Table of Contents

Secure Systems

Preface

Internet security is a very dynamic field and eCommerce has simply brought the whole issue into sharper focus. Anyone working in the eCommerce field today must live on a day-to-day basis with the problems and threats posed by poorly secured systems. In the light of this, most professionals find themselves constantly needing to acquaint themselves with the latest loop-holes that have been discovered and the tools to fix them. Sometimes this is just a debate that rages about the potential problems and sometimes it is more serious. Most larger eCommerce companies will hold a weekly seminar where all the interested parties gather to discuss the current security status of the system. If a big break-in occurs, then it is a sure thing that someone's head will roll.

This assignment is my opportunity to experience the ease (or not) of getting hold of useful information about emerging and, in some cases, established technology and solutions to the security issue. For my report, I will outline the technical basis and deployment context of the following subjects:

- Hacker tools as a means of self-evaluating system security
- Kerberos
- System Complexity Management
- Intrusion Detection Systems

The list of areas that I will explore and document for each selected subject include,

- An introduction and description to the technology and the security issue(s) it addresses/causes.
- How the technology would be deployed/managed?
- What form and degree of protection does the technology provide?
- What is the shelf-life of the technology? (i.e. Is it (or is it likely to become) an industry standard or is it some back-yard solution?)

Hacking tools as a means of self-evaluating system security

You cannot deny that the Internet has opened up the information border in our world. Information in the media has vastly increased during the past 20 to 30 years. Remote Access, eCommerce, telephony, video conferencing, information distribution and freedom of speech evolved and was the result of this technological revolution. Similarly, the good as well as the bad has been migrated as well. During the 1960's, there was a famous robbery called "The Great Train Robbery". It was a dirty and violent crime where several million pounds were taken. Nowadays, clean and non-violent electronic robbery exists while the number of physical crime reduced. Theft over the Internet has grown fast, where millions of dollars could be transferred through a click of a button. This had lead to an increase in security over the World Wide Web. The issues of security in relation to guarding information against loss or danger are primarily concerned with ensuring that information keeps its integrity.

Most cases of hacking or cracking attempts occur from hardware and software flaws, bugs, or gaps, caused both by the attacker, and by the user or programmer, sometimes intentionally while at other times by mistake. The early hackers (few in numbers) had a talent, or 'a knack' for finding such holes. They documented as many flaws as they could, and also created hacking/cracking tools to automate the infiltration. These tools and documents can work to a security manager's advantage; after all,

> *"To defeat the enemy, one must first know the enemy"*[1]

Outlining and analysing all the tools will be beyond the scope of this report, but the theory could be summed up into two sections.

Know the exploits & **Fix/Prevent the exploit**

It is very necessary for a security administrator to have a documented report on all the possible exploits, bugs, and flaws found on his software, hardware, and network topology so that he would be prepared for any attacks that may happen, and have an advantage or be one step ahead in prevention and patching. It is also important for the administrator to have the necessary software and hardware tools needed to prevent, fix, and monitor the system.

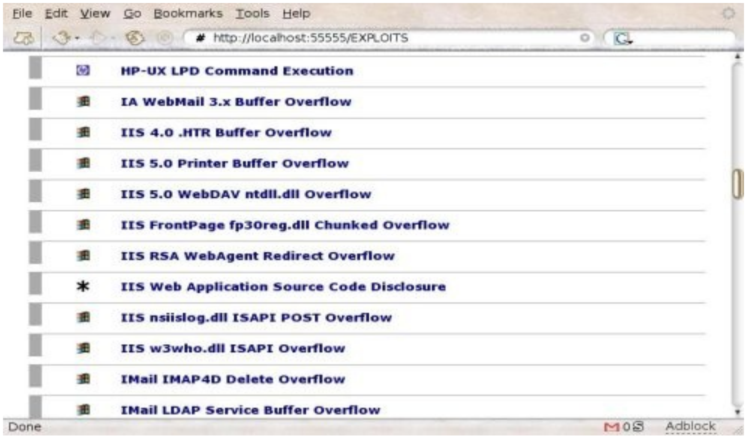
Know The Exploits

Of all the research I have done, Metasploit emerges at the top of the list. 'The Metasploit Framework is an advanced open-source platform for developing, testing, and using vulnerability exploit code. The goal is to provide useful information to people who perform penetration testing, IDS signature development, and exploit research. It is created to fill the gaps in the information publicly available on various exploitation techniques and to create a useful resource for exploit developers'..[2]

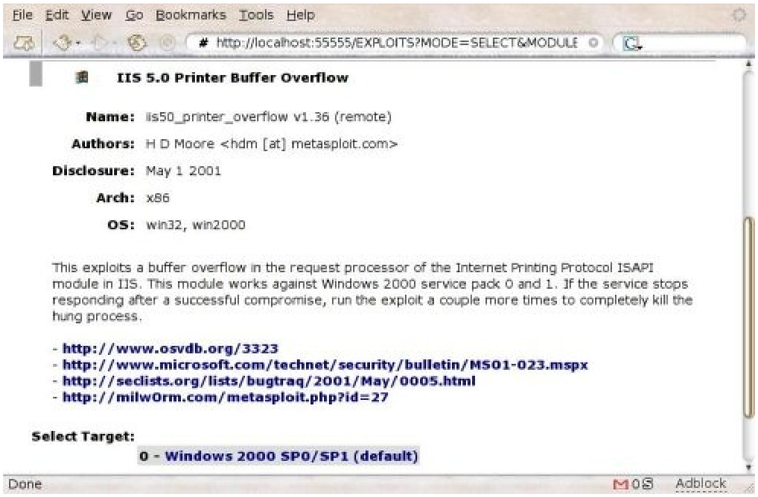
[1] Sun Tzu – The Art of War

[2] Metasploit - http://www.metasploit.com/

The following page shows two screen shots of Metasploit and a possible exploit.


File Edit View Go Bookmarks Tools Help
http://localhost:55555/EXPLOITS
Done
Adblock

HP-UX LPD Command Execution
IA WebMail 3.x Buffer Overflow
IIS 4.0 .HTR Buffer Overflow
IIS 5.0 Printer Buffer Overflow
IIS 5.0 WebDAV ntdll.dll Overflow
IIS FrontPage fp30reg.dll Chunked Overflow
IIS RSA WebAgent Redirect Overflow
IIS Web Application Source Code Disclosure
IIS nsiislog.dll ISAPI POST Overflow
IIS w3who.dll ISAPI Overflow
IMail IMAP4D Delete Overflow
IMail LDAP Service Buffer Overflow


File Edit View Go Bookmarks Tools Help
http://localhost:55555/EXPLOITS?MODE=SELECT&MODULE
Done
Adblock

IIS 5.0 Printer Buffer Overflow

Name: iis50_printer_overflow v1.36 (remote)
Authors: H D Moore <hdm [at] metasploit.com>
Disclosure: May 1 2001
Arch: x86
OS: win32, win2000

This exploits a buffer overflow in the request processor of the Internet Printing Protocol ISAPI module in IIS. This module works against Windows 2000 service pack 0 and 1. If the service stops responding after a successful compromise, run the exploit a couple more times to completely kill the hung process.

- http://www.osvdb.org/3323
- http://www.microsoft.com/technet/security/bulletin/MS01-023.mspx
- http://seclists.org/lists/bugtraq/2001/May/0005.html
- http://milw0rm.com/metasploit.php?id=27

Select Target:
0 - Windows 2000 SP0/SP1 (default)

In this example, a security administrator running IIS 5.0 now knows of that particular exploit, and can repair a fix, or proper filters to reduce any damage or prevent such an attack. One could argue that an attacker also has access to such information, but security is a cat and mouse game. One must always be prepared, and tool providing such knowledge has a deep impact and gives a high degree of protection. It is frequently updated, giving it a long lasting industrial standard shelf-life.

Moderation, Prevention, & Repair

I have tested many different varieties of software and hardware self evaluation tools, and here is a sum of the most commonly used ones,

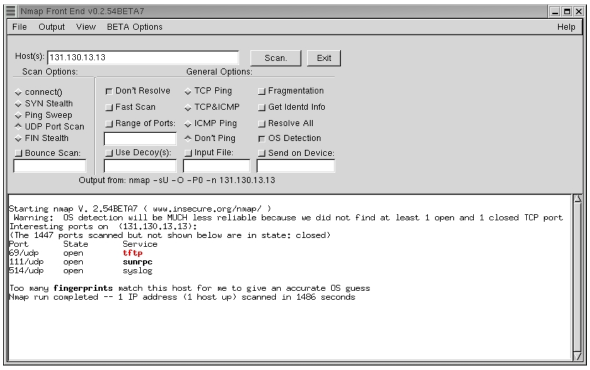
- Scanning
    - Port & Packet Scanning
        - NMAP
        - Ethereal
- Monitoring & Repairing
    - AA Tools
    - Spybot Search & Destroy

Port & Packet Scanners

NMAP:

NMAP is a free open source utility for network exploration, security scanning, security exploitation, and server testing. It is open source, allowing experienced users to refine the software, fix errors and bugs, as well as improve its quality and power by distributing its source code to the public. NMAP is designed to both scan single hosts, as well as giant networks of computers. When scanning computers, NMAP can find what hosts are available on the network, what operating system is installed and running, and what type of filters and firewalls are in use. An advantage of NMAP is that it can run on many different platforms, such as Windows and Unix operating systems. Its multi platform availability, along with ease of use and power makes it one of the best scanning software you can use.

NMAP can be used by a security administrator to search for loopholes, unnecessary open ports, and other security flaws that could be running on a network. For my example, I deliberately infected my virtual machine with a back-door Trojan[3]. When scanning with NMAP, not only did I discover its open port, but I also discovered other peer-to-peer ports[4] that, in a professional and secure environment, should not be running as well.



Host(s): 131.130.13.13

Scan Options:

○ connect()
○ SYN Stealth
○ Ping Sweep
● UDP Port Scan
○ FIN Stealth
□ Bounce Scan:

General Options:

☑ Don't Resolve
□ Fast Scan
□ Range of Ports:
□ Use Decoy(s):
○ TCP Ping
○ TCP&ICMP
○ ICMP Ping
● Don't Ping
□ Input File:
□ Fragmentation
□ Get Identd Info
□ Resolve All
☑ OS Detection
□ Send on Device:

Output from: nmap -sU -O -P0 -n 131.130.13.13

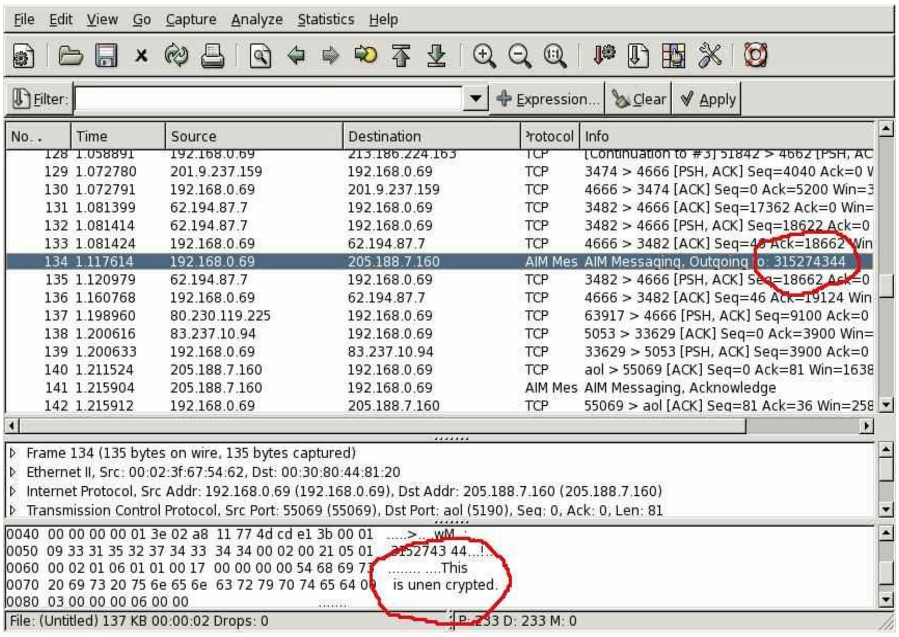
```
Starting nmap V. 2.54BETA7 ( www.insecure.org/nmap/ )
 Warning:  OS detection will be MUCH less reliable because we did not find at least 1 open and 1 closed TCP port
Interesting ports on  (131.130.13.13):
(The 1447 ports scanned but not shown below are in state: closed)
Port       State       Service
69/udp     open        tftp
111/udp    open        sunrpc
514/udp    open        syslog

Too many fingerprints match this host for me to give an accurate OS guess
Nmap run completed -- 1 IP address (1 host up) scanned in 1486 seconds
```

[3] Sub7 - http://www.hackpr.net/~sub7/main.shtml

[4] Ports 6882 (edonkey network), and 6667 (IRC)

Ethereal

Ethereal is a network packet analyser used for troubleshooting, and analysis of a network. It can capture over 700 different protocols "off the wire" from a live network connection, and give a complete overview of the packet's contents, of which includes source and destination addresses, type of packet, and data. In the wrong hands a hacker may capture unencrypted passwords, or even obtain enough information to inject his own poisonous packet for further penetration. However, in the right hands, it could be used to detect malicious packets through the use of advanced filtration. For a full list of features, visit the Ethereal website[5].



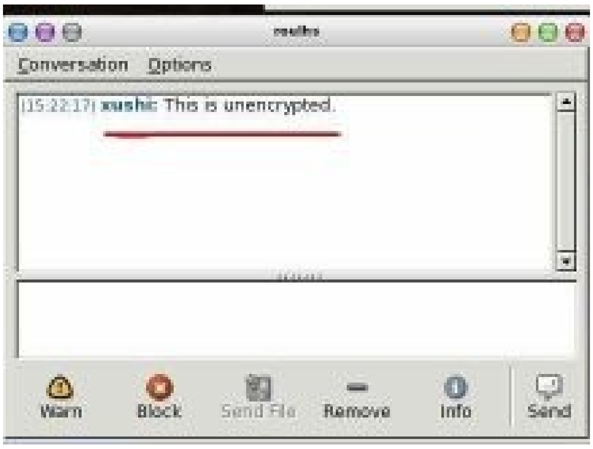
No. .	Time	Source	Destination	Protocol	Info
128	1.058891	192.168.0.69	213.186.224.163	TCP	[Continuation to #3] 51842 > 4662 [PSH, AC
129	1.072780	201.9.237.159	192.168.0.69	TCP	3474 > 4666 [PSH, ACK] Seq=4040 Ack=0 W
130	1.072791	192.168.0.69	201.9.237.159	TCP	4666 > 3474 [ACK] Seq=0 Ack=5200 Win=3
131	1.081399	62.194.87.7	192.168.0.69	TCP	3482 > 4666 [ACK] Seq=17362 Ack=0 Win=
132	1.081414	62.194.87.7	192.168.0.69	TCP	3482 > 4666 [PSH, ACK] Seq=18622 Ack=0
133	1.081424	192.168.0.69	62.194.87.7	TCP	4666 > 3482 [ACK] Seq=46 Ack=18662 Win
134	1.117614	192.168.0.69	205.188.7.160	AIM Mes	AIM Messaging, Outgoing o: 315274344
135	1.120979	62.194.87.7	192.168.0.69	TCP	3482 > 4666 [PSH, ACK] Seq=18662 Ack=0
136	1.160768	192.168.0.69	62.194.87.7	TCP	4666 > 3482 [ACK] Seq=46 Ack=19124 Win
137	1.198960	80.230.119.225	192.168.0.69	TCP	63917 > 4666 [PSH, ACK] Seq=9100 Ack=0
138	1.200616	83.237.10.94	192.168.0.69	TCP	5053 > 33629 [ACK] Seq=0 Ack=3900 Win=
139	1.200633	192.168.0.69	83.237.10.94	TCP	33629 > 5053 [PSH, ACK] Seq=3900 Ack=0
140	1.211524	205.188.7.160	192.168.0.69	TCP	aol > 55069 [ACK] Seq=0 Ack=81 Win=1638
141	1.215904	205.188.7.160	192.168.0.69	AIM Mes	AIM Messaging, Acknowledge
142	1.215912	192.168.0.69	205.188.7.160	TCP	55069 > aol [ACK] Seq=81 Ack=36 Win=258

```
▷ Frame 134 (135 bytes on wire, 135 bytes captured)
▷ Ethernet II, Src: 00:02:3f:67:54:62, Dst: 00:30:80:44:81:20
▷ Internet Protocol, Src Addr: 192.168.0.69 (192.168.0.69), Dst Addr: 205.188.7.160 (205.188.7.160)
▷ Transmission Control Protocol, Src Port: 55069 (55069), Dst Port: aol (5190), Seq: 0, Ack: 0, Len: 81
```

```
0040  00 00 00 00 01 3e 02 a8  11 77 4d cd e1 3b 00 01   .....>.. .wM...;..
0050  09 33 31 35 32 37 34 33  34 34 00 02 00 21 05 01   .3152743 44...!..
0060  00 02 01 06 01 01 00 17  00 00 00 00 54 68 69 73   ........ ....This
0070  20 69 73 20 75 6e 65 6e  63 72 79 70 74 65 64 00    is unen crypted.
0080  03 00 00 00 06 00 00                               .......
```

```
File: (Untitled) 137 KB 00:00:02 Drops: 0        P: 233 D: 233 M: 0
```

The images show a live capture of all packets generated in a typical network. In this particular example, a packet was captured of an unencrypted message being sent through the Internet.

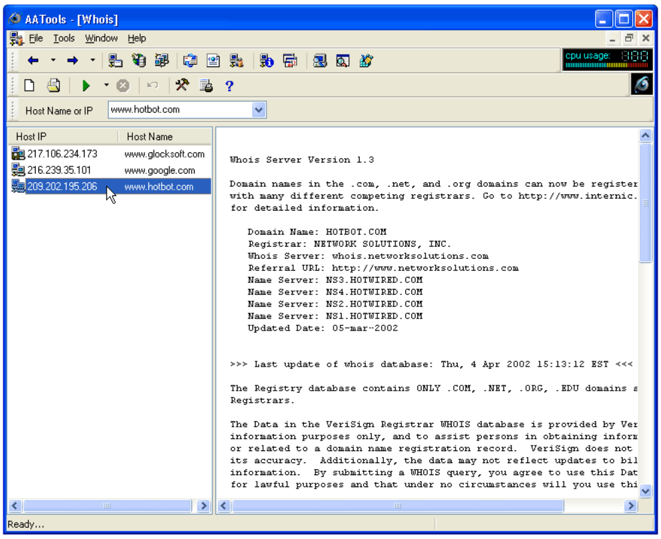


(15:22:17) xushi: This is unencrypted.

[5] Ethereal Features - http://www.ethereal.com/introduction.html#features

Monitoring & Repair

AATools

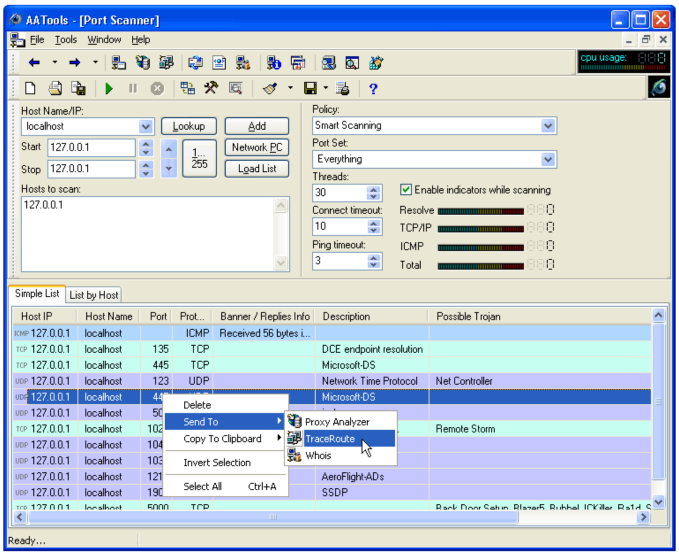


Host Name or IP: www.hotbot.com

Host IP	Host Name
217.106.234.173	www.glocksoft.com
216.239.35.101	www.google.com
209.202.195.206	www.hotbot.com

```
Whois Server Version 1.3

Domain names in the .com, .net, and .org domains can now be register
with many different competing registrars. Go to http://www.internic.
for detailed information.

   Domain Name: HOTBOT.COM
   Registrar: NETWORK SOLUTIONS, INC.
   Whois Server: whois.networksolutions.com
   Referral URL: http://www.networksolutions.com
   Name Server: NS3.HOTWIRED.COM
   Name Server: NS4.HOTWIRED.COM
   Name Server: NS2.HOTWIRED.COM
   Name Server: NS1.HOTWIRED.COM
   Updated Date: 05-mar-2002


>>> Last update of whois database: Thu, 4 Apr 2002 15:13:12 EST <<<

The Registry database contains ONLY .COM, .NET, .ORG, .EDU domains a
Registrars.

The Data in the VeriSign Registrar WHOIS database is provided by Ver
information purposes only, and to assist persons in obtaining inform
or related to a domain name registration record.  VeriSign does not
its accuracy.  Additionally, the data may not reflect updates to bil
information.  By submitting a WHOIS query, you agree to use this Dat
for lawful purposes and that under no circumstances will you use thi
```

Ready...



Host Name/IP: localhost

Start: 127.0.0.1

Stop: 127.0.0.1

Hosts to scan: 127.0.0.1

Policy: Smart Scanning

Port Set: Everything

Threads: 30

☑ Enable indicators while scanning

Connect timeout: 10

Ping timeout: 3

Resolve

TCP/IP

ICMP

Total

Host IP	Host Name	Port	Prot...	Banner / Replies Info	Description	Possible Trojan
ICMP 127.0.0.1	localhost		ICMP	Received 56 bytes i...		
TCP 127.0.0.1	localhost	135	TCP		DCE endpoint resolution	
TCP 127.0.0.1	localhost	445	TCP		Microsoft-DS	
UDP 127.0.0.1	localhost	123	UDP		Network Time Protocol	Net Controller
UDP 127.0.0.1	localhost	44			Microsoft-DS	
UDP 127.0.0.1	localhost	50				
TCP 127.0.0.1	localhost	102				Remote Storm
UDP 127.0.0.1	localhost	104				
UDP 127.0.0.1	localhost	103				
UDP 127.0.0.1	localhost	121			AeroFlight-ADs	
UDP 127.0.0.1	localhost	190			SSDP	
TCP 127.0.0.1	localhost	5000	TCP			Back Door Setup, Blazer5, Bubbel, ICKiller, Ra1d, S

Ready...

AATools is an advanced security diagnostic tool used to protect a computer against malicious hackers and crackers by scanning for malicious ports opened, as well as other security issues one may have in his system. AATools has many different functions you can use,

- Port Scanner: scans ports for suspicious ones, usually opened by malicious programs such as Trojan horses or worms.
- Proxy Analyser: tests lists of proxies and/or verifies a list of addresses on present proxy servers.
- Trace Route: shows you the path a packet sent from your machine to some other machine on the network takes as it hops from router to router. It will show you the IP address and the actual name of each router, line by line.
- Network Monitor: Shows you information on outbound and inbound network connections.
- Process Monitor: displays the process running on a system, along with some information on what they are.
- Whois: a network information utility that allows you to find out all the available information about IP addresses, host names, location, NSP name.

However, this powerful software could be a great tool for hackers when used in the wrong hands. A hacker can scan a network for possible holes or bugs, and use them to gain unauthorized access, or to bring down the network. But overall, if a security administrator uses such software on their system, they can find out all the holes they have first, and with the help of NMAP, along with proper anti virus and firewall software, they could keep one step ahead and safely secure their networks against any unauthorized access.

Conclusion

No system is 100% secure, but with proper training, knowledge, and the right choice of tools, an administrator can greatly reduce the chances of penetration, theft, or data loss. The tools mentioned here are but the tip of the iceberg. Many alternatives exist, and many different technologies can be used, of which can range from anti virus scanners, to backups redundancy and fallbacks, to hardware firewalls and network / security policies. The main point to note is that these utilities provide a great advantage when it comes to evaluating and securing systems.

Kerberos

Kerberos is a network authentication protocol system based on private-key cryptography, designed to provide a strong authentication mechanism for client/server based applications and network infrastructures. It allows clients to prove there identity to the server by means of a trusted third-party server to issue session keys for interactions between the server and the clients. Kerberos is built by MIT upon an open source infrastructure, and the underlying code, as well as the technology itself is freely available to the public. That has given it an advantage where commercial software systems incorporated the technology into their own products. Most modern operating systems, such as Windows 2000, Linux and Unix distributions, such as Redhat and Sun's Solaris come built in or include Kerberos. Even Cisco routers support Kerberos authentication for telnet connections.

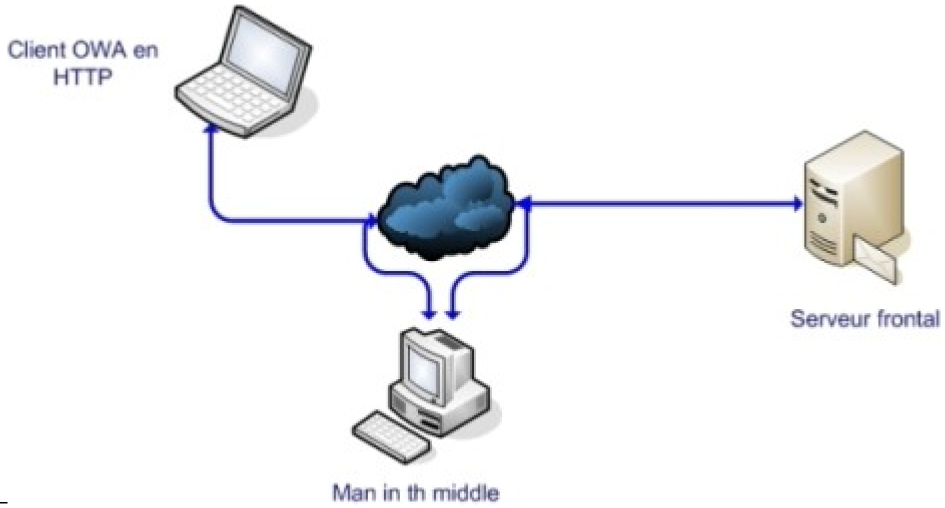
The security risks

It has been known that the Internet is an insecure worldwide network. It's protocols were built to guarantee successful delivery of information from the source to its destination. However, the main priority of the protocols is reliability, and not security. Any hacker standing in the middle of the transfer, or even pretending to be the destination can easily obtain the information he needs. The hacker could sniff and obtain unencrypted passwords in both a local and a wide area network with ease. Even the encrypted passwords could be broken into with the right tools, social engineering, and time.

There are many different methods of attack used nowadays. Below is a summary of the most popular methods used of which Kerberos aims to prevent.

Eavesdropping

Eavesdropping is the unauthorised interception of a data transmission, also known as the 'Man in the Middle' attack, where a hacker intercepts specific packets to obtain information needed either in part or whole, or to use to break into a secure system, as shown below[6].



[6] Image - http://www.laboratoire-microsoft.org/articles/win/OWA/images/man%20in%20the%20middle.jpg

Masquerading

A user presents him/herself to the system as another user. This may be done in order to gain unauthorized access to information or resources, to disseminate (mis)information in another's name, or to block or deny a system from operating correctly.[7] It could be done from a high level (a user physically impersonating), and can go as deep as masquerading cryptographic key signatures, IP Addresses, or MAC Addresses.

Infiltration

Unauthorised access caused by weak security points in a system or network.

Why use Kerberos ?

Firewalls were introduced to reduce unauthorised access and increase security of the 'internal' side of the network. However, most firewalls assume that the attacker is on the outside of the network, while it has been researched that a very large percentage of unauthorised access and damage is done from the inside too. Firewalls also cannot distinguish perfectly the difference between a user, and a hacker pretending to be that user to gain information.

> *"firewalls are simply a less extreme example of the dictum that there is nothing more secure then a computer which is not connected to the network --- and powered off! In many places, these restrictions are simply unrealistic and unacceptable."*[8]

This is where Kerberos emerges. Kerberos aims to plug the missing hole found in between server and client authentication. It uses a combination of cryptography and encryption so that a client and server can prove their identities to one another, and send their data securely.

The Key Aspects

The following are the key aspects of Kerberos, in slight detail.

Kerberos Server (S):

This is the master security server, which verifies logins for users and clients. The Kerberos master server knows all the private keys of the users and all keys that will be granted by the ticket granting service, it's a main server that the system works around.

Authentication Server (AS):

The Authentication Server (AS) deals with login details and requests from different user clients. The AS also authenticates users giving them a key to be used to set up a secure channel with the server.

[7] http://ac.bcc.ctc.edu/Policies/definitions.htm

[8] Center for Computational Science - http://www.cmf.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html#whatis

Key Distribution Centre: (KDC)

The KDC deals with setting up a secure channel, safe from eavesdropping, by giving out a message known as a ticket, used to gain a server's trust in a user. This way the user is trying to prove they are who they say they are. The procedure is done via issuing a nonce – a random key with different number and each number is different every time. The client and server do not share the same encryption key. This generated key is known as a session key and the ticket is what helps both end points receive the key securely.

The Kerberos ticket is a certificate that has been generated by the authentication server and its is encrypted using the servers key. This ticket contains information on who the ticket is intended for as well as the session key and the time stamp being used for the validity of the ticket.

Timestamps:

Timestamps are needed for directory replication and conflict resolution, as well as authentication.

Kerberos Encryption:

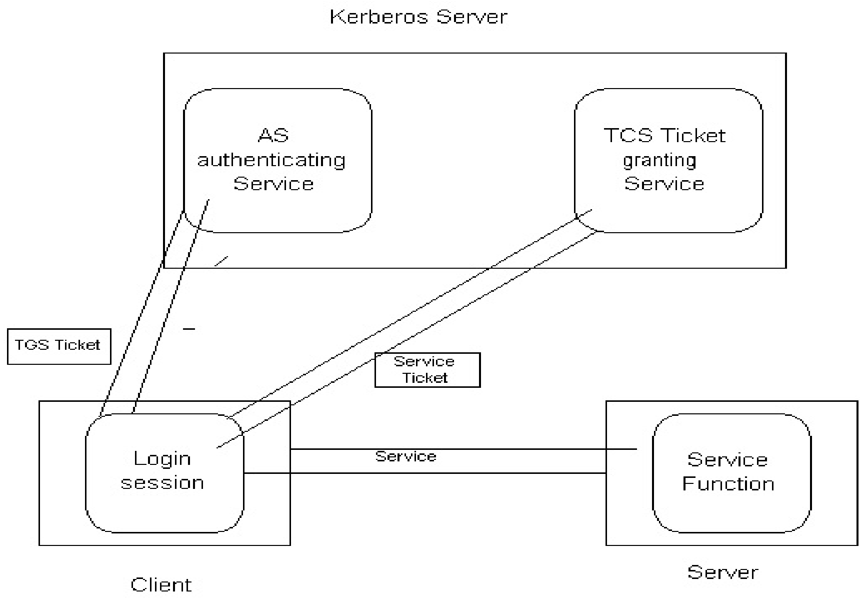
> *"The Kerberos protocol uses **strong cryptography** so that a client can prove its identity to a server (and vice versa) across an insecure network connection. After a client and server has used Kerberos to prove their identity, they can also encrypt all of their communications to assure privacy and data integrity as they go about their business."*[9]

To be more specific, Kerberos uses the 'Needham & Schroeder' authentication protocol, a method of distribution of a shared symmetric key by a trusted server and mutual authentication. There are five basic steps for authentication and secure connection. Below is the mathematical terms used to describe the steps, and a diagram that aids in the definition.

A, B, S	Two clients, **A, B**, and a Secure server **S**
Na, Nb	Nonce for A and B
Kas, Kbs, Kab	keys
dec	nonce -> nonce. decryption.

1. A-> S : { A, B, Na }
2. S -> A: {Na, B, Kab, {Kab, A}Kbs}Kas
3. A-> B: {Kab,A}Kbs
4. B-> A: {Nb}Kab
5. A-> B: {dec(Nb)}Kab

[9] MIT - http://web.mit.edu/kerberos/www/#what_is



Weaknesses of Authentication with Kerberos

Kerberos is designed to be used in a small network, or set of networks. There are other limitations to security, of which include the fact that the passwords used by the user could be easily discovered.

> *"Kerberos makes no provisions for host security; it assumes that it is running on trusted hosts with an untrusted network. If your host security is compromised, then Kerberos is compromised as well."*[10]

One could argue that basing the security trust onto one trusted machine could be problematic. If that machine was targeted for an attack and was brought down, then the whole security infrastructure goes down as well. It poses a great risk if the authentication server is compromised. Further examples of Kerberos vulnerabilities and flaws are discussed in Appendix A.

The shelf life and future of Kerberos

Its is a known fact then when using the Kerberos network protocol you attempt to switch platforms. Each time this is done you need a new password. This is given by the ticket granting service. The life time of this ticket is five days from time of initial authentication. Using Kerberos five authenticate the US Atlas SSH Gateways is a step forward. The new proposal will let you in the very near future use Kerberos 5 passwords that will replace legacy native AFS and legacy NIS/UNIX passwords.

Conclusion

One can come to a conclusion that no matter how hard you try there will always be a flaw in the system and there will be always new technology made in order to prevent attacks. In a man made world no system is perfect, we just have to try our best to be safe. Kerberos is a security measure that has more good than bad. This standard can be kept with constant upgrades and new methods to deal with security

[10] http://www.cmf.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html#weakness

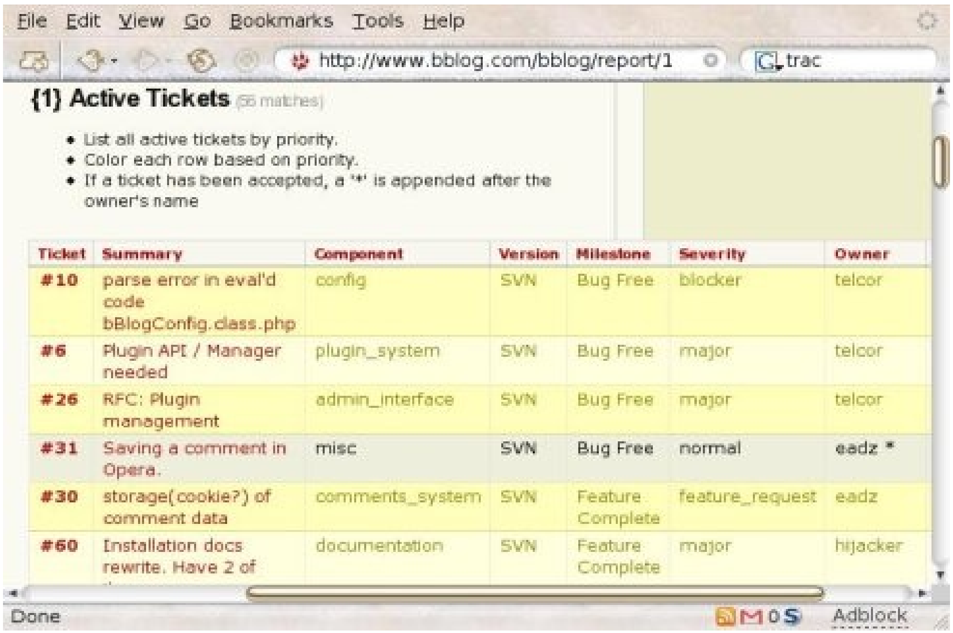
System Complexity Management

When securing a network, one should make note of the complexity of modern technology. For managing all the software, hardware, theoretical and technical cracks in a network to result in a 100% secure and stable environment is virtually impossible.

The term 'complexity' is formally defined as, a system of many parts which are coupled in a non-linear fashion, where a change on one side is not proportional to change on the other [11]. A

If we are to take an example in chess, if you leave a game of chess over night, and someone approaches the board and changes a character's move, the difficulty of you arriving the next day to track back all the moves and discover the modified one is the complexity an administrator faces in reality in a network to back trace bugs and security flaws, as well as repair them or find a solution. One should also note that in real terms, a hacker never leaves over night.

Complexity management is not about fixing the actual security holes or bugs in a system, but it is about making the process feasible and plausible. Physically having the ability to fix a mistake in a piece of code is one thing, but having the advantage of simplicity, organisation, and redundancy analysis to make the job possible is where complexity management fits in. For example, Trac[12], is an enhanced ticketing system used in many open source projects that allows for software complexity management. Below are two images of how complexity management eases tracking both a problem in a system as well as a change within the system's code.

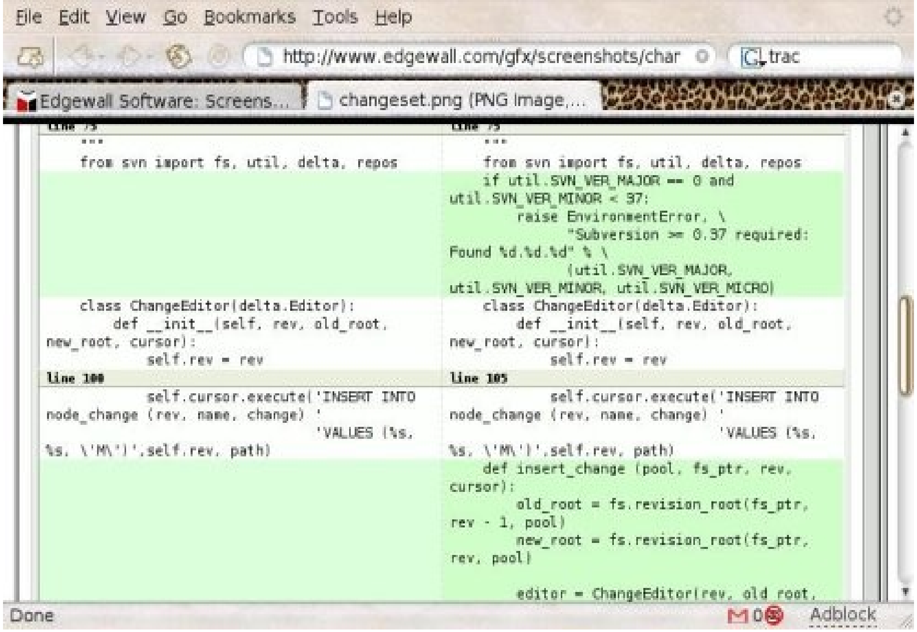

File Edit View Go Bookmarks Tools Help
http://www.bblog.com/bblog/report/1
trac

{1} Active Tickets (56 matches)

- List all active tickets by priority.
- Color each row based on priority.
- If a ticket has been accepted, a '*' is appended after the owner's name

Ticket	Summary	Component	Version	Milestone	Severity	Owner
#10	parse error in eval'd code bBlogConfig.class.php	config	SVN	Bug Free	blocker	telcor
#6	Plugin API / Manager needed	plugin_system	SVN	Bug Free	major	telcor
#26	RFC: Plugin management	admin_interface	SVN	Bug Free	major	telcor
#31	Saving a comment in Opera.	misc	SVN	Bug Free	normal	eadz *
#30	storage(cookie?) of comment data	comments_system	SVN	Feature Complete	feature_request	eadz
#60	Installation docs rewrite. Have 2 of	documentation	SVN	Feature Complete	major	hijacker

[11] Wikipedia - http://en.wikipedia.org/wiki/Complex_system
[12] Trac – http://www.edgewall.com/trac/


File Edit View Go Bookmarks Tools Help
http://www.edgewall.com/gfx/screenshots/char
trac
Edgewall Software: Screens...
changeset.png (PNG Image,...
Done
Adblock

```
line 75
    """
    from svn import fs, util, delta, repos

    class ChangeEditor(delta.Editor):
        def __init__(self, rev, old_root,
new_root, cursor):
            self.rev = rev
line 100
            self.cursor.execute('INSERT INTO
node_change (rev, name, change) '
                                'VALUES (%s,
%s, \'M\')',self.rev, path)
```

```
line 75
    """
    from svn import fs, util, delta, repos
    if util.SVN_VER_MAJOR == 0 and
util.SVN_VER_MINOR < 37:
        raise EnvironmentError, \
              "Subversion >= 0.37 required:
Found %d.%d.%d" % \
              (util.SVN_VER_MAJOR,
util.SVN_VER_MINOR, util.SVN_VER_MICRO)
    class ChangeEditor(delta.Editor):
        def __init__(self, rev, old_root,
new_root, cursor):
            self.rev = rev
line 105
            self.cursor.execute('INSERT INTO
node_change (rev, name, change) '
                                'VALUES (%s,
%s, \'M\')',self.rev, path)
    def insert_change (pool, fs_ptr, rev,
cursor):
        old_root = fs.revision_root(fs_ptr,
rev - 1, pool)
        new_root = fs.revision_root(fs_ptr,
rev, pool)

        editor = ChangeEditor(rev, old_root,
```

However, is the reduction of complexity always a good thing? Without complexity a system would cease to evolve. If you have a system with ten functions, having the ability to enhance it into fifty functions give the system a chance for expansion, even though it would mean that a user has forty more functions to learn.

> *“One thing to keep in mind is that what is simple for one person, may be complex to another - and vice versa, of course. Also, the addition of options, features, versatility, and expansion capabilities, naturally breads a certain amount of complexity . And, efforts to keep the operation of anything simple, usually limits the item's options, features, versatility, and expansion capabilities. “*[13]

As seen, complexity offers a high degree of protection at the upper levels of management and development within a system. When the upper levels are secure and enforced by policies, the lower levels can follow. Sometimes there are exceptions to the rule, and this can be seen with the complexity of the Unicode standard for a system's character encoding set.

[13] Loystoys - http://www.loystoys.com/info/system-complexity.html

Unicode exploit

Definition

The general ASCII table allows for 256 characters to be used. It, however, cannot cope with the amount of alphabetical, numerical, and general characters that exist in our multilingual world. Unicode attempts to provide a solution, allowing for the mapping of over 65,000 characters.

Unicode by default comes pre-configured and installed in IIS 4.0 and 5.0, allowing for non English characters to be recognised by web servers.

The Danger

The Unicode allows users to run arbitrary commands onto the web server through the web browser. It is a serious exploit found on web servers and services. It uses a legitimate mechanism and there for in most cases cannot be detected using an Intrusion Detection Systems.

Attack

In modern web servers (Microsoft's IIS as an example), the web daemon by default is run under the administrator account with full privileges. This is a major security issue because if an exploit is found within the http server, then a cracker will not only have enough privileges to bring down the daemon, but also have full read and write access to the system itself. Using the Unicode exploit, a hacker could simply type in the following link on his browser at his end of the system and have access to the system.

http://www.mysite.com/..%c0%af../scripts/windows/system32/cmd.exe?/c+dir+c:\

Although many fixes have been issued out, and Intrusion Detection Systems (IDS) have managed to scan and detect such intrusions, the complexity Unicode is high enough to trick the web server and IDS into thinking that the URL is legitimate, and to make it extremely difficult to rectify the problem once and for all.

And in conclusion,

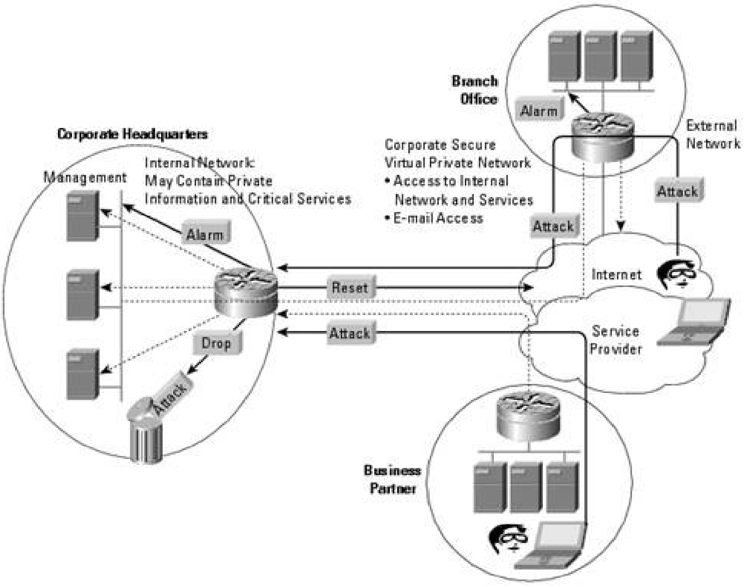
To summarise, system complexity is an ever growing issue in our modern technological world. However, complexity has its advantages in pushing forward the IT industry. System complexity management is necessary to keep a balance between the introduction of new functions and the reduction of errors and complications, as well as to ease the process of doing so.

Intrusion Detection Systems

It is well known that the number of cracking incidents, unauthorised network and system infiltration, and electronic theft is ever growing with each coming year. On one hand hardware and software technologies become more sophisticated and aware of security, while on the other hand, complexity of modern systems allowed for many bugs and exploits that could render a system insecure. It has always been a 'cat & mouse' game between hackers and security administrators.

One set of components in systems that aid in security are Intrusion Detection Systems (IDS), both software and hardware based. An IDS is a security tool designed to detect inappropriate, incorrect, or unauthorised activity. It monitors both incoming and outgoing network traffic, analysing the data packets for any malicious or ones that do not comply with the security policy of the system.

Modern ID Systems are not too difficult to manage. They range from software tools such as ZoneAlarm[14], a software firewall that is user friendly and can be easily managed, and can be as sophisticated as a dedicated hardware firewall. An example of a powerful hardware IDS is Cisco's Secure PIX Firewall[15]. Such systems act as a solid gateway between the inner secure network, and the outer open network.



The main advantage of using an Intrusion Detection System is that the system acts on behalf of a security manager, provided the manager compiles a strict set of security policies. Software and hardware ID Systems nowadays run on fast processors that are capable of analysing vast numbers of packets at extreme speeds, efficiency and accuracy. This supersedes what any human being could do, and if used properly, greatly reduces the number of intrusions.

Although ID Systems are powerful, they do have disadvantages. As intelligent as they are, they base their analysis on security policies set by a security administrator. Such policies need to be frequently revised and updated since new security holes are discovered on a daily basis. If the administrator fails

[14] ZoneAlarm - http://www.zonelabs.com/store/content/home.jsp
[15] Cisco PIX 500 - http://www.cisco.com/en/US/products/hw/vpndevc/ps2030/

to include a certain set of flaws in the policy, then the IDS will fail to protect against such infiltrations. ID Systems have a limit to what they can learn by themselves.

IDS Limitations

- ID Systems cannot analyse encrypted packets.
- There are yet many loopholes that exist, such as the Unicode exploit mentioned, that an IDS cannot prevent.
- A continuous scanning of network traffic reduces the total throughput of the segment on which the IDS is protecting.
- A bottleneck effect can happen since all traffic travelling both ways has to pass through this single point. If for any reason the system fails, then the network could be broken.

One could argue that the ID Systems provide a high degree of protection, but all a hacker needs is just one flaw, one bug, one small misconfiguration in order to penetrate. An IDS can protect against 10,000 types of attacks, but mis-configure one, or find a new hole and it is rendered useless. This reduces its shelf life dramatically. It's major or critical issue to date, is the fact that ID Systems assume that an attacker is always on the outside of a network. However, recent surveys show that over 35% of discovered attacks actually occur internally[16], and this is where an IDS is almost completely vulnerable.

To conclude, an ID System provides substantial security from outside attacks, however, it is not enough to completely base a network's security on it. No network or system is 100% error free, bug free, and secure. A combination of strict policies, ID Systems, packet analysing tools, user rights, and security enhanced network operating systems, is the best decision a security administrator can use. It is always a good idea to combine several technologies together, with each set concentrating on specific areas of security.

[16] Information Security Magazine (Nov 05) - http://informationsecurity.techtarget.com/magItem/0,291266,sid42_gci1137925,00.html

Bibliography

System Complexity

Loystoys [online]. Available: http://www.loystoys.com/info/system-complexity.html

The Internet Services Providers' Association [online]. Available: http://www.ispa-cost.org/c7_pres/

Trac [online]. Available: http://www.edgewall.com/trac/screenshots.html

Veryard Projects [online]. Available: http://www.users.globalnet.co.uk/~rxv/system/complexity.htm

Hacking Tools

Silberschatz Galvin Gagne. (2000). Applied Operating System Concepts (1st ed.). Wiley. ISBN 0471365084.

Ethical Hacking [online]. Available: http://www.research.ibm.com/journal/sj/403/palmer.html
http://neworder.box.sk/tomread.php?newsid=921

TCP/IP Limitations [online]. Available: http://www.cs.ubc.ca/~krasic/publications/krasic-idms2001.pdf

Ethereal [online]. Available: http://www.ethereal.com/

G-Lock’s AATools [online]. Available: http://glocksoft.com/

NMAP [online]. Available: http://insecure.org/nmap/

Slashdot [online]. Available:http://slashdot.org

Kerberos

Coulouris, Dollimore, Kindberg (2nd ed.). Distributed Systems, Concepts and Design. Addison Wesley, ISBN 0201624338

Needham & Schroeder Symmetric Cryptography [online]. Available: http://www.lsv.ens-cachan.fr/spore/nssk.html

Ammended Needham Schroeder Symmetric Key [online]. Available: http://www.lsv.ens-cachan.fr/spore/nssk_amended.html

Denning-Sacco shared key [online]. Available: http://www.lsv.ens-cachan.fr/spore/denningSacco.html

Kerberos V5 [online]. Available: http://www.lsv.ens-cachan.fr/spore/kerberos.html

Citeseer. Kerberos Flaws [online]. Available: http://citeseer.ist.psu.edu/context/13238/0 http://citeseer.ist.psu.edu/lowe96breaking.html

Techrepublic - http://techrepublic.com.com/5100-1035-5366280.html#

Intrusion Detection Systems

Netscreen Technologies. The Disappearance of the Trusted Network (January 2002) [online]. Available: http://www.sss.co.nz/pdfs/netscreen/disappearance_of_trusted_networks.pdf

Top Layer Networks. NFR Security [online]. Available: http://www.toplayer.com/pdf/TLNI_NFR_Joint_solution.pdf

DShield Distributed Intrusion Detection [online]. Available: http://www.dshield.org/

Intrusion Detection Systems Group – UK [online]. Available: http://www.intrusion-detection-system-group.co.uk/

Sans Trusted Computer Security [online]. Available: http://www.sans.org/resources/idfaq/

Window Security. Intrusion Detection Systems (IDS) Part 2 - Classification; methods; techniques [online]. Available: http://www.windowsecurity.com/articles/IDS-Part2-Classification-methods-techniques.html

Word Count
4150 words.

Appendix A

Further Readings

The following topics are articles and books that are useful and interesting to read for further knowledge on the subjects mentioned in the report.

Kerberos

Details of Kerberos Vulnerabilities Leaked (March 17, 2003) [online]. Available: http://www.eweek.com/article2/0,1895,1654988,00.asp

Kerberos Flaws Allow Access to Protected Networks (September 1, 2004) [online]. Available: http://www.eweek.com/article2/0,1895,1641643,00.asp

Kerberos Holes Could Bring Serious Exploits (September 1, 2004) [online]. Available: http://www.eweek.com/article2/0,1895,1641839,00.asp

Kerberos Flaw Leavs Code Vulnerable (October 28, 2002) [online]. Available: http://www.eweek.com/article2/0,1895,1654989,00.asp

Critical Kerberos Flaws Could Open Networks to Attack (July 13, 2005) [online]. Available: http://www.eweek.com/article2/0,1895,1836591,00.asp

Hacker Tools

Chris Null. Google: Net Hacker Tool du Jour (March 4, 2003) [online] Available: http://www.wired.com/news/infostructure/0,1377,57897,00.html

System Complexity Management

Steve Johnson. Everything Bad Is Good For You. Allen Lane Publishing. ISBN: 0713998024

Intrusion Detection Systems

Bob Toxen. Real World Linux Security: Intrusion Prevention, Detection, and Recovery. Prentice Hall PTR. ISBN 0130464562