http://www.ideahamster.org/

Open-Source Security Testing Methodology Manual

Written and Created by
Pete Herzog [pete@ideahamster.org]

current version:	osstmm.en.1.5 (Anniversary Edition)
notes:	This update is beyond a bug fix because it is significant enough to warrant internal document updates. The entire manual has been re-edited and cleaned up significantly. The deadline for version 2.0 is May 21st. Also, I want to say happy one-year anniversary to my wife and best friend, Marta.
date of current version:	Saturday, May 5, 2001
date of original version:	Monday, December 18, 2000
created by:	Pete Herzog [pete@ideahamster.org]
key supporters:	Marta Barceló [marta@marta.com] Clement Dupuis [cdupuis@cccure.org]
key contributors:	Don Bailey [baileydl@mitre.org] Michael S. Hines [mshines@purdue.edu] Miguel Angel Dominguez Torres [miguel-angel.dominguez@db.com] Angel Luis Urunuela [angel-luis.urunuela@db.com] Peter Klee [klee@de.ibm.com] Rich Jankowski [richj@lucent.com] Felix Schallock [felix.schallock@e-security-net.de] Vincent IP [VincentIpTingPong@hotmail.com]
edited by:	Drew Simonis [simonis@myself.com] Emily K. Hawthorn [ekh@mitre.org] Jordi Martinez i Barrachina [jordi.martinez-barrachina@db.com]
insightful commentary:	Robert Bagnall [robert.bagnall@veridian.com] Christopher H. Ray [solutions@ttlunlimited.com]
support:	Irwin [itprc@itprc.com]

Foreword

This manual is to set forth a standard for Internet security testing. Disregarding the credentials of many a security tester and focusing on the how, I present a solution to a problem that exists currently. Regardless of firm size, finance capital, and vendor backing, any network or security expert who meets the outline requirements in this manual is said to have completed a successful security snapshot. Not to say one cannot perform a test faster, more in depth, or of a different flavor. No, the tester following the methodology herein is said to have followed the standard model and therefore if nothing else, has been thorough.

I say security snapshot above because I believe an Internet security test is no more than a view of a system at a single moment in time. At that time, the known vulnerabilities, the known weaknesses, the known system configurations has not changed within that minute and therefore is said to be a snapshot. But is this snapshot enough?

The methodology proposed herein will provide more than a snapshot if followed correctly with no short-cuts and except for known vulnerabilities in an operating system or application, the snapshot will be a scattershot--encompassing perhaps a few weeks rather than a moment in time.

I have asked myself often if it is worth having a central standard for security testing. As I began to write down the exact sequence of my testing to share synchronously the active work of a penetration test, it became clear that what I was doing is not that unique. All security testers follow one methodology or another. But are all methodologies good?

All security information I found on the Internet regarding a methodology was either bland or secret. “We use a unique, in-house developed methodology and scanning tools….” This was a phrase found often. I remember once giving the advice to a CIO that if a security tester tells you his tools include ISS, Cybercop, and “proprietary, in-house developed tools” you can be sure he mainly uses ISS and Cybercop. That's not to say many don't have proprietary tools. I worked for IBM as an ethical hacker. They had the Network Security Auditor (NSA) that they now include in their firewall package. It was a good, proprietary tool with some nice reporting functions. Was it better than ISS or Cybercop? I couldn't say since we also used ISS to revalidate the NSA tests. This is due to the difficulty of keeping a vulnerability scanner up-to-date.

I feel it is valid to be able to ask companies if they meet a certain standard. I would be thrilled if they went above the standard. I would also know that the standard is what they charge a certain price for and that I am not just getting a port scan to 10,000 ports and a check of 4,800 vulnerabilities. Especially since most of which only apply to a certain OS or application. I'd like to see vulnerability scanners break down that number by OS and application. I know if I go into Bugtraq (the only true vulnerability checking is research on BT) that I will be able to find all the known vulnerabilities by OS and application. If the scanner checks for 50 Redhat holes in a certain flavor and 5 Microsoft NT holes and I'm an NT shop; I think I may try a different scanner.

So following an open-source, standardized methodology that anyone and everyone can open and dissect and add to and complain about is the most valuable contribution we can make to Internet security. And if you need a reason to recognize it and admit it exists (whether or not you follow it to the letter) it’s because you, your colleagues, and your fellow professionals have helped design it and write it. Supporting an open-source methodology does not reduce your ability – rather it shows you are just as good as all the other security testers. The rest is about firm size, finance capital, and vendor backing.

Contributions

Those who have been contributed to this manual in valuable ways have been listed at the top of this document. Each person receives recognition for the type of contribution although not as to what was contributed. The use of contribution obscurity in this document is for the prevention of biases. Only in translations of this document are persons actually noted since the translator becomes the contact of the translated document that includes not only keeping the document up to date but also FAQs, submissions and comments in that language to be translated into English for the original when appropriate.

Terms

Throughout this manual we refer to words and terms that may be construed with other intents or meanings. We will attempt to clarify most of them in the glossary at the end of this manual, however; it is important to note that there are a few which we make universal to fit our scope. They are as follows:

black-box
any testing that is done without prior knowledge, blindly but not randomly.

hacker
good or bad, novice or expert, a person who attempts to exploit or trick a computer system.

Internet presence
the thin veil which separates systems, services, and information between a network and the Internet.

invasive
trespassing by probing or attaching to non-public parts of a system or network.

passive
data collection by not probing or attaching to non-public parts of a system or network.

Red Team
the person or persons conducting a black-box penetration test or ethical hacking engagement.

white-box
any testing completed with privileged knowledge, i.e. having the source code for a program while testing.

Intended Audience

This manual is written for the Internet security professionals both developers and testers. Terms, skills, and tools mentioned in here may not make much sense to the novice or those not directly involved in Internet security. Networking professionals may also find this manual of use since much of security blurs between the IT networking department and the security professionals.

This manual does not examine the proper way to use particular software or network protocols or how to read the results. Evil hackers-in-the-making will find this a disappointing feature of the manual. Peoples concerned with this being another guide in how to hack for fun are mistaken. Evil hackers need to find only one hole. Security testers need to find them all. We are caught between the lesser of two evils and disclosure will at least inform in a structured, useful way those who need to defend their systems. So to disclose with this manual or not is truly a damned if you do and damned if you don't predicament. We choose disclosure. In choosing disclosure we have been sure not to include specific vulnerabilities or problems that can be abused and only offer a standard methodology.

Developers will find this manual useful in building better networks, firewalls, applications, and testing tools. Many of the tests do not currently have a way to automate them. Many of the automated tests do not follow a methodology in an optimal order. This manual will address these issues. Developers may feel free to address them as well. *Appendix D* addresses some tools that could be developed to assist some of the testing within this methodology.

Scope

This is a document of Internet security testing methodology, a set of rules and guidelines for solid penetration testing, ethical hacking, and information security analysis including the use of open source testing tools for the standardization of security testing and the improvement of automated vulnerability testing tools. It is within the scope of this document to provide a standardized approach to a thorough security assessment of an Internet presence of an organization. Within this standardized approach for thoroughness, we achieve an Open Standard for Internet Security Testing and use it as a baseline for all security testing methodologies known and unknown.

End Result

The ultimate goal is to set a standard in testing methodology which when used in either manual or automated security testing results in meeting operational security requirements for securing the Internet presence. The indirect result is creating a discipline that can act as a central point in all Internet security tests regardless of the size of the network, type of systems, or Internet applications.

Analysis and the Business Risk Assessment

Analysis is not within the scope of this document. This document maintains a "business" perspective that relates to the risk assessment. This document by no means forces the hand of the analyst but rather guides the hand of the auditor. The analysis of collected data is completely within the control of the security testing organization. While the business perspective helps form the scope with the assumption that the organization being tested is concerned about security, privacy, image, brand, time, and all the things where loss of money could be the inevitable result. Therefore, the security tester in this document takes the extended position as information security tester, privacy tester, systems security tester, policy tester, and marketing/business defenses tester. This is the position in which this manual will achieve its thoroughness.

BS7799 and ISO17799 Compliance

This document does not yet fully comply with all of the remote auditing and testing requirements of BS7799 (and its International equivalent ISO 17799) for information security testing. Once it is compliant, BS7799 (and ISO 17799) consultants will find this manual of great assistance in completing an information security audit on networked systems.

Risk and Sensitivity Assessment

This manual will treat risk assessment as the analysis of collected data. The security level to which this complies may depend on the country of the organization. The next section presents a sample of countries with strong data privacy and information security legislation and regulation for the framework of this manual and the quality of testing data to be analyzed.

Another aspect of this manual is to introduce offense measures to conduct market/business intelligence gathering. This document uses offensive and defensive market/business intelligence gathering techniques known as Competitive Intelligence as per the Society of Competitive Intelligence Professionals (SCIP) and the technique known as "Scouting" to compare the target organization's market/business positioning to the actual position as seen from other intelligence professionals on the Internet.

This document is also in compliance to the control activities found in the US General Accounting Office's (GAO) Federal Information System Control Audit Manual (FISCAM) where they apply to network security.

Legislation and Regulation Compliance

This manual was developed to satisfy the testing and risk assessment for personal data protection and information security in the following bodies of legislation. The tests performed provide the necessary information to analyze for data privacy concerns as per most Governmental legislations due to this manual's thorough testing stance. Although not all country statutes can be detailed herein, this manual has explored the various bodies of law to meet the requirements of strong examples of individual rights and privacy.

- USA Government Information Security Reform Act of 2000 section 3534(a)(1)(A)

- Deutsche Bundesdatenschutzgesetz (BDSG)-- Artikel 1 des Gesetzes zur Fortentwicklung der Datenverarbeitung und des Datenschutzes from 20. December 1990, BGBl. I S. 2954, 2955, zuletzt geändert durch das Gesetz zur Neuordnung des Postwesens und der Telekommunikation vom 14. September 1994, BGBl. I S. 2325
- Spanish LOPD Ley orgánica de regulación del tratamiento automatizado de los datos de carácter personal Art.15 LOPD -. Art. 5,
- Provincial Law of Quebec, Canada Act Respecting the Protection of Personal Information in the Private Sector (1993).

Process

A security test is performed with two types of attack. A passive attack is often a form of data collection which does not directly influence or trespass upon the target system or network. An intrusive attack however does trespass upon the target system or network and can be logged and used to alarm the target system or network.

The process of a security test concentrates on evaluating the following areas:

Visibility

Visibility is what can be seen on your Internet presence. This includes - but is not limited to - open or filtered ports, the types of systems, the architecture, the applications, email addresses, employee names, the skills of the new sys admin being hired through a job search online, the circulation your software products, and the websites visited by your employees and everything they download. Being invisible includes being able to step on wet sand and leave no footprint.

Access

Access is why people visit your Internet presence. This includes but is not limited to a web page, an e-business, a P2P server to content map, a DNS server, streaming video, or anything in which a service or application supports the definition of quasi-public, where a computer interacts with another computer within your network. Limiting access means denying all except what is expressly justified in the business plan.

Trust

Trust is the most important concept in Internet security. It is a measure of how much people can depend on what the system offers. Trust depends on the kind and amount of authentication, nonrepudiation, access control, accountability, data confidentiality, and data integrity employed by the system(s).

Sometimes trust is the basis for a service, for example when one computer links to another. Some trust “partnerships” include VPNs, PKIs, HTTPS, SSH, B2B connectors, database to server connections, e-mail, employee web surfing, or any communication between two computers which causes interdependency between two computers whether server/server, server/client, or P2P.

Alarm

Alarm is the timely and appropriate notification of activities that violate or attempt to violate Visibility, Access, or Trust. This includes but is not limited to log file analysis, port watching, traffic monitoring, intrusion detection systems, or sniffing/snooping. Alarm is often the weakest link in appropriate security measures.

Parameters

The methodology is broken down into *parameters* and *tasks*. The parameters are the flow of the methodology from one Internet Presence Point to the other. Each parameter has an input and an output. The input is the information used in performing each task. The output is the result of completed tasks. Output may or may not be analyzed data (also known as intelligence) to serve as an input for another parameter. It may even be the

case that the same output serves as the input for more than one parameter such as IP addresses or domain names.

Some tasks yield no output; this means that parameters will exist for which there is no input. Parameters which have no input can be ignored during testing. Ignored parameters do not necessarily indicate an inferior test; rather they may indicate superior security.

Parameters that have no output as the result can mean one of three things--

1. The tasks were not properly performed.
2. The tasks revealed superior security.
3. The task result data has been improperly analyzed.

It is vital that impartiality exists in performing the tasks of each parameter. Searching for something you have no intention of finding may lead to you finding exactly what you want. In this methodology, each parameter begins as an input and output exactly for the reason of keeping bias low. Each parameter gives a direction of what should be revealed to move further down the flow.

Time is relative. Larger projects mean more time spent at each parameter and on each task. The amount of time allowed before returning with output data depends on the tester and the scope of the testing. Proper testing is a balance of time and energy where time is money and energy is the limit of man and machine power.

Internet Presence Points

Security testing is a strategic effort. While there may be different ways and different tools to test many of the same parameters, there are few variations in the order in which to test them. Although some of the parameters mentioned here (specifically 2, 11, and 13) are not Internet presence points, they are worth noting due to the electronic nature and the lack of places of where they may fit in as a test of their own.

Internet presence points are every point in the Internet where an organization interacts with the Internet. These presence points are developed to offer as parameters in the methodology flow. Some of these parameters are:

1. Network Surveying
2. Port Scanning
3. System Fingerprinting
4. Wireless Leak Tests
5. Services Probing
    - Web Tracks
    - Mail Tracks
    - Name Services
    - Visible Documents
    - Anti-Virus and Trojan
6. Redundant Automated Vulnerability Scanning
7. Exploit Research
8. Manual Vulnerability Testing and Verification
9. Application Testing
10. Firewall & ACL Testing
11. Security Policy Review
12. Intrusion Detection System (IDS) Testing
13. Wardialing, Voicemail, & PBX Testing
14. Doc Grinding (Electronic Dumpster Diving)
    - News, Trade, and Business Sources
    - Job, Board, and Chat Searches
    - Newsgroups
    - Cracks, Serials, and Underground
    - FTP, Gopher
    - Web
    - P2P
15. Social Engineering
16. Trusted Systems Testing
17. Password Cracking
18. Denial of Service Testing
19. Privacy Policy Review
20. Cookie & Web Bug Analysis
21. IDS & Server Logs Review

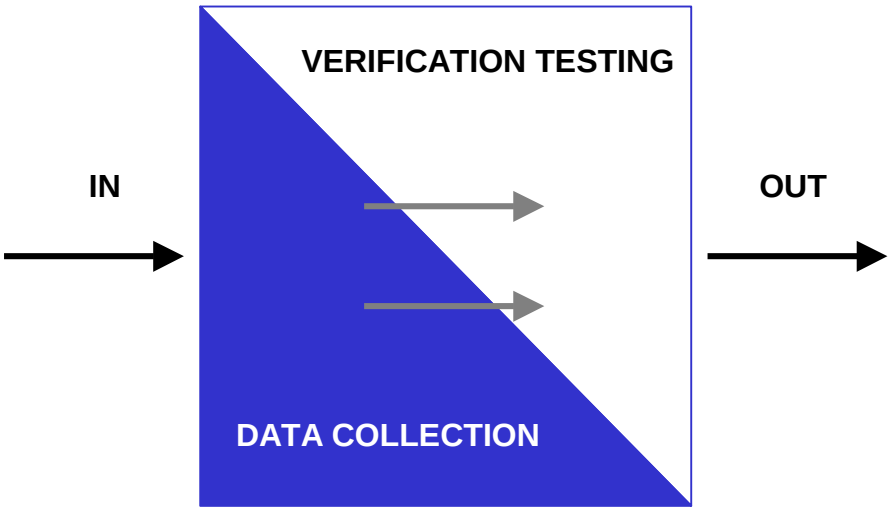
As you see there is a great amount of data to collect and analyze. The above steps can be graphed into a more visual form to help understand the flow of the testing.

Methodology

The methodology flows from the point of Network Surveying to the final report. An example of this flow would allow a separation between data collection and verification testing of and on that collected data. The flow would also determine the precise points of when to extract and when to insert this data.

In defining the methodology of testing, it is important to not constrict the creativity of the tester by introducing standards so formal and unrelenting that the quality of the test suffers. Additionally, it is important to leave tasks open to some interpretation where exact definition will cause the methodology to suffer when new technology is introduced. For example, verifying that the system uses proper encryption does not specify the techniques to be used for verification nor does it specify what kind of encryption. This is done on purpose. The parameter on vulnerability testing is especially open due to the dynamic nature of exploits.

In this methodology, we define *parameters* and *tasks*. Each parameter has a relationship to the one before it and the one after it. Security testing begins with an input that is ultimately the addresses of the systems to be tested. Security testing ends with the beginning of the analysis phase and the final report. This methodology does not affect the form, size, style, or content of the final report nor does it specify how the data is to be analyzed. That is left to the security tester or organization.



Parameters are the variables in security testing. The parameter requires an input to perform the tasks of the parameter. Tasks are the security tests to perform depending upon the input for the parameter. The results of the tasks may be immediately analyzed to act as a processed result or left raw. Either way, they are considered the output of the parameter. This output is often the input for a following parameter or in certain cases such as newly discovered hosts, may be the input for a previous parameter.

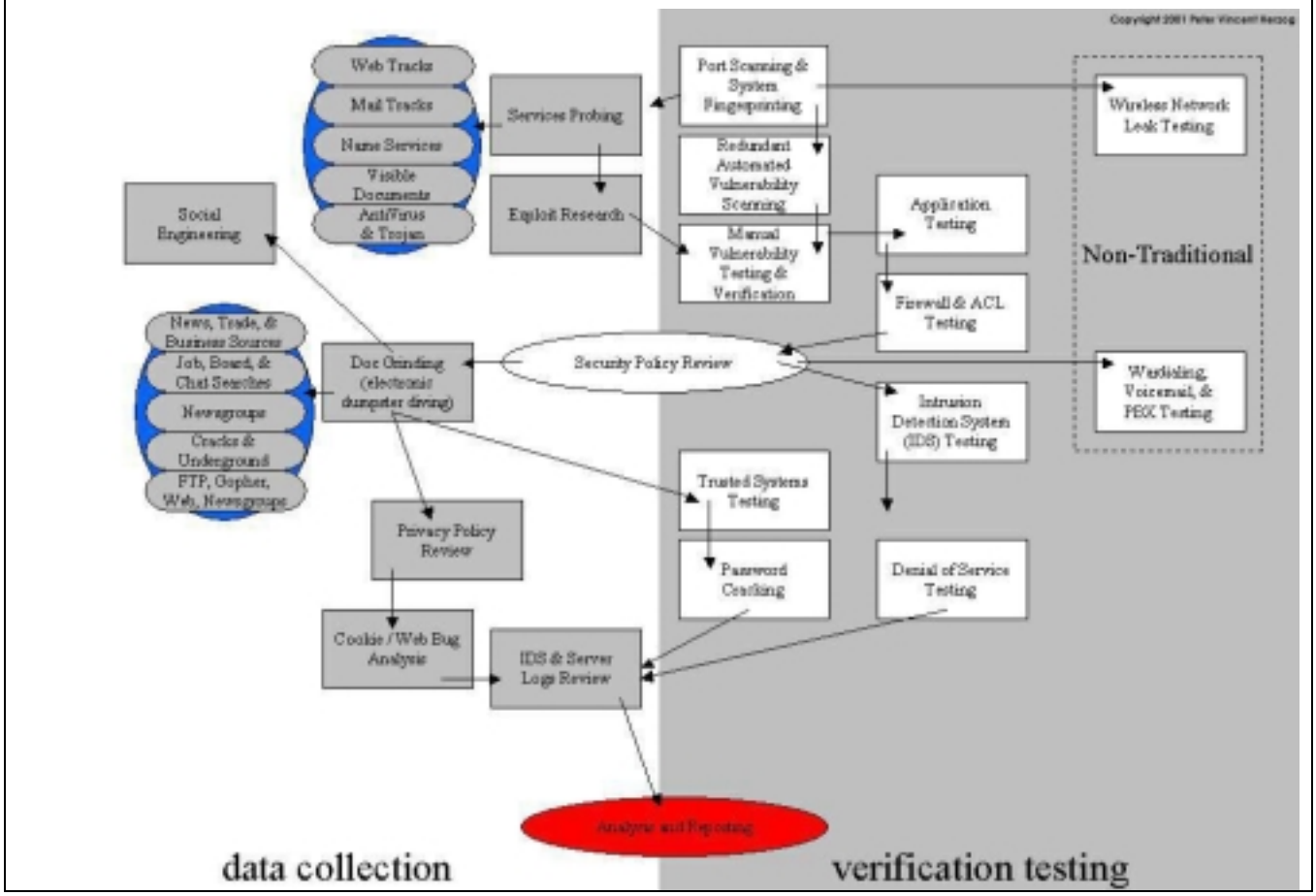
Non-traditional parameters refer to conditions where the components to test are not clearly defined as Internet Presence Points. A wireless LAN for example may bleed to the street where one can "remotely" test the network however must still be within close range. Telephone switches of organizations have long been the target of zealous phreakers (phone hackers) and therefore also the Security Tester. These two examples are considered part of a thorough security test.

Parameter Interdependency

In the above methodology we see a certain order in the flow that presents the possibility of running certain tests in parallel. For instance, IDS testing does not interfere with wardialing; also, neither test depends on the results of the other. However, both depend upon the review of the security policy to define certain parameters.

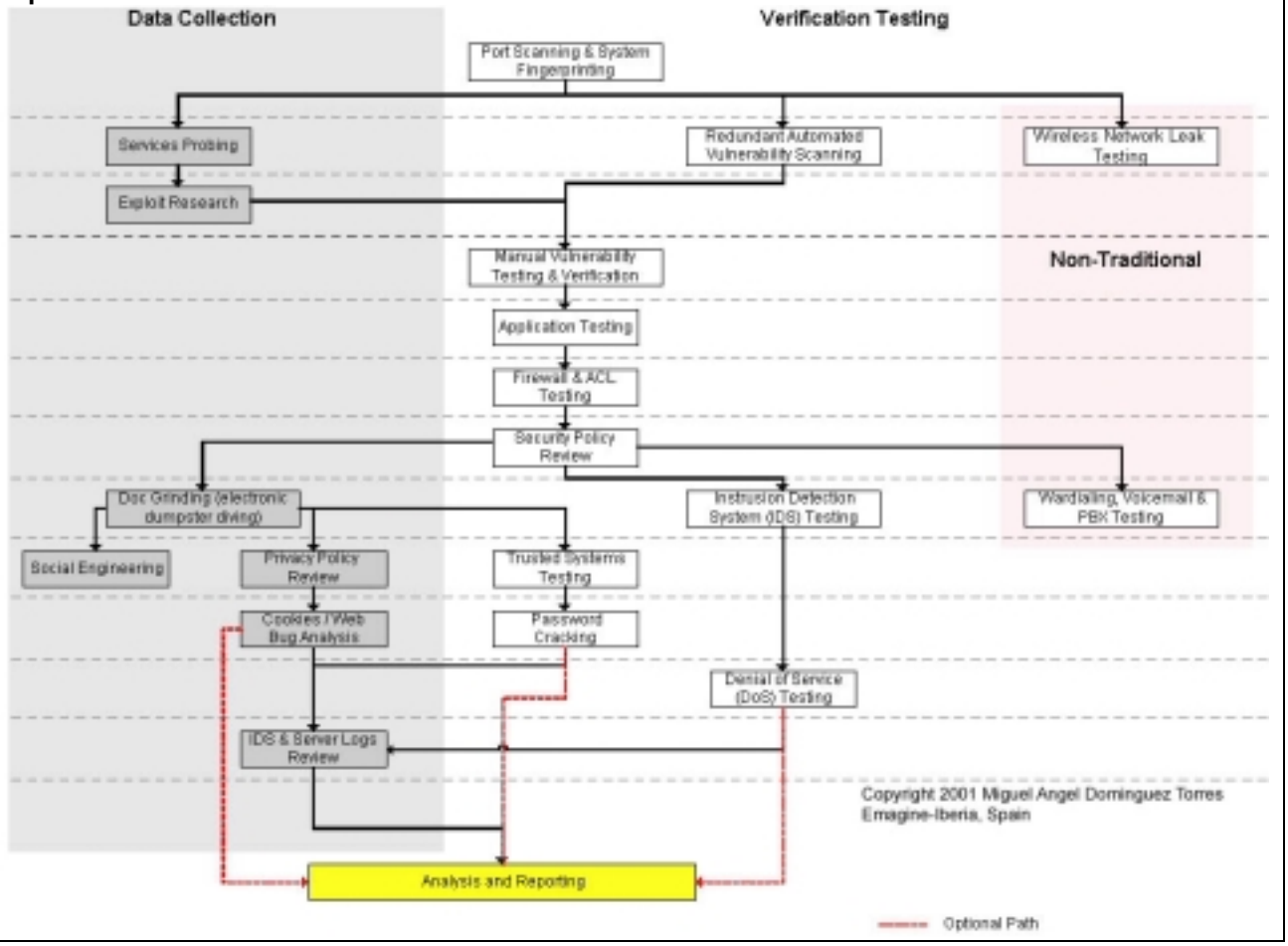
Example 1 shows the relationship of one test to the other in terms of dependencies. Each parameter may depend on a parameter before it to have the best result. This is the input / output model discussed previously. Therefore, each parameter begins with the explanation of what the expected output is to be (*Expected Results*). The input is a bit more complicated. For certain tests, it is up to the security tester to decide what input is best. In some tasks there is a default and an alternative. For example, in TCP port scanning, one can choose to scan all 65,536 ports or can scan what has been determined to be the standard set of problematic ports. In others such as Electronic Dumpster Diving (EDD) it is not so clear since the dependencies on the depth of the testing are so greatly varied.

Example 1



Example 2 shows the same tasks for parallel testing and data collection. This is very useful in team testing and automated testing tool design.

Example 2



Test Parameter Definition

The test parmeters are the steps to a complete and thorough Internet security test corresponding to the open standard. The path of the methodology can be explained best by describing the tasks to be performed, an explanation of each task, and the possible resources involved in the task.

If no public-domain tool or resource exists for a certain task, the explanation of the task will so state, and full details will be entered into **Appendix D -- Needed Resources** until such time as the request is fulfilled.

Currently, many more Internet services and applications exist (some dynamically I might add) than there are tests to specifically address them. For the sake of the methodology, it will be assumed that the tester addresses each service and application in kind wherever the terms Services or Applications are used with the appropriate tools or analysis for completing a thorough test.

Test Parameters and Tasks

Network Surveying

A network survey serves an introduction to the systems to be tested. It is best defined as a combination of data collection and information analysis. Although it is often advisable from a legal standpoint to define contractually exactly which systems to test if you are a third-party auditor or even if you are the system administrator, you may not be able to start with concrete system names or IP addresses. In this case you must survey and analyze. The point of this exercise is to find the number of reachable systems to be tested without exceeding the legal

limits of what you may test. Therefore the network survey is just one way to begin a test; another way is to be given the IP range to test. In this parameter, no intrusion is being performed directly on the systems except in places considered a quasi-public domain.

In legal terms, the quasi-public domain is a store that invites you in to make purchases. The store can control your access and can deny certain individuals entry but for the most part is open to the general public (even if it monitors them). This is the parallel to an e-business or web site.

Although not truly a parameter in the methodology, the network survey is a starting point. Often times, more hosts are detected during actual testing. Please bear in mind that the hosts discovered later may be inserted in the testing as a subset of the defined testing and often times only with permission or collaboration with the target organization's internal security team.

Expected Results	Domain Names Server Names IP Addresses Network Map ISP / ASP information System and Service Owners Possible test limitations

Tasks to perform for a thorough network survey include:

Name server responses.

- Examine Domain registry information for servers.
- Find IP block owned.
- Question the primary, secondary, and ISP name servers for hosts and sub domains.

Examine the outer wall of the network.

- Use multiple traces to the gateway to define the outer network layer and routers.

Examine tracks from the target organization.

- Search web logs and intrusion logs for system trails from the target network.
- Search board and newsgroup postings for server trails back to the target network.

Information Leaks

- Examine target web server source code and scripts for application servers and internal links.
- Examine e-mail headers, bounced mails, and read receipts for the server trails.
- Search newsgroups for posted information from the target.
- Search job databases and newspapers for IT positions within the organization relating to hardware and software.

Port Scanning

Port scanning is the invasive probing of system ports on the transport and network level. Included here is also the validation of system reception to tunneled, encapsulated, or routing protocols. This parameter is to enumerate live or accessible Internet services as well as penetrating the firewall to find additional live systems. The small sample of protocols here is for clarity of definition. Many protocols are not listed here. Testing for different protocols will depend on the system type and services it offers. For a more complete list of protocols, see Appendix F.

Each Internet enabled system has 65,536 TCP and UDP possible ports. However, it is not always necessary to test every port for every system. This is left to the discretion of the test team. Port numbers that are important for testing according to the service are listed with the task. Additional port numbers for scanning should be taken from the Consensus Intrusion Database Project Site.

Expected Results	Open, closed or filtered ports IP addresses of live systems

	List of discovered tunneled and encapsulated protocols List of discovered routing protocols supported Active services Network Map

Tasks to perform for a thorough Port Scan:

Error Checking

- Check the route to the target network for packet loss
- Measure the rate of packet round-trip time
- Measure the rate of packet acceptance and response on the target network
- Measure the amount of packet loss or connection denials at the target network

Enumerate Systems

- Collect broadcast responses from the network
- Probe past the firewall with strategically set packet TTLs (Firewalking) for all IP addresses.
- Use ICMP and reverse name lookups to determine the existence of all the hosts in a network.
- Use a TCP source port 80 and ACK on ports 3100-3150, 10001-10050, 33500-33550, and 50 random ports above 35000 for all hosts in the network.
- Use TCP fragments in reverse order with FIN, NULL, and XMAS scans[1] on ports 21, 22, 25, 80, and 443 for all hosts in the network.
- Use a TCP SYN on ports 21, 22, 25, 80, and 443 for all hosts in the network.
- Use DNS connect attempts on all hosts in the network.
- Use FTP and Proxies to bounce scans to the inside of the DMZ for ports 22, 81, 111, 132, 137, and 161 for all hosts on the network.

Enumerating Ports

- Use TCP SYN (Half-Open) scans to enumerate ports as being open, closed, or filtered on the default TCP testing ports in Appendix B for all the hosts in the network.
- Use TCP fragments in reverse order to enumerate ports and services for the subset of ports on the default Packet Fragment testing ports in Appendix B for all hosts in the network.
- Use UDP scans to enumerate ports as being open or closed on the default UDP testing ports in Appendix B if UDP is NOT being filtered already. [Recommended: first test the packet filtering with a very small subset of UDP ports.]

Encapsulated and Tunneled Protocols

- Verify and examine the use of SMB (Server Message Block) via IP.
- Verify and examine the use of NBT (NetBIOS-over-TCP).
- Verify and examine the use of IPX or IPX/SPX (Novell's network protocol) via TCP/IP.
- Verify and examine the use of RPC (Remote Procedure Call) and DCE RPC in the Internet presence
- Verify and examine the use of PPTP (Point to Point Tunneling Protocol).
- Verify and examine the use of L2TP (Layer 2 Tunneling Protocol).
- Verify and examine the use of IP in IP encapsulation.
- Verify and examine the use of SNMP (Simple Network Management Protocol).
- Verify and examine GRE (Generic Routing Encapsulation), IPSEC, and Radius support.

Routing Protocols

- Verify and examine the use of ARP (Address Resolution Protocol).
- Verify and examine the use of RIP (Routing Information Protocol).
- Verify and examine the use of OSPF (Open Shortest Path First) and Link State Advertisements (LSA).
- Verify and examine the use of BGP (Border Gateway Protocol).

System Fingerprinting

System fingerprinting is the active probing of a system for responses that can distinguish unique systems to operating system and version level.

[1] See nmap, Appendix A, for a description of xmas scans.

Expected Results	OS Type Patch Level System Type

Tasks to perform for a thorough System Fingerprint:

- Examine system responses to determine operating system type and patch level.
- Examine application responses to determine operating system type and patch level.
- Verify the TCP sequence number prediction for each live host on the network.
- Search job postings for server and application information from the target.
- Search tech bulletin boards and newsgroups for server and application information from the target.
- Match information gathered to system responses for more accurate results.

Services Probing

This is the active examination of the application listening behind the service. In certain cases more than one application exists behind a service where one application is the listener and the others are considered components of the listening application. A good example of this is PERL installed for use in a Web application. In that case the listening service is the HTTP daemon and the component is PERL.

Expected Results	Service Types Service Application Type and Patch Level Network Map

Tasks to perform for a thorough service probe:

- Match each open port to a service and protocol.
- Identify server uptime to latest patch releases.
- Identify the application behind the service and the patch level using banners or fingerprinting.
- Verify the application to the system and the version.
- Identify the components of the listening service.

Automated Vulnerability Scanning

Testing for vulnerabilities using automated tools is an efficient way to determine existing holes and system patch level. Although many automated scanners are currently on the market and in the underground, it is important for the tester to identify and incorporate the current underground scripts/exploits into this testing.

Expected Results	List of system vulnerabilities Type of application or service by vulnerability Patch levels of systems and applications

Tasks to perform for a thorough Vulnerability Scan:

- Measure the target organization against the currently popular scanning tools.
- Attempt to determine vulnerability by system type.
- Attempt to match vulnerabilities to applications.
- Attempt to determine application type and service by vulnerability.
- Perform redundant testing with at least 2 automated vulnerability scanners.

Exploit Research

This parameter covers the research involved in finding vulnerabilities up until the report delivery. This involves searching online databases and mailing lists specific to the systems being tested. Do not confine yourself to the web-- consider using IRC, Newsgroups, and underground FTP sites.

Expected Results	Patch levels of systems and applications List of possible denial of service vulnerabilities

Tasks to perform for a thorough Exploit Research:

- Identify all vulnerabilities according to applications.
- Identify all vulnerabilities according to operating systems.
- Identify all vulnerabilities from similar or like systems that may also affect the target systems.

Manual Vulnerability Testing and Verification

This parameter is necessary for eliminating false positives, expanding the hacking scope, and discovering the data flow in and out of the network. Manual testing refers to a person or persons at the computer using creativity, experience, and ingenuity to test the target network.

Expected Results	List of areas secured by obscurity or visible access List of actual vulnerabilities minus false positives List of Internal or DMZ systems List of mail, server, and other naming conventions Network map

Tasks to perform for a thorough Manual Testing and Verification:

- Verify all vulnerabilities found during the exploit research phase for false positives.
- Verify all positives (be aware of your contract if you are attempting to intrude or perform a denial of service).

Application Testing

This parameter refers to the testing of non-daemon[2] applications accessible from the Internet. These applications can be written in any language or script. They generally provide a business process, for example to receive user queries and provide responses. For example, a banking application might support queries to a checking or savings account.

Expected Results	List of applications List of application components List of application vulnerabilities List of application system trusts

Tasks to perform for a thorough Internet Application test:

- Decompose or deconstruct if necessary to access the source code.
- Examine the processes of the application.
- Test the inputs of the application.
- Examine the outputs of the application.
- Examine the communications, trusts, and relationships of the application.
- Determine the limits of authentication and access control.
- Measure the limitations of the defined variables.
- Examine the use of cacheing.

Firewall & Access Control List Testing

The Firewall and Screening Router are two defences often found on a network that control the flow of traffic between the enterprise network and the Internet. Both operate on a security policy and use ACLs (Access Control Lists). This parameter is designed to assure that only that which should be expressly permitted be allowed into the network; all else should be denied. However, this is often difficult when no written security policy exists and the analyst is forced to make assumptions as to the acceptable risk. This is not the job of the tester. The security tester must attempt to find the limits of the firewall and/or the screening router both as a system and as a service.

Expected Results	Information on the Firewall as a service and a system Information on the routers as a service Outline of the network security policy by the ACL

[2] A *daemon* is a system process that runs on a regularly scheduled basis to perform a system task. For example, FTPD is a daemon that handles file transfers with the FTP protocol.

	List of the types of packets which may enter the network List of the types of protocols with access inside the network List of live systems found.

Tasks to perform for a thorough Firewall & ACL Test:

- Verify the Firewall type with information collected from intelligence gathering.
- Verify the router types and configurations.
- Test the ACL against the written security policy or against the "Deny All" rule.
- Verify that the firewall is egress filtering local network traffic
- Verify that the firewall and/or router is performing address spoof detection
- Verify the penetrations from inverse scanning completed in the Port Scanning parameter.
- Verify the penetrations from strategically determined packet TTL settings (Firewalking) completed in the Port Scanning parameter.

Intrusion Detection System (IDS) Testing

This test is focused on the performance and sensitivity of an IDS. Much of this testing cannot be properly achieved without access to the IDS logs. Some of these tests are also subject to attacker bandwidth, hop distance, and latency that will affect the outcome of these tests.

Expected Results	Type of IDS Note of IDS performance under heavy load Type of packets dropped or not scanned by the IDS Type of protocols dropped or not scanned by the IDS Note of reaction time and type of the IDS Note of IDS sensitivity Rule map of IDS

Tasks to perform for a thorough IDS Test:

- Verify the IDS type with information collected from intelligence gathering.
- Test the IDS for configured reactions to multiple, varied attacks.
- Test the IDS for configured reactions to obfuscated URLs.
- Test the IDS for configured reactions to speed adjustments in packet sending.
- Test the IDS for configured reactions to source port adjustments.
- Test the IDS for the ability to handle fragmented packets.
- Test the IDS for configured reactions to the network traffic listening configuration in the designated network segment(s).
- Test the IDS for alarm states.
- Test the signature sensitivity settings over 1 minute, 5 minutes, 60 minutes, and 24 hours.
- Test the effect and reactions of the IDS against a single IP address versus various addresses.

Security Policy Review

The security policy noted here is the written human-readable policy document outlining the mitigated risks an organisation will handle with the use of specific types of technologies. This security policy may also be a human readable form of the ACLs. There are two functions to be performed: first, the testing of the against the actual state of the Internet presence and other non internet related connections; and second, to assure that the policy exists within the business justifications of the organisation, local and federal legal statutes, and personal privacy ethics.

These tasks require that the testing and verification of vulnerabilities is completely done and that all other technical reviews have been performed. Unless this is done you can't compare your results with the policy that should be met by measures taken to protect the operating environment.

Expected Results	List of all policy points differing from the actual state of the Internet presence Show non- approval from management List Inbound connections rules not met List Outbound connections rules not met

	List Security measures not met List of all policy points differing from the actual state of none internet connections List Modems rules not met List Fax machines rules not met List PBX rules not met

Tasks to perform for a thorough Security Policy review:

- Measure the security policy points against the actual state of the Internet presence.
  - *Approval from Management* -- Look for any sign (e.g. signature) that reveals that the policy is approved by management. Without this approval the policy is useless because staff is not required to meet the rules outlined within. From a formal point of view you could stop investigating the policy if it is not approved by management. However, testing should continue to determine how effective the security measures are on the actual state of the internet presence.
  - *Inbound connections* -- Check out any risks mentioned on behalf of the Internet inbound connections (internet->DMZ, internet -> internal net) and measures which may be required to be implemented to reduce or eliminate those risks. These risks could be allowed on incoming connections, typically HTTP, HTTPS, FTP, VPNs and the corresponding measures as authentication schemes, encryption and ACL. Specifically, rules that deny any stateful access to the internal net are often not met by the implementation.
  - *Outbound connections* -- Outbound connections could be between internal net and DMZ, as well as between internal net and the Internet. Look for any outbound rules that do not correspond to the implementation. Outbound connections could be used to inject malicious code or reveal internal specifics.
  - *Security measures* **--** Rules that require the implementation of security measures should be met. Those could be the use of IDS, firewalls, DMZs, routers and their proper configuration/implementation according to the outlined risks to be met.
- Measure the security policy points against the actual state of non-Internet connections.
  - *Modems* -- There should be a rule indicating that the use of modems that are not specially secured is forbidden or at least only allowed if the modems are powered down when not in use, and configured to disallow dial- in. Check whether a corresponding rule exists and whether the implementation follows the requirements.
  - *Fax machines* -- There should be a rule indicating that the use of fax machines which can allow access from the outside to the memory of the machines is forbidden or at least only allowed if the machines are powered down when not in use. Check whether a corresponding rule exists and whether the implementation follows the requirements.
  - *PBX* -- There should be a rule indicating that the remote administration of the PBX system is forbidden or at least only allowed if the machines are powered down when not in use. Check whether a corresponding rule exists and whether the implementation follows the requirements.

Document Grinding (Electronic Dumpster Diving)

The parameter here is important in the verification of much of the tested information and pertains to many levels of what is considered information security. The amount of time granted to the researching and extraction of information is dependent upon the size of the organisation, the scope of the project, and the length of time planned for the testing. More time however, does not always mean more information but it can eventually lead to key pieces of the security puzzle.

Expected Results	(See Appendix E for the default profile template) A profile of the organization A profile of the key employees A profile of the organization's network

Tasks to perform for a thorough Document Grind:

- Examine web databases and caches concerning the target organization and key people.
- Investigate key persons via personal homepages, published resumes, and organizational affiliations.
- Compile e-mail addresses from within the organization and personal e-mail addresses from key people.
- Search job databases for skill sets technology hires need to possess in the target organization.
- Search newsgroups for references to and submissions from within the organization and key people.
- Search documents for hidden codes or revision data.

Competitive Intelligence

CI Scouting is the scavenged information from an Internet presence that can be analysed as business intelligence. Different than the straight-out intellectual property theft found in industrial espionage or hacking, CI lends to be non-invasive and much more subtle. It is a good example of how the Internet presence extends far beyond the hosts in the DMZ. Using CI in a penetration test gives business value to the components and can help in finding business justifications for implementing various services.

Expected Results	A measurement of the organization's network business justifications Size and scope of the Internet presence A measurement of the security policy to future network plans

Tasks to perform for a thorough Competitive Intelligence Scouting:

- Map and weigh the directory structure of the web servers
- Map the weigh the directory structure of the FTP servers
- Examine the WHOIS database for business services relating to registered host names
- Estimate the IT cost of the Internet infrastructure based on OS, Applications, and Hardware.
- Estimate the cost of support infrastructure based on regional salary requirements for IT professionals, job postings, number of personnel, published resumes, and responsibilities.
- Measure the buzz (feedback) of the organization based on newsgroups, web boards, and industry feedback sites
- Estimate the number of products being sold electronically (for download)
- Estimate the number of products found in P2P sources, wares sites, available cracks up to specific versions, and documentation both internal and third party about the products

Trusted Systems Testing

The purpose of testing system trusts is to affect the Internet presence by posing as a trusted entity of the network. The testing scenario is often more theory than fact and does more than blur the line between vulnerability testing and Firewall/ACL testing-- it is the line.

Expected Results	Map of systems dependent upon other systems Map of applications with dependencies to other systems Types of vulnerabilities which affect the trusting systems and applications

Tasks to perform for a thorough Trusted Systems test:

- Verify possible relationships determined from intelligence gathering, application testing, and services testing.
- Test the relationships between various systems through spoofing or event triggering.
- Verify which systems can be spoofed.
- Verify which applications can be spoofed.

Password Cracking

Password cracking is the process of validating password strength through the use of automated password recovery tools that expose either the application of weak cryptographic algorithms, incorrect implementation of cryptographic algorithms, or weak passwords due to human factors. This parameter should not be confused with password recovery via sniffing clear text channels, which may be a more simple means of subverting system security, but only due to unencrypted authentication mechanisms, not password weakness itself. [Note: This parameter could include manual password guessing techniques, which exploits default username and password combinations in applications or operating systems (e.g. Username: System Password: Test), or easy-to-guess passwords resulting from user error (e.g. Username: joe Password: joe). This may be a means of obtaining access to a system initially, perhaps even administrator or root access, but only due to educated guessing. Beyond manual password guessing with simple or default combinations, brute forcing passwords for such applications as Telnet, using scripts or custom programs, is almost not feasible due to prompt timeout values, even with multi-connection (i.e. simulated threading) brute force applications.]

Once gaining administrator or root privileges on a computer system, password cracking may assist in obtaining access to additional systems or applications (thanks to users with matching passwords on multiple systems) and is a valid technique that can be used for system leverage throughout a security test. Thorough or corporate-wide password cracking can also be performed as a simple after-action exercise and may highlight the need for stronger encryption algorithms for key systems storing passwords, as well as highlight a need for enforcing the use of stronger user passwords through stricter policy, automatic generation, or pluggable authentication modules (PAMs).

Expected Results	Password file cracked or uncracked List of login IDs with user or system passwords List of systems vulnerable to crack attacks List of documents or files vulnerable to crack attacks List of systems with user or system login IDs using the same passwords

Tasks to perform for a thorough Password Cracking verification:

- Obtain the password file from the system that stores usernames and passwords
  - For Unix systems, this will be either /etc/passwd or /etc/shadow
  - For Unix systems that happen to perform SMB authentication, you can find NT passwords in /etc/smbpasswd
  - For NT systems, this will be /winnt/repair/Sam._ (or other, more difficult to obtain variants)
- Run an automated dictionary attack on the password file
- Run a brute force attack on the password file as time and processing cycles allow
- Use obtained passwords or their variations to access additional systems or applications
- Run automated password crackers on encrypted files that are encountered (such as PDFs or Word documents) in an attempt to gather more intelligence and highlight the need for stronger document or file system encryption.
- Verify password aging.

Denial of Service Testing

Denial of Service (DoS) is a situation where a circumstance, either intentionally or accidentally, prevents the system from functioning as intended. In certain cases, the system may be functioning exactly as designed however it was never intended to handle the load, scope, or parameters being imposed upon it.

It is very important that DoS testing receives additional support from the organization and is closely monitored.

Expected Results	List weak points in the Internet presence including single points of failure Establish a baseline for normal use List system behaviors to heavy use List DoS vulnerable systems

Tasks to perform for a thorough DoS test:

- Verify that administrative accounts and system files and resources are secured properly and all access is granted with "Least Privilege".
- Check the exposure restrictions of systems to non-trusted networks
- Verify that baselines are established for normal system activity
- Verify what procedures are in place to respond to irregular activity.
- Verify the response to SIMULATED negative information (propaganda) attacks.
- Test heavy server and network loads.

Privacy Policy Review

The privacy policy is the focal point of the organisation's stance on customer privacy. This policy must be publicly viewable. In cases where this policy does not exist, it is necessary to use the local privacy legislation of the target organization.

Expected Results	List any disclosures List compliance failures between public policy and actual practice

	List systems involved in data gathering List data gathering techniques List data gathered

Tasks to perform for a thorough Privacy Policy review:

- Compare publicly accessible policy to actual practice
- Identify database type and size for storing data
- Identify data collected by the organization
- Identify storage location of data
- Identify cookie types
- Identify cookie expiration times
- Identify information stored in cookie
- Verify cookie encryption methods
- Identify server location of web bug(s)
- Identify web bug data gathered and returned to server

IDS & Server Logs Review

Reviewing the server logs is needed to verify the tests performed on the Internet presence especially in cases where results of the tests are not immediately visible to the tester. Many unknowns are left to the analyst who has not reviewed the logs

Expected Results	List of IDS false positives List of IDS missed alarms List of packets which entered the network by port number List of protocols which entered the network List of unmonitored paths into the network

Tasks to perform for a thorough IDS and Server Log review:

- Test the Firewall, IDS, and Server logging process.
- Match IDS alerts to vulnerability scans.
- Match IDS alerts to password cracking.
- Match IDS alerts to trusted system tests.
- Verify TCP and UDP scanning to server logs.
- Verify automated vulnerability scans.
- Verify services' logging deficiencies.

Social Engineering

This is a method of gaining valuable information about a system by querying personnel. For example pretend to be an authority figure, call an administrator and tell him/her that you forgot your password and need immediate access so as to not lose a very important client (money). Many situations can be made up, depending what information you already gained about the organisation you are auditing. In some cases it is good to construct a situation which creates a lot of pressure on the victim (to get information fast). This way of gathering information is often very time-consuming and therefore only applicable if enough resources are available.

Expected Results	Useful information for obtaining access or about insecurities

Tasks to perform for a thorough Social Engineering test:

- Select victim from information already gained about personnel
- Examine the contact methods (via telephone, e-mail, Newsgroups, Chat etc.) for victim from the target organisation.
- Gather information about victim (position, habits, preferences)
- Make up a situation to get in contact (telephone, pretend to be an authority or restaurant, socialize with victim)
- Gather information from victim
- Verify levels of information insecurity susceptibility based on total non-disclosure as the baseline.

Wireless Leak Tests

Wireless Leak Testing

Expected Results	Find the outer-most wireless edge of the network Find access points into the network

Tasks to perform for a thorough Wireless Network test:

- Verify the distance in which the wireless communication extends beyond the physical boundaries of the organization
- Verify that the communication is secure and cannot be challenged or tampered
- Probe network for possible DoS problems

PBX Testing

Securing your organisation's Private Branch Exchange (PBX) systems will help prevent toll-fraud and theft of information.

Expected Results	Find voice mailboxes that are world accessible Find PBX Systems that are allowing remote administration List systems allowing world access to the maintenance terminal List all listening and interactive telephony systems.

Tasks to perform for a thorough PBX test:

- Verify that voicemail PINS are changed often.
- Review call detail logs for signs of abuse.
- Ensure administrative accounts don't have default, or easily guessed, passwords.
- Make sure OS is up to date and patched.
- Check for remote maintenance access to system.
- Check the physical security of maintenance terminal.
- Identify modems, faxes, and automated operators.
- Test dial-in authentications.
- Verify remote dial-in authentication.

Appendix A - Open Source Software Tools

The tools in the section are related to Internet Security Testing. Many of these tools are open source. We recommend you review the code of whatever you implement. Be aware that any tool you download and execute may have spying or Trojan features.

Please be advised that currently some of the descriptions found here have been taken directly off the website of the tool or the security site linking to the tool.

www

md-webscan-1.0.1.tar..> a high quality CGI vulnerability scanner. It is well written, easily extensible, and has a few nifty options. Changes: 106 new checks. Homepage here. By Mordrian

rivat.tgz Rivat is a distributed CGI scanner written in perl which scans for over 405 vulnerabilities. Homepage: http://www.r00tabega.com. By Xtremist

whisker-1.4.0.tar.gz whisker is what I've dubbed a 'next generation' CGI scanner. It is Scriptable. It's a programming-ish language that is tailored to do lots of flexible web scanning. Very stealthy. I've implemented anti-IDS checks into the scan. Includes over 200 checks. Lots of options. Reads in nmap output, files full of domains, or single host. Virtual host support. Proxy support. Can be used as a CGI. Changes: Includes 10 anti-IDS tactics, brute force user names, brute force basic authentication guessing, now uses perl modules if available for extra speed, HTTP return values can be redefined, can now be used as a CGI, html output, SSL support, more vulnerabilities in the scan.db, and bug fixes. Homepage: http://www.wiretrip.net. By Rain Forrest Puppy
Project 2068 Allows for the testing of passwords for simple authentication on web servers. By supplying a host and a dictionary file, the program will try to brute force the username and password on the webserver, and return the successful password when found.

cgi-check99 0.3 This is one of the worlds most cross platform cgi scanners, running on 37 operating systems! Even Palmos soon! Will check for hundreds of common cgi and other remote issues. Plus it will report you the Bugtraq ID of some vulnerabilities. Get the rebol interpreter at http://www.rebol.com.

Flatline 0.80 Web Server vulnerability scanner, beta version for linux, BSD. Options include mass host scanning, scanning through proxies, Detection evasion, quick banner grab scans, interactive mode to send specific url's. Also includes sample exploit database if a vulnerable file is found it will print a BugTraq ID or way to exploit the file. This is a semi beta release lots of new things to come.

dns

zodiac-0.4.9.tar.gz Zodiac is a portable, extensible and multithreaded DNS tool. It is meant to be used as a DNS packet monitor and DNS protocol test and debuging tool. It's basic features are: sniffing of DNS datagrams on an ethernet device, decoding of all types of DNS packets, including safe decompression (partly finished, SOA record are, for example, not decoded yet), nice display and gui, if you like ncurses and text based frontends, always interactive in all situations through built in command line, threaded and flexible design. Advanced features include: local DNS spoof handler, jizz DNS spoof, exploiting a weakness in old bind implementations, determines jizz-weakness, id-prediction and resolver type remotely, id spoofing, exploiting a weakness in the dns protocol itself, implements some advanced DNS denial of service attacks, including flood, label compression and unres attack, advanced DNS smurf. Changes: Now runs on *BSD, and fixed some bugs. Homepage: http://www.team-teso.net. By Scut

Toplevel DNS Scanner 0.02 The Toplevel Domain Scanner, commonly referred to as TDS, is a tool for scanning through DNS records. It allows you to plug scanned data into security software that checks your networks for holes, or look for weaknesses in other networks.

pop

trypop3.c Some code I put together to do some testing on the POP3 daemons on some machines installed at work. Attempts to overflow user/password variables. Homepage here. By Missnglnk

smtp

relaycheck.pl Relay Check.

scanners

sara-3.3.5.tar.gz Security Auditor's Research Assistant (SARA) is a security analysis tool based on the SATAN model. It is updated twice a month to address the latest threats. Checks for common old holes, backdoors, trust relationships, default cgi, common logins, open shares, and much more. Changes: Now detects the SunOS snmpXdmid remote root bug, and te lion worm. We also enhanced our SSH detection logic to address not only the new vulnerabilities but also its use as a backdoor. Homepage: http://www-arc.com/sara. By Advanced Research Corporation.

Nmap Network and Host scanner, which reveals open, filtered or closed ports. Has the ability to make OS assumptions based on packet signatures. The tool uses intrusive detection which can be revealed by IDS.

- Vanilla TCP connect() scanning,
- TCP SYN (half open) scanning,
- TCP FIN, Xmas, or NULL (stealth) scanning,
- TCP ftp proxy (bounce attack) scanning
- SYN/FIN scanning using IP fragments (bypasses some packet filters),
- TCP ACK and Window scanning,
- UDP raw ICMP port unreachable scanning,
- ICMP scanning (ping-sweep)
- TCP Ping scanning
- Direct (non portmapper) RPC scanning
- Remote OS Identification by TCP/IP Fingerprinting, and
- Reverse-ident scanning.

nessus Active vulnerability scanner which is actively maintained. It has a database for known vulnerabilities and includes CVE links. It uses intrusive detection which can be revealed by IDS.

The "Nessus" Project aims to provide to the internet community a free, powerful, up-to-date and easy to use and remote security scanner. A security scanner is a program which will audit remotely a given network and determine whether bad guys (aka 'crackers') may break into it, or misuse it in some way.

DataPool (DoS) *This is a really cool DoS tool. Use it with caution*. An extensive collection of 3 shell scripts used in conjunction with 69 DoS tools to analyze vulnerabilities of a certain system. Options to scan only, log to disk, select port range. Ability to work on a range of IPs, and to loop the script until a weakness is found. Compiled in Linux with sources. Only brief testing done so far on local machine, and only on Linux.

Egressor 1.0 MITRE has released a freeware tool that allows a company to check the configuration of their Internet point-of-presence router. The tool will help companies determine whether their routers are configured to the Help Defeat Denial of Service Attacks guidelines. This configuration of egress filtering reduces the chance that their computers can unwittingly contribute to a distributed denial of service attack. The tool has two parts; a generator and a receiver. The test generator (or "client") is being provided as C source code and the test receiver (or "server") is a PERL script. Both are currently known to work on LINUX, and the server also works on Solaris.

MNS 0.91beta MNS-v.91beta is the Multifunctional Network Scanner. It is *THE* classless network auditing and vulnerability logging package. It uses the latest methods of stealthing, os detection, and vulnerability checking. This is the last beta release before official release 1.0, so please download it, test it, and tell us your thoughts. You may need to download and install libpcap-0.4 which can be accessed through the 'libs' link on this page.

Narrow Security Scanner (Unix/Perl) 2000pre12 Narrow Security Scanner 2000 (Unix /Perl) searches for 367 remote vulnerabilities.

Pandora for Linux v4 beta 2.1 Offline password auditing and Online attack for the X Windows platform on Linux. (includes source code) The Online software requires an IPX-aware kernel and root. THIS SOFTWARE IS SUBJECT TO CHANGE FAIRLY QUICKLY, SO CHECK http://www.nmrc.org/pandora/ OFTEN. Gunzip then untar in your fave directory and build an icon to online/Pan-on in your fave X Windows manager. Hack Netware 4 and 5 from Linux!

SAINT - Security Administrator's Integrated Network Tool 3.1.1 SAINT (Security Administrator's Integrated Network Tool) is a security assessment tool based on SATAN. It is updated regularly and scans for just about all remotely detectable vulnerabilities. Features include scanning through a firewall, updated security checks from CERT & CIAC bulletins, 4 levels of severity (red, yellow, brown, & green) and a feature rich HTML interface.

Scowl CGI scanner You can easily add new bugs. Very fast using threads. Warns you for hosts that return false positive answers. Currently scans for more than 400 bugs. Freeware.

Messala An advanced vulnerability scanner.

Vetescan Vetescan is a bulk vulnerability scanner which contains programs to check for and/or exploit many remote network security exploits that are known for Windows or UNIX. It includes various programs for doing different kinds of scanning. Fixes for vulnerablities are included along with the exploits.

Firewalk Firewalking is a technique developed by MDS and DHG that employs traceroute-like techniques to analyze IP packet responses to determine gateway ACL filters and map networks. Firewalk the tool employs the technique to determine the filter rules in place on a packet forwarding device. The newest version of the tool, firewalk/GTK introduces the option of using a graphical interface and a few bug fixes.

RvScan (http://www.ussrback.com/UNIX/scanners/fts-rvscan.v2-b3.tgz)
(remote vulnerability scanner) determines the remote operating system, then procedes to find common vulnerabilites. New features: dual OS guessing [telnet banner grabbing + nmap OSScan], remote exploit checks [bind, imap, wuftpd, rpc.mountd, qpop, sendmail, iquery], multiple pop3 authentication, anonymous ftp services, httpd exploits [cold fusion, website pro, frontpage extensions, 52 vulnerable cgis], icmp echo filters, nfs exports, and over 10 sendmail holes. By ben-z.

misc

aes-netcat.tgz aes-netcat is a patch with some includes for netcat 1.10 that adds an option to do password encrypted sessions using strong AES encryption. Doesn't include an AES algorithm but urls where to get them (15 candidates available).By Mixter

Netcat (http://www.l0pht.com/~weld/netcat/) Linux/Unix tool made by Hobbit. Can provide you a remote shell on self defined ports. Very good for systems where firewalls allowing connections only on defined ports, e.g. kill the webserver process and let netcat run on port 80, there you go with a remote shell. An encrypted (twofish) version is available too (farm9.com/content/Free_Tools/Cryptcat)

bing-104.tar.bz2 Bandwidth Ping. Estimates bandwidths between network hosts and routers.

casl20.tgz Custom Auditing Scripting Language (CASL) implements a packet shell environment for the Custom Auditing Scripting Language that is the basis for the Cybercop(tm) line of products by Network Associates. The CASL environment provides an extremely high performance environment for sending and receiving any normal and/or morbid packet stream to firewalls, networking stacks and network intrusion detection systems as well as being sufficiently rich of a language to write honeypots, virtual firewalls, surfer hotel, phantom networks and jails. By Timothy Newsham and Thomas Ptacek

hping2 Beta 54 A network tool able to send custom ICMP/UDP/TCP packets and to display target replies like ping does with ICMP replies. It handles fragmentation and arbitrary packet body and size, and can be used to transfer files under supported protocols. Using hping2, you can: test firewall rules, perform [spoofed] port scanning, test net performance using different protocols, packet size, TOS (type of service), and fragmentation, do path MTU discovery, tranfer files (even between really Fascist firewall rules), perform traceroute-like actions

under different protocols, fingerprint remote OSs, audit a TCP/IP stack, etc. hping2 is a good tool for learning TCP/IP.

Merlin Merlin is a tool which was written to correlate and manage the output from other security tools. It is written in perl, and is easily configurable to add tools, and modify its reporting formats.

NLog 1.6.0 NLog is a set of PERL scripts for managing an analyzing your nmap 2.0 log files. It allows you to keep all of your scan logs in a single searchable database. The CGI interface for viewing your scan logs is completly customizable and easy to modify and improve. The core CGI script allows you to add your own extension scripts for different services, so all hosts with a certain service running will have a hyperlink to the extension script.

Overloaded Jport 2.0 Jport is a port scanner written in Java that will scan upto 65535 ports in under 30 seconds. It also contains new features such as port identification and a compleatly new threading system that does not allow any thread to be idle, hence increasing speed.

p0f 1.7 p0f performs passive OS detection based on SYN packets. Unlike nmap, p0f does recognition without sending any data. Additionally, it is able to determine distance to remote host, and can be used to determine the structure of a foreign or local network. When running on the gateway of a network it is able to gather huge amounts of data and provide useful statistics. On a user-end computer it could be used as powerful IDS add-on. p0f supports full tcpdump-style filtering expressions, and has an extensible and detailed fingerprinting database. New command-line functionality and getopt() support, fixes for bugs in verbose mode and the last big-endian problems, many new OS signatures, and a new die_nicely() routine.

SpiderMap 0.1 Spidermap is a collection of perl scripts which enable you to launch precisely tuned network scans. The goal of this project is to create an integrated suite of tools for low-impact network reconnaisance with features including custom packet rates and scan types for each network with increased efficiency by mapping multiple networks in parallel. The target users are system administrators and network security professionals seeking a non-destructive way to inventory network services and do so in a reasonable amount of time.

Dsniff A collection of tools for network auditing and penetration testing. dsniff, filesnarf, mailsnarf, msgsnarf, urlsnarf, and webspy passively monitor a network for interesting data (passwords, e-mail, files, etc.). arpspoof, dnsspoof, and macof facilitate the interception of network traffic normally unavailable to an attacker (e.g, due to layer-2 switching). sshmitm and webmitm implement active monkey-in-the-middle attacks against redirected SSH and HTTPS sessions by exploiting weak bindings in ad-hoc PKI.

Hunt Advanced packet sniffer and connection intrusion. Hunt is a program for intruding into a connection, watching it and resetting it. . Note that hunt is operating on Ethernet and is best used for connections which can be watched through it. However, it is possible to do something even for hosts on another segments or hosts that are on switched ports.

Fragrouter Fragrouter is aimed at testing the correctness of a NIDS,according to the specific TCP/IP attacks listed in the Secure Networks NIDS evasion paper. [2] Other NIDS evasion toolkits which implement these attacks are in circulation among hackers or publically available, and it is assumed that they are currently being used to bypass NIDSs

Tcpdump (http://www.tcpdump.org) Network sniffer with a lot of features.

snmp

snmpscan-0.05.tar.gz snmpscan scans hosts or routers running SNMPD for common communities (passwords). Communities on routers and hosts running snmpd (the simple network management protocol daemon) often have simple communities set. With a community, you can view various stats about a given machine or router, and often times actually make changes to the host. Use this tool to test and eventually secure your snmp devices. Changes: First release. By Knight, phunc.

crackers

Pwdump3 (http://www.ebiz-tech.com/pwdump3/)
Dumps NT/2000 passwords from registry/sam remotely.

John the Ripper, http://www.openwall.com/john/
John is a Unix password cracker which is available for Unix, DOS and NT.

l0phtcrack, (http://www.securitysoftwaretech.com/lc3/)
A Windows NT password sniffer and cracker, very usable for easy internal LAN hacking.

ftp

ftpscan.c This is useful for see if there are any world writeable directories.

ssl

bjorb A ssl wrapper (for using with brute-force tools).
sslwrap Another ssl wrapper.

packet generator

ipsend - generates TCP/IP packets with a scripting language.

SPAK Module generates TCP/IP packets by a shell pipe (IP, TCP, UDP).

udpprobe Send and receive UDP Packets.

Libnet is an API to help with the construction and handling of network packets. It provides a portable framework for low-level network packet writing and handling (use libnet in conjunction with libpcap and you can write some really cool stuff).

SendIP Project Purple's Command Line IP Packet Sender (large amount of options) (Mike Ricketts)

nemesis is a command-line UNIX network packet injection suite based on libnet.

Appendix B - Standard Testing Ports

Listed here are the standard ports to be tested based on the testing task in the Port Scanning parameter. Many organizations wish to or need to have all 65,536 ports scanned for completeness. What is offered here is a representation of the ports a thorough scan must involve. Please consider your resources and your objectives before deciding whether to use the standard or to actually test all ports.

Starting Point

The standard testing ports also include dynamic ports of which it is not possible to list in this static document. The dynamic ports chosen are from the list of "All Destination Ports for past 30 days" as found at the CID (Consensus Intrusion Database) Project.

Standard UDP ports

```
1, 2, 3, 5, 7, 9, 11, 13, 17, 18, 19, 20, 21, 22, 23, 24, 25, 27, 29, 31, 33, 35, 37, 38, 39, 41, 42,
43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 61, 62, 63, 64, 65, 66, 67, 68, 69,
70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 88, 89, 90, 91, 92, 93, 94, 95, 96,
97, 98, 99, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 117, 118,
119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135, 136, 137, 138,
139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158,
159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175, 176, 177, 178,
179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 198,
199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215, 216, 217, 218,
219, 220, 221, 222, 223, 242, 243, 244, 245, 246, 247, 248, 256, 257, 258, 259, 260, 261, 262, 263,
264, 280, 281, 282, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 321, 344, 345, 346, 347, 348,
349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365, 366, 367, 368,
369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388,
389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408,
409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428,
429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445, 446, 447, 448,
449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 468,
469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485, 486, 487, 488,
489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508,
509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525, 526, 527, 528,
529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548,
549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565, 567, 568, 569,
570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589,
590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 606, 607, 608, 609, 610, 611, 634, 635, 640,
650, 666, 704, 709, 729, 730, 731, 737, 740, 741, 742, 744, 747, 748, 749, 750, 751, 752, 753, 758,
759, 760, 761, 762, 763, 764, 765, 767, 769, 770, 771, 772, 773, 774, 775, 776, 780, 781, 782, 783,
786, 800, 801, 888, 996, 997, 998, 999, 1000, 1008, 1012, 1025, 1030, 1031, 1032, 1058, 1059, 1067,
1068, 1080, 1083, 1084, 1110, 1155, 1167, 1212, 1222, 1248, 1346, 1347, 1348, 1349, 1350, 1351, 1352,
1353, 1354, 1355, 1356, 1357, 1358, 1359, 1360, 1361, 1362, 1363, 1364, 1365, 1366, 1367, 1368, 1369,
1370, 1371, 1372, 1373, 1374, 1375, 1376, 1377, 1378, 1379, 1380, 1381, 1383, 1384, 1385, 1386, 1387,
1388, 1389, 1390, 1391, 1392, 1393, 1394, 1395, 1396, 1397, 1398, 1399, 1400, 1401, 1402, 1403, 1404,
1405, 1406, 1407, 1408, 1409, 1410, 1411, 1412, 1413, 1414, 1415, 1416, 1417, 1418, 1419, 1420, 1421,
1422, 1423, 1424, 1425, 1426, 1427, 1428, 1429, 1430, 1431, 1432, 1433, 1434, 1435, 1436, 1437, 1438,
1439, 1440, 1441, 1442, 1443, 1444, 1445, 1446, 1447, 1448, 1449, 1450, 1451, 1452, 1453, 1454, 1455,
1456, 1457, 1458, 1459, 1460, 1461, 1462, 1463, 1464, 1465, 1466, 1467, 1468, 1469, 1470, 1471, 1472,
1473, 1474, 1475, 1476, 1477, 1478, 1479, 1480, 1481, 1482, 1483, 1484, 1485, 1486, 1487, 1488, 1489,
1490, 1491, 1492, 1493, 1494, 1495, 1496, 1497, 1498, 1499, 1500, 1501, 1502, 1503, 1504, 1505, 1506,
1507, 1508, 1509, 1510, 1511, 1512, 1513, 1514, 1515, 1516, 1517, 1518, 1519, 1520, 1521, 1522, 1523,
1524, 1525, 1526, 1527, 1528, 1529, 1530, 1531, 1532, 1533, 1534, 1535, 1536, 1537, 1538, 1539, 1540,
1541, 1542, 1543, 1544, 1545, 1546, 1547, 1548, 1549, 1550, 1551, 1552, 1600, 1645, 1646, 1650, 1651,
1652, 1661, 1662, 1663, 1664, 1665, 1666, 1667, 1668, 1669, 1670, 1671, 1672, 1698, 1699, 1812, 1813,
1986, 1987, 1988, 1989, 1990, 1991, 1992, 1993, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002,
2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020,
2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2030, 2032, 2033, 2034, 2035, 2038, 2040, 2041, 2042,
2043, 2044, 2045, 2046, 2047, 2048, 2049, 2065, 2067, 2103, 2104, 2105, 2106, 2108, 2140, 2201, 2232,
2241, 2307, 2401, 2430, 2431, 2432, 2433, 2500, 2501, 2627, 2784, 2989, 3049, 3130, 3141, 3150, 3264,
3333, 3421, 3455, 3456, 3457, 3801, 3900, 3984, 3985, 3986, 4008, 4045, 4132, 4133, 4321, 4343, 4444,
4500, 4672, 5000, 5001, 5002, 5010, 5011, 5050, 5145, 5190, 5191, 5192, 5193, 5236, 5300, 5301, 5302,
5303, 5304, 5305, 5308, 5500, 5540, 5555, 5632, 5713, 5714, 5715, 5716, 5717, 6110, 6111, 6141, 6142,
6143, 6144, 6145, 6146, 6147, 6148, 6558, 6838, 6969, 7000, 7001, 7002, 7003, 7004, 7005, 7006, 7007,
7008, 7009, 7010, 7028, 7100, 7200, 7201, 7648, 7649, 7650, 7651, 7983, 8787, 8879, 9325, 9535, 9876,
10067, 10080, 10167, 10498, 17007, 18000, 18753, 20433, 21554, 26274, 27374, 27444, 27573, 31335,
31337, 31338, 31787, 31789, 31790, 31791, 32770, 32771, 32772, 32773, 32774, 32775, 32776, 32777,
32778, 32779, 32780, 33390, 34555, 47262, 47557, 49301, 54320, 54321, 57341
```


Standard TCP ports

1, 2, 3, 5, 7, 9, 11, 13, 15, 17, 18, 19, 20, 21, 22, 23, 24, 25, 27, 29, 31, 33, 35, 37, 38, 39, 41,
42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 61, 62, 63, 64, 65, 66, 67, 68,
69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94,
95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115,
116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135,
136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155,
156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175,
176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195,
196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215,
216, 217, 218, 219, 220, 221, 222, 223, 242, 243, 244, 245, 246, 247, 248, 256, 257, 258, 259, 260,
261, 262, 263, 264, 280, 281, 282, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 321, 344, 345,
346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365,
366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385,
386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405,
406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425,
426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445,
446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465,
466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485,
486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505,
506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525,
526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545,
546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565,
566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585,
586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 606, 607, 608, 609, 610,
611, 628, 634, 666, 704, 709, 729, 730, 731, 740, 741, 742, 744, 747, 748, 749, 750, 751, 752, 753,
754, 758, 759, 760, 761, 762, 763, 764, 765, 767, 769, 770, 771, 772, 773, 774, 775, 776, 777, 780,
781, 782, 783, 786, 799, 800, 801, 871, 888, 911, 989, 990, 992, 993, 994, 995, 996, 997, 998, 999,
1000, 1001, 1008, 1011, 1012, 1015, 1024, 1025, 1026, 1027, 1029, 1030, 1031, 1032, 1033, 1042, 1045,
1058, 1059, 1067, 1068, 1080, 1083, 1084, 1090, 1103, 1109, 1110, 1112, 1127, 1155, 1170, 1178, 1207,
1212, 1222, 1224, 1234, 1241, 1243, 1245, 1248, 1269, 1346, 1347, 1348, 1349, 1350, 1351, 1352, 1353,
1354, 1355, 1356, 1357, 1358, 1359, 1360, 1361, 1362, 1363, 1364, 1365, 1366, 1367, 1368, 1369, 1370,
1371, 1372, 1373, 1374, 1375, 1376, 1377, 1378, 1379, 1380, 1381, 1383, 1384, 1385, 1386, 1387, 1388,
1389, 1390, 1391, 1392, 1393, 1394, 1395, 1396, 1397, 1398, 1399, 1400, 1401, 1402, 1403, 1404, 1405,
1406, 1407, 1408, 1409, 1410, 1411, 1412, 1413, 1414, 1415, 1416, 1417, 1418, 1419, 1420, 1421, 1422,
1423, 1424, 1425, 1426, 1427, 1428, 1429, 1430, 1431, 1432, 1433, 1434, 1435, 1436, 1437, 1438, 1439,
1440, 1441, 1442, 1443, 1444, 1445, 1446, 1447, 1448, 1449, 1450, 1451, 1452, 1453, 1454, 1455, 1456,
1457, 1458, 1459, 1460, 1461, 1462, 1463, 1464, 1465, 1466, 1467, 1468, 1469, 1470, 1471, 1472, 1473,
1474, 1475, 1476, 1477, 1478, 1479, 1480, 1481, 1482, 1483, 1484, 1485, 1486, 1487, 1488, 1489, 1490,
1491, 1492, 1493, 1494, 1495, 1496, 1497, 1498, 1499, 1500, 1501, 1502, 1503, 1504, 1505, 1506, 1507,
1508, 1509, 1510, 1511, 1512, 1513, 1514, 1515, 1516, 1517, 1518, 1519, 1520, 1521, 1522, 1523, 1524,
1525, 1526, 1527, 1528, 1529, 1530, 1531, 1532, 1533, 1534, 1535, 1536, 1537, 1538, 1539, 1540, 1541,
1542, 1543, 1544, 1545, 1546, 1547, 1548, 1549, 1550, 1551, 1552, 1600, 1604, 1650, 1651, 1652, 1661,
1662, 1663, 1664, 1665, 1666, 1667, 1668, 1669, 1670, 1671, 1672, 1698, 1699, 1720, 1723, 1741, 1807,
1830, 1835, 1981, 1986, 1987, 1988, 1989, 1990, 1991, 1992, 1993, 1994, 1995, 1996, 1997, 1998, 1999,
2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016,
2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 2027, 2028, 2030, 2032, 2033, 2034, 2035,
2038, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049, 2064, 2065, 2067, 2105, 2106, 2108,
2111, 2112, 2115, 2120, 2140, 2155, 2201, 2232, 2241, 2283, 2307, 2401, 2430, 2431, 2432, 2433, 2500,
2501, 2564, 2565, 2583, 2600, 2601, 2602, 2603, 2604, 2605, 2627, 2638, 2716, 2721, 2766, 2784, 2801,
2855, 2865, 3000, 3005, 3006, 3024, 3028, 3049, 3086, 3128, 3129, 3141, 3150, 3264, 3306, 3333, 3389,
3421, 3455, 3456, 3457, 3459, 3462, 3700, 3791, 3900, 3984, 3985, 3986, 4008, 4045, 4092, 4100, 4132,
4133, 4144, 4321, 4333, 4343, 4444, 4500, 4557, 4559, 4567, 4590, 4672, 4950, 5000, 5001, 5002, 5010,
5011, 5031, 5032, 5050, 5145, 5190, 5191, 5192, 5193, 5232, 5236, 5300, 5301, 5302, 5303, 5304, 5305,
5308, 5321, 5400, 5401, 5402, 5432, 5510, 5512, 5520, 5521, 5530, 5540, 5550, 5555, 5556, 5557, 5569,
5631, 5632, 5637, 5638, 5680, 5713, 5714, 5715, 5716, 5717, 5741, 5742, 5800, 5801, 5977, 5978, 5979,
5997, 5998, 5999, 6000, 6001, 6002, 6003, 6004, 6005, 6006, 6007, 6008, 6009, 6110, 6111, 6112, 6141,
6142, 6143, 6144, 6145, 6146, 6147, 6148, 6346, 6400, 6558, 6666, 6667, 6668, 6669, 6670, 6671, 6711,
6712, 6713, 6723, 6771, 6776, 6912, 6939, 6969, 6970, 7000, 7001, 7002, 7003, 7004, 7005, 7006, 7007,
7008, 7009, 7010, 7028, 7100, 7200, 7201, 7300, 7301, 7306, 7307, 7308, 7326, 7597, 7789, 8080, 8888,
9100, 9200, 9201, 9202, 9203, 9400, 9535, 9872, 9873, 9874, 9875, 9876, 9878, 9989, 9999, 10005, 10067,
10082, 10083, 10167, 10520, 10607, 10666, 11000, 11050, 11223, 12076, 12223, 12345, 12346, 12361,
12362, 12456, 12631, 12701, 12754, 13000, 13700, 15104, 16484, 16660, 16959, 16969, 17007, 17300,
18000, 20000, 20001, 20005, 20034, 20203, 20331, 20432, 20433, 21544, 21554, 22222, 22273, 22289,
22305, 22321, 23456, 23476, 23477, 26208, 26274, 27374, 27573, 27665, 29891, 30029, 30100, 30101,
30102, 30133, 30303, 30999, 31335, 31336, 31337, 31338, 31339, 31666, 31785, 31787, 31789, 31791,
32418, 32770, 32771, 32772, 32773, 32774, 32775, 32776, 32777, 32778, 32779, 32780, 33270, 33333,
33911, 34324, 37651, 40412, 40421, 40422, 40423, 40425, 40426, 43118, 43210, 47252, 47557, 50505,
50766, 50776, 53001, 54320, 54321, 57341, 59998, 60000, 61348, 61466, 61603, 63485, 65000, 65301

Standard Frag ports

1, 2, 3, 5, 7, 9, 11, 13, 15, 17, 18, 19, 20, 21, 22, 23, 24, 25, 27, 29, 31, 33, 35, 37, 38, 39, 41,
42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 61, 62, 63, 64, 65, 66, 67, 68,
69, 70, 71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 89, 90, 91, 92, 93, 94,
95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115,
116, 117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130, 131, 132, 133, 134, 135,
136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155,
156, 157, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 171, 172, 173, 174, 175,


```
176, 177, 178, 179, 180, 181, 182, 183, 184, 185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195,
196, 197, 198, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211, 212, 213, 214, 215,
216, 217, 218, 219, 220, 221, 222, 223, 242, 243, 244, 245, 246, 247, 248, 256, 257, 258, 259, 260,
261, 262, 263, 264, 280, 281, 282, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 321, 344, 345,
346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 360, 361, 362, 363, 364, 365,
366, 367, 368, 369, 370, 371, 372, 373, 374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385,
386, 387, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405,
406, 407, 408, 409, 410, 411, 412, 413, 414, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425,
426, 427, 428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441, 442, 443, 444, 445,
446, 447, 448, 449, 450, 451, 452, 453, 454, 455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465,
466, 467, 468, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 484, 485,
486, 487, 488, 489, 490, 491, 492, 493, 494, 495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505,
506, 507, 508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 522, 523, 524, 525,
526, 527, 528, 529, 530, 531, 532, 533, 534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545,
546, 547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562, 563, 564, 565,
566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584, 585,
586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 606, 607, 608, 609, 610,
611, 628, 634, 666, 704, 709, 729, 730, 731, 740, 741, 742, 744, 747, 748, 749, 750, 751, 752, 753,
754, 758, 759, 760, 761, 762, 763, 764, 765, 767, 769, 770, 771, 772, 773, 774, 775, 776, 777, 780,
781, 782, 783, 786, 799, 800, 801, 871, 888, 911, 989, 990, 992, 993, 994, 995, 996, 997, 998, 999,
1000, 1001, 1008, 1011, 1012, 1015, 1024, 1025, 1026, 1027, 1029, 1030, 1031, 1032, 1033, 1224, 1723,
2865, 9200, 9201, 9202, 9203
```


Appendix C - Public Internet Resources

This resource is by no means a end-all, be-all for who's who in online security resources. This section is a list of sites used in researching information for this methodology. Although no specific documents were referenced to make this manual, many were read to be sure that thoroughness and quality were achievable objectives. In addition, the sites here have provided a public service which security testers will find both useful and beneficial.

CISSP Open Study Guides Web Site

http://www.cccure.org/

This site is an open resource for individuals studying to become a Certified Information Systems Security Professional (CISSP). Publicly contributed information, security documents, study guides for each of the 10 domains, forums, and online quizzes attempt to cover the topics outlined in the CISSP Common Body of Knowledge Study Guide. All of the above for free and for everyone without discrimination.

Security Focus

http://www.securityfocus.com/

This is an online database which contains sections about vulnerabilities, searchable via OS, software, version and customizable queries. Additionally you can find references for auditing and other security related tools, papers about security issues, configuration manuals, news and mailing lists.

Packet Storm

http://packetstorm.securify.com/

Packet Storm is a large, searchable collection of exploits, tools, and advisories that is constantly updated. This site provides a backend interface that can be accessed through automated scripts, keeping you aware of the latest exploits and tools, or for creating and updating your own exploit and tool collection.

INFOSYSSEC

http://www.infosyssec.com/

This is a portal to computer and network security related information containing quick access to news, alerts, vendor patches, and various search engines. The amount of security related information that can be reached from this site is massive.

IT Professional's Resource Center

http://www.itprc.com/

The Information Technology Professional's Resource Center" Technology Resources for IT Professionals by IT Professionals.

SANS Institute Online

http://www.sans.org/

The System Administration, Networking, and Security (SANS) Institute is comprised of over 96,000 security practitioners. SANS is involved with providing security news, research, publications, education and certification.

The Traceroute Organization

http://www.traceroute.org/

A portal linking to online traceroute utilities throughout the world. Here you can perform traceroutes using your browser, view network performance statistics for some networks, and reach other online network utilities.

Dshield

http://www.dshield.org/

A site that provides a means for users of firewalls to share intrusion information. As a SANS partner, their information is relayed to the Internet Storm Center, which is described below.

Internet Storm Center

http://www.incidents.org/cid/

SANS and partner's (namely Dshield, myNetWatchman, and NetSquared) CID Project is based on the same methodology. Intrusion data is collected from several sites and stored in databases and openly shared, including lists of malicious source IP addresses, and intrusion methods (Protocols, Destination Ports etc..). The purpose of the data is to provide a better understanding of malicious activity on the Internet.

CVE Database

http://cve.mitre.org

Common Vulnerabilities and Exposures (CVE) is a list of standardized names for vulnerabilities and other information security exposures. Several vulnerability databases and security tools, such as SecurityFocus and Snort respectively, use CVE to share data across other vulnerability databases and security tools. Consideration should be given to using CVE as a reference for report generation, particularly for correlating discovered vulnerabilities to known vulnerabilities.

Appendix D - Needed Resources

If necessity is the mother of development here than shortsightedness is the deadbeat dad of non-modifiable, non-stackable, closed-source testing tools.

A number of tools may exist for the task we are attempting to achieve but currently they either don't offer the features needed and can't be modified. This list is of software is that which needs development support. All of the tools listed here are supplements to this manual. These tools are not intended to be used by non-security professionals. If you are interested in being part of development of any of these tools, please contact info@ideahamster.org.

Springboard

This tool has the ability to search public databases on key word or phrase and download a set parameter of documents according to that key word or phrase. All downloads go into a directory type hierarchy that can be manually or automatically searched. This tool is beneficial to the searching and downloading of documents for analyzing information security leaks as in the Electronic Dumpster Diving parameter in the Security Testing Methodology.

Key Features

- reads URLs in text format with the %s variable for search parameter and can be easily updated with new URL strings by just inserting them into this flat file. For example, adding the search from the site http://www.fuckedcompany.com/ would be as simple as inserting the search string: "http://www.fuckedcompany.com/archives/search.cfm?search=%s" where %s is the key word or phrase.
- downloads found documents and store them in a directory hierarchy
- chooses what kinds of objects/documents to download
- can restrict the download by object size
- can download only X number of documents from the search
- follows links up to X deep
- logs into sites which need member access to download results
- can download alternate documents if the original is not found (such as cached pages available in Google)
- also uses HTTPS and FTP
- uses command line parameters

Seeker

This tool has the ability to search a directory tree for key words, phrases, or patterns in many documents of which few are flat ASCII or text files and create a single text file for the results. This tool will satisfy the Document Grinding parameter of the Security Testing Methodology.

Key Features

- supports doc formats such as pdf, MS .doc, Lotus .lwp, LaTex, PS as examples
- automatically uncompress/extract documents first before reading
- supports spreadsheet formats and spreadsheet reading by extracting in human readable format
- supports the size of the extraction by sentence, line, or number of words before and after the key word/phrase/pattern.
- reads key words, phrases, and patterns from text file
- some built in commands for default patterns like Names, For example, two or three words together starting with capitals is can be called by writing "Names" in the text file.
- works from a command line

Privacide

Pages that use VB or JavaScript to extract additional info from a user is not new. What Privacide does is to take the best features of user data grabbing including a configurable form for posting and package it into a combo HTML / Perl set that will extract the user info including the web server log data and insert it into a log file by IP address. The HTML is changeable of course and can be anything that would be innocuous. This will prevent the need from matching the collected data with information from the web server log matched by date and time.

Key Features

- grabs all information possible from a user through scripting
- combines web server log data
- can run external commands such as Traceroute, NMAP or HPING back at the IP address
- forces user to activate JavaScript and frames to enjoy the page
- fails if it is not from one of the vulnerable browsers with a fake 403
- includes a form which can be filled out by the user
- includes a "Tell a friend button" to support the forwarding of the page to other users
- includes tests for some known browse vulnerabilities
- saves loading times of the pages and other performance information
- saves info and all input to a text file by IP address or other variable

Metis

This is a tool to satisfy the Competitive Intelligence and business side of information gathering in a security test. This will download information that will be beneficial in enumerating the traffic of a site, the users, and the utilization. Additional features gather information for password cracking, e-mail attacks, and social engineering.

Key Features

- counts the number of objects/documents in a given server sortable by type
- returns the size and date of all files on a given server
- counts the number of postings in a bulletin board or forum
- extracts the subjects, names, e-mail addresses from postings in a bulletin board or forum
- saves the results in a comma delimited text file

Scanning Tools in the Wild Database

This is a database/website that lists the current vulnerability scanning tools being used in the wild by popularity. This is measured by matching the signatures from firewall logs and honey pots around the world to existing tools. These tools can then be tested against a target organization or scanning tools can implement the same tests as these tools. The number of times a tool's signature has been found in the wild attributes to its rank.

Key Features

- organized and printable
- links to the tools used
- the vulnerabilities each tool tests for
- searchable
- all signatures found are ranked
- integration with other signature databases
- replicable (mirroring)

Appendix E - Profile Template

This profile template is an example of the information that should be collected to the best of the testers ability. The objective of this template is to instruct and not to serve as the ultimate template for information collection. Nor is this template a draconian must. The use of this template is to enhance the collection of information and not the analysis on whether or not this information is a security threat at any level.

Business Name:
Business Address:
Business Telephone:
Business Fax:

Primary Contacts:
Method of Contact:
Employee Names and Positions:
Employee Personal Pages:
Employee Information:
Departments and Responsibilities:
Hierarchy Model:
Office Hierarchy:
Line of Business:
Operations:
Legal Structure:
Year Started:
Company History:
Partners:
Resellers:

Outsourcer Web:
Outsourcer Email:
Outsourcer Tech Support:
Outsourcer Firewall:
Outsourcer IDS:
Outsourcer Help Desk:
ISPs:
ASPs:

Domain Names:
Network Blocks:
Network Block Owner:
Record Created:
Record Last Updated:

Number of Network Accounts:
Network Account Standard:
Network Account Creation Standard:

Number of Systems:
System Names Standard:
System Names:
Types of Systems:
Operating Systems:
Services provided:

Noted Business Phone Numbers:
Phone Number Block:
Phone Number Type:
Number of Modems:
Modem Phone Numbers:
Modem Connect Speeds:
Number of Fax Machines:
Fax Phone Numbers:
Unusual Phone Numbers:

Email Server Address:
Email Server Type:
Email Client:
Email System:
Email Address Standard:
E-mail Footer:
Encryption / Standard:
Bounced mails:
SMTP server path:
Automatic Vacation Returns:
Anti-virus Capabilities:

Website Address:
Web Server Type:
Server Location:
Dates Listed:
Date Last Modified:
Web Links Internal:
Web Site Searchability:
Web Links External:
Web Server Directory Tree:
Technologies Used:
Encryption standards:
Web-Enabled Languages:
Form Fields:
Form Variables:
Method of Form Postings:
Keywords Used:
Company contactability:
Meta Tags:
Comments Noted:
e-commerce Capabilities:
Services Offered on Net:
Products Offered on Net:
Features:
Search Engines Identified:
Search Engine Ranking:
Daily/Weekly/Monthly Hits:
Link Popularity:
Link Culture:

Web Clients Used:
Screen Size:
Security Settings in Browser:

FTP Server Address:
Server Location:
Server Type:
Directory Tree:
Files Sitting:

Primary (Authoritative) Name Server:
Secondary:
Last Update:
Additional Name Servers:

Firewall Address:
Firewall Type:
ID system:

Router Addresses:
Router Types:
Router Capabilities:

VPN Capabilities:
VPN Type:

Network Services Noted:

Newsgroup Postings:
Bulletin Board Postings:
Business Wire Postings:
Help Wanted Ads:

Customer List:
Target Market:
Product List:

Appendix F - Protocols

This list of protocols comes from various sources (www.protocols.com, www.acterna.com, www.cisco.com).

AARP	AppleTalk Address Resolution Protocol/AppleTalk Phase 2 Protocol Specification, document ADPA #C0144LL/A
AEP	AppleTalk Echo Protocol
AH	IP Authentication Header/RFC 1826
ARP	Address Resolution Protocol message (request or response); does not include Reverse ARP (RARP) packets which are counted separately./RFC 826
ATALK	AppleTalk Protocol/AppleTalk Phase 2 Protocol Specification, document ADPA #C0144LL/A
ATMP	Ascend Tunnel Management Protocol/RFC 2107
BGP	Border Gateway Protocol/RFC 1267
BOOTPC	Bootstrap Protocol Client Protocol; BOOTP server replies are sent to the BOOTP client using this destination port./RFC 951
BOOTPS	Bootstrap Protocol Server Protocol; BOOTP clients send requests (usually broadcast) to the bootps port./RFC 951
BRIDGE	MAC Bridge Protocol Entity/Bridge Protocol defined in
CATALYST	Synchronization protocol used between Cisco Catalyst switches
CDP	Cisco Discovery Protocol
CGMP	Cisco Inter-Process Communication
CIPC	Cisco Group Management Protocol
CSTB	Cisco Spanning Tree BPDU
DBASE	dBASE UNIX
DISL	Dynamic Inter-Switch Link
DLSRPN	Data Link Switch (DLSw) Read Port Number/RFC 1795
DLSWPN	Data Link Switch (DLSw) Write Port Number/RFC 1795
DOMAIN1	Domain Name Service Protocol; DNS may be transported by UDP (RFC768) or TCP (RFC793). If transport is UDP, DNS requests restricted to 512 bytes in length may be sent to this port./RFC 1035
DOMAIN2	Domain Name Service Protocol; DNS may be transported by UDP (RFC768) or TCP (RFC793). If transport is UDP, DNS requests restricted to 512 bytes in length may be sent to this port./RFC 1035
DOOM	DOOM Game; Id Software
DRP	DEC Routing Protocol/DECnet Digital Network Architecture Phase IV Routing Layer Functional Specification
EGP	Exterior Gateway Protocol (*historical*)/RFC 904
ESMP	Even Simpler Management Protocol
ESP	IP Encapsulating Security Payload/RFC 1827
EXEC	Remote Exec
FINGER	Finger User Information Protocol/RFC 1288
FTP	File Transfer Protocol Control Port; an FTP client initiates an FTP control connection by sending FTP commands from user port (U) to this port./RFC 959
FTP-DATA	File Transfer Protocol Data Port; FTP server process default data-connection port./RFC 959; See section 3.2 for details about FTP data connections.
GARP	General Attribute Registration Protocol/ISO/IEC 15802-3
GDP	Cisco Gateway Discovery Protocol
GOPHER	Internet Gopher Protocol/RFC 1436
GRE	General Routing Encapsulation/
H323-GAT	H.323 control protocol/
HSRP	Cisco Hot Standby Router Protocol/RFC 2281
HTTPS	Secure HTTP; HTTP over TLS/SSL/
ICMP	Internet Message Control Protocol/RFC 792

IDP	Xerox IPX/Xerox Corporation, Document XNSS 028112, 1981
IDRP	Inter-Domain Routing Protocol/RFC 1745
IGMP	Internet Group Management Protocol; used by IP hosts to report host group memberships to any immediately neighboring multicast routers./RFC1112, Appendix A
IGRP	IGRP; Cisco routing protocol
IMAP3	Interactive Mail Access Protocol v3 (*historical*)./RFC 1203
INGRES-N	Network PostScript
IP	Protocol identifiers for Internet Protocol (IP); may be encapsulated within itself, so more than one identifier may be present in a protocolDir ID string./RFC 791
IPIP	Authoritative repository for PROTOCOL NUMBERS is at:
IPIP4	IP-within-IP Encapsulation Protocol
IPV6	Method for the Transmission of IPv6 Packets over Ethernet Networks/
IPX	Novell IPX/Defined by Novell Corporation
IP-XNS-R	XNS-RIP
IPX-TUNN	Tunneling IPX Traffic through IP Networks/RFC 1234
IRC	Internet Relay Chat Protocol/RFC 1459
ISO-TP4	ISO Transport Protocol Specification/RFC 905; ISO DP 8073
ISP-IP	ISO Internet Protocol
KERBEROS	Kerberos Network Authentication Service V5/RFC 1510
L2F	Cisco Layer Two Forwarding (Protocol)-- L2F/RFC 2341
L3SW	Layer 3 IP and IPX switching
LAT	DEC Local Area Transport Protocol
LDAP1	Lightweight Directory Access Protocol/
LDAP2	Lightweight Directory Access Protocol/
LOGIN	BSD Rlogin; remote login via telnet/RFC 1282
MOBILE	Minimal Encapsulation within IP
MSCP	Multicast Shortcut Control Protocol
NBP	AppleTalk Name Binding Protocol
NBT-DA1	NetBIOS-over-TCP datagram protocol/RFC 1002
NBT-DA2	NetBIOS-over-TCP datagram protocol/RFC 1002
NBT-NA1	NetBIOS-over-TCP session protocol/RFC 1002
NBT-NA2	NetBIOS-over-TCP session protocol/RFC 1002
NBT-SE1	NetBIOS-over-TCP name protocol/RFC 1002
NBT-SE2	NetBIOS-over-TCP name protocol/RFC 1002
NCP_NAM	Netware Core Protocol
NDS	Netware Directory Services sub-protocol
NETBEUI	LAN Manager NetBEUI protocol
NEWS1	News
NEWS2	NewS
NFS1	NBMA Next Hop Resolution Protocol
NFS2	Sun Network File System/RFC 1813
NHRP	Sun Network File System/RFC 1813
NLSP	NLSP
NNTP	Network News Transfer Protocol/RFC 977
NOTES1	Lotus Notes Protocol
NOTES2	Lotus Notes Protocol
NOTES3	Lotus Notes Protocol
NOTES4	Lotus Notes Protocol
NOTES5	Lotus Notes Protocol
NOTES6	Lotus Notes Protocol
NOV-PEP	Novell Packet Exchange Protocol. A null protocol layer as all IPX packets contain the relevant fields for this protocol. Defined so socket-based decoding has a point of attachment in the decode tree while allowing packet type-based decoding.

NOV-RIP	Novell Routing Information Protocol
NOV-SAP	Novell Service Advertising Protocol; binds applications on a particular host to an IPX/PEP or IPX/SPX socket number. Although not used as a transport protocol, is used to establish sessions between clients and servers. Is the only reliable method, besides well-known sockets, to determine the protocol running over a particular socket on a particular system./
NOV-SPX	Novell Sequenced Packet Exchange Protocol; an extension of IPX/PEP (shares a common header).
NSP	DEC Network Services Protocol/DECnet Digital Network Architecture Phase IV NSP Functional Specification
NTP	Network Time Protocol
OSPF	Open Shortest Path First Interior GW Protocol (OSPFIGP)/
PAGP	Port Aggregation Protocol
PNNI	PNNI over IP
POP2	Post Office Protocol--Version 2. Clients establish connections with POP2 servers using this destination port number./RFC 937
POP31	Post Office Protocol--Version 3. Clients establish connections with POP3 servers using this destination port number./RFC 1725
POP32	Post Office Protocol--Version 3. Clients establish connections with POP3 servers using this destination port number./RFC 1725
PORTMA1	SUNRPC PORTMAPPER program; SUNRPC program used to locate the UDP/TCP ports on which other SUNRPC programs can be found./RFC 1057, Appendix A
PORTMA2	SUNRPC PORTMAPPER program; SUNRPC program used to locate the UDP/TCP ports on which other SUNRPC programs can be found./RFC 1057, Appendix A
PRINT-1	Network PostScript
PRINT-2	Network PostScript
PRINTER	Printer Spooler
Q931	H.323 call-signalling protocol/ITU-T Recommendation Q.931 Digital Subscriber Signalling System No. 1 (DSS 1)--ISDN User-Network Interface Layer 3 Specification for Basic Call Control./ ITU-T Recommendation H.225.0 Media Stream Packetization and Synchronization on Non-Guaranteed Quality of Service LANs.
RARP_NAM	Reverse Address Resolution Protocol/RFC 903
RAUDIO	Real Audio
RSVP1	Resource Reservation Setup Protocol/RFC 2205
RSVP2	Resource Reservation Setup Protocol/RFC 2205
RSVP3	Resource Reservation Setup Protocol/RFC 2205
RTCP	Real Time Control Protocol/RFC 1889
RTMP1	AppleTalk Routing Table Maintenance Protocol
RTMP2	AppleTalk Routing Table Maintenance Protocol
RTP	Real Time Protocol/RFC 1889
SCCP	Skinny Client Control Protocol
SMB1	Microsoft Server Message Block Protocol
SMB2	Microsoft Server Message Block Protocol
SMB3	Microsoft Server Message Block Protocol
SMB4	Microsoft Server Message Block Protocol
SMTP	The Simple Mail Transfer Protocol; SMTP control and data messages are sent on this port./RFC 821
SNMP	Simple Network Management Protocol; includes SNMPv1 and SNMPv2 protocol versions; does not include SNMP trap packets./
SNMPTRAP	Simple Network Management Protocol Trap Port/
SQL*NET	Oracle SQL*NET
SQL-NET	SQL-NET
SQLSRV	SQL Services; protocol to talk to Oracle databases (*historical*).
SSH	SSH Remote Login Protocol
SSTB	Shared Spanning Tree BPDU
STP	Bridge Spanning Tree Protocol/ISO/IEC 15802-3

SUNRPC1	SUN Remote Procedure Call Protocol; port mapper function requests are sent to this destination port./RFC 1831
SUNRPC2	The authoritative list of RPC Functions is identified by the URL:
SYSLOG	SUN Remote Procedure Call Protocol; port mapper function requests are sent to this destination port./RFC 1831
SYSTAT	syslog
TACACS	Retrieves the active users list; debugging tool for TCP and UDP transports./RFC 866
TAGSWITC	Tag Switching
TCP	Transmission Control Protocol/RFC 793
TELNET	Provides a general, bidirectional, eight-bit, byte-oriented communications facility that allows a standard method of interfacing terminal devices and terminal-oriented processes to each other. /RFC 854
TFTP_NAM	Trivial File Transfer Protocol; only the first packet of each TFTP transaction is sent to port 69. If the tracksSessions attribute is set, packets for each TFTP transaction are attributed to tftp instead of the unregistered port numbers encoded in subsequent packets.
UDP	User Datagram Protocol/RFC 768
VARP	Banyan Vines Address Resolution Protocol/Vines Protocol Definition--Part# 092093-001, Order# 003673
VDOLIVE	VDOLive
VICP	Banyan Vines Internet Control Protocol/Vines Protocol Definition--Part# 092093-001, Order# 003673
VINES	VINES
VIP1	Banyan Vines Internet Protocol/Vines Protocol Definition--
VIP2	Banyan Vines Internet Protocol/Vines Protocol Definition--
VIPC	Vines Protocol Definition--Part# 092093-001, Order# 003673
VRTP	Banyan Vines Routing Update Protocol/Vines Protocol Definition--Part# 092093-001, Order# 003673
VSI	Virtual Switch Interface
VSPP	Banyan Vines Sequenced Packet Protocol/Vines Protocol Definition--Part# 092093-001, Order# 003673
VTP	VLAN Trunking Protocol/Cisco VLAN Trunk Protocol
WHO	rwho; shows logged-in users
WWW-HTTP	Hypertext Transfer Protocol/
XDMCP	X Display Manager Control Protocol
XNS-ECHO	XNS echo protocol
XNS-ERRO	XNS error-handler protocol
XNS-PEP	XNS Packet Exchange Protocol
XNS-RIP	Routing Information Protocol
XNS-SPP	Sequenced Packet Protocol
XWIN1	X Windows Protocol
XWIN2	X Windows Protocol
ZIP	AppleTalk Zone Information Protocol

