An Overview of Network Security Analysis and Penetration Testing

A Guide to Computer Hacking and Preventative Measures

The MIS Corporate Defence Solutions Ltd., Network Security Team.
nst@mis-cds.com, http://www.mis-cds.com
Tel +44 (0)1622 723400, Fax +44 (0)1622 728580

August 1st 2000

Published Electronically by MIS Corporate Defence Solutions Ltd. at http://www.mis-cds.com

Table of Contents

Introduction to MIS Corporate Defence Solutions

Global Corporate Defence

Since 1991, MIS Corporate Defence Solutions have been pioneers in the specialist IT Security arena. From our Head Office in Kent, England, we have expanded our operations in the UK and Europe. We will be opening further offices across Europe and the United States.

Long Lasting Protection

With computers in universal use, often in multiple locations within the organisation, today's computer systems may present major security problems. The growth of networking, the profusion of keyboards and the friendliness of the computer environment have all outgrown the use of traditional passwords. The old solutions can no longer prevent infiltration to your most strategic asset - *business information*.

It is one of our aims to educate executive-level management to the range of potential cyber attacks and related information protection initiatives. MIS Consultants can also illustrate to customers how IT security represents an enabling enhancement to their business systems, rather than an inhibiting technology, thus providing a solution that addresses the current and future needs of the organisation.

The purchase of hardware and software represents only part of the solution to your security concerns. In fact, many security products can restrict the potential of your business systems, making them less user-friendly, slowing down response times and limiting flexibility for further development. This need not be the case.

MIS Consultants have considerable experience of matching security needs to real life operations, and this is key to our business. Our philosophy is to share our knowledge of proven security products and practices with our customers, and to work with them to provide pragmatic and workable security solutions, backed up by a flexible ongoing support service.

Secure Business Solutions for a Competitive Advantage

Many organisations have already taken their first steps towards securing their valuable and sensitive data. Most have implemented some solutions to reduce the threat of hackers, thieves, dishonest employees, viruses, bug-infested illegal software or the myriad dangers of the Internet.

However, the most forward-looking organisations no longer regard IT Security as just a necessary evil - a mere preventative measure to protect their business information. They now acknowledge it as a means of improving productivity and *enabling* the technology of the future, both of which represent *measurably* increased profitability and *genuine* business advantage.

Understanding the Threats

Everyone now recognises the power of the Internet as a valuable information source and communications medium. With the advent of Electronic Commerce, business and private trading practices are rapidly evolving as this new technology gains popularity. No-one can afford to ignore this innovative and profitable opportunity - and MIS can help you to implement it, safely and affordably.

The scope of e-commerce crime stretches far beyond the security of a single credit card transaction over the World Wide Web. Potential losses due to computer-based financial fraud are devastating, whether perpetrated by intruders or dishonest employees. Theft of proprietary information, historically conducted through the “turning” of employees, is increasingly performed via hacking. Information warfare attacks on infrastructure targets such as the power grid, the telecommunications public switch networks and the air traffic control system may be only a few keystrokes away.

Unparalleled Knowledge and Experience

The MIS organisation consists of specialists in leading edge business systems (business analysis & systems development), IT security products & services, BS 7799 security compliance, business continuity and disaster recovery, data protection & encryption laws, military systems defence and computer fraud.

The Technology of the Future

Our newly researched and updated product portfolio is described in the MIS Corporate Defence Solutions Product Guide. This provides your organisation with a comprehensive guide to some of the latest IT security products from around the world. Our ‘Best of Breed’ range have all met our stringent selection criteria and have been fully tested in a commercial environment. They conform to international regulations and standards and they have unique features that set them apart from similar products. Moreover, they all represent exceptional value for money.

Ongoing Support and Training

MIS offers a global technical support service 24 hours a day, 365 days a year. Operated by our Technical Security Consultants, this service can be tailored to a customer’s individual needs, and includes user training, the provision of new software releases, as well as on-site and telephone hotline support.

Best Practice Approach

Utilising industry ‘Best Practice’ methods, we can identify the strengths and weaknesses of a customer’s security policy. Our security professionals will examine our customers’ operational requirements, physical layout, business goals and objectives, and even their corporate culture, then they design a custom Enterprise Security Management Plan. This custom plan provides the foundation for developing a comprehensive information security plan that addresses the specific needs of the organisation. It identifies budget and resource requirements, establishes criteria for selecting products and standard security tools, provides metrics for measuring improvement, and helps the customer to determine an acceptable risk profile.

Large or Small Solutions - According to Your Needs

Whether you need to secure your communications and information assets, or to develop your organisation’s overall information security strategy, you should talk to MIS first. If you need to understand the latest legal issues, run a simple security check or test an existing firewall, one of our Consultants would be happy to discuss this, or indeed any other security problem that concerns you. MIS will address all IT security issues, efficiently and cost-effectively.

The Business of the Future

We are confident that our corporate infrastructure, combined with our unrivalled portfolio of products and services, positions MIS Corporate Defence Solutions at the forefront of the IT security market. With continued investment in the growth of our global organisation, we are committed to providing business enabling solutions into the 21st century.

Part I, The Basic Concepts of Penetration Testing and Network Security Analysis

This section of the document lays down much of the Information Security foundations, documenting the rationale behind Penetration Testing and the threats to businesses with Internet presence.

Chapter 1
The Internet – The New Wild West

Since it was born in the early 1980's, the Internet has become the world's largest computer network, with millions of individual users the world over. The Internet is currently a thriving forum for free speech and self-expression; this is mainly due to the anonymity of the Internet.

When a user connects to the Internet currently, he could be anyone. Browsing web sites and talking to users over ICQ and IRC (Internet Relay Chat), the user can choose his own identity. It is currently virtually impossible for law enforcement agencies to successfully identify the real user from an IP address alone.

Hackers are a completely new breed; the Internet generation. Knowledgeable in networking and TCP/IP, hackers can exploit vulnerabilities in networked computer systems to gain control over that system and the way in which it operates. This is the essence of computer hacking, taking a system and through feeding it data in such a way that the system performs a task that is useful to the hacker.

To ensure anonymity, many hackers will use a complex network of backdoored and misconfigured hosts, such as proxy servers and hosts in countries that are historically weak from an Information Security perspective, usually including Korea and Japan. Upon building such an intricate network of useful hosts the world over, hackers can bounce attacks through such networks to hide the true source of the attack (ie. the IP address of their dialup modem account in most cases).

Law enforcement agencies have a waiting game on their hands. Many hackers will make little mistakes over time, or tell other hackers about their actions. It's up to the FBI, the Scotland Yard Computer Crimes Unit and other organisations to track these hackers over time and log their actions. Due to the global nature of the Internet, a hacker could be in any country with Internet access. The Internet does not have national boundaries with passport control systems like those in the real world; it is a seamless, giant computer network spanning the globe. If the FBI traces a hacker back to Japan, it is usually the responsibility of Japanese law enforcement officers to apprehend the hacker and deal with him. All this red tape regarding law enforcement and the Internet makes it extremely difficult for hackers to be brought to justice unless they make some serious mistakes.

Chapter 2
The Threats to Businesses and Organisations Connected to the Internet

The majority of companies with Internet presence use the Internet on a daily basis for the following purposes –

- To host the company web site
- To send and receive e-mail
- To allow online ordering of products

This relationship with the Internet allows the company to operate in a more efficient manner, being able to access information instantly, and send e-mails across the world in a matter of seconds. But the sword is a double-edged one, as electronic channels are created between end-user PCs and the Internet which usually rely on trust.

Hackers with a goal to break into a company's internal networks can take advantage of these channels and the trust relationships between networked computer systems. Most companies have external network segments consisting of public servers, including e-mail and web servers.

A key point to remember is this –

> "It is never impossible for a hacker to break into a network, only improbable."

Imagine if the hacker knew all your passwords, he could simply walk into your networks through the proverbial front door. There is a fine balance between a highly secure network and one that is not end-user friendly. Network security is often overlooked by many organisations that do not recognise or understand the risks involved. Public awareness is important, as more and more people become aware of the threat that hackers pose to their organisation's network security and integrity, more measures will be taken to deter such Internet-based attackers.

Hackers with access to business critical hosts and networks can cause havoc. Upon breaching such hosts, hackers will usually do all they can in order to mask their presence. Backdoors and *rootkits* are commonplace, as they allow hackers to access hosts without necessarily being logged or detected. Due to today's businesses becoming more and more dependant on computer networks, the business losses that could be incurred as a result of a security breach are phenomenal. Even if hackers don't access confidential data or read user's e-mail, systems administration staff have to assume the worst case scenario and usually take the entire network segment and trusted hosts off-line in order to perform computer forensics and assess the damage caused.

Chapter 3
What is Penetration Testing?

Penetration Testing is the process of emulating determined hackers when assessing the security or target hosts and networks. Penetration Testing is also known as Ethical Hacking, due to obvious comical reasons regarding the phrase ‘Penetration Testing’.

There is a distinct difference between Penetration Testing and Network Security Analysis or assessment. A Penetration Test will include an exploit phase with which the testing team can assess the real-world impact of a hacker compromising an e-mail or web server, by attempting to circumvent security measures in place. Assessing the security of a network using tools such as ISS Internet Scanner or NAI CyberCop is effective to a degree, but do not always highlight risks that determined hackers will identify and exploit, especially in the case of more complex network topologies. The business relevance of the report generated is also questionable, as most reports contain pages of statistics, which may not be relevant to the client. A Penetration Test will give a client a crystal clear idea of the real-world threats that his business faces, whereas a Network Security Scan will simply identify open services and banners, not forgetting the amount of false positive results that such scanners can bring up.

A Security Assessment or Penetration Test will be the first thing an organisation will look to do in order to help manage their Information Security risk. By identifying the vulnerabilities that exist in their networks, an organisation can then look at deploying an Information Security solution, such as a firewall or IDS (Intrusion Detection System).

Information Security is a moving target, with hackers certainly leading the way in terms of offensive technologies that exploit vulnerabilities in systems. Information Security companies are always behind the hackers, trying to keep up-to-date with the latest threats to host and network security. A Penetration Test Report is only as good as the day it was published, as new risks and exploits are being identified on a daily basis. It is therefore important that companies adopt a more pro-active stance regarding Information Security and network integrity. Pro-active security strategies usually include the deployment of systems such as adaptive IDS solutions and full-time Information Security staff who can constantly assess new threats to the business and it’s mission critical hosts and networks.

Chapter 4
The Equipment and Tools Required to Perform Penetration Testing

Determined hackers and Information Security enthusiasts will be knowledgeable in the running of Operating Systems such as Linux, Solaris and Windows NT. Many hackers choose to run Linux on their home systems. Linux is a hacker's Operating System, it is a highly customisable Unix-based Operating System, and makes a very good launch platform for attacks against other Unix-based systems.

If a hacker wanted to run a remote exploit in order to compromise a Sun Microsystems Sparc-based Solaris host remotely, in most cases he would have to run the exploit program from a similar Sun Microsystems Sparc-based host in order for the exploit to work correctly. Due to this fact, many hackers will have access to various compromised hosts running a variety of Operating Systems, including IRIX, AIX, BSDi, Solaris, and others. Such hosts act as effective launch pads for exploits and attacks that hackers launch to compromise target hosts and networks.

Information Security companies providing Network Security Assessment services often use a small cluster of Windows NT servers to perform network testing and then generate reports. Penetration Testing usually involves compromising vulnerable hosts in order to assess the vulnerabilities present in real terms. Access to Solaris hosts running on Sun Sparc hardware and IRIX hosts running on SGI hardware is required to launch attacks and exploits against target hosts and networks running similar Sun Sparc and SGI hardware. Companies performing large-scale Penetration Testing exercises invest heavily in such launch pads running various Operating Systems. It is important to have a good testing infrastructure so that testing can be conducted against even the most complex target networks.

Penetration Testing teams seldom rely on commercial network scanning systems such as ISS Internet Scanner and NAI CyberCop, primarily due to the fact that such systems are not at the cutting edge in the checks they perform. New vulnerabilities and threats to organisations are being published on a daily basis, and it is vitally important that Information Security companies position themselves as close the cutting edge as possible in terms of Information Security risk intelligence. Most teams use a combination of scanning tools available primarily to underground groups and computer hackers themselves, such as nmap, whisker and various toolkits by security groups including ADM and Rhino9. Due to the fact that reports generated by Penetration Testing teams have to be relevant to the client and it's business, many reports are hand-written to highlight serious vulnerabilities.

Many of the powerful scanning tools available run under specific Operating Systems, below is a list of systems we would recommend you take a look at –

Linux and Unix-based systems

Nmap	http://www.insecure.org/nmap/
Whisker	http://www.wiretrip.net/rfp/bins/whisker/whisker.tar.gz (source code)
	http://www.wiretrip.net/rfp/bins/whisker/whisker.txt (documentation)
ADM tools	ftp://adm.isp.at/ADM/
Other scanners	http://packetstorm.securify.com/UNIX/scanners/

Win32 based systems

eEye Retina	http://www.eeye.com/html/Products/Retina.html
Rhino9 tools	ftp://ftp.technotronic.com/rhino9-products/
Other scanners	http://packetstorm.securify.com/NT/scanners/

Chapter 5
The Security Lifecycle

The security lifecycle is a model documenting the steps that should be taken to work towards a secure network environment. Many Information Security companies publicise this model in order to educate users in the relevance of each stage. This chapter of the document will briefly cover the security lifecycle way of thinking and how Penetration Testing performs an integral part of the security assessment segment of the cycle.

The cycle follows this path –

Assessment -> Design -> Deployment -> Management

All models are based on the same 4 points, regarding the assessment, planning, deployment and management of Information Security risk and countermeasures.

Assess

This stage of the security lifecycle involves the assessment of Information Security risks and threats to the client hosts and networks. Penetration Testing emulates the external threat of hackers and attackers based on the Internet, and gives a crystal clear assessment of the risk to the target organisation.

Design

Designing and planning a secure network strategy is of paramount importance, as the foundations are laid down for a secure network that can be managed in an efficient manner.

Deploy

Deployment of a secure network will ensure a high level of security and efficient security systems that suit the business need of the organisation.

Manage

It’s all well and good having a secure network in place, but the Information Security risk needs to be managed to ensure ongoing improvement of security. Management brings support to the organisations networked infrastructure and Information Security systems, including firewall and IDS solutions.

Assessment of the Information Security risk to the target organisation is the first stage in the security lifecycle and vitally important to the rest of the cycle. Risks identified at the assessment stage will then be quashed through secure network design and implementation, and future risks and threats identified by managed security solutions.

Part II, Penetration Testing

This section of the book will cover Penetration Testing and the techniques involved when performing testing and Network Security Analysis in an accurate and effective way.

Chapter 6
Footprinting the Target Organisation

Depending on the level of blindness you have when it comes to a Penetration Test, you may or may not be required to perform footprinting. Some clients will only give you a company name or address of a building in which mission-critical servers are housed. It is important to identify routes into the target organisation and target servers, which could exist at various levels –

- The physical level
- The telephone level
- The Internet level

The physical level will cover physical access to the building and it’s computer networks. We have performed physical Penetration Tests against buildings before, and social engineering plays a large part of this.

Telephone level identification of routes to target networks would include the identification of telephone number ranges used by the target organisation. If the target organisation has a fax machine on 020 728 5520, and the direct dial number for the switchboard is 020 728 5000, the 020 728 5xxx range of numbers should be checked for the presence of modems or terminal servers. Many companies use terminal servers to allow dial-in access to their internal networks, this access can however be abused to give unauthorised access to internal hosts.

The Internet is currently the hackers choice of domain over which to launch attacks against companies. It provides an anonymous playground on which hackers can scan and probe hosts and networks to their hearts content with a low risk of being identified. Internet-level footprinting would simply include the identification of company networks and domain names.

Chapter 7
Host Enumeration and Network Identification

Assuming that you now have an idea of company Internet presence, domain names and IP address ranges in use. There are a handful of extremely useful techniques that can be adopted in order to identify other target networks and hosts.

DNS querying

Using nslookup, you can perform various DNS query functions in order to retrieve network information that can be used in turn to help map the target network space.

Below is an example of how you would list the mail exchange and DNS hosts for the domain example.com from using the nslookup command under a Unix-based environment –

```
$ nslookup
Default Server:  localhost
Address:  127.0.0.1

> set querytype=any
> example.com
Server:  localhost
Address:  127.0.0.1

Non-authoritative answer:
example.com     nameserver = NS.ISI.EDU
example.com     nameserver = VENERA.ISI.EDU

Authoritative answers can be found from:
example.com     nameserver = NS.ISI.EDU
example.com     nameserver = VENERA.ISI.EDU
> server ns.isi.edu
Default Server:  ns.isi.edu
Address:  128.9.128.127

> example.com
Server:  ns.isi.edu
Address:  128.9.128.127

example.com     nameserver = VENERA.ISI.EDU
example.com     nameserver = NS.ISI.EDU
example.com
        origin = VENERA.ISI.EDU
        mail addr = iana.ISI.EDU
        serial = 950301
        refresh = 43200 (12H)
        retry   = 3600 (1H)
        expire  = 1209600 (2W)
        minimum ttl = 86400 (1D)
example.com preference = 10, mail exchanger = VENERA.ISI.EDU
example.com preference = 20, mail exchanger = IANA.ISI.EDU
example.com     nameserver = VENERA.ISI.EDU
example.com     nameserver = NS.ISI.EDU
VENERA.ISI.EDU  internet address = 128.9.176.32
NS.ISI.EDU      internet address = 128.9.128.127
>
```

From querying the authoritative DNS server for the example.com domain (ns.isi.edu), we deduce that the e-mail relay host for the example.com domain is venera.isi.edu.

DNS zone files for domains are very useful, as they document sub-domains and other interesting information that we can use to build a good map of the target networks. The *host* command found on most Linux distributions allows us to glean DNS zone information for specific domains easily –

```
$ host –l example.com
EXAMPLE.COM name server VENERA.ISI.EDU
EXAMPLE.COM name server NS.ISI.EDU
DUMMY-HOST.EXAMPLE.COM has address 10.0.0.0
$
```

Large organisations with many networks will return copious amounts of DNS zone information, including the names of sub-domains, key servers and test or development hosts and networks.

More secure networks denying external access to DNS zone information will return the following –

```
$ host –l ibm.com
Server failed: Query refused
$
```

NIC querying

If you do not know the domain names or network ranges that the target organisation uses, you can perform *whois* queries to identify network ranges and domain names registered by the target organisation.

RIPE (the European NIC) has a very useful whois search engine on it's website that is very powerful when it comes to identifying networks and hosts owned by specific organisations, check it out at http://www.ripe.net/ripencc/search.html. Other NICs also have search engines, including ARIN (the Asia-Pacific NIC) at http://www.arin.net/whois/index.html and Network Solutions, who cover most of the other networks at http://www.networksolutions.com/cgi-bin/whois/whois.

ICMP ping-sweeping

Upon identifying all the IP addresses and network ranges owned and used by the target organisation, it is sensible to perform an ICMP ping-sweep to identify live accessible hosts in the network ranges.

Nmap is a useful tool for performing ICMP ping-sweeps, as it resolves the names of the hosts and identifies subnet broadcast and network addresses, below is an example of how to perform an nmap ICMP ping-sweep of a network range –

```
$ nmap -sP 192.168.7.1-48

Starting nmap V. 2.12 by Fyodor (fyodor@dhp.com, www.insecure.org/nmap/)
Host   (192.168.7.16) seems to be a subnet broadcast address (returned 1 extra
pings).  Skipping host.
Host cube.mis-cds.com (192.168.7.17) appears to be up.
Host onyx.mis-cds.com (192.168.7.18) appears to be up.
Host darkside.mis-cds.com (192.168.7.21) appears to be up.
Host   (192.168.7.31) seems to be a subnet broadcast address (returned 1 extra
pings).  Skipping host.
Host   (192.168.7.32) seems to be a subnet broadcast address (returned 1 extra
pings).  Skipping host.
Host test1.testbed.org (192.168.7.33) appears to be up.
Host dev1.testbed.org (192.168.7.35) appears to be up.
Host pdc.testbed.org (192.168.7.46) appears to be up.
Host   (192.168.7.47) seems to be a subnet broadcast address (returned 1 extra
pings).  Skipping host.
Host   (192.168.7.48) seems to be a subnet broadcast address (returned 3 extra
pings).  Skipping host.
Nmap run completed -- 48 IP addresses (11 hosts up) scanned in 7 seconds
$
```

From the results of the nmap scan, live hosts responding to ICMP can be identified and subnet information also. The subnet broadcast and network address information is extremely useful, as you may have ping-sweeped the entire class-c network that you find a target web server on, only to find that the target organisation owns 16 IP addresses of the block. As with the above example, the target domain that we are scanning may be mis-cds.com, and the testbed.org hosts and network range may belong to another organisation entirely.

Certain security-conscious organisations filter ICMP to mission-critical hosts and networks so that ping-sweeping in this fashion is not effective. Domains including microsoft.com and cert.org filter ICMP at their border routers in this way, so to identify active hosts each IP address in the network space has to be portscanned. It should be noted that forcefully scanning hosts in this fashion can be extremely time consuming.

Chapter 8
Network Scanning

The primary purpose of network scanning is to identify active TCP and UDP services running on hosts, the portscan results can also be used during further analysis to assess firewall and filter rulesets and identify the Operating Systems of the target hosts via. TCP/IP fingerprinting techniques.

Standard connect() TCP scanning

Vanilla TCP portscanning as it is sometimes known, is the most simple type of portscan to conduct. There is no stealth whatsoever involved in this form of scanning, as a TCP/IP connection is attempted to port 1 of the target host, then port 2, 3, 4 and so on, you get the idea. Due to the reliability of TCP/IP as a protocol, vanilla portscanning in this fashion is a very accurate way of determining which services are active on the target host.

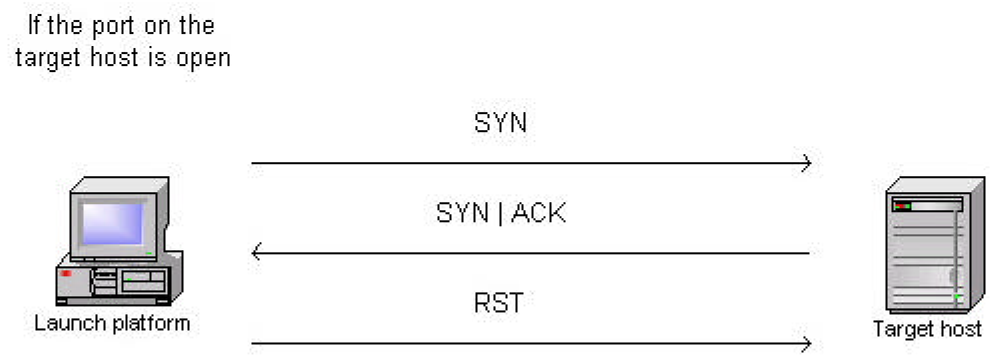
The majority of Windows-based portscanners available from PacketStorm perform scanning in this way, tcpprobe.c is a good example of a simple Unix-based connect() portscanner, which is available from http://packetstorm.securify.com/Exploit_Code_Archive/tcpprobe.c.

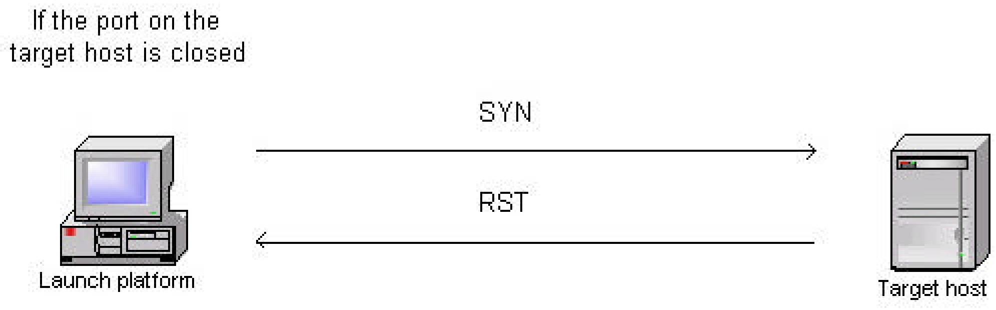
Stealth portscanning techniques

Half-open or SYN TCP scanning

Stealth scanning in this fashion evades some logging systems because of the fact that a full TCP/IP connection is never established. Usually a three-way handshake is initiated to synchronise a connection between two hosts. The client sends a SYN packet to the server, the server responds with SYN | ACK if the port is open and accepting connections, and the client sends an ACK to complete the handshake.

Below is a simple diagram –



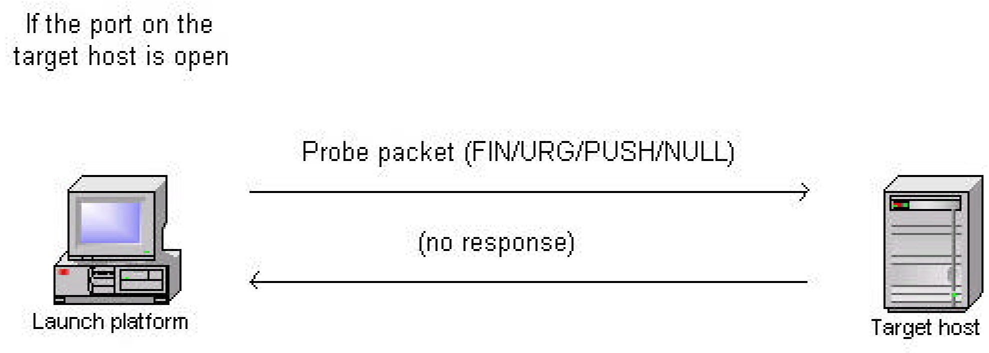


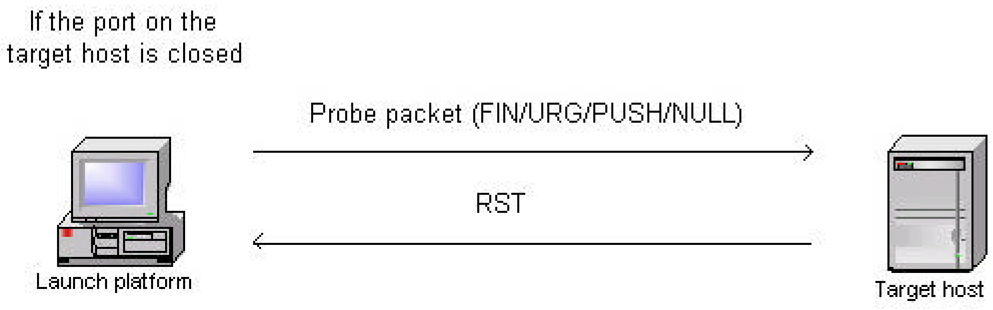
In the case of SYN portscanning, an RST packet is sent as the third part of the handshake, which resets the connection. Due to the fact that you have not completed the three-way handshake, the attempt of the connection is often not logged.

FIN, Xmas tree and Null scanning

Filtering systems such as firewalls can usually pick up on things like SYN packets being sent to sensitive ports on target hosts, programs are also available to log half-open scan attempts, including synlogger and Courtney. Probe packets with strange TCP flags set can sometimes pass through filters undetected.

Below is a simple diagram and explanation of this –


If the port on the target host is open
Probe packet (FIN/URG/PUSH/NULL)
(no response)
Launch platform
Target host


If the port on the target host is closed
Probe packet (FIN/URG/PUSH/NULL)
RST
Launch platform
Target host

The idea is that closed ports are required to reply to your probe packet with an RST, while open ports must ignore the packets in question (see RFC 793). The FIN scan uses a bare FIN packet as the probe, while the Xmas tree scan turns on the FIN, URG, and PUSH flags. The Null scan turns off all flags. Microsoft Operating Systems completely ignore this standard and FIN/Xmas/Null scans will not be effective against Windows hosts. Nmap supports all of these scanning types.

Spoofed portscanning

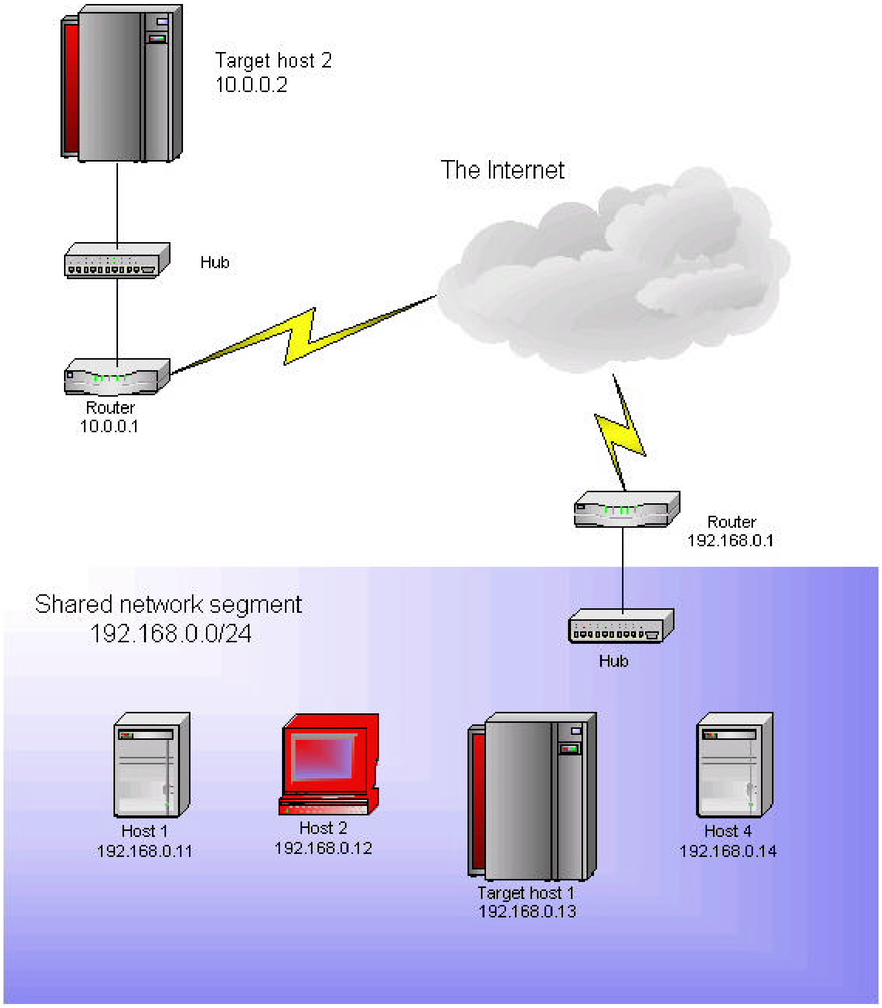
A new breed of publicly available scanner is spoofscan.c by jsbach, which is available from http://packetstorm.securify.com/UNIX/scanners/spoofscan.c. Spoofscan takes advantage of a fundamental vulnerability in shared network segments which allows such spoofing to take place.

Spoofscan works by sending out spoofed TCP/IP packets with a different source IP address to your own, and then sniffs the responses as they come back to your network segment. For this to work however, you have to either be on –

- The same shared network segment as the host you want to fake the scans from
- The same shared network segment as the target host that you want to scan
- Somewhere in between, on the same network segment as the router or gateway host which connects the target host directly or in-directly to the Internet

It also has a distinct benefit when evading pro-active IDS systems which may block scans from IP addresses that have been logged. If you have root access to a host on a shared class-c network segment of 254 IP addresses, you can spoof your portscan as originating from each and every routable IP address in the address space. There are various other scenarios when using spoofed portscans in this way, use your imagination.

The three basic scenarios are explained below with the following diagram –



In the diagram, we have root access to Host 2 and jsbach's spoofscan utility installed. Due to the fact that spoofscan sends out spoofed probe packets and then sniffs the responses using the shared network segment, we can spoof portscans from any host in the 192.168.0.* address space launched against the target host 2 across the Internet at 10.0.0.1.

In the same way, we could spoof a portscan launched against Target host 1 from any IP address, including 1.2.3.4 and 1.3.3.7. This technique can be used as a nifty DoS if portsentry or a pro-active security system has been deployed and is configured incorrectly. We could systematically spoof portscans from trusted and depended hosts, which would then be written into the hosts.deny file on Target host 1, and not be able to connect to the server later.

A slightly more theoretical way of performing spoofed portscans in this fashion would be to gain root access to a host that lies on a static route between the 192.168.0.0 and 10.0.0.0 networks, there are various possibilities depending on networking conditions in place.

Chapter 9
Information Gathering and Network Reconnaissance

By this stage you should already be aware of the target organisations networks and hosts and their IP addresses. The information gathering and network reconnaissance segment of the testing process is where relationships and paths of trust between hosts and other networks are identified.

All 'hacking' is based on exploiting vulnerabilities in established systems. Information Gathering and Network Reconnaissance is simply a process of exploiting vulnerabilities in network services such as fingerd in order to glean information that could prove useful. A basic concept of security is to identify and exploit the weakest link in the proverbial chain, one such way of achieving this is to identify trusted hosts and networks.

Fingerd

The fingerd service that runs on TCP port 79 by default can be queried in order to learn more about the target host, it's users and surrounding networks. Finger clients are part of most TCP/IP program suites nowadays, a technique that we adopt when performing reconnaissance against hosts with TCP port 79 open, is to issue finger requests with key words. The goal of this is to identify test or guest user accounts that have been set up, as these often have very weak passwords, or none whatsoever!

Below is a good example of an effective finger query issued from a Unix-based host –

```
$ finger user@target-host.com
Login: ftp                              Name: FTP User
Directory: /home/ftp                    Shell: /bin/sh
Never logged in.
No mail.
No Plan.

Login: samba                            Name: SAMBA user
Directory: /home/samba                  Shell: /bin/null
Never logged in.
No mail.
No Plan.

Login: test                             Name: test user
Directory: /home/test                   Shell: /bin/sh
Never logged in.
No mail.
No Plan.
$
```

From this we have been able to identify the test user 'test', who has never logged in. It is probable that the password for this account is weak.

Other keywords to issue as finger queries include –

user admin account guest test

Vulnerabilities exist in some Unix-based finger daemons, such as with IRIX and Solaris. If you issue a query of *finger 0@target-host* , it will return a complete listing of user's that have never logged into the host. With the Solaris fingerd, issuing a request such as *finger "1 2 3 4 5 6 7 8 9 0"@target-host* , will list many of the user accounts. There are various small vulnerabilities in fingerd implementations, it is recommended that you check security sites such as http://packetstorm.securify.com in order to identify these other vulnerabilities. It's often forgotten that simply performing *finger @target-host* will list all of the users currently logged into the host.

Below is an example of the Solaris fingerd bug –

```
$ finger "1 2 3 4 5 6 7 8 9 0"@example.com
[example.com]
Login       Name               TTY         Idle    When    Where
root     Super-User            console      <Jun  3 17:22> :0
admin    Super-User            console      <Jun  3 17:22> :0
daemon          ???                         < .  .  .  . >
bin             ???                         < .  .  .  . >
sys             ???                         < .  .  .  . >
adm      Admin                              < .  .  .  . >
lp       Line Printer Admin                 < .  .  .  . >
uucp     uucp Admin                         < .  .  .  . >
nuucp    uucp Admin                         < .  .  .  . >
listen   Network Admin                      < .  .  .  . >
nobody   Nobody                             < .  .  .  . >
noaccess No Access User                     < .  .  .  . >
nobody4  SunOS 4.x Nobody                   < .  .  .  . >
bob           ???            pts/0           1 Tue 00:08  nexus.sec
bob           ???            pts/1          3d Thu 01:57  nexus.sec
```

When issuing finger queries against hosts with many users, it is possible to identify weak trusted hosts on other networks. If you finger a user that has recently logged in, the finger daemon will return the last IP address that the user logged in from. Such information can be used to build a clear picture of the network and trusted hosts.

SMTP Services

The EXPN and RCPT TO: options within e-mail systems such as Sendmail can be exploited in order to learn more about users and internal networks. If fingerd is not found running on a target host, the EXPN command can be executed through the target SMTP server. Below is an example of this (although it should be noted that EXPN has to be enabled in the Sendmail configuration file) –

```
[bob@lisa bob]$ telnet example.com 25
Trying 192.168.0.190...
Connected to example.com.
Escape character is '^]'.
220 example.com ESMTP Server (Microsoft Exchange Internet Mail Service
5.5.2448.0) ready
helo
501 helo requires domain address
helo world.com
250 purple.flumps.org Hello mis-cds.com [207.155.248.7] (may be forged),
pleased to meet you
expn root
250 root <root@example.com>
expn test
550 test... User unknown
expn bob
250 Bob Sheppard <bob@mis-cds.com>
```

From EXPN querying it is possible to identify test users, e-mail aliases and true e-mail addresses. It is possible to use EXPN to identify all users on a box by issuing brute-force like queries. It should be noted that querying SMTP in this fashion will put a lot of junk into the logs of the target host and is a very loud way of checking for valid user accounts.

Looking at the above dump of the SMTP session on example.com, it should also be noted that the host is not running Microsoft Exchange. Many administrators attempt to mask the version of the SMTP service they are running by changing the banner that is displayed. It is possible however to identify the type of service (Sendmail, Qmail, Exchange, et al) by issuing commands such as HELO and checking the response that is given. Sendmail responds to HELO with '*501 helo requires domain address*', whereas Microsoft Exchange will respond to a HELO command with '*250 OK*'. Another way of determining the true type of SMTP service present is to issue unrecognised commands, such as this –

Sendmail 8.9.3

```
$ telnet nexus 25
Trying 192.168.0.26...
Connected to nexus.
Escape character is '^]'.
220 nexus ESMTP Sendmail 8.9.3/8.9.3; Mon, 26 Jun 2000 16:40:43 +0100
helo domain.com
250 nexus.mis-cds.com Hello cube.mis-cds.com [192.168.0.4], pleased to meet you
blaah
500 Command unrecognized: "blaah"
```

Microsoft Exchange

```
$ telnet darkside 25
Trying 192.168.0.9...
Connected to darkside.
Escape character is '^]'.
220 darkside.mis-cds.com ESMTP Server (Microsoft Exchange Internet Mail Service
5.5.2448.0) ready
helo domain.com
250 OK
blaah
500 Command not recognized.
```

Rusersd

The RPC service 'rusersd' can be queried to list all the users currently logged into the target host. Below is a quick example of this –

```
$ rpcinfo -p example.com
   program vers proto   port
    100000    2   tcp    111  portmapper
    100000    2   udp    111  portmapper
    100004    2   udp    991  ypserv
    100004    1   udp    991  ypserv
    100004    2   tcp    994  ypserv
    100004    1   tcp    994  ypserv
    100007    2   udp   1007  ypbind
    100007    2   tcp   1009  ypbind
    100009    1   udp    763  yppasswdd
    100002    1   tcp    998  rusers
$ rusers example.com
root jimmy bob
$
```

From this we can see that 'root', 'jimmy' and 'bob' are logged into example.com.

Chapter 10
The Checking of Network Services

Upon identifying active TCP and UDP network services, it is important to understand the services and exactly what they mean. Below is a matrix we have drawn up to help you understand the relevance of network services. It is recommended that you keep up-to-date with the BugTraq mailing list (at http://www.securityfocus.com under forums -> bugtraq) and security sites such as Packetstorm and eSecurityOnline (http://packetstorm.securify.com and http://www.esecurityonline.com).

The TCP and UDP Network Services Matrix

Port Number	Protocol	Service Name	Service Security Notes	Operating System that the service is usually found on
1	tcp	tcpmux	TCP Multiplexer service, runs by default on IRIX installations. Many hackers simply sweep network ranges for this open port to identify IRIX hosts	IRIX
7	tcp,udp	echo	Internal service used to test network connectivity by echoing values sent to the service	Unix-based
9	tcp,udp	discard	Internal service	Unix-based
11	tcp	systat	Displays systat information for the host	Unix-based
13	tcp,udp	daytime	Internal service	Unix-based
15	tcp	netstat	Displays netstat information for the host	Unix-based
17	tcp	qotd	Quote of the Day, novelty service used to generate random quotes for users	Unix-based
19	tcp,udp	chargen	Internal service used to generate random characters to test network connectivity	Unix-based
20	tcp	ftp-data	FTP data port, used to send and receive data from the File Transfer Protocol server	All
21	tcp	ftp	File Transfer Protocol command service, many Unix-based FTP services including WU-FTP and ProFTP have remote vulnerabilities	All
22	tcp,udp	ssh	Secure Shell, used as an encrypted telnet replacement. All login information is sent to the server in an encrypted for to prevent network sniffing	Unix-based

Port Number	Protocol	Service Name	Service Security Notes	Operating System that the service is usually found on
23	tcp	telnet	Standard command-line access service, usually used with Unix-based hosts to access and use them, default login accounts exist on various hosts and devices.	All
25	tcp	Smtp	Simple Mail Transfer Protocol, the e-mail relay service. Vulnerabilities exist in old Sendmail releases, and there are DoS issues in many mail services nowadays	All
37	tcp,udp	time	Time service	All
42	tcp	nameserver	Usually used by the Microsoft WINS name resolution service running from Windows NT hosts	Windows NT
53	tcp,udp	domain	Domain Name Service, there are many remote BIND exploits for Linux-based DNS services	All
67	tcp,udp	bootps	BOOTP server, used to boot workstations remotely	Unix-based and some hardware
68	tcp,udp	bootpc	BOOTP client	Unix-based and some hardware
69	udp	tftp	Trivial File Transfer Protocol, very weak implementation of the FTP protocol, commonly used by routers and hardware devices to upload new firmware	Network devices such as Routers
70	tcp,udp	gopher	Internet GOPHER, used before the WWW became popular	Unix-based
79	tcp	finger	Finger is used by many network Operating Systems to return information on logged in users. Fingerd can be exploited in some cases (see chapter 9) to return copious amounts of useful information	All
80	tcp	http	World Wide Web service, used to serve web pages, servers running Apache and IIS have alsorts of security issues with sample CGI scripts and features	All

Port Number	Protocol	Service Name	Service Security Notes	Operating System that the service is usually found on
88	tcp,udp	kerberos	Kerberos, used as a secure encrypted authentication service when users log into hosts	All
109	tcp	pop2	Post Office Protocol v.2, used before v.3 was released	All
110	tcp	pop3	Post Office Protocol v.3, used by end-users to pick up e-mail. POP3 can be abused to brute-force user login/password combinations, as many POP3 daemons do not log failed login attempts	All
111	tcp,udp	rpcbind	The rpcbind, or RPC portmapper service returns a listing of the active RPC services running a host, rpcbind can be queried from a Unix-based host by running the *rpcinfo* command	Unix-based
113	tcp	auth	Identd, used to authenticate login names to sockets on networked hosts, can be queried in conjunction with a portscanner to identify the users of processes running on high ports, see nmap's –I option.	All
115	tcp	sftp	Secure File Transfer Protocol, an encrypted version of FTP	Unix-based
119	tcp	nntp	Network News Transfer Protocol, used to serve Usenet information to users, some Linux-based NNTP daemons are vulnerable to remote compromise	All
123	tcp,udp	ntp	Network Time Protocol, used to synchronise networked device clocks	All
135	tcp,udp	loc-srv	Location service	Windows NT
137	tcp,udp	netbios-ns	NetBIOS name service, used in Windows networking and filesharing	Primarily Windows, although SAMBA runs on many Unix-based platforms.

Port Number	Protocol	Service Name	Service Security Notes	Operating System that the service is usually found on
138	tcp,udp	netbios-dgm	NetBIOS datagram service, used in Windows networking and filesharing	Primarily Windows, although SAMBA runs on many Unix-based platforms.
139	tcp,udp	netbios-ssn	NetBIOS session service, used in Windows networking and filesharing, login and password information for NetBIOS shares can be brute forced using ADMsmb available from (ftp://adm.isp.at/ADM/ADMsmb-v0.2.tgz)	Primarily Windows, although SAMBA runs on many Unix-based platforms.
143	tcp,udp	imap2	Internet Message Access Protocol v.2, allows users to pick up e-mail whilst retaining the original message on the server. Many sites run POP3 instead. Various remote vulnerabilities exist in Unix-based implementations of IMAP2.	Unix-based
161	udp	snmp	Simple Network Management Protocol, often runs on Hardware such as Routers, Switches and Network Printers. Tools such as ADMsnmp (available from ftp://adm.isp.at/ADM/ADMsnmp.0.1.tgz) are good for brute-forcing SNMP community strings (the equivalent of passwords for SNMP)	Unix-based and Network devices such as Routers, Switches and Printers
162	udp	snmptrap	SNMP trap service, used to manage SNMP enabled devices and their operation	Unix-based and Network devices
389	tcp,udp	ldap	Lightweight Directory Access Protocol, used in x.500 networks, querying LDAP can be used to gain useful information	All
443	tcp,udp	https	Secure HTTP service, used in secure transactions with SSL	All
512	tcp	exec	rexecd, used to execute commands remotely	Unix-based
513	tcp	login	rlogind, uses .rhosts to authenticate users, the r-services can be abused and spoofed by determined attackers to access hosts	Unix-based

Port Number	Protocol	Service Name	Service Security Notes	Operating System that the service is usually found on
513	udp	who	rwho displays information of logged in users	Unix-based
514	tcp	shell	rshd, uses .rhosts to authenticate users as with rlogind. Hackers often abuse rshd to access hosts without being logged, rcp can also be used to transfer files between hosts running rshd unlogged	Unix-based
514	udp	syslog	syslogd, used to log to the syslog file across networks	All
515	tcp	lpd	Line Printer Daemon, used to print across TCP/IP networks, a vulnerability exists in Linux LPD that can result in a remote compromise	Network Printers, Windows NT and Unix-based
517	udp	talk	Used in Unix environments for communication between users on different hosts	Unix-based
520	udp	route	Used to update routing tables dynamically, as with RIP. A serious vulnerability exists in IRIX and other BSD-derived systems which can be used as an effective DoS against hosts running routed, see http://rootshell.com/archive-j457nxiqi3gq59dv/199801/riptrace.c.html for exploit information	All
540	tcp	uucp	Unix-to-Unix Copy Protocol, used to copy files between Unix hosts, fairly primitive with weak authentication	Unix-based
1080	tcp	socks	Socks proxy service, used to proxy TCP/IP traffic, can be exploited if misconfigured to access trusted hosts and networks, mileage may vary	All
1433	tcp	ms-sql	Microsoft SQL server port	Windows NT
1524	tcp	ingreslock	Ingreslock, used by many hackers as a ‘backdoor’ port as in the case of many Solaris remote exploits	Unix-based
1999	tcp	cisco-discovery	Cisco discovery protocol, many Cisco devices have this port open by default	Cisco Devices

Port Number	Protocol	Service Name	Service Security Notes	Operating System that the service is usually found on
3128	tcp	squid-http	Squid webproxy service, performs caching of pages to increase efficiency	Unix-based
3306	tcp,udp	mysql	MySQL SQL server port	Unix-based
6667	tcp	irc	Internet Relay Chat server port	All
8080	tcp	webcache	Webcache servers use this port to perform proxying and caching functions to increase web-browsing efficiency on large networks	All

There are many other services which have not been listed here. The above listing is a to-the-point breakdown of important services that should be identified and checked. Please check other security resources for more information about network services.

RPC Services

RPC services should also be checked if the RPC portmapper service is found running on port 111. Below is a matrix of common RPC services identified with relevant information regarding the security risks inherent when running the systems –

RPC Service Number	RPC Service Name	Security Notes and Information	Operating Systems commonly found on
100001	rstatd	Displays system memory and CPU information	Unix-based
100002	rusersd	Returns the login names of users currently logged in	Unix-based
100003	nfsd	Network File System service	Unix-based
100004, 100007, 100009, 100028	YP services	Yellow Pages services, superseded by NIS. YP is used to share user login and account information across a network and maintain synchronisation of account information	Unix-based
100005	mountd	The NFS mountd service, handles mount requests for exported directories. A *showmount* command can be executed from a Unix-based host in order to list exported directories	Unix-based
100008	walld	Walld, displays messages to all logged in users, can be abused to flood users across the Internet	Unix-based

RPC Service Number	RPC Service Name	Security Notes and Information	Operating Systems commonly found on
100017	rexd	Remote execution daemon, easily exploited to gain remote ‘bin’ access to hosts, a **very** dangerous service to be running	Unix-based
100024	status	Handles NFS status information. There are remote exploits available for statd running on Solaris 2.4 and 2.5 systems	Unix-based
100068	cmsd	The Solaris Calendar Management System, vulnerable to remote root compromise on earlier Solaris versions (2.6 and before)	Solaris primarily
100083	ttdbserverd	The Tooltalk Database Server, vulnerable on most platforms to remote compromise (HP-UX, Solaris, IRIX, et al)	Unix-based
100232	sadmind	Solaris Solsuite remote administration system, can be exploited in Solaris 2.7 and before to gain remote root access	Solaris primarily
100300	nisd	The Network Information Service Daemon, superseded Yellow Pages. A remote vulnerability exists in Solaris 2.5 nisd	Unix-based
150001	pcnfsd	PCNFS, used to allow PC’s and Windows workstations to access NFS exports on Unix-based hosts. There are various remotely exploitable vulnerabilities in pcnfsd, and it’s recommended that the latest version is deployed	Unix-based

There are many RPC services which are vulnerable to remote compromise, including those running on IRIX and Linux-based platforms. Check security resource sites such as http://packetstorm.securify.com and http://www.SecurityFocus.com in order to identify relevant risks to the target hosts and networks you are testing.

Chapter 11
Assessing the Risks and Vulnerabilities

All active network services should be checked for vulnerabilities and security risks. The next step of the testing process is to assess the risks and the impact to business in the event of an external threat exploiting the vulnerabilities and compromising client hosts and networks. Testing of services in this fashion usually follows the following path –

Identify open network port -> Identify type of service and function -> Identify release and version of service -> Identify relevant enabled options

Upon identifying the exact services running, their function and enabled options, the service information should be cross-referenced against a vulnerability matrix to identify risks. Risks and threats usually fall into the following 3 categories; High, Medium and Low risk.

High risk vulnerabilities are also known as *showstoppers*, such vulnerabilities can jeopardise host or network security if exploited and present a serious threat to the organisations integrity. Vulnerabilities that are classified as high risk in most cases would include –

- A remote DoS (Denial of Service) risk that crashes the entire server.
- A remote vulnerability in a network service that if exploited, will give hackers immediate root or super-user access to the server.
- A misconfiguration or default service or feature enabled that can easily be exploited to gain user access to the server.
- Any vulnerability that can lead to unauthorised access of network resources

Depending on the client organisation, vulnerabilities that are usually considered as a medium risk, may be placed in the high risk category. If a threat is known that prevents the organisation from conducting it's business, it is considered high risk.

Medium risk vulnerabilities are not as serious, usually presenting only a nuisance to Network Operations and Systems Administration staff. Examples of medium risk vulnerabilities would include –

- A service such as fingerd or rusersd running, which can be used to glean important network and user information
- A service such as walld or talkd that can be abused to flood users

Low risk vulnerabilities are usually commonplace in networks, and cannot be exploited to directly gain unauthorised access to network resources. Examples of low risk vulnerabilities include –

- Services such as telnetd being configured in a way that the exact Operating System of the target host can be identified
- Misconfigurations in services which can allow user login names to be identified

When assessing the risks present on a target network, it is important to keep the assessment relevant to the client organisation. Large financial sites usually will be threatened by more determined online attackers than a company that makes pencils for example. It is important to be realistic in the analysis of the threats to the target organisation.

Chapter 12
Exploiting the Vulnerabilities

To exploit vulnerabilities running on various platforms and hardware types, it is important to have access to a variety of hosts running similar Operating Systems and hardware as the target hosts you wish to exploit. For example a remote exploit for Solaris rpc.statd, will usually only compile and run from a Solaris host, this is also usually true for exploiting systems running on platforms such as as IRIX and Windows NT.

There are primarily three types of tool and program used to exploit weaknesses in networks and hosts remotely; buffer overflows, denial of service tools, simple exploits and brute-force tools. Below is a brief overview of such exploits and how they work.

Buffer Overflows and Denial of Service

Buffer overflows exist in programs usually because of poor programming and bounds checking of variables passed to programs as arguments. In essence these vulnerabilities can be exploited in two ways –

- To perform a function that is useful to the attacker, such as spawn a remote super-user prompt from where the attacker can access the host

- To make the machine crash through inducing a memory fault. This type of vulnerability is commonly exploited under systems such as Windows NT where it is difficult for attackers to write machine code to perform functions such as spawn command-prompts.

In a nutshell, a buffer overflow exploit will write an amount of machine, or shellcode, into the target hosts memory stack, and then proceed to execute the code. Many Unix-based services run as the super-user root, and so any shellcode that is passed into the target host's memory stack is subsequently executed as root. The shellcode is usually sent to the server by passing it as an argument to the program with padding data to ensure it is injected into the right region of the memory stack.

Such vulnerabilities exist in many Unix-based services; primarily in DNS, FTP and RPC services. Since the source code for commercial Operating Systems such as Solaris and IRIX has been available to hackers through various channels, many new vulnerabilities in remote services have been identified. Exploits usually have to be compiled on the same platform as the target platform that will be exploited. If a hacker has cmsd.c, a Solaris rpc.cmsd remote exploit for Solaris 2.6, then he will require access to a similar Solaris host in order to compile the exploit and use it. Below is a list of remote exploits that are used by hackers to compromise Unix-based servers running vulnerable services to give remote super-user access to the target host –

http://www.technotronic.com/horizon/statd.tar.gz	rpc.statd/automountd remote exploit for Solaris 2.5(1)
http://www.technotronic.com/horizon/pcnfsd_remote.tar.gz	rpc.pcnfsd remote exploit for Unix-based platforms
http://rootshell.com/archive-j457nxiqi3gq59dv/199810/rpc.ttdbserver.c.html	rpc.ttdbserver remote exploit for various Unix-based platforms
http://packetstorm.securify.com/9905-exploits/w00f.c	WU-FTP 2.4.2 remote exploit for Linux

It is always good to keep up-to-date with the latest exploits and tools as they are released. Security sites such as PacketStorm and SecurityFocus offer a good starting point. Below is a listing of recommended security sites which many professionals access daily in order to keep up-to-date with tools and exploits –

http://packetstorm.securify.com
http://www.rootshell.com
http://mixter.warrior2k.com
http://www.securityfocus.com
http://www.technotronic.com
http://www.w00w00.org
http://ADM.freelsd.net/ADM/
http://www.wiretrip.net/rfp/

Windows NT DoS

Useful information such as shellcode can be written to the executable stack which then performs a function that is useful to the attacker, this is only possible if the memory locations are known and the data is executed correctly. In the case of Windows NT remote buffer overflows, it is very difficult to execute commands in the memory of the target Windows NT host in this fashion. Due to this difficulty, many vulnerabilities identified in Windows NT platforms are remote Denial of Service vulnerabilities, as the information written to the memory stack is often junk, which then crashes the server process or the server itself.

In the case of buffer overflows, there is a fine line between executing code that is useful to you, and crashing the service by writing bad information into memory.

Simple Exploits

Vulnerabilities exist primarily in web-based systems such as CGI scripts which are easily exploited with simple HTTP GET requests. Below is an example of the now outdated PHF technique syntax –

http://www.example.com/cgi-bin/phf?Qalias=x%0a/bin/cat%20/etc/passwd

PHF is a search-like CGI script that is run from default Apache installations, it can however be exploited because of the incorrect bounds checking that the program performs. As part of the PHF query, the Qalias value is set to x, and then a carriage return (%0a) is passed. The arbitary command we want to run is then passed, which in this case is /bin/cat /etc/passwd, httpd is run as the nobody user in most cases, and the /etc/passwd file is subsequently displayed.

There are many other CGI vulnerabilities that can be exploited simply by using web browsers such as Netscape Navigator. By passing malformed query strings, it is possible in many cases to retrieve system files which can lead to access of the target host being gained. New CGI exploits are being published on security sites such as http://packetstorm.securify.com which can be easily used by most users with access to a networked machine with a web browser.

Simple exploits such as these take advantage of incorrect bounds checking as Buffer Overflow exploits do. The conditions in place when it comes to CGI exploits however allow attackers to run commands through the HTTP query string that is passed to the script, as opposed to having to craft complex shellcode that is relevant to the architecture of the target host.

Brute Force Tools

Brute force techniques are adopted if you have a lot of time on your hands during a Penetration Test. The goal of a brute force attack is usually to gain access to a network resource or server by guessing the password of a user. Many of the latest Operating Systems log failed login attempts in order to flag brute force password cracking attempts against some services such as telnet. Historically, services including POP3 do not log failed login attempts.

Professional Penetration Testing teams often have access to hosts on fast networks to launch such brute force attacks from. Distributed computer clusters can also be configured to offer a highly effective stage from which brute forcing can be achieved. It is very important that you have a good dictionary file when brute forcing in this fashion, an extensive archive of dictionary files is available from http://packetstorm.securify.com/Crackers/wordlists/.

Windows and SAMBA

If the target host is running Windows Filesharing or SAMBA using NetBIOS on ports 137 and 139, you can launch a brute force attack against the host and it's shares to gain access to the host. ADMsmb and NAT are useful tools for brute-forcing Windows and SAMBA NetBIOS shares. You can download them from the following locations –

ADMsmb

ftp://ADM.freelsd.net/pub/ADM/ADMsmb-v0.2.tgz	(Unix-based)

SNI NetBIOS Auditing Tool (NAT)

http://packetstorm.securify.com/NT/scanners/nat10.tar.gz	(Unix-based)
http://packetstorm.securify.com/NT/scanners/nat10bin.zip	(Windows)

SNMP

The Simple Network Management Protocol runs on UDP port 161 by default. SNMP is used by Network Operations and Systems Administration staff to monitor network devices such as routers, switches and network printers. SNMP community strings are use as authentication and can be brute forced easily. Primarily there are two types of SNMP community string, public and private. The public community string allows the user to read the SNMP information from the host, and the private string allows users to change the SNMP information. ADMsnmp is a powerful and flexible tool for brute forcing SNMP community strings, it can be downloaded from –

ftp://ADM.freelsd.net/pub/ADM/ADMsnmp.0.1.tgz

Upon knowing the SNMP community strings, the *snmpwalk* and *snmpset* tools under Unix-based environments can be used to read and modify the SNMP tables of the target device.

POP3

POP3 services and daemons run on TCP port 110 and usually do not log failed login attempts. Due to the fact that no logging occurs, brute forcing POP3 is perfect choice for hackers looking to guess a valid user login and password for the target host. Upon identifying a target user or list of valid users through querying services such as fingerd and rusersd, one can launch a brute force attack in order to guess the user’s password. The following POP3 brute force systems are recommended –

http://rootshell.com/archive-j457nxiqi3gq59dv/199707/pop3.c.html
http://packetstorm.securify.com/Crackers/hypno.zip
http://packetstorm.securify.com/Crackers/hv-pop3crack.pl

Others

There are brute force systems available for all services that require simple authentication in the form of a login and password combination, below are some more examples –

Unix-based

ftp://ADM.freelsd.net/pub/ADM/ADMftpforce.tgz
http://rootshell.com/archive-j457nxiqi3gq59dv/199710/brute_ssl.c.html
http://rootshell.com/archive-j457nxiqi3gq59dv/199707/brute_web.c.html
http://packetstorm.securify.com/UNIX/scanners/mailbrute.c

Windows-based

http://packetstorm.securify.com/Crackers/sqlbf.zip
http://packetstorm.securify.com/groups/thc/thc-lh11.zip
http://packetstorm.securify.com/Win/sslcrack.zip
http://packetstorm.securify.com/Crackers/ntsweep.zip

Misc

http://packetstorm.securify.com/Crackers/

Chapter 13
Upon Compromising Host Security

Upon compromising host security you have many options. The goals of the Penetration Testing phase are usually set out before testing commences, tokens such as trophies are often used to prove vulnerabilities and risks to organisations from a real-world standpoint. For example the goal of a Penetration Test may be to retrieve the payroll information for the target organisation.

Depending on the goals of your Penetration Test, the following avenues are usually explored upon compromising host security –

- Backdooring the local host
- Circumventing security measures and accessing trusted hosts
- Performing Denial of Service (DoS)

Backdooring the Local Host

In order to gain access to the compromised host in the future without being logged or seen by Systems Administration or Network Operations Staff, *rootkits* and *backdoors* can be deployed in order to replace programs on the target host with our own, which operate in a fashion that is beneficial to us. Most backdoors give access to hosts, others hide processes or network connections.

The standard /usr/bin/login program under Solaris allows users to log into the host and also writes information to the utmp, wtmp and lastlog files. Due to the fact that such information has been written to the system logs by the login program, users can see when they are logged in by typing the *who* command for example.

A backdoored Solaris /usr/bin/login program, is based on the original Solaris 2.x source tree. An effective backdoored login binary will operate normally, until someone logs in with the trigger username, which is usually '0wnership' or 'check_mate'. Upon a user logging in with this username, he is assigned a terminal number as usual, but nothing is written to the utmp, wtmp or lastlog files. The attacker does not appear in the system logs when he is accessing the host, although his connection and processes can be tracked using *netstat* and *ps* commands. To get around this, we can deploy *netstat* and *ps* backdoors which hide processes when logged in as the trigger username. Different triggers can be used to give remote super-user access to hosts, including connecting to a service from a specific source port as in the case of inetd backdoors.

The trigger in most cases is hardcoded into the backdoored binary. Even if all of the user passwords in the /etc/passwd and /etc/shadow files are changed, access will still be granted via. the backdoored login binary upon a user logging in with the trigger username. All authentication systems that are trusted in this way can be abused upon compromising a host to give access later.

Loadable Kernel Modules can be used to backdoor the host kernel in order to outsmart the Systems Administration staff. A good kernel backdoor is extremely effective depending on the security systems in place. Backdoors that take advantage of different network protocols such as ICMP can also be used to evade logging by TCP/IP based systems such as netstat.

Backdoors and rootkits can be downloaded from security sites including PacketStorm at the following URLs –

http://packetstorm.securify.com/UNIX/penetration/rootkits/
http://packetstorm.securify.com/trojans

More simple Windows-based backdoors include –

http://www.bo2k.com/indexdownload.html	cDc Back Orifice 2000
http://packetstorm.securify.com/trojans/NB20Pro.exe	NetBus
http://subseven.slak.org/download.html	Sub Seven

Circumventing Security Measures and Accessing Trusted Hosts

From having local access to a host, security measures such as IP-based filtering systems can be circumvented and access to other hosts gained by gleaning useful information from the compromised host. We have previously performed Penetration Tests where Cisco Router configuration information for a DMZ had been stored in a user's home directory. This configuration information was then used to access the Cisco routers that formed the DMZ, and reconfigure them to allow our traffic through.

The following techniques and methodologies can be adopted to circumvent security measures and access other hosts –

- Checking of the local filesystems for useful information
- Network sniffing
- Spoofing to circumvent network-based filtering systems
- Spoofing to hide the true source of aggressive network probes

Checking of the local filesystems can be used to good effect depending on the Information Security awareness of the Network Operations staff who use the system on a daily basis. Systems Administrators often leave very interesting networking configuration information in their home directories which can be used to compromise other hosts and networks. It is always a good idea to download the /etc/passwd and /etc/shadow files in order to crack the passwords, as many users use the same passwords across networks and hosts. Also we have encountered situations where the latest patches have been installed and access to the compromised host lost, where logging in as a normal user (through the proverbial front door) is the only way to compromise the host.

Network sniffing is a massively effective technique adopted by many hackers upon compromising a host. A sniffer can be used to capture user login and password information as it is sent across a shared network segment such as a hub or thin ethernet. Sniffers work by putting the network card of the compromised host into promiscuous mode, where it picks up all information flowing across the shared network segment. Sniffers work by logging sessions that use TCP/IP in plaintext, such as telnet, ftp and rlogin/rsh. My favorite sniffer right now is Greedy Dog, which compiles on Solaris, Linux and BSD-based Operating Systems, it is available from http://packetstorm.securify.com/sniffers/gdd13.c other sniffers are available from PacketStorm's archives at http://packetstorm.securify.com/sniffers/.

By installing switched networks, sniffers will not be effective. ARP spoofing however, can be adopted to modify routes in place so that all the hosts on the local network think you are the gateway host for that network. IP forwarding must then be enabled on your host so that the packets are passed on to the real gateway and out to the Internet or other networks. These techniques are covered with diagrams in other papers available from the MIS website.

If the host you have compromised is on the same shared network segment as other target hosts, spoofing and hijacking techniques can be used to compromise such systems. Spoofing and hijacking in this way are covered in the aforementioned paper (hubs-and-switches.doc). Portscanning systems such as spoofscan by jsbach can be used to launch spoofed portscans and network probes against other hosts in order to mask the true source of the probes.

There are countless scenarios that you could get into upon compromising hosts, with limited access to certain hosts and networks in some cases. It’s really a case of working with what you have access to and attempting to circumvent the security of other hosts, in order to achieve your goals.

Performing Denial of Service (DoS)

Upon compromising a host and having access to the local network, some effective Denial of Service attacks can be launched against local hosts. An effective form of Denial of Service across a local network would be to spoof ARP replies in a malicious manner with a tool such as hunt. Hunt can be downloaded from –

http://packetstorm.securify.com/sniffers/hunt/hunt-1.5.tgz

There are many, many other effective DoS techniques that can be adopted in this scenario, it’s important that you keep up-to-date with these threats by checking security sites such as PacketStorm and SecurityFocus.

Part III, Secure Network Design Guidelines

This section of the book gives very brief pointers and introduces concepts that will help you to understand the methods and techniques adopted in designing and implementing secure networks. If you are looking for a detailed book documenting the pro's and con's of security architectures and how they work, you should read books such as –

> Network Intrusion Detection : An Analysis Handbook
> by Stephen Northcutt
> New Riders Publishing; ISBN: 0735708681

Chapter 14
The 'Hurdles' Approach

If we go back to chapter 2 we will remember that it is never impossible for a hacker to break into a network, only improbable. Due to this fundamental fact, a network can never be 100% secure. To currently combat hackers and Internet-based attackers it is important that Systems Administration and Network Operations Staff are aware of the risks and threats to their business and how to protect their infrastructure.

The hurdles approach is a concept that we have always used to design secure networks depending on the client's business need and Information Security budget.

So a network can never be 100% secure, it's all about understanding the risks at a network level. If you install a firewall which filters all ports on your hosts except for 22 (ssh), 25 (smtp), 80 (http) and 110 (pop3), you may think you're secure. Infact all you're doing is limiting the amount of services that attackers can access. If a hacker finds a vulnerable CGI script running on your web servers, he can compromise the host, even with a firewall in place. You may then install an Intrusion Detection System, to monitor the traffic that isn't filtered by your firewall, but hackers can use tools such as whisker, fragrouter and ides, to evade detection by IDS systems –

> Whisker is available from http://www.wiretrip.net/rfp/bins/whisker/whisker.tar.gz
> Fragrouter is available from http://www.anzen.com/research/nidsbench/
> IDES is available from http://mixter.warrior2k.com/ides.c
>
> A good paper that documents the IDS evasion techniques that whisker uses is available from http://packetstorm.securify.com/papers/IDS/whiskerids.html.

The key point to remember here is that of time. The more security systems that you deploy which can complement each other, the longer it will take an attacker to circumvent them. The longer an attacker spends circumventing all your security measures, the more he will be logged by your Intrusion Detection Systems.

Hackers are usually opportunists, and the more hurdles you lay down on the track that the hacker has to run in order to compromise your network, the longer it will take. The idea is to make the race so long, that the hacker takes one look at your network and gives up.

Chapter 15
Firewalling Concepts

Firewalls should be configured in a way that is relevant to the organisation that the firewall will be protecting. This is achieved by first understanding the realistic threats and risks that are posed to the network, and gearing the configuration of the firewall around those risks in order to minimise and manage them.

The way that firewalls should be usually configured is to filter all traffic, then open up ports on the firewall in line with the business need. If you have a web server behind your firewall, you should open up traffic to port 80 of the server. Many firewalls are deployed with all ports unfiltered, then filters put in on port 23 (telnet) and 111 (rpcbind) to minimise access. This kind of filtering is ineffective against determined attackers.

Chapter 16
DMZ Configuration

De-Militarised Zones should be configured in an aggressive way as with firewalls. All traffic allowed onto your DMZ should be filtered and then access to specific ports given as the need arises. DMZ networks are often quite complicated, it should be remembered that a DMZ is more-or-less a quarantine zone on your network, containing publicly accessible servers. In order to minimise the risk of an internal network compromise, correct filtering should exist between your internal and DMZ hosts. We have seen many DMZ's where traffic can only flow onto the DMZ from specific trusted theatre hosts on the Internal network. It's a case of minimising the visibility that a hacker has of your internal networks, in the event that the DMZ is compromised.

Chapter 17
Defeating Portscanning Techniques

To effectively defeat portscanning techniques adopted in 'the wild', it is important to understand the tools and systems that attackers use to probe your networks.

Many portscanning systems including the highly popular nmap, are very 'loud' when they probe target networks. In order to identify hosts that are alive, an ICMP ping-sweep of the target network range is conducted. Hosts that respond to the ICMP pings sent by nmap are then probed with TCP and UDP portscans.

By filtering ICMP at your border routers, portscanning in this fashion will prove extremely time consuming for the attacker, as to accurately scan your networks he will have to forcefully check every single IP address and it's TCP/UDP ports. Scanning in this way can take days instead of minutes if ICMP is not filtered.

If you run a grass-roots network of sorts and you're the only Systems Administrator, you could run key services such as telnet and ftp on very high ports. Most hackers won't scan your ports over 2000, so it's usually a good idea to run services such as telnet and ftp on very high ports (I've seen them running on 60023 and 60021 before). This is practising security through obscurity, but it is effective in most cases.

Chapter 18
Pro-active Security Systems

Pro-active security systems can be used to assess and manage risks in real time. Hackers can be hindered by pro-active systems which perform the following functions –

- Identify Threat
- Assess Risk
- Manage Risk

Threat identification is usually in the form of detecting a portscan or network probe or attack from an IP address on the Internet, network and host-based Intrusion Detection Systems are capable of this.

Assessing the risk is the next step that the system has to take, in order to reach a sound outcome for the management phase. The system may assess that there is no risk, and simply ignore the threat. If the threat continues and grows, it is re-assessed and may be managed differently.

In order to manage the risk, a network reconfiguration is usually undertaken. A pro-active security system will assess the risk and then act accordingly. Most pro-active systems will filter the attackers IP address at a network or host level, and then raise relevant alarms to indicate that an attack has been attempted.

Misconfigured pro-active security systems can also be used to perform Denial of Service against target hosts and networks. It is important that trusted servers (ie. servers on the same network segment) are placed into the configuration files so that if an attacker spoofs an attack from a trusted IP address, it isn't placed into the hosts.deny file and can no longer access the server.

Good examples of Intrusion Detection Systems are –

Portsentry	http://www.psionic.com/abacus/portsentry/
Snort	http://www.snort.org
Various Unix-based IDS systems	http://packetstorm.securify.com/UNIX/IDS/
Various Windows NT IDS systems	http://packetstorm.securify.com/NT/IDS/