Interested in learning more about security?

SANS Institute
InfoSec Reading Room

This paper is from the SANS Institute Reading Room site. Reposting is not permitted without express written permission.

Forced Evolution of Security on Redhat Linux Server due to System Compromise

This practical assignment describes my experiences in setting up the office computer network system for a small engineering company in Hong Kong and my experiences of handling the system when it was compromised. I will outline the setup of the original server and list, in highlight the mistakes made in the configuration and the impact of these errors will be reviewed. I will demonstrate an effective review of these mistakes and show how a change in implementation by research and training in systems security, helped to ...

Copyright SANS Institute

Author Retains Full Rights

AD

Forced Evolution of Security on Redhat Linux Server due to System Compromise

Sans GSEC Practical Assignment V 1.4 –
A case Study in Information Security

Author: Alec Wood
30th September 2002

1.0 Abstract

This practical assignment describes my experiences in setting up the office computer network system for a small engineering company in Hong Kong and my experiences of handling the system when it was compromised.

I will outline the setup of the original server and list, in highlight the mistakes made in the configuration and the impact of these errors will be reviewed. I will demonstrate an effective review of these mistakes and show how a change in implementation by research and training in systems security, helped to rectify the situation and implement a more robust and dependable system.

The final sections will review the current status and look at future methods of improving the security of the system and balance these against the time and cost of implementing them. In conclusion I will summarize the mistakes made and outline the lessons learned.

2.0 Before Snapshot - Implementing an office server under time and budget constraints.

We had a rather archaic system of dialup to the Internet via a shared modem (Modem Share) on a hybrid Windows 95 and Windows 3.1 peer-to-peer network. Sharing was determined by the holder of the 'token' who was then the only person 'authorized' to dial up and perform their email transactions. When they had completed their Internet connections they passed the token on to the next person in the queue. This set up was totally inefficient in the face of the increasing email traffic and web usage that hit the business landscape in the late 90's.

The choice of operating system was most clearly defined by cost of purchase and with HK$ 15,000- HK$18,000 for a Windows NT Server ten user license or HK$ 300 for one of the many Linux distributions. The choice was clear.

Redhat Linux was chosen for the simple reason of economics and the fact that updates and utilities were free from the Internet. The RPM utility made obtaining and installing

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

upgrades and utilities straightforward. Redhat 5.2 was duly purchased and the system installed with an 'out of the box' default configuration.

For Local file sharing and printing, SAMBA was chosen and the default supplied smb.conf file was edited appropriately.

Principle mistakes to note are the modification of the default Directory Create and File Create Modes to 0777 and not using encrypted passwords (Windows 95 doesn't have encrypted passwords as a default) and not limiting the access from the local network or private interface - 192.168.0.*/24 in the smb.conf file. Note:- Microsoft has a registry setting for making Win 98 transmit passwords in clear text, which I had used on those workstations running it. In hindsight, this was a mistake, as it gave no real layered defense.

To quote from the Microsoft Knowledge Base article: *"WARNING: If you enable plain text password use in Windows 98, all passwords are sent on the network in an unencrypted format."*[1]

Broadband connectivity was not required and indeed not available in the building where our office was situated. Thus a standard 56k dial up connection was used which automatically dialed up whenever a packet destined outside of the local network was sent to the server. Eric Shenk's great little utility called DIALD[2].

The sole security feature that was added to the default install was to put the modem on an electrical time switch so that the machine was disconnected from the Internet outside office hours.

The mistake here was relying on security through obscurity in believing that a dial up account with a changing IP address would provide a satisfactory level of protection.

The procedures that were followed were the Masquerading HOWTO (Network Address Translation based on IPFWADM), SAMBA HOWTO, DIALD HOWTO, PPPD Man-pages, and the CHAT man-pages. Some of these instructions included some basic security features, for example adding the 'ALL:ALL' entry in the hosts.deny file.

Note: This is now depreciated and the preferred 'deny all' method is to add the entry ALL:ALL: DENY in the hosts.allow file.

These were followed purely because they were part of the implementation instructions and not because I appreciated the security need for them.

[1]Microsoft Knowledge Base Article Q187228 "Unable to connect to a SAMBA server with Windows 98", Published June 8th 1998.
URL: http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q187228& (28th Sept 2002)

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

The DIALD author who provided a sample firewall script for masquerading that I used admitted, *"If you need advanced security considerations, it may be a little limited."* [2]

Amongst the services left running on the server were:

SMBD, NMBD, X, the RPC processes, BIND (named) YPSERVE, DHCPD, TCPD, FTPD, Sendmail, telnetd and Apache amongst others (RSHD etc….

There was no process auditing and removal of unneeded and redundant processes, especially dangerous was relying on the default installation of BIND, Sendmail and Apache without applying latest patches.

DHCP was not used for the reason that if the server failed and I was out of the office then nothing would work. The windows workstations in the office were set up with NETBEUI protocol as well as TCP/IP (with NETBEUI set as the default protocol) so that any two staff could turn on their machines and share files even if the server was down.

In hindsight, a proper review of what the server was required to do for the business would have helped decide the services to implement and would have kick started a company security policy. Instead, it was an unplanned implementation concentrating on functionality rather than security.

In addition, our head office in the UK produced a CDROM of the company's Internal Intranet Website that was forwarded to us every month. No problem, we were running Apache web server and by putting a symbolic link on the public shared directory into the HTTPD root directory it would enable our office administrator to copy the HTML CD direct to the web server.

Here there was no evaluation of the content (value) of the Intranet web pages nor was there any kind of risk analysis of putting it on a web server that could potentially be compromised. An ad-hoc method of adding services with no real segregation of functions in the directory hierarchy left the system wide open to a root compromise. With hindsight the gaping errors in these assumptions became clear.

Total time to set up the system was two weeks most of which were spent on reading the MAN pages for the various utilities necessary to get the core required services working (internet connection and sharing with NAT, IPFWADM, DIALD, PPPD, CHAT and SAMBA).

The original task had been accomplished, we could all now browse the Internet at the same time; we didn't have to wait for our turn. Linux seemed really stable (the final uptime for the system before I pulled the plug was 235 days). In addition it "felt" secure

[2] Seco, Andrés 'The DIALD HOWTO', v1.13 April 17th 2000.
URL: http://www.tldp.org/HOWTO/Diald-HOWTO.html - toc3 , (28th Sept 2002)

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

because no one wanted to touch it (the text based "Login:" prompt is not welcoming to uninformed users).

Every thing seemed fine and I felt I could leave that now and get on with my real work.

3.0 During Snapshot – "I can't get my email and the printer is very slow".

After about 3 to 4 months, I noticed that occasionally the server would be a little slow. By this time, I had another machine (running Mandrake 6.0) that was used to test X server connections over the office network and I "administered" the server by using StarNet's excellent X-Win32 software[3] to connect to the XDM on the server from my windows laptop.

The X session should really have been connected using SSH, without this simple security feature the X session was sent unencrypted over our local network and carried the usernames and passwords of the root account and several others. To quote the Xwin32 product description: - "*To remain effective as a remote access tool, today's PC X server must offer a Secure Shell (SSH) connect option.*" [3]

On December 16th 2000, I was browsing the /etc files (which was my ad hoc security audit) and noticed that the hosts.deny file had been altered and the ALL:ALL entry had been removed. I checked the SAMBA access logs and found a total of 35 different machine names having browsed our shared directories. This would account for the slow response of the server.

Further examination of the system logs showed that monitoring of logins had stopped sometime back in July. Immediately I changed the root password and my own login password with our ISP and within 5 minutes received an anonymous email with the DOS PIF file "is_linux_good_enough!.txt.pif". Which was a virus as defined by the Symantec Virus Encyclopedia: - "*W95.MTX has a virus component and a worm component. It propagates by email.*"[4]

My immediate conclusion was that we had been compromised and were being monitored remotely. The coincidence of this email arriving within minutes of the passwords being changed was too much to be ignored.

Further investigation found that the modem timer switch had been bypassed by one of our sales staff whilst they were working out of office hours. This weekend work corresponded with the cessation of some of the logging facilities in July. It was then

[3] Starnet Communications product information
URL: http://www.starnet.com/products/ , (26th Sept 2002)

[4] Symantec Security Response Virus Encyclopaedia. 17th August 2000.
URL:http://securityresponse.symantec.com/avcenter/venc/data/w95.mtx.html. (23rd Sept 2002)

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

necessary to tell the office members that the server had been compromised (hacked) and that they should change their passwords on their mail accounts and on any other account that had been accessed from the office especially any commercial logins (e.g. Amazon) and online banking facilities.

Here I realize that a regular security review and talk to the users would have reinforced the fact that (little use that it was) the modem being on a timer switch was actually part of the security system. User education would have helped them to leave it in the manner they found it. One of the drawbacks in using DIALD for automated dial up is that the login ID and password are kept in clear text in a CHAT script file, which was easily read by any intruder (world readable). In the current configuration of the server this script file has more restrictive access rights.

An assessment of the compromise did not show up anything but then any number of the standard Linux commands could have been trojan'ed to hide tracks. For example the T0rn root kit[5].

The entire system needed to be treated as suspect. Rather than attempt to analyze the break in and define the extent of the damage the decision was made to remove the Data drive (which housed the shared files handled by SAMBA and represented the bulk of users work), reformat the system drive and start again. It would have been a challenging exercise to review the system, work out what happened, trace the compromises and try to track those responsible. In reality it would never happen and economic and commercial considerations forced the server to be up and running again in the shortest possible time.

Thus the plug was pulled on the system and all users were told to use their modems for timeshared access to the Internet as before.

In hindsight I could have been more prepared for a security incident and burned a toolkit onto a CDROM that would have statically linked utilities for analyzing a compromised machine (e.g. "*The Coroner's Toolkit (TCT)*[6] *is a collection of tools that are either oriented towards gathering or analyzing forensic data on a Unix system*")

In effect, not having an incident response plan meant that there was no reasoned response to the compromise and 'worrying' about the action to take potentially wasted several days. Furthermore there was a trade off in time to analyze the compromise and the business requirement was to get the system up and running again. This is always a difficult decision, but at the very least steps must be taken to prevent a reoccurrence. To quote the authors of The Coroners Toolkit: - *"A tremendous amount of time can be*

[5] Miller, Toby 'Analysis of the T0rn rootkit' GIAC Special Notice 1999-2000
URL http://www.sans.org/y2k/t0rn.htm (25th Sept 2002)

[6] The Coroners Toolkit Venema, Wietse & Farmer, Dan August 1999
URL: http://www.fish.com/tct). (26th Sept 2002)

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

consumed taking care of the problem at hand, but as a rule of thumb if you don't spend at least a day or two you're probably short changing yourself and your system"[7]

I began research into the methodology to be used for the next incarnation of the server after the Christmas holidays. Faced with the humbling thought that we could have had our internal company intranet revealed to the world and our passwords for the dialup accounts compromised, I needed to make sure it would be right the next time.

For me the main lesson learned during this period was that I needed to know if an incident has happened. I had no sign nor warning that anything was amiss and the maxim *"Prevention is ideal, but detection is a must*"[8] quickly came to my attention while doing research on the web for suitable tools.

In addition one of the very real dangers of leaving the default configuration of a system unchanged is that very often you don't know what that configuration is. For example, while reviewing the log files I could not be sure what should have been logged by default. The whole experience of being compromised and having to inform management and users that their system and data may have leaked into the wild was a very humiliating one. It was at this point that I resolved to gain the knowledge and experience to prevent a reoccurrence.

I began the New Year by carefully defining the functions that the server would need to perform within the office environment. This functional specification was narrowed down to the following:

1) Network File sharing (static configuration, no DHCP)	SAMBA
2) Network Printing	SAMBA
3) Dial on demand to our ISP	DIALD
4) Sharing Internet connection	NAT / IPMasq (IPchains)

In particular the following services and features would be specifically disabled on the basis that they provide services that are not part of the functional specification above.

1) Web server (Apache)
2) NIS services
3) Named (No DNS at all, not even caching in forward only mode)
4) Telnet and 'R' protocols
5) FTP server.
6) X Window functionality – and all associated network services (e.g. X font server).
7) No user would be allowed to log in to the server and execute commands.

[7] Venema, Wietse & Farmer, Dan 'A bit of help if you've just been broken into' August 1999
URL: http://www.fish.com/tct/help-when-broken-into (26th Sept 2002)

[8] DeFrance, Fred, 'A Case for Centralised Logging' December 7th, 2001
URL: http://ebuzzsaw.com/whitePapers/Case_for_Centralize_Logging.htm (21st September 2002)

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

8) NFS server and client services.

A note about DNS services; I decided against a local caching DNS facility as it is a regular source of vulnerabilities and compromises, and the management overhead of maintaining this set up correctly would be more than I could afford with my workload. The main drawback was that the machine it self cannot resolve URL's on the Internet and so downloading updates and patches would have to be done on a workstation. I also felt that this would slow down an attacker, as they would have to enable some form of DNS to download any utilities or launch further attacks.

The main risks identified with this server implementation were as follows:

1) Allowing internal services to be made available to the public (Internet) interface.
2) Up to date Anti Virus software not installed on all workstations.
3) Laptops that have no personal firewall installed being used out of the office.

Additionally the following security policies and practices were going to be put into effect within the office network environment. In effect this list is a combined Program, Issue-Specific and System Specific policy. As there was only one server and one administrator with four other users it was not felt that separate documents needed to be generated.

1) **No plaintext passwords over the local network or over the Internet.** As we had a central hub instead of a switch it would be trivial for one compromised machine to harvest passwords from the rest. This meant every machine had to be using at least Windows 98, and all install media for Windows 95 were tracked down and locked up to prevent users from reinstalling. All Windows 98 workstations had the relevant registry entry modified. Then the machines were rebooted, and the registry entry removed. I found out the hard way that Win98 machines will still send unencrypted passwords if the registry entry is simply removed. Operation was checked with SAMBA running the 'Require Encrypted Passwords' option.

2) **Workstations to have individual firewalls / IDS and up to date AV software.** Blackice Defender was chosen for the office machines but some users had laptops and roamed out of the country, reconfigured their machines, reloaded the OS etc… So they also ended up using ZoneAlarm, as it was free to download and did not require registration keys. Anti-Virus software depended on the machines as some users had Norton Anti Virus bundled and some had PC-Cillin.

3) **Passwords to be a minimum mixture of letters and numbers and at least 8 characters long.** There are more stringent rules that can be followed for passwords but the users would resist such changes. I would manually reset passwords every 3 months. (Note: I decided not to automate the password expiry as this could happen while I was traveling and the whole office could grind to a halt in my absence. Remember these users were not used to 'good' passwords and were used to dialing up

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

to our ISP on a timeshare basis. If left to themselves they would choose their surnames spelt backwards.

4) **The server and workstations to have a default set of firewall rules specific to the office network.** Rather than relying on the default installation of the different firewall software, a policy of specifically allowing and disallowing ports and addresses was implemented on every machine. I had decided that one of the biggest advantages I had against an intruder was the fact that I knew the network topology and they didn't. Only connections from the local network were allowed on ports 137,138 and 139. No other connections were allowed and the steps outlined in "Top Ten Blocking recommendations for Ipchains"[9] rules were manually written into Black Ice Defender and copied to each workstation. Very importantly these rules were also used to block specific ports and addresses in the FORWARD chain of IPCHAINS on the server. This provided an additional level of egress filtering at the server.

5) **The server and workstations to be updated with the latest security patches monthly.** At least every month for the server and 3 months for the workstations. Once again as I traveled at least 1 week in every month and sometimes more it was not realistic to schedule goals that could not be achieved. As some patches can cause a system to fail when applied they were not applied to the server unless they were relevant to the services being used and a backup had been taken of the system.

6) **Review of system logs and running Tripwire[10] report at least every week.** As previously stated this could be automated however it was decided that the Tripwire reporting would be done manually by myself as an automated process could fill up the /usr partition (or wherever Tripwire reports are chosen to be saved) if I was away traveling for an extended period of time. Additionally I regularly check the free disk space (DF command) and search for large files (greater than Two Megabytes) to see what the system and the users are doing.

7) **Allowed services from workstations to the Internet would be: DNS, POP, SMTP, FTP and HTTP only.** If necessary all other ports or services may be closed down at any time and are not explicitly guaranteed as a user right. In effect the NAT modules for other services (apart from FTP) would not be installed.

8) **The server will not accept (would silently drop) all connections attempted to the external interface and will offer no open ports to the Internet.** That is ppp0 or slip0 (as configured by DIALD at this time) or could be eth1 if ADSL was added to our network. Using the free port scanners on the Internet I regularly check this still true. Portsentry[11] is also deployed with the default response to drop the offending IP address

[9] Tiedemann, Paul, August 8th 2000.
URL: http://rr.sans.org/firewall/blocking_ipchains.php (17th Sept 2002)

[10] Tripwire Inc, Tripwire Open source Project homepage
URL: http://www.tripwire.org/ (21st Sept 2002)

[11] Psionic Technologies Inc., Portsentry Product Info 1.1 & 2.0b1 beta

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

with a "IPCHAINS -I input -s $TARGET$ -j DENY" rule. Note that DIALD has 'if-up' and 'if-down' scripts that can be used to automate the implementation of firewall rules. I have written these to run the 'bastille-firewall' script and enable forwarding and to flush the IPCHAINS rules respectively.

9) **The server will only accept Windows file and printer sharing connections (ports 137,138 & 139) and SSH (port 22) from the internal network (eth0 192.168.0.X) only.** This limited the open services and allowed for secure remote administration from my desktop all within the private IP address range we used in the office. All other ports and services are denied.

These basic policies represented a start point for the system. They defined the services offered to the business and then describe the manner in which these will be provided. Without such a defined starting point the system would have no real focus nor direction and services offered to users would have no particular priority or precedence. In fact without such a defined policy, there is no differentiation between legitimate users (who can inherently use the "Allowed Services") and intruders for who most services are denied.

4.0 After Snapshot – Setting up a secure server for a small office network.

The latest version of Redhat server was purchased (Version 7.1) and installation began. In the following manner:

1) Prepare the hardware for installation. Define the drives being used for /dev/hda root partition (/ which was further partitioned), /dev/hdb ('/home'), /dev/hdc ('/public') and the CDROM drive. Ensure that the network hardware was correctly installed and make sure that the partition mapping was worked out before commencing the install.

2) Install the custom configuration instead of the default Server configuration for the Redhat install process. Selecting appropriate utilities to be loaded and specifically not loading other utilities as mentioned above.

3) Remove unused users and groups from the default installation, e.g. ftp, slip, sys, uucp, nuucp, listen etc…. Leaving just one root login and the required accounts for needed services. No user accounts were added at this stage apart from my normal (non root) account. This was all accomplished by using linuxconf as root from the command prompt.

4) Set up the network interfaces correctly. Configure and set up SAMBA for remote printing and file sharing using custom shares. This was tested with my normal login account only. This step proved to be a little problematic with the removal of a DNS

URL: http://www.psionic.com/products/portsentry.html (18th Sept 2002)

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

service on the server. SAMBA requires a defined method of resolving all NETBIOS names to IP addresses or it will not function and any references to DNS were removed in the SMABA search order. It is possible to enable the WINS component of SAMBA and this does function adequately. As a backup to WINS, each workstation had a locally defined LMHOSTS file that was centrally copied to their Windows directory manually (listing the static IP addresses of each of the workstations). This is cumbersome for large networks but we have a small user base and a low staff turnover rate so it was deemed acceptable.

5) Baseline the system by setting up and running Tripwire with the final policy and database files stored on a write protected floppy.

6) Set up DIALD to auto connect to the Internet and check local connectivity only by using console access to ping our ISP mail servers IP address directly (not by URL as no local DNS was setup as /etc/resolve.conf was empty and hosts.conf had 'Order hosts' only and the named process was halted on all run levels). In addition the named binaries were moved from /usr/sbin to /root/named.

7) Configure Network Address Translation with IPCHAINS and test for shared connectivity by pinging direct IP addresses from the workstations. By never allowing the machine to be connected to the Internet for more than 10-20 minutes, I reasoned this was enough time to test ping from a client machine to the Internet but hopefully not long enough to be scanned and attacked. In the words of the Honeynet Project Team members *"…I connected the system to the network. Within 15 minutes my system had been probed identified and exploited."*[12]

8) Re-run Tripwire and compare to the backed up copy of the database and manually check that all the changed entries reflect the files installed for the required service since the database was generated.

9) Secure the system with Bastille Linux[13]. Then perform other tests on the system such as searching for world writeable files and changing the mount settings for the shared 'data' drives through SAMBA (/home on /dev/hdb and /public on /dev/hdc) to be 'NOSUID' and 'NOEXEC'. The Bastille Linux hardening kit is a great utility for bringing a system up to a defined security standard but it does need manual adjustment to make sure that the exact configuration of your system is protected in the best possible manner.

In addition the firewall rules were amended to have the specific rules detailed in "Top Ten Blocking Recommendations Using IPchains'. In particular ports 22 (SSH), 514 (Syslogd), 515 (LPD), 137, 138, 139 (SMB and NETBIOS) were specifically blocked on

[12] The Honeynet Project, Know your Enemy. Reading: Adison Wesley Longman, Inc. October 2001, page 3.

[13] Lasser, Jon; Beale, Jay et al, Bastille Linux Homepage URL: http://www.bastille-linux.org/. (21st Sept 2002)

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

the external interface connecting to the Internet (ppp0) These services are specifically served internally on eth0 (except LPD which is required for printing by SAMBA) to the client machines in the office.

10) Re-test the required services listed above and ensure logs are being generated correctly.

11) Using a workstation machine only, download the latest updates and patches for the OS (Red hat 7.1). A working CDROM of all tested and applied patches (RPM's and SRPM's) is kept so that the system can be recovered if it is ever compromised and the backups fail. In addition a snapshot of the system drive (/dev/hda) was taken with the **dd**[14] command: - "dd if=/dev/hda of=/public/backups/system/snapshot-yy-mm-dd".

12) Put machine into production on the internal network and make backups of the system drive that are then transferred to a designated windows client machine, as they are too big to be burnt onto CD ROM.

13) Set users passwords (with Shadow passwords enabled) using 8 characters with a mixture of letters and numbers. The root password was set to 11 characters long using a memory technique based on mnemonics derived from the lyrics to a favorite song. The login shell was disabled in /etc/passwd to prevent any user gaining access

14) As a final check, the machine was scored against The Center for Internet Security's Linux Benchmark v1.0.0[15]. This is particularly useful as the CISscan command (installed by RPM) generates a score out of ten and lists the negative and positive points of the system. By comparing the recommendations of the two tools, (Bastille Linux and the CISscan) it is possible to have a very clear indication of the potential areas of weakness in the system. For example, CISscan found several SUID and SGID programs that Bastille Linux (Version 1.3) did not pick up in /usr/lib/amanda and recommended tightening the permissions for /etc/crontab.

The server was put into service and users were told their new passwords for logging onto the network. The only teething problems were on the shared directories via SAMBA where users could not delete other users files. Whilst this was the recommended setup of shared directories (sticky bit set), users of Windows 9X machines are not used to file and directory ownership concepts and they complained that they could not delete certain files.

[14] BackupCentral, Example of using dd as a backup
URL: http://www.backupcentral.com/dd-backup.html (25th Sept 2002)

[15] The Center For Internet Security, CIS Level 1 Benchmark and Scoring Tool for Linux, April 2002
URL: http://www.cisecurity.org/ (18th Sept 2002)

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

Rather than allowing global write access to files the users were placed in a group called 'office' and default create modes for files in SAMBA were set to 0660, and directories set to 0770 (so directories show up in a file listing).

The server was audited by trying one of the online scans on the Internet (e.g.: Steve Gibsons' 'Shields Up'[16]). I also intend to ask a trusted colleague to run a more effective scan against the server to more effectively check for a wider range of open ports.

Future Improvements

The biggest single limitation in securing the server more comprehensively is the time available in which to perform such tasks. If more resources become available in the future then implementing the following actions would be desirable:

1) **Segregation of functions**. The most logical step to make is to add another machine specifically to be used for Internet connection. This would mean that it could be locked down very comprehensively or I could use one of the many "Linux Firewall on a Floppy" packages (For example FloppyFW or the Linux Router Project) and have an old 486 machine boot solely from a floppy or CDROM. Another machine would then provide the internal services such as file and printer sharing.

2) **Additional egress filtering**. I feel that the current egress filtering from the server is adequate but not the masqueraded connections from the workstations. In the near future I intend to review the outgoing connections and generate a list of IPCHAIN rules to further block outgoing packets.

3) **Remote Logging**. Use a separate machine for remote logging and use one of the utilities for automated log review such as Swatch. This would require the implementation of NTP synchronization between machines.

4) **Add a dedicated backup machine**. The sole use of which is to hold backup data from the others. As very large hard disks are now quite cheap this could easily be achieved and the backup process automated. Current backup procedures are made to other workstations or to writeable media such as CDR or Panasonic PD (Power Drive).

5) **Set up SAMBA as a Primary Domain Controller**. Then implement automated logon scripts for configuring users machines in a defined manner. This could help immensely by synchronizing the date and time settings on all workstations, provide automatic update to AV software, and map the necessary shared drives as the same drive letters on all machines.

[16] Gibson, Steve Gibson Research Corporation. Shields Up
URL: http://grc.com/default.htm (21st Sept 2002)

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

In addition to adding and configuring more hardware there is more that can be achieved in terms of planning for an incident. I hope to develop a policy to review the value of data that users store on the server and estimate the cost of a compromise.

I also feel that my current auditing of the server is inadequate and I intend to run regular scans for open ports and a more structured Tripwire check and reporting process.

Finally I feel the largest area of weakness is from the users who roam with their laptops abroad and connect to local dialup numbers in foreign countries. I believe that this is a significant risk of compromise and my immediate plans are to implement a straightforward review of the workstations and users backup regimes and enforce a stricter password policy.

Summary and Conclusion

In setting up a server to connect to the Internet it is imperative that a methodical and structured approach is undertaken. There are many aspects of systems security and almost all of them can be made easier by using commonly available tools and utilities. None of these will completely secure a machine but they will ensure that an adequate baseline is achieved in a broad range of areas. This prevents one area of neglect undermining the whole system. For example, the storage of our dial up account and passwords in a world readable file by an early version of DIALD probably allowed the first attacker that gained entry to our system to get into our ISP's as a legitimate user.

Extensive logging and review is also important. This should be the first indication of an event that an administrator will receive (as opposed to retroactively working out why something doesn't work). To read and review them in a timely manner before an intruder doctors or deletes them is absolutely critical and allows an administrator to respond to an incident early on before it escalates to a more serious compromise. Reviewing log files can be a time consuming task so some form of automated log monitoring utility is highly recommended. (E.g. **swatch**[17] or **Logsentry**[18])

When a security incident occurs preparation can be everything, including saving your reputation. Having been responsible for a system that was compromised and not having a clear direction of what to do was a humbling experience. I now have regular personal and system backup scripts and securely stored recovery CDR's with system snapshots (taken with **dd**) and analysis tools (**The Coroners Toolkit**) in preparation for any future compromise. I verify these backups and tools on other systems (my Linux network server at home) to ensure that I know what they do should I ever need to use them.

---

[17] Atkins, Todd. Swatch Homepage 8th Nov 2001
URL: http://www.oit.ucsb.edu/~eta/swatch/ (21st Sept 2002)

[18] Psionic Technologies Inc. Logsentry Product Overview.
URL: http://www.psionic.com/products/logsentry.html (21st Sept 2002)

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

References

[1]Microsoft Knowledge Base Article Q187228 "Unable to connect to a SAMBA server with Windows 98", Published June 8th 1998.
URL: http://support.microsoft.com/default.aspx?scid=KB;EN-US;Q187228& (28th Sept 2002)

[2]Seco, Andrés 'The DIALD HOWTO', v1.13 April 17th 2000.
URL: http://www.tldp.org/HOWTO/Diald-HOWTO.html - toc3 , (28th Sept 2002)

[3] Starnet Communications product information
URL: http://www.starnet.com/products/ , (26th Sept 2002)

[4] Symantec Security Response Virus Encyclopaedia. 17th August 2000.
URL:http://securityresponse.symantec.com/avcenter/venc/data/w95.mtx.html. (23rd Sept 2002)

[5] Miller, Toby 'Analysis of the T0rn rootkit' GIAC Special Notice 1999-2000
URL http://www.sans.org/y2k/t0rn.htm (25th Sept 2002)

[6] The Coroners Toolkit Venema, Wietse & Farmer, Dan August 1999
URL: http://www.fish.com/tct). (26th Sept 2002)

[7] Venema, Wietse & Farmer, Dan 'A bit of help if you've just been broken into' August 1999
URL: http://www.fish.com/tct/help-when-broken-into (26th Sept 2002)

[8] DeFrance, Fred, 'A Case for Centralised Logging' December 7th, 2001
URL: http://ebuzzsaw.com/whitePapers/Case_for_Centralize_Logging.htm (21st September 2002)

[9] Tiedemann, Paul, August 8th 2000.
URL: http://rr.sans.org/firewall/blocking_ipchains.php (17th Sept 2002)

[10] Tripwire Inc, Tripwire Open source Project homepage
URL: http://www.tripwire.org/ (21st Sept 2002)

[11] Psionic Technologies Inc., Portsentry Product Info 1.1 & 2.0b1 beta
URL: http://www.psionic.com/products/portsentry.html (18th Sept 2002)

[12] The Honeynet Project, Know your Enemy. Reading:
Adison Wesley Longman, Inc. October 2001, page 3.

[13] Lasser, Jon; Beale, Jay et al, Bastille Linux Homepage
URL: http://www.bastille-linux.org/. (21st Sept 2002)

[14] BackupCentral, Example of using dd as a backup
URL: http://www.backupcentral.com/dd-backup.html (25th Sept 2002)

[15] The Center For Internet Security, CIS Level 1 Benchmark and Scoring Tool for Linux, April 2002
URL: http://www.cisecurity.org/ (18th Sept 2002)

[16] Gibson, Steve Gibson Research Corporation. Shields Up
URL: http://grc.com/default.htm (21st Sept 2002)

[17] Atkins, Todd. Swatch Homepage 8th Nov 2001
URL: http://www.oit.ucsb.edu/~eta/swatch/ (21st Sept 2002)

[18] Psionic Technologies Inc. Logsentry Product Overview.
URL: http://www.psionic.com/products/logsentry.html (21st Sept 2002)

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

Last Updated: May 25th, 2013

Upcoming SANS Training

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

SANS Malaysia @ MCMC 2013	Cyberjaya, MY	Jun 03, 2013 - Jun 08, 2013	Live Event
SANS Pen Test Berlin 2013	Berlin, DE	Jun 03, 2013 - Jun 08, 2013	Live Event
Industial Control Systems Security Training - Houston	Houston, TXUS	Jun 10, 2013 - Jun 15, 2013	Live Event
Security Impact of IPv6 Summit 2013	Washington, DCUS	Jun 14, 2013 - Jun 16, 2013	Live Event
SANSFIRE 2013	Washington, DCUS	Jun 14, 2013 - Jun 22, 2013	Live Event
SANS Canberra 2013	Canberra, AU	Jul 01, 2013 - Jul 13, 2013	Live Event
Digital Forensics & Incident Response Summit 2013	Austin, TXUS	Jul 09, 2013 - Jul 16, 2013	Live Event
SANS London Summer 2013	London, GB	Jul 09, 2013 - Jul 16, 2013	Live Event
SANS Rocky Mountain 2013	Denver, COUS	Jul 14, 2013 - Jul 20, 2013	Live Event
SANS Mumbai 2013	Mumbai, IN	Jul 22, 2013 - Jul 27, 2013	Live Event
SEC528 SANS Training Program for the CompTIA® New Advanced Security Practitioner (CASP) Certification	Washington, DCUS	Jul 22, 2013 - Jul 26, 2013	Live Event
SANS San Francisco 2013	San Francisco, CAUS	Jul 29, 2013 - Aug 03, 2013	Live Event
SANS SEC 560: Network Penetration Testing @ Bangalore 2013	Bangalore, IN	Aug 05, 2013 - Aug 10, 2013	Live Event
SANS Boston 2013	Boston, MAUS	Aug 05, 2013 - Aug 10, 2013	Live Event
Critical Security Controls Summit	Washington, DCUS	Aug 12, 2013 - Aug 18, 2013	Live Event
Industrial Control Systems Security Training - DC	Washington, DCUS	Aug 12, 2013 - Aug 16, 2013	Live Event
SANS Thailand 2013	Bangkok, TH	Aug 19, 2013 - Aug 31, 2013	Live Event
SANS Virginia Beach 2013	Virginia Beach, VAUS	Aug 19, 2013 - Aug 30, 2013	Live Event
Mobile Device Security Summit 2013	OnlineCAUS	May 30, 2013 - Jun 06, 2013	Live Event
SANS OnDemand	Books & MP3s OnlyUS	Anytime	Self Paced