone transfer occurs when a name server is queried for a full domain listing, and the name server responses with the listing of the entire domain. This is a serious vulnerability as the attacker then has the information about all the key servers within the organisation.

```
03/24/03 23:52:05 Zone transfer zerotrust.com.au@ns1.zerotrust.com.au
Zone transfer zerotrust.com.au@ns1.zerotrust.com.au (192.168.100.1)
...
 Query for zerotrust.com.au type=252 class=1
  zerotrust.com.au SOA (Zone of Authority)
        Primary NS: ns1.zerotrust.com.au
        Responsible person: jbloke@zerotrust.com.au
        serial:3000000034
        refresh:10800s (3 hours)
        retry:3600s (60 minutes)
        expire:3600000s (410 days)
        minimum-ttl:86400s (24 hours)
  zerotrust.com.au NS (Nameserver) ns1.zerotrust.com.au
  zerotrust.com.au NS (Nameserver) ns2.zerotrust.com.au
  zerotrust.com.au MX (Mail Exchanger) Priority: 10
mail.zerotrust.com.au
  zerotrust.com.au A (Address) 192.168.100.1
  accounting.zerotrust.com.au CNAME (Canonical Name)
internaldb.zerotrust.com.au
  intranet.zerotrust.com.au CNAME (Canonical Name)
internalweb.zerotrust.com.au
  ns1.zerotrust.com.au A (Address) 192.168.100.1
  ns2.zerotrust.com.au A (Address) 192.168.100.2
  gateway.zerotrust.com.au A (Address) 192.168.100.254
  customer.zerotrust.com.au A (Address) 192.168.100.3
  ftp.zerotrust.com.au CNAME (Canonical Name) www.zerotrust.com.au
  owa.zerotrust.com.au CNAME (Canonical Name)
exchange.zerotrust.com.au
  internaldb.zerotrust.com.au A (Address) 192.168.100.5
  mail.zerotrust.com.au A (Address) 192.168.100.6
  exchange.zerotrust.com.au A (Address) 192.168.100.4
  customer.zerotrust.com.au A (Address) 192.168.100.7
  extranet.zerotrust.com.au CNAME (Canonical Name)
customer.zerotrust.com.au
  internalweb.zerotrust.com.au A (Address) 192.168.100.8
  www.zerotrust.com.au A (Address) 192.168.100.10
  zerotrust.com.au SOA (Zone of Authority)
        Primary NS: ns1.zerotrust.com.au
        Responsible person: jbloke@zerotrust.com.au
        serial:3000000034i
        refresh:10800s (3 hours)
        retry:3600s (60 minutes)
        expire:3600000s (410 days)
        minimum-ttl:86400s (24 hours)
```

System information exposed on the Internet

Checklist Item 8 – Identify system information inadvertently posted on the Internet by IT and Systems Support Staff

Using a search on www.google.com, the auditor identified several postings made by the IT staff at ZTI, requesting for help to fix internal system problems. Some of the postings contained information about the systems that should not have been on the postings.



Ping Sweep

Checklist Item 11: Identify live systems on the network

After identifying the server IP addresses, the auditor ran a ping sweep across the entire subnet that the servers resided on. There were two objectives behind the ping sweep. One was to identify which of the listed servers were alive. The other was to see if there were other devices beside the listed one deployed within the network.

The ping sweep also should have alerted any intrusion detection system, if one was deployed.

The tool used for the ping sweep was nmap.
 # nmap –sP –T Sneaky 192.168.1.100-254

```
Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
Host  (192.168.100.1) appears to be up.
Host  (192.168.100.3) appears to be up.
Host  (192.168.100.4) appears to be up.
Host  (192.168.100.5) appears to be up.
Host  (192.168.100.6) appears to be up.
Host  (192.168.100.7) appears to be up.
Host  (192.168.100.8) appears to be up.
Host  (192.168.100.10) appears to be up.
Host  (192.168.100.254) appears to be up.
Nmap run completed -- 254 IP addresses (9 hosts up) scanned in 16002
seconds
```

Port Scan and OS fingerprinting

Checklist Item 12: Identify applications and listening ports

Following is the results of a port scan run against the IP addresses identified as live from the ping sweep. The information gathered through the zone transfer enabled the auditor to narrow the scan to a specific range of IP addresses, thereby saving time spent on the scan.

The auditor ran several scan types including TCP connect scans and SYS Stealth scans. The auditor also ran a scan against the full subnet to test if any intrusion detection system identifies the scan as an intrusion.

The port scan helped the auditor to identify the services running on the servers, and those that accept connections.

The tool used for the port scan is nmap. Several nmap queries were run against the IP Addresses that were identified live within the network. The following results are from a normal nmap scan conducted at the latter end of the audit to test intrusion detection and incident response capabilities of the organisation. Normal scan timing and default ports were used rather Sneaky

or a Paranoid mode scan, as objective was to identify if the organisation could identify the scan as a potential attack.

Commands used:
nmap –sT 192.168.100.0/24 – for the TCP connect scan
nmap –sS 192.168.100.0/24 – for the TCP SYN Stealth scan

```
Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
Host   (192.168.100.0) seems to be a subnet broadcast address
(returned 2 extra pings).  Still scanning it due to ping response
from its own IP.
Initiating Connect() Scan against  (192.168.100.0)
The Connect() Scan took 1 second to scan 1601 ports.
Warning:  OS detection will be MUCH less reliable because we did not
find at least 1 open and 1 closed TCP port
All 1601 scanned ports on  (192.168.100.0) are: closed
Too many fingerprints match this host for me to give an accurate OS
guess
TCP/IP fingerprint:
SInfo(V=3.00%P=i386-redhat-linux-gnu%D=6/23%Time=3EF641E9%O=-1%C=1)
T5(Resp=N)
T6(Resp=N)
T7(Resp=N)
PU(Resp=N)

Host ns1.zerotrust.com.au (192.168.100.1) appears to be up ... good.
Initiating SYN Stealth Scan against ns1.zerotrust.com.au
(192.168.100.1)
 Adding open port 1025/tcp
 Adding open port 111/tcp
 Adding open port 22/tcp
 Adding open port 6000/tcp
 Adding open port 1024/tcp
 Adding open port 631/tcp
 Adding open port 1241/tcp
 Adding open port 25/tcp
 Adding open port 37/tcp
 Adding open port 110/tcp
 Adding open port 80/tcp
 Adding open port 53/tcp
 Adding open port 113/tcp

The SYN Stealth Scan took 3 seconds to scan 1601 ports.
For OSScan assuming that port 22 is open and port 1 is closed and
neither are firewalled
Interesting ports on ns1.zerotrust.com.au (192.168.100.1):
(The 1593 ports scanned but not shown below are in state: closed)
Port       State       Service
22/tcp     open        ssh
25/tcp     open        smtp
37/tcp     open        time
53/tcp     open        domain
80/tcp     open        http
110/tcp    open        pop3
111/tcp    open        sunrpc
113/tcp    open        auth
```

```
631/tcp    open        ipp
1024/tcp   open        kdm
1025/tcp   open        NFS-or-IIS
1241/tcp   open        msg
6000/tcp   open        X11
Remote operating system guess: Linux Kernel 2.4.0 - 2.5.20
Uptime 0.104 days (since Sun Jun 22 13:40:55 2003)
TCP Sequence Prediction: Class=random positive increments
                         Difficulty=5356883 (Good luck!)
IPID Sequence Generation: All zeros

Host  (192.168.100.2) appears to be down, skipping it.
Host  (192.168.100.3) appears to be up ... good.
Initiating Connect() Scan against  (192.168.100.3)
Adding open port 23/tcp
Adding open port 25/tcp
The Connect() Scan took 2 seconds to scan 1601 ports.
For OSScan assuming that port 23 is open and port 1 is closed and
neither are firewalled
Interesting ports on  (192.168.100.3):
(The 1599 ports scanned but not shown below are in state: closed)
Port       State       Service
23/tcp     open        telnet
25/tcp     open        smtp
Remote OS guesses: Cisco 801/1720 running 12.2.8, Cisco IOS 12.2(8)T
TCP Sequence Prediction: Class=truly random
                         Difficulty=9999999 (Good luck!)
IPID Sequence Generation: All zeros

Host exchange.zerotrust.com.au(192.168.100.4) appears to be up ...
good.
Initiating SYN Stealth Scan against 192.168.100.3)
Adding open port 3268/tcp
Adding open port 119/tcp
Adding open port 80/tcp
Adding open port 3389/tcp
Adding open port 389/tcp
Adding open port 53/tcp
Adding open port 10000/tcp
Adding open port 691/tcp
Adding open port 6103/tcp
Adding open port 593/tcp
Adding open port 5800/tcp
Adding open port 5900/tcp
Adding open port 993/tcp
Adding open port 25/tcp
Adding open port 636/tcp
Adding open port 3269/tcp
Adding open port 464/tcp
Adding open port 1026/tcp
Adding open port 6667/tcp
Adding open port 143/tcp
Adding open port 1212/tcp
Adding open port 443/tcp
Adding open port 1030/tcp
Adding open port 110/tcp
Adding open port 6668/tcp
Adding open port 563/tcp
Adding open port 3372/tcp
Adding open port 995/tcp
The SYN Stealth Scan took 3 seconds to scan 1601 ports.
```

```
For OSScan assuming that port 25 is open and port 1 is closed and
neither are firewalled
Interesting ports on exchange.zerotrust.com.au(192.168.100.4):
(The 1569 ports scanned but not shown below are in state: closed)
Port       State       Service
25/tcp     open        smtp
53/tcp     open        domain
80/tcp     open        http
88/tcp     open        kerberos-sec
110/tcp    open        pop-3
119/tcp    open        nntp
135/tcp    open        loc-srv
139/tcp    open        netbios-ssn
143/tcp    open        imap2
389/tcp    open        ldap
443/tcp    open        https
445/tcp    open        microsoft-ds
464/tcp    open        kpasswd5
563/tcp    open        snews
593/tcp    open        http-rpc-epmap
636/tcp    open        ldapssl
691/tcp    open        resvc
993/tcp    open        imaps
995/tcp    open        pop3s
1026/tcp   open        LSA-or-nterm
1030/tcp   open        iad1
1212/tcp   open        lupa
3268/tcp   open        globalcatLDAP
3269/tcp   open        globalcatLDAPssl
3372/tcp   open        msdtc
3389/tcp   open        ms-term-serv
6667/tcp   open        irc
6668/tcp   open        irc
10000/tcp  open        snet-sensor-mgmt
Remote operating system guess: Windows Millennium Edition (Me), Win
2000, or WinXP
TCP Sequence Prediction: Class=random positive increments
                         Difficulty=3592 (Formidable)
IPID Sequence Generation: Incremental

Host internaldb.zerotrust.com.au(192.168.100.5) appears to be up ...
good.
Initiating SYN Stealth Scan against
internaldb.zerotrust.com.au(192.168.100.5)
Adding open port 445/tcp
Adding open port 80/tcp
Adding open port 3389/tcp
Adding open port 1433/tcp
Adding open port 10000/tcp
Adding open port 6103/tcp
Adding open port 5800/tcp
Adding open port 1050/tcp
Adding open port 5900/tcp
Adding open port 135/tcp
Adding open port 6667/tcp
Adding open port 443/tcp
Adding open port 6668/tcp
Adding open port 139/tcp
The SYN Stealth Scan took 2 seconds to scan 1601 ports.
For OSScan assuming that port 80 is open and port 1 is closed and
neither are firewalled
```

```
Interesting ports on internaldb.zerotrust.com.au(192.168.100.5):
(The 1587 ports scanned but not shown below are in state: closed)
Port       State       Service
80/tcp     open        http
135/tcp    open        loc-srv
139/tcp    open        netbios-ssn
443/tcp    open        https
445/tcp    open        microsoft-ds
1050/tcp   open        java-or-OTGfileshare
1433/tcp   open        ms-sql-s
3389/tcp   open        ms-term-serv
5800/tcp   open        vnc-http
5900/tcp   open        vnc
6103/tcp   open        RETS-or-BackupExec
6667/tcp   open        irc
6668/tcp   open        irc
10000/tcp  open        snet-sensor-mgmt
Remote operating system guess: Windows Millennium Edition (Me), Win
2000, or WinXP
TCP Sequence Prediction: Class=random positive increments
                         Difficulty=11037 (Worthy challenge)
IPID Sequence Generation: Incremental


Host mail.zerotrust.com.au(192.168.100.6) appears to be up ... good.
Initiating SYN Stealth Scan against
mail.zerotrust.com.au(192.168.100.5)
Adding open port 111/tcp
Adding open port 22/tcp
Adding open port 25/tcp
The SYN Stealth Scan took 2 seconds to scan 1601 ports.
For OSScan assuming that port 22 is open and port 1 is closed and
neither are firewalled

Interesting ports on mail.zerotrust.com.au (192.168.100.6):
(The 2046 ports scanned but not shown below are in state: closed)
Port       State       Service
22/tcp     open        ssh
111/tcp    open        sunrpc
22/tcp     open        ssh
Remote OS guesses: Linux Kernel 2.4.0 - 2.5.20, Linux 2.4.19-pre4 on
Alpha
Uptime 0.104 days (since Sun Mar 22 13:40:55 2003)
TCP Sequence Prediction: Class=random positive increments
                         Difficulty=5356883 (Good luck!)
IPID Sequence Generation: All zeros

.....................................................

Host  (192.168.100.11) appears to be down, skipping it.
Host  (192.168.100.12) appears to be down, skipping it.

.....................................................

Host  (192.168.100.253) appears to be down, skipping it.
Host  (192.168.100.254) appears to be up ... good.
Initiating Connect() Scan against  (192.168.100.254)
Adding open port 21/tcp
Adding open port 23/tcp
The Connect() Scan took 5 seconds to scan 1601 ports.
```

```
For OSScan assuming that port 21 is open and port 1 is closed and
neither are firewalled
Interesting ports on  (192.168.100.254):
(The 1599 ports scanned but not shown below are in state: closed)
Port       State       Service
21/tcp     open        ftp
23/tcp     open        telnet
Remote operating system guess: Router/Switch/Printer (LanPlex
2500/Cisco Catalyst 5505/CISCO 6509/Trancell Webramp/Xylan Omni
Switch)/Epson Stylus (100BTX-NIC HP Secure Web Console, Sonicwall
firewall appliance 3.3.1)
TCP Sequence Prediction: Class=64K rule
                         Difficulty=1 (Trivial joke)
IPID Sequence Generation: Busy server or unknown class


Host   (192.168.100.255) seems to be a subnet broadcast address
(returned 2 extra pings).  Still scanning it due to ping response
from its own IP.
Initiating Connect() Scan against  (192.168.100.255)
The Connect() Scan took 1 second to scan 1601 ports.
Warning:  OS detection will be MUCH less reliable because we did not
find at least 1 open and 1 closed TCP port
All 1601 scanned ports on  (192.168.100.255) are: closed
Too many fingerprints match this host for me to give an accurate OS
guess
TCP/IP fingerprint:
SInfo(V=3.00%P=i386-redhat-linux-gnu%D=6/23%Time=3EF6427A%O=-1%C=1)
T5(Resp=N)
T6(Resp=N)
T7(Resp=N)
PU(Resp=N)

Nmap run completed -- 256 IP addresses (8 hosts up) scanned in 170
seconds
```

Identify Internal Network Topology

Checklist Item: None

The auditor used the SING utility to gain information about the IP subnet mask used on the gateway router/firewall.

Command Used:

sing –mask –c 192.168.100.254

Result

```
#sing -mask -c 3 192.168.100.254

SINGing to 192.168.100.254 (192.168.100.254): 12 data bytes
12 bytes from 192.168.100.254: seq=0 DF! ttl=255 TOS=0
mask=255.255.255.0
12 bytes from 192.168.100.254: seq=1 DF! ttl=255 TOS=0
mask=255.255.255.0
12 bytes from 192.168.100.254: seq=2 DF! ttl=255 TOS=0
mask=255.255.255.0
```

```
--- 192.168.100.254 sing statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 0.217/0.322/0.457 ms
```

Identifying software and versions from software banners

Checklist Item 13: Identify software and versions used on the systems through application banners and headers

The auditor connected to services identified as open, and collected software version information from the banners displayed upon connecting.

The tool used for collecting banner information is scanline (sl.exe).

Command Used:

sl –bt 1-10000 192.168.100.1-12,254

Result

```
Scan of 13 IPs started at Mon Mar 24 12:37:21 2003

------------------------------------------------------------------
192.168.100.1
Responded in 0 ms.
23 hops away
Responds with ICMP unreachable: No
TCP ports: 25 80 110 443

TCP 22:
[220 ProFTPD 1.2.0pre3 Server ready.]

TCP 25:
[220 ns1.zerotrust.com.au ESMTP Sendmail 8.8.7/8.8.7; Mon, 24 Mar
2003 12:37:21 +1000 ]

TCP 80:
TCP 80:
[HTTP/1.1 403 Forbidden Date: Thu, 26 Jun 2003 07:48:39 GMT Server:
Apache/1.3.3 (Unix) (Red
Hat/Linux) Accept-Ranges: bytes Content-Length: 2898 Connection:
close]

TCP 110:
[+OK POP3 ns1.zerotrust.com.au v6.50 server ready ]

------------------------------------------------------------------
192.168.100.3
Responded in 0 ms.
23 hops away
Responds with ICMP unreachable: No
TCP ports: 21 25 80 119


TCP 21:
[220 win2ksrv Microsoft FTP Service (Version 5.0).]

TCP 25:
```

```
[220 win2ksrv Microsoft ESMTP MAIL Service, Version: 5.0.2195.1600
ready at Mon,
 24 Mar 2003 12:37:21 +1000]

TCP 80:
[HTTP/1.1 200 OK Server: Microsoft-IIS/5.0 Content-Location:
http://192.168.100.3/index.html
Date: Mon, 24 Mar 2003 12:37:21 GMT Content-Type: text/html Accep]

TCP 119:
[200 NNTP Service 5.00.0984 Version: 5.0.2195.1608 Posting Allowed]

------------------------------------------------------------------------



------------------------------------------------------------------------
192.168.100.4
Responded in 0 ms.
23 hops away
Responds with ICMP unreachable: No
TCP ports: 25 80 443


TCP 25:
[220 exchange.zeroconf.com Microsoft ESMTP MAIL Service, Version:
5.0.2195.2966 ready at
Mon, 23 Jun 2003 12:38:39 +1000]

TCP 80:
[HTTP/1.1 200 OK Server: Microsoft-IIS/5.0 Content-Location:
http://192.168.100.8/infoindex.htm Date: Mon, 23 Jun 2003 02:38:39
GMT Content-Type:
text/html Ac]

------------------------------------------------------------------------
192.168.100.5
Responded in 0 ms.
23 hops away
Responds with ICMP unreachable: No
TCP ports: 80 443


TCP 80:
[HTTP/1.1 200 OK Server: Microsoft-IIS/5.0 Date: Mon, 23 Jun 2003
02:38:41 GMT Connection:
Keep-Alive Content-Length: 1270 Content-Type: text/html Set-Cookie:]

------------------------------------------------------------------------
192.168.100.6
Responded in 0 ms.
23 hops away
Responds with ICMP unreachable: No
TCP ports: 22 23 25


TCP 23:
[Red Hat Linux release 7.1 (Seawolf) Kernel 2.4.18-26.7.x on an i686]

------------------------------------------------------------------------
192.168.100.7
```

```
Responded in 0 ms.
23 hops away
Responds with ICMP unreachable: No
TCP ports: 80 443


TCP 80:
[HTTP/1.1 200 OK Server: Microsoft-IIS/5.0 Date: Mon, 23 Jun 2003
02:38:40 GMT Connection:
Keep-Alive Content-Length: 1270 Content-Type: text/html Set-Cookie:]

------------------------------------------------------------------------
192.168.100.8
Responded in 30 ms.
23 hops away
Responds with ICMP unreachable: No
TCP ports: 23 80


TCP 23:
[User Access Verification Password:]

TCP 80:
[HTTP/1.0 401 Unauthorized Date: Tue, 06 Apr 1993 13:08:50 EST
Content-type: text/html
Expires: Thu, 16 Feb 1989 00:00:00 GMT WWW-Authenticate: Basic
realm="l]

------------------------------------------------------------------------
192.168.100.9
Responded in 0 ms.
23 hops away
Responds with ICMP unreachable: No
TCP ports: 23 80


TCP 23:
[User Access Verification Password:]

TCP 80:
[HTTP/1.0 401 Unauthorized Date: Sun, 21 Mar 1993 14:36:01 EST
Content-type: text/html
Expires: Thu, 16 Feb 1989 00:00:00 GMT WWW-Authenticate: Basic
realm="a]

------------------------------------------------------------------------
192.168.100.10
Responded in 10 ms.
23 hops away
Responds with ICMP unreachable: No
TCP ports: 23 80


TCP 23:
[User Access Verification Password:]

TCP 80:
[HTTP/1.0 401 Unauthorized Date: Thu, 20 May 1993 01:18:25 UTC
Content-type: text/html
Expires: Thu, 16 Feb 1989 00:00:00 GMT WWW-Authenticate: Basic
realm="l]
```

```
------------------------------------------------------------------------
192.168.100.254
Responded in 0 ms.
23 hops away
Responds with ICMP unreachable: No
TCP ports: 23


TCP 23:
[User Access Verification Password:]

------------------------------------------------------------------------


Scan finished at Mon Jun 23 12:37:25 2003
```

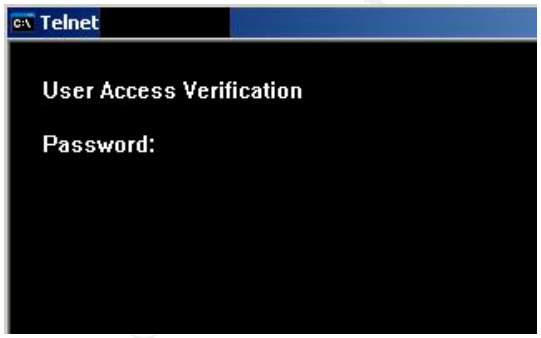
Testing Open Services for insure logon mechanisms

Checklist item 15: Identify insecure logon mechanisms used in applications exposed to the Internet.

The auditor attempted connecting to telnet ports that were visible from the outside. Several responded with a logon prompt. The auditor identifies that a Microsoft Exchange Outlook Web Access Server was running within the network and attempted connecting to it using standard HTTP. The server responded with a logon prompt indicating that at least the web mail authentication was transmitted across the Internet unencrypted.

Clear Text Telnet sessions

Command used – **c:\telnet 192.168.100.254**

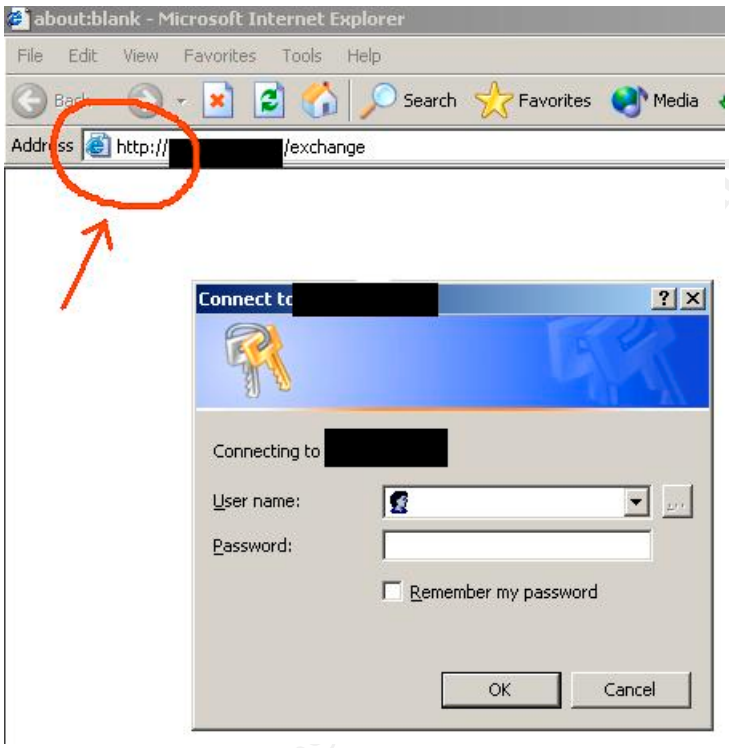


Testing Open Services for insure logon mechanisms and unencrypted web services that expose sensitive information

Checklist item 15: Identify insecure logon mechanisms used in applications exposed to the Internet.

Checklist item 17: identify unsecured web pages containing sensitive information

Unencrypted Outlook Web Access Server logons



Identifying known vulnerabilities

Checklist Item 16: Check web, mail and other servers exposed to the Internet to ensure correct patch levels are maintained

The auditor then deployed Nessus, the vulnerability scanning tool, against the server that were exposed to the Internet. Nessus was run against all the servers identified through the DNS zone transfer and port scans. However, only the report of a single scan has been included in this assignment.

Nessus Scan Report

This report gives details on hosts that were tested and issues that were found. Please follow the recommended steps and procedures to eradicate these threats.

Scan Details

Hosts which where alive and responding during test
1

Number of security holes found
1

Number of security warnings found
6

Host List

Host(s)	Possible Issue
192.168.100.1	Security hole(s) found

Analysis of Host

Address of Host	Port/Service	Issue regarding Port
192.168.100.1	ftp (21/tcp)	Security warning(s) found
192.168.100.1	ssh (22/tcp)	Security warning(s) found

192.168.100.1
domain (53/tcp)
Security hole found

192.168.100.1
http (80/tcp)
Security warning(s) found

192.168.100.1
sunrpc (111/tcp)
No Information

192.168.100.1
ldap (389/tcp)
No Information

192.168.100.1
h323hostcall (1720/tcp)
No Information

192.168.100.1
x11 (6000/tcp)
No Information

192.168.100.1
general/udp
Security notes found

192.168.100.1
domain (53/udp)
Security notes found

192.168.100.1
general/tcp
Security notes found

Security Issues and Fixes: 192.168.100.1

Type
Port
Issue and Fix

Warning
ftp (21/tcp)
This FTP service allows anonymous logins. If you do not
want to share data with anyone you do not know, then you should deactivate
the anonymous account, since it can only cause troubles.
Under most Unix system, doing :
echo ftp >> /etc/ftpusers
will correct this.

Risk factor : Low
CVE : CAN-1999-0497
Nessus ID : 10079

Informational
ftp (21/tcp)
Remote FTP server banner :
220 ready, dude (vsFTPd 1.1.0: beat me, break me)

Nessus ID : 10092

Warning
ssh (22/tcp)

The remote SSH daemon supports connections made
using the version 1.33 and/or 1.5 of the SSH protocol.

These protocols are not completely cryptographically
safe so they should not be used.

Solution :
If you use OpenSSH, set the option 'Protocol' to '2'
If you use SSH.com's set the option 'Ssh1Compatibility' to 'no'

Risk factor : Low
Nessus ID : 10882

Warning
ssh (22/tcp)

You are running OpenSSH-portable 3.6.1p1 or older.

If PAM support is enabled, an attacker may use a flaw in this version
to determine the existence or a given login name by comparing the times
the remote sshd daemon takes to refuse a bad password for a non-existant
login compared to the time it takes to refuse a bad password for an
existant login.

An attacker may use this flaw to set up a brute force attack against
the remote host.

*** Nessus did not check whether the remote SSH daemon is actually
*** using PAM or not, so this might be a false positive

Solution : Upgrade to OpenSSH-portable 3.6.1p2 or newer
Risk Factor : Low
CVE : CAN-2003-0190
Nessus ID : 11574

Informational
ssh (22/tcp)
The remote SSH daemon supports the following versions of the
SSH protocol :

. 1.33
. 1.5
. 1.99
. 2.0

Nessus ID : 10881

Vulnerability
domain (53/tcp)

The remote BIND 9 server, according to its
version number, is vulnerable to a buffer
overflow which may allow an attacker to
gain a shell on this host or to disable
this server.

Solution : upgrade to bind 9.2.2 or downgrade to the 8.x series
See also : http://www.isc.org/products/BIND/bind9.html
Risk factor : High
Nessus ID : 11318

Informational
domain (53/tcp)

A DNS server is running on this port. If you

do not use it, disable it.

Risk factor : Low
Nessus ID : 11002

Warning
http (80/tcp)

The remote host appears to be running a version of
Apache 2.x which is older than 2.0.43

This version allows an attacker to view the source code
of CGI scripts via a POST request made to a directory
with both WebDAV and CGI enabled.

*** Note that Nessus solely relied on the version number
*** of the remote server to issue this warning. This might
*** be a false positive

Solution : Upgrade to version 2.0.43
See also : http://www.apache.org/dist/httpd/CHANGES_2.0
Risk factor : Medium
CVE : CAN-2002-1156, CAN-2003-0083
BID : 6065
Nessus ID : 11408

Warning
http (80/tcp)

The remote web server seems to have its default welcome page set.
It probably means that this server is not used at all.

Solution : Disable this service, as you do not use it
Risk factor : Low
Nessus ID : 11422

Warning
http (80/tcp)

The remote host appears to be running a version of
Apache 2.x which is older than 2.0.45

This version is vulnerable to various flaws :

- There is a denial of service attack which may allow
an attacker to disable this server remotely

- The httpd process leaks file descriptors to child processes,
such as CGI scripts. An attacker who has the ability to execute
arbitrary CGI scripts on this server (including PHP code) would
be able to write arbitrary data in the file pointed to (in particular,
the log files)

Solution : Upgrade to version 2.0.45
See also : http://www.apache.org/dist/httpd/CHANGES_2.0
Risk factor : Medium
CVE : CAN-2003-0132
BID : 7254, 7255
Nessus ID : 11507

Informational
http (80/tcp)
The remote web server type is :

Apache/2.0.40 (Red Hat Linux)

Solution : You can set the directive 'ServerTokens Prod' to limit
the information emanating from the server in its response headers.
Nessus ID : 10107

Informational
general/udp
For your information, here is the traceroute to 203.xx.xx.xx:
xx.xx.xx.xx
xx.xx.xx.xx
xx.xx.xx.xx
xx.xx.xx.xx
xx.xx.xx.xx
xx.xx.xx.xx
xx.xx.xx.xx
xx.xx.xx.xx
xx.xx.xx.xx
xx.xx.xx.xx
xx.xx.xx.xx
xx.xx.xx.xx
192.168.100.1

Nessus ID : 10287

Informational
domain (53/udp)

A DNS server is running on this port. If you
do not use it, disable it.

Risk factor : Low
Nessus ID : 11002

Informational
domain (53/udp)
The remote bind version is : 9.2.1
Nessus ID : 10028

Informational
general/tcp
Remote OS guess : Axis 2100 Network Camera running Linux/CRIS v2.32

CVE : CAN-1999-0454
Nessus ID : 11268

4.4 Determining the risk to the system

Once the vulnerability testing has been concluded, and the results have been analysed to identify existing and potential vulnerabilities, a risk analysis may be conducted to determine the risks posed by the identified vulnerabilities. The risk analysis process in a zero knowledge vulnerability assessment is a highly subjective process. There are several contributory factors contributing the subjectiveness of the risk analysis. The fact that the auditor evaluates based on his/her interpretation of the organisation and its business based on the research s/he conducted is one such factor.

There are several definitions developed to quantify, and thereby reduce the subjectiveness of a risk assessment conducted as part of a vulnerability assessment. A common formula used to quantify the risk is:

Risk = threat x vulnerability x cost of the asset

Where, a vulnerability is the weakness of the system, and the threat is the likelihood of the vulnerability being exploited. Cost of the asset could be interpreted as either the material value of the asset, or the cost of the asset being unavailable due to the vulnerability being exploited. There are two approaches to assigning values to the threats, vulnerabilities and cost (variables). One is to use a predetermined scale of values, for example a scale of 1-10 and assign these values to the variables in the above equation. The other is to assign subjective values such as high, medium and low to the variables.

Some factors to consider when determining values for threats, vulnerabilities and costs of the assets:

Threats

- Level of difficulty in exploiting the vulnerability
- Costs involved in exploiting the vulnerability
- What is to be gained by exploiting the vulnerability
- Where the threat comes from

Costs

- Loss of reputation
- Lost Business
- Cost of repair
- Cost of downtime
- Cost of replacement

The proposed methodology will employ the above formula in determining the risk posed by the identified vulnerabilities.

4.4.1 Limitations of the Risk assessment methodology

The use of the above formula to quantify the risk is intended to reduce the subjectiveness of the risk assessment. However, the values assigned to the threats, and costs would typically be based on the auditor's interpretation of the business intelligence s/he has gathered on the organisation through the research s/he conducted. While the auditor may make every attempt possible to make an objective assessment of risk,

> The owners of the assets will analyse the possible threats to determine which ones apply to their environment. The results are known as risks. This analysis can aid in the selection of countermeasures to counter the risks and reduce it to an acceptable level (CME Supplement p.14)

Another important thing to consider in assessing risk through a vulnerability assessment us that vulnerabilities are identified as individual entities. Isolated vulnerabilities may be assigned low risks, based on the nature of the vulnerability and the exploit associated with the vulnerability. However, there is the possibility that a low risk vulnerability, combined or aggregated with another vulnerability may present a higher level of risk. The assessor need to be aware of this fact when assigning values to vulnerabilities and threats, as vulnerability scanners cannot infer this type of information.

4.4.2 Example Risk Assessment

Following is an adaptation of the above formula to quantify the risk.

Consider the following output from an nmap scan.

Interesting ports on **internaldb.zerotrust.com.au**(192.168.100.5):
(The 1587 ports scanned but not shown below are in state: closed)

Port	State	Service
80/tcp	open	http
135/tcp	open	loc-srv
139/tcp	open	netbios-ssn
443/tcp	open	https
445/tcp	open	microsoft-ds
1050/tcp	open	java-or-OTGfileshare
1433/tcp	open	ms-sql-s
3389/tcp	open	ms-term-serv
5800/tcp	open	vnc-http
5900/tcp	open	vnc
6103/tcp	open	RETS-or-BackupExec
6667/tcp	open	irc
6668/tcp	open	irc
10000/tcp	open	snet-sensor-mgmt

Several vulnerabilities immediately stand out from the result of this scan. However, for the purposes of this exercise the following vulnerabilities will be considered.
The name of the server indicates that it is a database server used internally. Open port 1433 confirms this assumption, as it is the port used by Microsoft SQL server. This server also has open NetBIOS sessions visible to the Internet. In addition to the above vulnerabilities, VNC has been installed on

the server. A web server is running on port 80, which was confirmed as an IIS server through its http banner.

Output from Scanline:

```
192.168.100.5
Responded in 0 ms.
23 hops away
Responds with ICMP unreachable: No
TCP ports: 80 443


TCP 80:
[HTTP/1.1 200 OK Server: Microsoft-IIS/5.0 Date: Mon, 23 Jun 2003
02:38:41 GMT Connection:
Keep-Alive Content-Length: 1270 Content-Type: text/html Set-Cookie:]
```

Vulnerabilities (at a glance)

- Information leakage identified the server and several services running on the server.
- Several services with known vulnerabilities are exposed to the Internet.
- Several known exploits to the exposed services exist.
- Internal database server exposed to the Internet.

The exploitation of vulnerabilities identified could result in:

- Unauthorised access
- Disclosure of data, or unauthorised modification of data, or denial of services by
- Systems being used for illegal activities such as attacks on other systems external to the organisation.

Based on the above information the assessor may assign a **high** rating to the vulnerabilities discovered on the above server.

4.4.3 Determining the threat

- Level of difficulty in conducting an attack on the server is low. Several known attacks against the applications on the server exist.
- Being a database server, it is an attractive target to the business competition, as well as other unscrupulous individuals who may use the information for personal gain.
- The organisation has been in the news lately, therefore the publicity of a successful attack may attract an attacker who may be seeking prestige among the attacker community.

Based on the above information the assessor may assign a **high** threat rating to the above server.

4.4.4 Estimation of the cost

The system may have confidential client information, as well as other confidential corporate information such as business strategies, company employee information, payroll information, loss and profit information etc.

The loss of such information, or the information not being available when required, could have financial impact such as loss of business.

The server being unavailable due to a denial of service attack, or a rebuild due to a system compromise could cause financial loss.

The negative publicity of a successful attack on the system may result in the loss of business due to clients losing faith about the security of their investments with the company.

Therefore, not addressing the vulnerability may have a **high** cost impact on the company.

Based on the formula used above, the risk created by the identified vulnerability is rated as high.

5 Assignment 4 - Follow Up

At the final stage of the audit, management at ZTI was presented with a report of the audit. The report targeted the management as the main audience, but it was envisaged that it would be passed on to the IT department of ZTI for addressing the technical issues. The structure of the report reflected this need. It contained an executive summary, a summary of the key findings and a summary recommendation for review by the Management at ZTI, followed on by a detailed analysis of the findings, including identified vulnerabilities and risks, as well as detailed recommendations to address those vulnerabilities and risks.

In preparing this report for the purposes this assignment, a selection of 10 findings to cover various aspects of the assessment was made.

Report of the External Perimeter Vulnerability Assessment of Zero Trust Incorporated

By

Hackville Security Consulting

1 September 2003

DOCUMENT CONTROL

Distribution List

Copy	To	Comments
1	Jane Doe COO, ZTI	Project Sponsor
2	Joe Blogg IT Manager ZTI	Technical Contact
3	Homer Simpson HSC	Internal Quality Control
4		

Version Control

Version	Date Updated	Updated By	Comments
0.1	15 Aug 2003	Senior Consultant, HSC	Initial draft for internal review
0.2	19 Aug 2003	Senior Consultant, HSC	Draft for client interviews
1.0	1 Sept 2003	Senior Consultant	Final Version

Executive Summary

This report details an external security review and vulnerability assessment (audit) of the Zero Trust Incorporated (ZTI) information systems undertaken by Hackville Security Consulting (HSC). The objective of the audit was to find out vulnerabilities of the systems at ZTI were exposed to the Internet that may be exploited by an outside attacker. The audit was conducted between the 24th and 26th March 2003, via the Internet. The auditor had no prior knowledge of the systems to be audited, and used the same methodology that a potential attacker would use to identify and compromise a target company's information systems.

A Significant finding of the assessment related to the lack of control mechanisms for traffic inbound to the ZTI network, or the poor configuration and implementation of the existing control mechanisms. The configuration of the gateway device allowed many types of non-business related traffic to enter the network unfiltered, enabling HSC to carry out many types of scans and probes and other simulated attacks. Using these scans and probes HSC was able to identify the network topology and critical servers within the network.

Another key finding of the assessment indicates the lack of policy, or the inefficient enforcement of policy governing Information security. Many servers appeared to run with default installs and configurations with insecure configurations. Services that may not be required for the business functionality of the servers were found to be installed and running, and some of them still had the un-patched default versions with several known vulnerabilities. Many of these servers also revealed information about themselves through the software banners displayed upon connection to these servers.

HSC also identified design flaws in the architecture of the network, which allowed servers that should only be accessible internally to be accessible to the Internet, thereby risking corporate information to be exposed. Both corporate and publicly accessible servers were installed on the same network segment thereby increasing the threat of internal servers being compromised by an attacker.

Tests that HSC conducted generated a substantial quantity of intrusive traffic, including some of the tests that were conducted quite aggressively on purpose. The aim was to identify the intrusion detection and incident response capability of ZTI. The fact that no reports of such detections were made to the IT Manager, with whom HSC liaised during the tests indicate that either there are no mechanisms for detecting such behaviour, or that the IT staff are not trained in incident response skills.

The vulnerabilities found in the assessment poses several risks to ZTI. The most significant of these risks are that several of the servers critical to the

business functions of the organisation exposed vulnerabilities that could result in the total compromise of these servers. Several of the servers were vulnerable to denial of service attacks. In summary, ZTI currently are exposed to the risk of having their corporate information stolen, modified or made unavailable for use by staff and clients.

In conclusion HSC strongly recommends that existing Information security policies be reviewed and enforced, or if such policies are not in place to immediately take action to implement such policies. These policies should address server hardening, configuration management, firewall management, and incident response at a minimum. HSC also recommends that a review of the architecture be carried out immediately to separate internal corporate servers from the publicly available ones. HSC considers training of staff in information security practices and incident response skills to be another important area that needs attention.

Findings and Recommendations

Detailed below are the findings of the tests conducted by HSC, the risks associated with the findings and recommendations and estimation of costs to address the vulnerabilities found.

HSC uses the following rating system to indicate the level of severity of the findings.

Critical – Critical vulnerability identified. Needs immediate attention and action.

High – High-risk vulnerability identified. Needs to be addressed as soon as possible.

Moderate – Needs to be addressed once critical and high rated vulnerabilities have been addressed.

Low – Could be addressed during scheduled maintenance

Information only – further tests may be necessary to determine severity and impact.

Vulnerabilities in Administrative Controls in place at ZTI

Finding 1 – Lack of enforcement of policy and procedural controls

During the tests HSC found several instances that indicated the lack of, or poor enforcement of policy and procedure. This is an inference made by HSC based on the firewall and server configuration and design flaws and IT staff's incident response capabilities. These individual items will be addressed separately under technical controls.

Vulnerability rating: **High**

Background/Risk:

Policies define the standard that should be maintained, in this case the information security infrastructure and management practices. Not having such policies open the organisation up for further vulnerabilities, as there are no standards against which to measure the security of the operational IT environment.

Potential Security Breach:

Without a security policy, it is not possible to enforce any form of security apart from arbitrary/ad hoc security mechanisms.

Recommendations:

Ensure at the following policies are in place and that they are enforced

1. Firewall Policy – defining the allowed ingress and egress traffic, the management of the firewall, log and event management.
2. Incident Response Policy – Define intrusions and incidents, and response procedures.
3. Configuration Management Policy – define disabling of services not required for functionality of the server or device, management of updates and patches.
4. Server Hardening Policy – Define the initial configuration of the server to suit its role and includes such information as the services allowed to run, hosts allowed to access server etc.

Cost:

Significant number of person hours in developing or reviewing the existing policy. It may be necessary to employ external expertise in developing these policies, in which case a rough financial cost of about $1000.00 per day may be required to engage such experts.

Finding 2 – lack of incident detection and response mechanisms and awareness

HSC conducted a series of test that should have generated a substantial amount of event entries in the firewall, intrusion detection systems and server logs. HSC also conducted a series of aggressive tests deliberately meant to generate such events and to possibly alerts. It appears that these events have

either not been logged, or if they were logged, not been identified as possible intrusion attempts and attacks.

Vulnerability Rating: **High**

Background/Risk:

Intrusions and systems compromises may go undetected, with the business operating with compromised servers, and corrupt or modified data.

Potential Security Breach:

Not being able to identify any security breaches on the systems.

Recommendations:

At a minimum, enable systems logging (syslog) features built in to the gateways, firewalls.

Install an Intrusion Detection System (IDS).

Cost:

Enabling systems logs and alerts, and testing the alerts may cost ~8-10 person hours. No monitory expenditure required.

An open source IDS system with alerting may be built and installed with little cost to the company. Expenses associated would be ~$1000.00-1500.00 for a personal computer, and about 6 hours person time if required skills are available within the company. If external engineers are used, it may cost ~100.00 an hour for the engineer.

Vulnerabilities in Network Architecture and Design

Finding 3 – Both internal corporate servers and publicly available servers reside on the same network segment (subnet).

Using various ICMP types such as ping, trace route and ICMP address mask requests on the gateway device and name resolution mechanisms, HSC was able to deduce that the internal mail, database and other servers were located on the same subnet as the publicly accessible servers.

Vulnerability Rating: **Critical**

Background/Risk:

Internal servers being publicly available exposes them to the risk of being compromised and confidential information stored in them to be stolen, modified or deleted by an attacker. They are also exposed to the threat of external denial of service attacks. Even if the firewall was reconfigured to block access to these servers, a compromised externally accessible server may be used as a beachhead to conduct an attack on them if they continued to live on the same subnet as the externally accessible servers.

Potential Security Breach:

Intrusion of the systems for network mapping and other reconnaissance probes which may result in further security breaches.

Recommendations:

- Reconfigure ZTI network infrastructure to include a screened subnet or a demilitarized zone.
- Place all publicly accessible servers in the new network segment.
- Place all internal servers in their own subnet using a non-routable IP address space.
- Filter out all traffic to internal servers at the firewall.
- If external servers need access to internal servers for business functionality, identify the IP addresses and application ports that need to be accesses and filter out the traffic from and to non-required IP addresses and ports

Please refer to Appendix A for a network diagram illustrating this recommendation.

Cost:

The planning of the reconfiguration will involve the Management, IT management, systems administration and application development teams to identify the impacts the change may have on the system, and the short term and long term impact on the system. It is imperative to understand the communication paths between the systems and applications before any change is made.

Therefore the cost of planning as estimated by HSC to be about 4 days, with a minimum of one representative from each of the groups present in the planning sessions.

The cost of a firewall is estimated at ~$3000-$5000 based on the number of staff at ZTI.

The implementation may need to be done after hours, and may require the assistance of external systems engineers. The cost of such external assistance would be ~$100.00 during daytime, and $150.00 after hours.

Cost of the system downtime needs to be estimated during the planning session.

Overtime, and/or time off in lieu cost to be estimated at planning time.

Vulnerabilities in Systems Configuration

Finding 4 – The Perimeter Firewall does not filter out traffic not required for business functionality

HSC was able to connect to many servers and services using several types of scans and probes. Using the scans and probes HSC was ale to map the network topology, and identify most of the configuration of the network.

Vulnerability Rating: **Critical**

Background/Risk: A potential attacker can gather a substantial amount of information about the network through various ICMP messages. The attacker may also deploy attacks to compromise and 'own' the systems, or cause denial of services on these servers. The attacker may also deploy attacks that lead to stealing, modification or making unavailable of corporate information.

Potential Security Breach:

Attacks on the systems, causing breach of confidentiality, integrity or availability of the information contained within the systems resulting in:

- Disclosure of clients' confidential information
- Disclosure of corporate confidential information
- Data altered on the systems
- Systems being used to store illegal or inappropriate material such as pirated software and pornography
- System being used for denial of services attacks on other organisations.

Recommendations:

Configure the firewall to filter out traffic not required for business functionality

Cost:

30-60 minutes of person hours by the firewall administrator if the permitted traffic is defined by a firewall policy. It may take longer if the permitted traffic is to be defined as part of the control mechanism.

Finding 5: A misconfigured name server allows unauthorised persons to gain a substantial amount of information about ZTI's systems including names and IP addresses of servers

HSC was able to complete a DNS zone transfer, which revealed not only the information that should be publicly available on the name server, but also information about the internal servers and some hosts.

Vulnerability Rating: **High**

Background/Risk:

This information arms a potential attacker with insight into ZTI's internal network and assists in identifying critical servers and services within the network. As ZTI uses a descriptive naming scheme for their servers (exchange.zerotrust.com, internaldb.zerotrust.com etc) it is also possible to guess at the services provided by these servers and even the importance of some of the servers to the organisation. Depending on the intent of the attacker s/he may pick the target they want to compromise or deploy a denial of service attack.

IT is also possible to poison the dns server through a zone transfer, thus resulting in a denial of service attack. This is a potential vulnerability that should be further investigated by ZTI.

Potential Security Breaches:

Attacks on the systems, causing breach of confidentiality, integrity or availability of the information contained within the systems. These security breaches may result in:

- Disclosure of clients' confidential information
- Disclosure of corporate confidential information
- Data altered on the systems
- Systems being used to store illegal or inappropriate material such as pirated software and pornography
- System being used for denial of services attacks on other organisations.

Recommendations:

Reconfigure name server to disallow zone transfers to unauthorised hosts. I Install separate name servers for internal and external use and implement a split-dns.

Cost:

Reconfiguring the DNS servers to disallow DNS zone transfers to unauthorised hosts may take 4 hrs of person time.

If hardware need to be purchased to implement a split DNS system, and have an internal-only name server system to service internal clients, it may cost up to AUD $2500.00 per server. For resilience it may be necessary to implement two servers.

Finding 6: Services not required for business functionality are installed and running on servers on the ZTI network.

Using a port scan for applications running on identified servers, HSC found that several applications installed by default were left running on these servers. These applications included remote access services such as Microsoft Terminal Services, VNC and X Windows, as well as other services with a number of known technical vulnerabilities such as NetBIOS services.

Vulnerability Rating: **High**

Background/Risk:

The more services left running on the servers, the more chances an attacker has in identifying an exploitable vulnerability.

Potential Security Breaches:

Attacks on the systems, using publicly known, and/or zero day exploits causing breach of confidentiality, integrity or availability of the information contained within the systems. Further security breaches may lead to:

- Disclosure of clients' confidential information
- Disclosure of corporate confidential information
- Data altered on the systems
- Systems being used to store illegal or inappropriate material such as pirated software and pornography

- System being used for denial of services attacks on other organisations.

Recommendations:

Enforce on server configuration policy and remove or shut down applications and services not required for business functionality.

Cost:

Based on the number of servers, HSC estimates 2-3 days of work by the IT staff to enforce this recommendation. As the removal of certain services may require system restarts, it may be required to schedule these actions for after hours work, and may have additional costs of overtime payments for IT staff, or time off in lieu implications.

Finding 7: Several Services on the ZTI servers leaked information about the software and operating system versions and patch levels.

Using various tools and software clients, HSC was able to identify several services and software version on servers with the ZTI network.

Vulnerability Rating: **High**

Background/Risk:

Identifying software and operating systems version and patch level information allows attackers to narrow down their search for exploitable vulnerabilities and exploits. This increases the threat of these servers being attractive to attackers.

Potential Security Breaches:

Attacks on the identified operating systems or applications using publicly known, and/or zero day exploits causing breach of confidentiality, integrity or availability of the information contained within the systems. Further security breaches may lead to:

- Disclosure of clients' confidential information
- Disclosure of corporate confidential information
- Data altered on the systems
- Systems being used to store illegal or inappropriate material such as pirated software and pornography
- System being used for denial of services attacks on other organisations.

Recommendations:

Edit the configuration files for these services, or use appropriate tools to change or obfuscate the banners so they do not reveal information about the software and operating system versions.

A server hardening policy as recommended in Finding 1 should typically include the removal of these services.

Note – There may be services that may not allow the modifications of such nature. Please refer to the administration/user manuals before carrying out these recommendations.

It may be possible to carry out the recommendations for Findings 6 and 7 at the same time if properly researched and planned. HSC recommends that ZTI IT staff investigate the feasibility of carrying out recommendations 6 and 7.

Cost:

Based on the number of services identified HSC estimates 2-3 days of person hours to implement these recommendations. As in the previous recommendation, it may be necessary to conduct some of these activities after hours.

Finding 8: Several servers and services on the ZTI network were running versions of software not patched to remedy various vulnerabilities with available known exploits.

Using vulnerability scanning tool and manual methods, HSC identified several vulnerabilities that had known vulnerabilities freely available on the Internet.

Vulnerability Rating: **Critical**

Background/Risk:

An attacker could exploit these vulnerabilities to totally compromise the systems and gain control over them, steal or modify the information stored on them, or cause a denial of services on these servers.

Potential Security Breaches:

Exploitation of these vulnerabilities, and compromising and taking control o these servers by malicious attackers using publicly known, and/or zero day exploits. Results of such an attack could be the breach of confidentiality, integrity or availability of the information contained within the systems. Other security breaches effected by these unpatched servers and services may include:

- Disclosure of clients' confidential information
- Disclosure of corporate confidential information
- Data altered on the systems
- Systems being used to store illegal or inappropriate material such as pirated software and pornography
- System being used for denial of services attacks on other organisations.

Recommendations:

Immediately patch these servers with vendor supplied patches and fixes.

Cost:

Further investigation into the services identified with this vulnerability is warranted prior to applying the patches. If these services are not required for business functionality as indicated in Finding 6, simply shutting them down would mitigate the risks posed by these vulnerabilities.

HSC estimates 2-3 days of person hours to remedy these vulnerabilities. It may be possible to address these vulnerabilities at the same time as remedial actions for findings 6 and 7.

Finding 9: Some remotely accessible applications used clear text based authentication mechanisms.

HSC found several severs and devices that allowed remote login used unencrypted authentication mechanisms that sent the username and password across the Internet in clear text. These services included telnet, http based logons and remote access service such as Virtual Network Computing (VNC) services and Microsoft Windows Terminal Services.

Vulnerability Rating: **High**

Potential Security Breaches:

Disclosure of sensitive authentication information that may lead to system compromise.

Background/Risk:

An attacker could eavesdrop on these authentication sessions and harvest user names and passwords, and then gain access to the systems to carry out unauthorised activities. These unauthorised activities could include stealing of modification of information and denial of services attacks.

Recommendations:

- Where possible replace these clear text logon sessions with encrypted logon sessions. For example use Secure Shell (SSH) to replace telnet session, introduce Secure Socket Layer (SSL) or Transport Layer Security (TLS) to HTTP based logons.
- If remote access services such as Terminal Services and VNC are required, secure the communication channel using a Virtual Private Network (VPN) or restrict access to predefined hosts only.

Cost:

Most operating systems provide built in secure communication mechanisms. Therefore no additional purchases may be required if using built in SSH servers, clients and SSL services are used.

Deploying a VPN mechanism may require purchase of additional hardware if the firewall in place does not support VPN technologies. Additional client licenses may also be required.

Implementing these options will require IT staff to spend several person hours on planning, configuring and deploying these secure communication mechanisms.

There will also be a requirement for end user training in the use of these secure communication mechanisms.

Vulnerabilities caused by user practices

Finding 10: System information was publicly made available on news groups and other postings on the Internet

Using several searches on the Internet HSC was able to identify information about the systems posted by IT staff requesting for assistance in troubleshooting systems issues encountered in their day to day operations. This information included versions of operating systems, IP address schemes, and information about the firewall.

Vulnerability Rating: **High**

Potential Security Breaches:

Disclosure of sensitive information about the systems and the security mechanisms. This may lead to a malicious attacker gaining an insight into not only system configuration information, but also operational information such as system administrator practices and skill levels.

Background/Risk:

Making this information publicly available entices potential attacker to attack the systems as s/he can has reliable information about the systems without any conducting any scans and probes that may alert the organisation.

An additional risk of being social engineered to provide more information about the system, and even access to the system is possible.

Recommendations:

HSC identifies this as a vulnerability caused by IT staff insufficiently trained in Security awareness and secure systems administration practices. It is also possible that the IT staff are over burdened with day to day operational activities and troubleshooting problems, that they do not have the time for self learning and research for a fix to systems issues.

HSC also identifies that this behaviour could be due to the lack of policy and procedural mechanisms that control such postings.

- HSC recommends that IT staff be trained in security practices as a mater of urgency
- HSC also recommends that further investigations be made to staffing requirements, and review staffing levels accordingly.

Cost:

Cost of training in Australia is typically about AUD 500-800 per person, per day.

In case staffing levels are to be increased, market salary rates differ based on the expertise and skill levels the position requires.

Other recommendations

HSC found that several of the vulnerabilities have been on the systems for at least a period of two years. The external assessment did not reveal that the systems had been compromised. Since there doesn't seem to be any

historical logs and alerts to indicate whether the systems had been attacked in the past or not, HSC strongly recommends a host audit to be carried out within the ZTI network to ensure that they are not functioning with a compromised system.

This vulnerability assessment is accurate at the time it was conducted. However, new vulnerabilities are discovered almost daily, and any change made to the system may introduce new vulnerabilities. HSC recommends that ZTI address the issues identify during this assessment, and using that as a baseline model, conduct periodic vulnerability assessments of their systems to ensure their systems functions with minimum vulnerabilities exposed to the Internet.

The overall risk assessment is based on the above findings, and the business intelligence gathered by HSC from various sources. Therefore it is a subjective assessment, and may not reflect the organisational definitions of risks.

Analysis of Risk to the Business

As the details of the systems were not revealed to the auditor at the commencement of the audit, it was difficult to do a technical risk assessment at the inception of the audit. However, using the initial interviews with the management and publicly available information about ZTI, the auditor was able to identify several risks associated with 'any' system in place at these organizations.

The following table contains a brief analysis of the possible security breaches from attacks against the systems, consequences of these security breaches and the potential impact on the company. Also included is a risk rating that was assigned to each security breach. The value assigned for the risk is based on the company management's perception of them being chosen as a target due to their recent high profile, and is subjective in this case. The following information was considered when identifying the potential threat to ZTI systems.

- The company has seen substantial growth in the past few years, thus making its profile more prominent and attractive to potential attackers seeking publicity.
- The company has outmanoeuvred its competition, and has managed to offer a wider variety of services than the competition.
- Several customers from competing companies have moved their business across to ZTI.
- Until recently, information security has not been a priority within the organisation, and the IT staff have not been provided with security training and tools they needed.
- According to the management, some security measures were in place. While it was not revealed what these measures were at the beginning of the engagement, the auditor found that the 'security measures' they were referring to was a firewall that had been installed some months before the audit was conducted.

The risk rating was assigned based on the formula

Risk = Vulnerability x Threat x Cost

Due to the exposure of the systems, and the recent publicity the company has received, the assessor assigned a **high** value to the threat of security breaches occurring against the organisation. This is a subjective value.

Some of the vulnerabilities identified during the assessment left the systems at ZTI open to several security breaches that may lead to the compromise of the systems and/or the information contained within those systems. Therefore these vulnerabilities have been assigned a **high** value by the assessor.

In determining the cost to the systems, the assessor considered the cost of replacement and repair, cost of downtime, and the possible damage to the reputation of the company and loss of faith in the company. The assessor assigned values ranging from **low** to **high** in determining the cost.

Please note that these values are subjective and is based on the assessors personal assessment. It is recommended that the company assess these risks internally and in depth.

For a mapping of the vulnerability to potential security breaches, please refer to the findings section of this report.

	Potential security breach	**Consequence**	**Impact on the Company**	**Risk**
1.	Disclosure of clients' confidential information	Negative publicity, loss of faith in the company resulting in loss of business, company open to litigation	High	High
2.	Disclosure of corporate confidential information	Negative publicity, corporate and trade secrets leaked to competition,	High	High
3.	Data altered on the systems	Loss of faith in the company, open to litigation, inaccurate business decisions made based on available data	High	High

	Potential security breach	**Consequence**	**Impact on the Company**	**Risk**
4.	Systems being used to store illegal or inappropriate material such as pirated software and pornography	Negative publicity, litigation	Medium	Medium
5.	System being used for denial of services attacks	Negative publicity, possible litigation for contributory negligence	High	High
6.	System being unavailable due to any of the above reasons	Inability to conduct business during system downtime, cost of sanitising or rebuilding systems	Medium to High depending on duration	High
7.	System being unavailable due to denial of service attacks.	Inability to conduct business during system downtime, cost of sanitising or rebuilding systems	Medium to High depending on duration	High
8.	Use of company resources by unauthorized individuals or groups	Storage space, system processes and memory, bandwidth etc. being unavailable for business purposes.	Low to medium	Medium

Conclusion

The external vulnerability assessment of Zero Trust Incorporated conducted by Hackville Security Consulting revealed several vulnerabilities. These vulnerabilities exposed the system of Zero Trust Incorporated to several risks that could affect their business objectives.

Hackville Security Consulting, in their report, have evaluated the identified vulnerabilities and made recommendations and suggestions to address these vulnerabilities, thereby mitigating or reducing the risks associated with them. It is envisaged that Zero Trust Incorporated will action these recommendations and suggestions based on the severity of the vulnerabilities in order to ensure secure operation of their systems.

Appendix A

Current Network Architecture

Based on the information gathered from the Internet and various tests, HSC was able to construct the following map of the ZTI network.

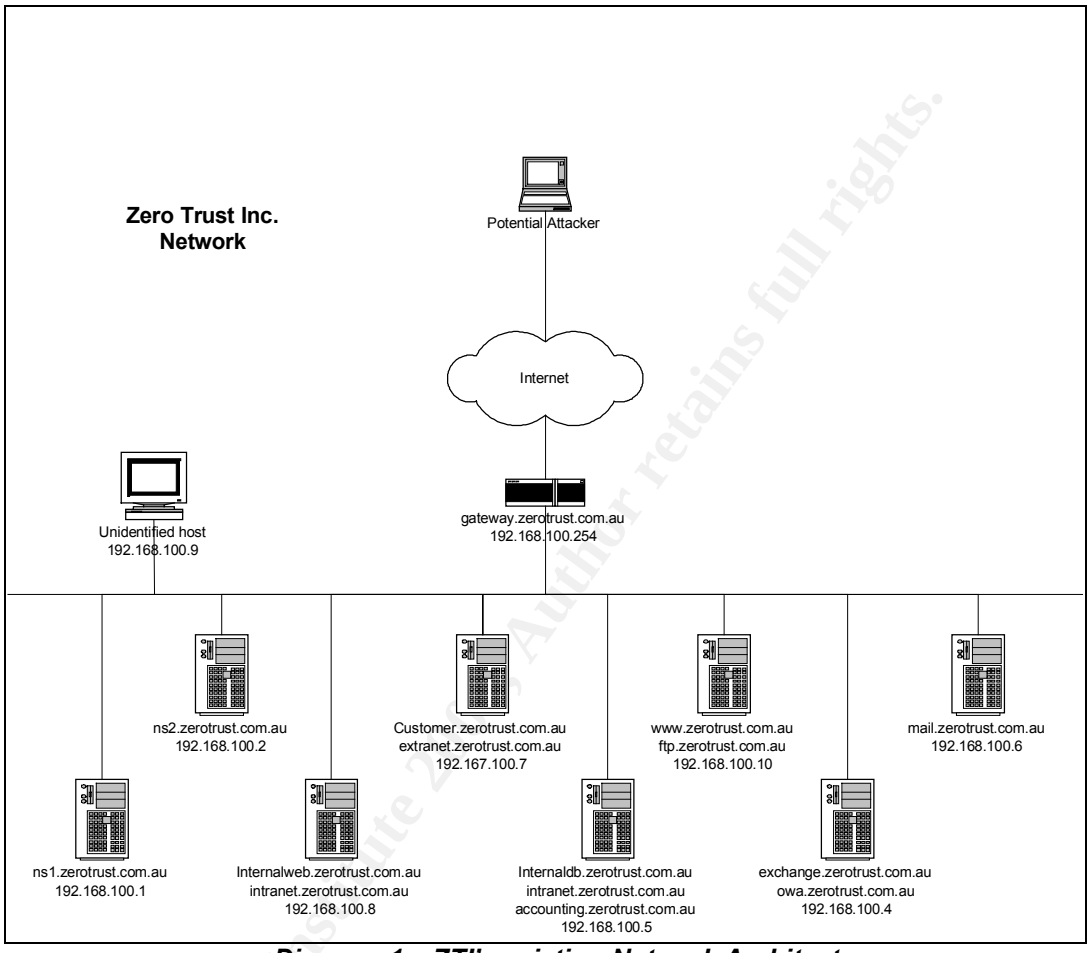


Diagram 1 – ZTI's existing Network Architecture

Suggested review of the ZTI Network Architecture

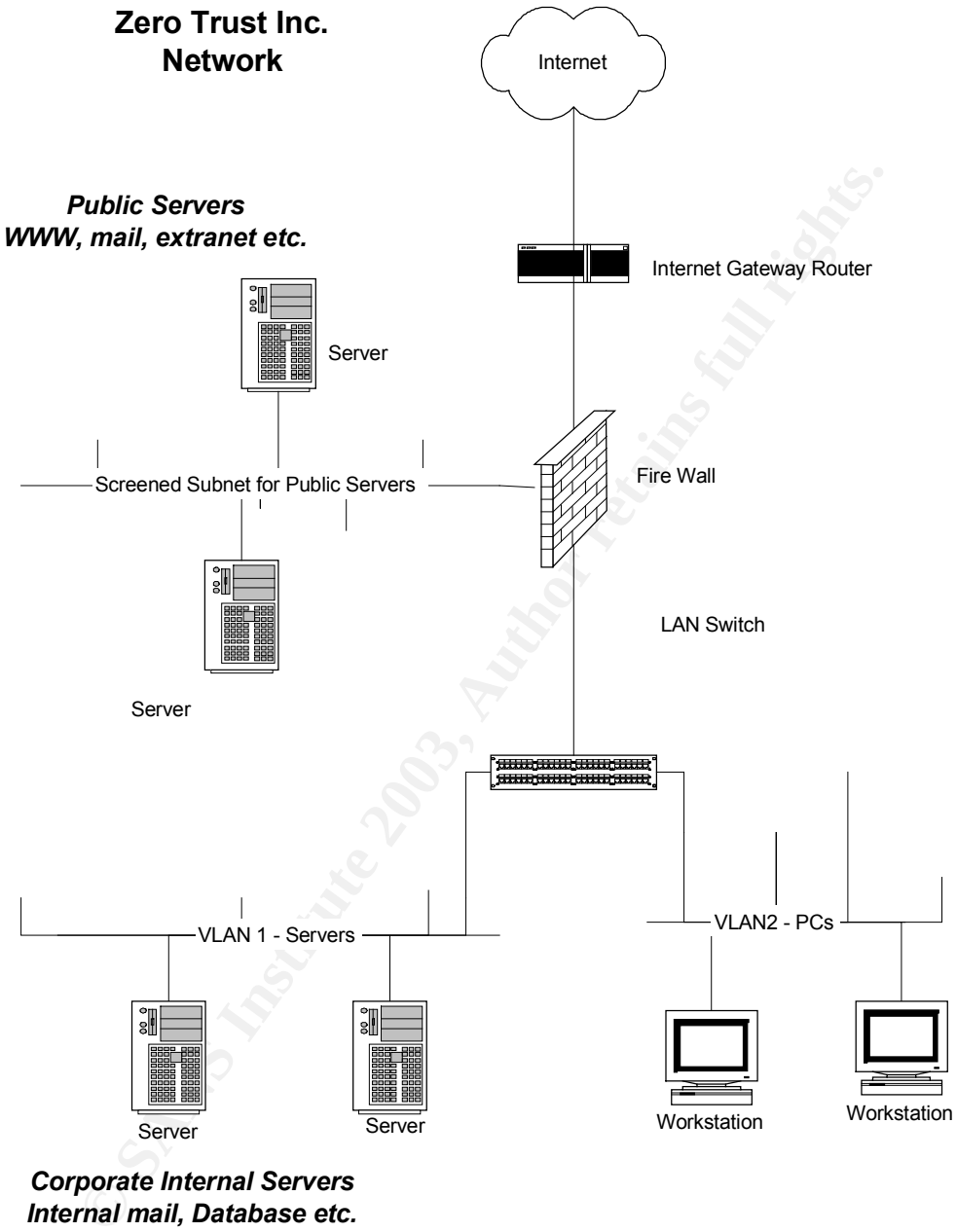


Diagram 2 – Suggested Architecture for ZTI network

6 Bibliography and Further Reading

CERT (2003), CERT Advisory CA-2003-04 MS-SQL Server Worm, http://www.cert.org/advisories/CA-2003-o4.html

CERT (Computer Emergency and Response Team) (2001), Securing Public Web Servers, http://www.cert.org/security-improvement/modules/m11.html#s.%20issues

Chirrillo, J. (2001), Hack Attacks Revealed John Wiley

General Accounting Office, United States (1999) Federal Information Systems Control Manual

Google, Hackers Best Friend" by Paris 2k Labs http://www.astalavista.com/library/basics/guides/GoogleHTool.pdf

Herzog, Pete (2002) Open Source Security Testing Methodology, www.ideahamster.org

http://csrc.nist.gov/publications/drafts/security-testing.pdf

http://www.commoncriteria.org/review_docs/docs/2002-07-001.pdf

Information Security Forum (November 2000), The Forum' s Standard of Good Practice: The Standard for Information Security

Institute of Internal Auditors, IT Audit Check Lists http://www.theiia.org/itaudit/index.cfm?fuseaction=catref&catid=18

International Information Security Foundation, (1999), Generally Accepted System Security Principals.

Johansson, K. (2001) Offensive Operations Model http://www.penetrationtest.com/1_Internet_security_information_security_downloads.htm

Klevinsky, T. J., Laliberte, S. and Gupta, A. (2002) Hack I.T. Security Through Penetration Testing, Addison-Wesley

McLure, S. Scambray, J. and Kurtz, George (2001) Hacking Exposed 3rd Edition, Osborne/McGraw Hill.

Naidu, K. (Date unknown) Firewall Check Lists http://www.sans.org/score/firewallchecklist.php

Naidu, K. (Date unknown) Web Application Check Lists http://www.sans.org/score/webappschecklist.php

SANS Reading Room, http://rr.sans.org

Schweitzer, D., (2003), Incident Response, Wiley

Skoudis, E. (2002), Hack Counter Hack Training Course, Prentice Hall.

Spitzner, L (2000) Auditing Your Firewall http://www.spitzner.net/audit.html

Swanson, Marianne and Guttman, B (1996), Generally Accepted Principles and Practices for Securing Information Technology Systems http://csrc.nist.gov/publications/nistpubs/800-14/800-14.pdf

The Open Web Application Security Project Guide to Building Secure Web Applications http://www.owasp.org/documentation/guide

7 Tools used in the audit

Brutus, http://www.hoobie.net/brutus/brutus-download.html

Google Search Engine, www.google.com

John the Ripper, http://www.bebits.com/app/2396

LC4, http://www.atstake.com/research/lc4

Microsoft Internet Explorer http://www.microsoft.com

Nessus, http://www.nessus.org/download.html

Netcat, http://www.atstake.com/research/tools/network_utilities/

Nmap, http://www.insecure.org/nmap/nmap_download.html

Sam Spade, http://www.samspade.org/ssw/download.html

Scanline (sl.exe), http://www.foundstone.com/index.htm?subnav=resources/navigation.htm&subcontent=/resources/scanning.htm

SING, http://sourceforge.net/projects/sing

Snscan.exe, http://www.foundstone.com/index.htm?subnav=resources/navigation.htm&subcontent=/resources/scanning.htm

SQLscan, http://www.foundstone.com/index.htm?subnav=resources/navigation.htm&subcontent=/resources/scanning.htm

8 Appendix 1 – configuration options to some of the tools and commands used in the audit.

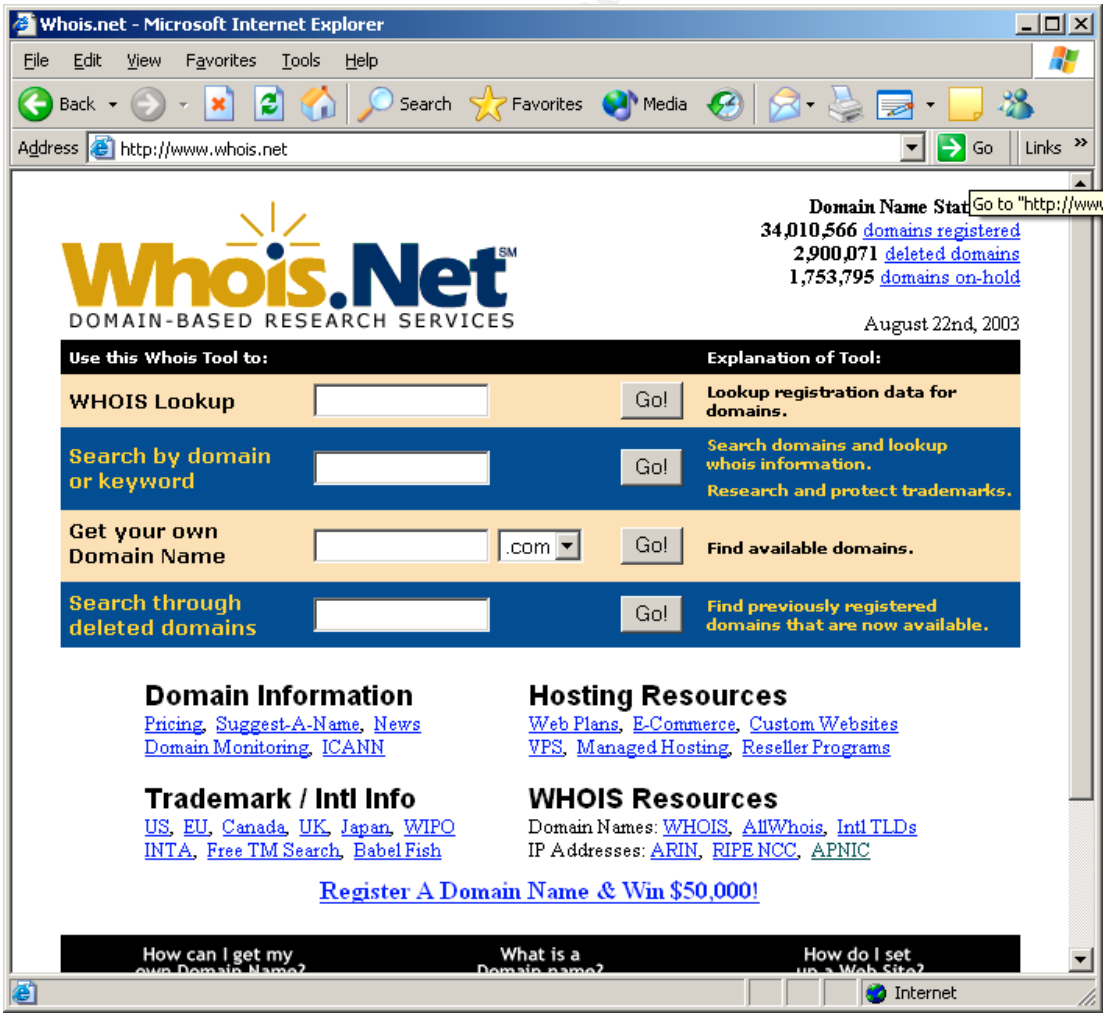
Using the whois command in Linux

$ whois -h whois.register.com sans.org

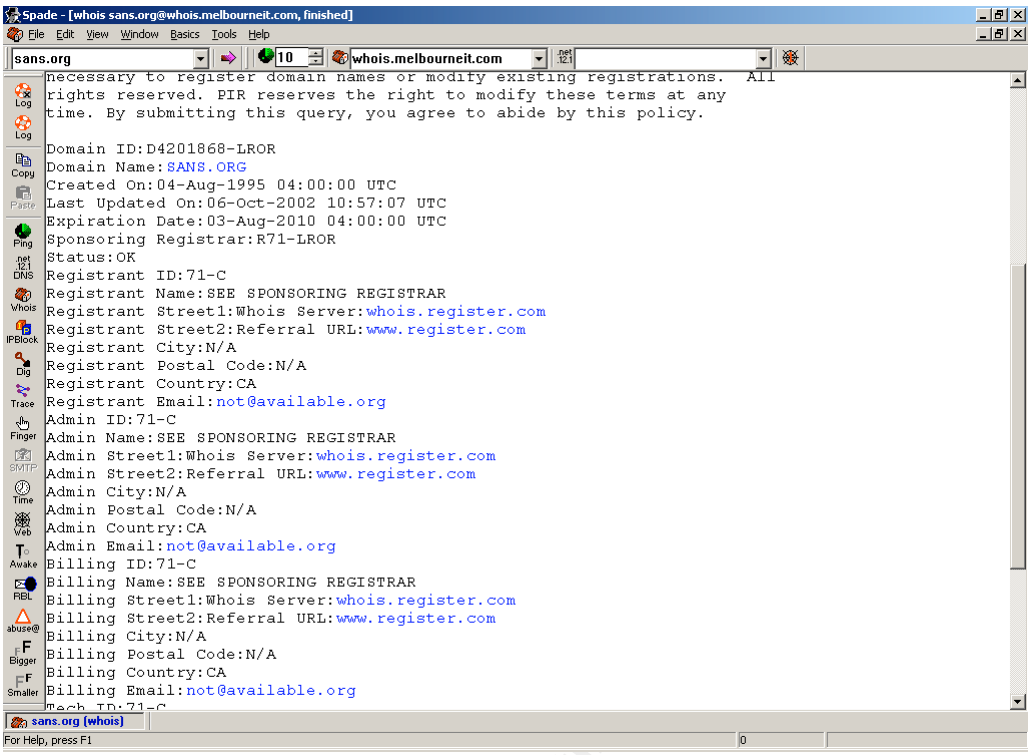
```
The data in Register.com's WHOIS database is provided to you by
Register.com for information purposes only, that is, to assist you in
obtaining information about or related to a domain name registration
record.  Register.com makes this information available "as is," and
does not guarantee its accuracy.  By submitting a WHOIS query, you
agree that you will use this data only for lawful purposes and that,
under no circumstances will you use this data to: (1) allow, enable,
or otherwise support the transmission of mass unsolicited, commercial
advertising or solicitations via direct mail, electronic mail, or by
telephone; or (2) enable high volume, automated, electronic processes
that apply to Register.com (or its systems).  The compilation,
repackaging, dissemination or other use of this data is expressly
prohibited without the prior written consent of Register.com.
Register.com reserves the right to modify these terms at any time.
By submitting this query, you agree to abide by these terms.


   Organization:
      SANS
      SANS SANS
      4610 Tournay Road
      Bethesda, MD 20816
      US
      Phone: 301-951-0102
      Fax..: 301-951-0104
      Email: hostmaster@sans.org

   Registrar Name....: Register.com
   Registrar Whois...: whois.register.com
   Registrar Homepage: http://www.register.com

   Domain Name: SANS.ORG

      Created on..............: Fri, Aug 04, 1995
      Expires on..............: Tue, Aug 03, 2010
      Record last updated on..: Fri, Aug 22, 2003

   Administrative Contact:
      SANS
      SANS SANS
      4610 Tournay Road
      Bethesda, MD 20816
      US
      Phone: 301-951-0102
      Fax..: 301-951-0104
      Email: hostmaster@sans.org

   Technical Contact:
      Register.Com
      Domain Registrar
      575 8th Avenue
```

```
   New York, NY 10018
   US
   Phone: 902-749-2701
   Fax..: 902-749-5429
   Email: domain-registrar@register.com

Zone Contact:
   Register.Com
   Domain Registrar
   575 8th Avenue
   New York, NY 10018
   US
   Phone: 902-749-2701
   Fax..: 902-749-5429
   Email: domain-registrar@register.com

Domain servers in listed order:

NS1.HOMEPC.ORG                                  66.129.1.102
NS2.HOMEPC.ORG                                  168.103.43.50
NS1.GIAC.NET                                    63.100.47.43
NS2.GIAC.NET                                    65.173.218.103
```

Online Tools for a domain name search – www.whois.net



Using Sam Spade Windows based GUI for a whois query



Using nmap

```
Nmap V. 3.00 Usage: nmap [Scan Type(s)] [Options] <host or net list>
Some Common Scan Types ('*' options require root privileges)
* -sS TCP SYN stealth port scan (default if privileged (root))
  -sT TCP connect() port scan (default for unprivileged users)
* -sU UDP port scan
  -sP ping scan (Find any reachable machines)
* -sF,-sX,-sN Stealth FIN, Xmas, or Null scan (experts only)
  -sR/-I RPC/Identd scan (use with other scan types)
Some Common Options (none are required, most can be combined):
* -O Use TCP/IP fingerprinting to guess remote operating system
  -p <range> ports to scan.  Example range: '1-1024,1080,6666,31337'
  -F Only scans ports listed in nmap-services
  -v Verbose. Its use is recommended.  Use twice for greater effect.
  -P0 Don't ping hosts (needed to scan www.microsoft.com and others)
* -Ddecoy_host1,decoy2[,...] Hide scan using many decoys
  -T <Paranoid|Sneaky|Polite|Normal|Aggressive|Insane> General timing
policy
  -n/-R Never do DNS resolution/Always resolve [default: sometimes
resolve]
  -oN/-oX/-oG <logfile> Output normal/XML/grepable scan logs to
<logfile>
  -iL <inputfile> Get targets from file; Use '-' for stdin
* -S <your_IP>/-e <devicename> Specify source address or network
interface
Example: nmap -v -sS -O www.my.com 192.168.0.0/16 '192.88-90.*.*'
SEE THE MAN PAGE FOR MANY MORE OPTIONS, DESCRIPTIONS, AND EXAMPLES
```

Using ScanLine

```
C:\>sl
ScanLine (TM) 1.01
Copyright (c) Foundstone, Inc. 2002
http://www.foundstone.com

sl [-?bhijnprsTUvz]
   [-cdgmq <n>]
   [-flLoO <file>]
   [-tu <n>[,<n>-<n>]]
   IP[,IP-IP]

 -?  - Shows this help text
 -b  - Get port banners
 -c  - Timeout for TCP and UDP attempts (ms). Default is 4000
 -d  - Delay between scans (ms). Default is 0
 -f  - Read IPs from file. Use "stdin" for stdin
 -g  - Bind to given local port
 -h  - Hide results for systems with no open ports
 -i  - For pinging use ICMP Timestamp Requests in addition to Echo
Requests
 -j  - Don't output "-----..." separator between IPs
 -l  - Read TCP ports from file
 -L  - Read UDP ports from file
 -m  - Bind to given local interface IP
 -n  - No port scanning - only pinging (unless you use -p)
 -o  - Output file (overwrite)
 -O  - Output file (append)
 -p  - Do not ping hosts before scanning
 -q  - Timeout for pings (ms). Default is 2000
 -r  - Resolve IP addresses to hostnames
 -s  - Output in comma separated format (csv)
 -t  - TCP port(s) to scan (a comma separated list of ports/ranges)
 -T  - Use internal list of TCP ports
 -u  - UDP port(s) to scan (a comma separated list of ports/ranges)
 -U  - Use internal list of UDP ports
 -v  - Verbose mode
 -z  - Randomize IP and port scan order

Example: sl -bht 80,100-200,443 10.0.0.1-200

This example would scan TCP ports 80, 100, 101...200 and 443 on all
IP addresses from 10.0.0.1 to 10.0.1.200 inclusive, grabbing banners
from those ports and hiding hosts that had no open ports.
```