Global Information Assurance Certification Paper

Introduction

Let me begin by stating up front, 'I am not a security specialist'. I am an administrator looking after a small site who has an interest in obtaining a reasonable level of proficiency in security. Why the interest? Well there are two reasons. The first is that in order for the Internet to fulfill its utopian promise people need to feel confident about performing their daily business using the medium. That means that all of us in the IT business are going to have to play our part in improving the day to day security, including administrators of small sites. Although it is highly unlikely we would be targeted directly we could quite easily fall to 'cyber vandals' or groups wishing to use us as a staging post in attacks against other sites. The second reason is more personal. Some years ago I was placed in the unenviable position of being part of a team that successfully laid a trap (I guess you could say a very simplified form of a honeypot) for someone abusing their position in order to access information without authorisation. The big sting was that the perpetrator was not only a friend, but also someone that I had looked up to. Believe me an experience like that leaves a lasting impression. I learned a couple of very important lessons, firstly a site is far more exposed to someone operating on the inside, who knows the site, and secondly people who are exposed to stress, or are suffering personal problems can act totally out of character. If we can keep the site secure then not only can we protect the Companies we work for from the bad guys, but also remove temptation from anyone who may temporarily suffer a character aberration.

Three years ago, I started at my current place of work. I guess it would be typical of many small businesses running a few NT servers and around 50 NT workstations on a fully switched network. Part of the operation is involved with trading on a 7 x 24 basis. The IT department consists of one developer and myself. We have some disadvantages when compared to larger sites in the area of security; for example, my role consists of anything from changing a faulty workstation patch lead through to writing a system utility and spending too much time on one job per day is difficult. I would love to devote a lot more time to security but unfortunately it is just one task amongst many. However there are many advantages too, for example I not only know everyone that works on the site personally, but I speak to most users each day. I am aware intimately of what is installed and running at the site.

Being necessarily brief, this document shows some of the steps we have taken (albeit gradually) to tighten up access to the site.

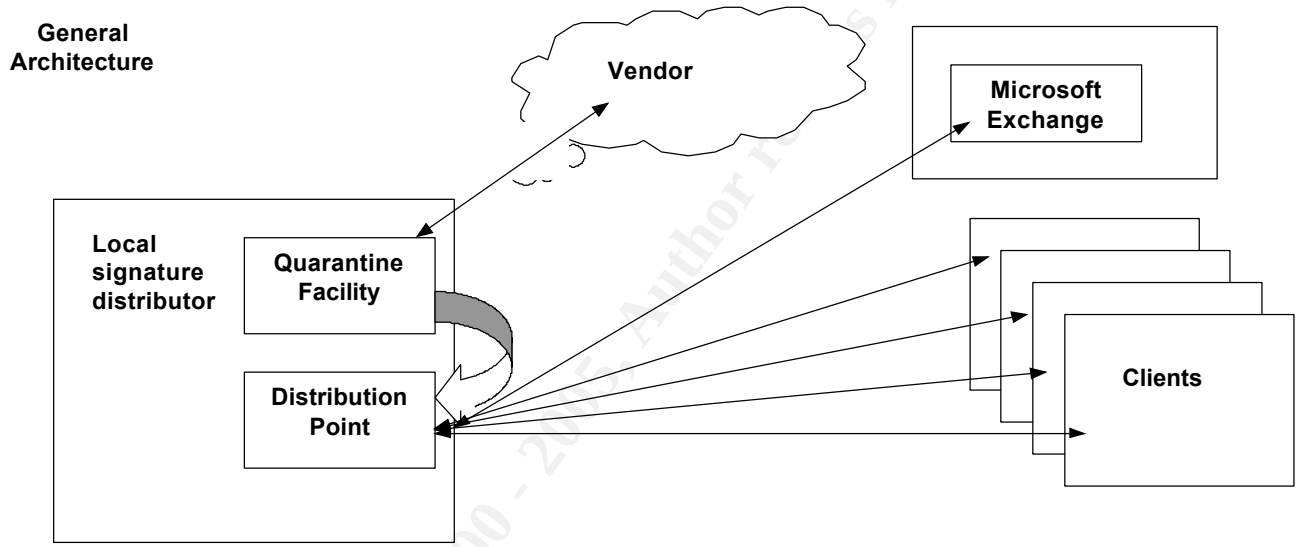
The Beginning

The Company had been formed from a large Corporation that did not have any form of security culture, indeed whenever I mentioned 'security' the general reaction left me with no illusions of how difficult the task was going to be. I looked at what (I felt) needed to be done and started to prioritise the tasks, this work had to slot in amongst all the rest of the not insubstantial work involved with starting up a new concern. There was a requirement for Internet access, but I had neither the time nor the necessary skill set to ensure that a secure service could be set up in a timely fashion. I arranged to

outsource the management of the Routers, Firewall, Web and Proxy services to specialists in this area, and these services, with the exception of the router, were hosted offsite. This left me with a number of tasks that I decided needed immediate attention. They included, but not limited to, Virus protection, tightening server access and upgrading the level of passwords. Physical security was already adequate for our needs

The AntiVirus solution

One thing that is very important in any company that is short on people resource is a product that offers the capability of being automated. Here is the solution we eventually set up to help protect against Viruses.



The Local Signature Distributor periodically polls the vendor for updated files (hourly). If an updated file is detected it is downloaded and dropped into a quarantine area. This is so that the files can be verified before they are distributed to the rest of the site. Once the predetermined quarantine period has expired, the files are automatically moved to the distribution point. The clients' poll the Distributor once per hour looking to see if there are updated files available. This does not put a large load onto the network (we are running a small number of 100 Mb workstations on a fully switched 100 Mb network) and ensures that the clients have the latest signature files available to them in a timely fashion. If the client discovers a newer version on the distributor then it closes down the virus services, downloads the new files and then restarts the services. Once the new files are installed, a full scan of the disk is performed. The service is only closed for a matter of seconds.

Protection on the client consists of two different mechanisms. A process runs which scans file access in real time, that is as a file is opened/saved by the client either via the network or local drives. This process is configurable and has two scan options, a fast scan and a more in-depth scan. The fast scan is a compromise between speed and protection and checks the header

and tail of the file. It is reputed to offer around a 95% protection, although I suspect this figure could vary as viruses become more sophisticated. The secure scan option verifies the whole file contents, but of course will consume much more resources on the client. Selecting which file extensions to test may modify the two different scan options further. We have erred on the side of safety by taking the option of the secure scan and testing all file extensions. The clients are also subjected to a quarantine facility. If a virus is detected on a client the administrator and user is informed and the file is effectively quarantined for a predetermined period allowing corrective action to be taken. We do, however, perform more testing on the clients each evening. A job is scheduled to run on each client each evening to perform a full secure scan of all files. There are a number of actions to set up for how to deal with an infected file, from notification through to deletion. My preference is for a cure to be attempted first, i.e. remove the infected part of the file, with a rename of the file and move to a quarantine area if the cure fails. The administrator is notified of any viruses that are discovered on the site via e-mail and a pop up window.
In case of accidental power off at night, all clients are configured to run the same type of secure scan on boot up. We prefer our client workstations to be left up at night (logged off) as we schedule defragmentation, virus scans etc. as much as possible out of office hours. At the weekend, each client runs a heuristical scan. A heuristical scan attempts to analyse a file's behaviour to see if it portrays characteristics that are consistent with that of a virus. In this mode, the program is set to notify only, as it is highly possible that mistakes will be made in this mode. In reality I have yet to see this scan bring up a failure in over three years but as it does not cause any user discomfort, running during an evening at the weekend, we will continue to run it.

There was an issue initially with the services on the clients dying for no apparent reason and was fixed with a patch from the vendor. However in the meantime I wrote a small visual basic program (utilising ADSI) that scans each client that was available on the network and verifies the status of the service. If the service is not started an attempt is made to restart the service and the administrator notified. Although this problem has now been fixed, I keep the routine running just as a sanity check.

One area that needs improvement, and being worked on right now, is a real time scanning option for mail. The mail database is scanned fully once per day and the real time process picks up any infected attachments that a user tries to run on the clients. I would feel much happier if I had this extra string to my bow. This would give us the chance to stop a mail borne virus for which we had no signature before it arrived at the user's desktop. The suspect virus could be moved directly into a quarantine area to be inspected in a safe manner.

The only manual virus related check I perform on a routine basis when I arrive at work in the morning is to verify that the clients are all running the same version of signature file, which can be done easily from my desktop. This implementation has worked well for us and although we have seen many viruses come to site, we have yet to see one infect any of our systems.

One important area that is often missed, for whatever reason, on small sites is that of user education. My personal opinion is that it is important for users have information on viruses that is aimed at fostering their understanding along with an explanation of how to handle suspected infections, hoaxes etc. We try to keep this documentation brief enough so that people will read it, yet detailed enough so that they feel comfortable with any recommendations contained within. Most people who work here have PC's at home and any that wish to be kept informed of the latest virus bulletins are added to a distribution list and I forward any information that I come across that I feel may be of use to them. I keep updated from a number of mailing lists including the vendors. I have been particularly pleased with how the perception of viruses has progressed amongst our users during the past three years.

Upgrading the quality of Passwords

Before I embarked on this cause, I knew that I would be in for a bumpy ride. For some reason, the length of password that a user has to type in tends to become a very emotive issue. I started, as is generally the case, by issuing a one-page document that covered three topics. The first was password length and complexity. I explained that the implemented scheme was far too weak, and left the site at risk. The second issue was how to protect passwords from social engineering. Much has been written about social engineering and my own feelings are that this will become a much more prevalent method of cracking sites as the general level of technical protection increases. The document basically stated that under no circumstances should anyone give their password to any other party, even the administrator, and that they should not enter their password at the PC in response to anything but the ctrl+alt+del sequence at logon, and to clear the screen saver. The third issue gave brief guidelines on how to choose a good password, one that should be difficult to guess, but easy to remember. Under no circumstance should a user choose a password that needs to be written down. Once a password is written down (unless the written copy is locked in a safe) then it is automatically weakened. For a general user account, I prefer that people choose something that they can remember without having to resort to pen and paper. The examples I gave consisted of easy to remember sequences containing words that are spelt incorrectly, and yet easy for all users to grasp. This helps to thwart a dictionary-based attack. For example I take the word Yellow and change the spelling to yeller remembering to check that I have not ended up with another valid word or the dictionary attack will succeed. Now I take that and change the case of some of the letters, yeLleR. Now I want to add something different that will still be easy to remember - 1yeLleR1. When choosing a password using this method I specify a few don'ts. Do not add a digit to the beginning or end and just increment this at password change times. Do not use something like a month abbreviation at the end such as Jan, Feb etc.

The resulting uproar took me by surprise. I wanted to migrate to a nine character minimum complex password, with a history list of five passwords

and an expiry time of 90 days. The responses I got included: -

“Who on earth would anyone want to hack into here”

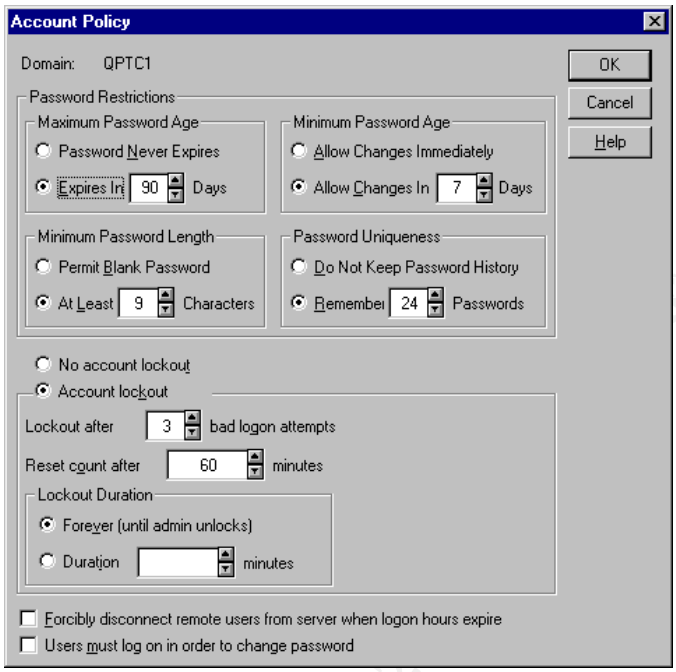
“I will never be able to remember a password that is longer than 6 characters”

“I will never be able to think up enough passwords if I have to change it every 90 days, and cannot reuse them”.
I countered this in a good humoured way (I wanted them on my side) by explaining to them that most of them regularly create documents in excess of 2000 words, and many of these documents include many words longer than 9 characters. Many users were still not convinced so I offered a challenge. I would run a password cracker and see how many passwords we could crack in a 12-hour period. To my surprise, everyone thought that this was a good idea. The CEO backed the audit and so users were given 7 days to change their passwords if they wished. From the results, I can only assume that not many users took up this offer. I used a cracking program from L0pht industries http://www.l0pht.com (Now http://www.atstake.com) and I can heartily recommend this program to any administrator of NT systems who wants to run password audits.

Please remember that you should always obtain permission before running a password audit. Whenever I am about to perform a run I first obtain permission from the CEO and then give all users 7 days notice before running the audit. I am using the tool to raise awareness, not to catch anyone out and I would much sooner perform a run and not crack anyone’s password in the allotted time.

The time came to perform the run. In the first 7 seconds, the program cracked 17 passwords. I had to rerun it as I thought there was a mistake. In the 12-hour period 26 passwords were discovered. Much to my great embarrassment, and the mirth of users, the administrator’s password was one of these. Once these results (with passwords removed) were circulated, everyone really got behind the push to improve the level of our password protection.

First step was to make changes to the basic account setup. For reference, I used the **NT Security Guidelines** written by **Steve Sutton** of **Trusted systems**. You can obtain a copy from http://www.trustedsystems.com/tss_nsa_guide. A word of warning; These days there are many good guides available, both in the book stores and on the Internet with the majority making good recommendations. Make sure that when you utilise this material you read it, understand it, and then instead of just copying all of the recommendations make a decision based upon what is



appropriate for the site you manage.
Let us look at each of the account Policy settings in turn.

Password restrictions:

There are four settings to be made within this area. In my opinion, the strength of the password scheme has to be considered with respect to **all** the parameters. Remember that all four parameters combine to give your password policy, not one setting in isolation (a discussion I have regularly with our auditors). For example, it is no good having a maximum password age of 10 days if you allow blank passwords and no history list.

The **Maximum Password Age** is the maximum length of time that a user can use the same password before it will expire. The setting is between 1 and 999 days. Most guides I have seen recommend a setting of 30 days. However, our maximum is set to 90 days. I was a little concerned that if I asked users to change their password more frequently some would end up writing their passwords down, effectively diluting any implemented password scheme, and 90 days is appropriate for us when considered within the general password policy.

The **Minimum Password Age** is the minimum length of time a user has to wait after changing his/her password before they may change it again. This mechanism is used to prevent a user 'cycling' their password. Without having a minimum setting a user can change their password a number of times

(until the password history list is exhausted) and then reuse their original password.
This would effectively mean that it is possible for a user to continually use the same password. The range is 1 to 999 days. Most of the published documents that recommend this setting to be utilised generally specify a setting of 1 day. I, however, like to set this to 7 days. This definitely discourages any cycling of passwords.

The **Password Uniqueness** is a setting that controls how many passwords the system will 'remember'. This setting prevents a user from re-using their favourite passwords. The range is from 1 to 24. Most guides recommend that you set this figure to 24 ensuring that users have to choose a new password every time. We use a setting of 24.

The final setting available in the password restriction area is the **Minimum Password Length.** Available choices range from blank through to 14 characters. This setting really has to come down to individual site preferences. I settled on 9 characters as being the appropriate figure for our site, and one that everyone can live with. I personally would not like to run below this figure, although I can see where some sites may need to increase it.

These figures are minimum standards set for the site as a whole and people are free to choose passwords that exceed these measures, indeed they are encouraged to do so. Unfortunately, there is no mechanism built into NT for differentiating account settings for privileged and non-privileged accounts. For any privileged account (services etc), I prefer to change the passwords on a monthly basis using a 14 character complex password. Password filters augment the password scheme along with other measures that will be discussed later.

On the Account policy screen some other settings deal with the lockout of accounts after a number of failed logon attempts. This mechanism is aimed at defeating a concerted attempt to guess the password online using methods such as a dictionary attack. This is where a password is effectively tested against a large number of known words under program control. If this setting is not enabled then an attacker may try unlimited numbers of sequences in order to try to break the password. If the attempt occurs over a public holiday, giving the attacker more time to run the attempted break in, then the chances of a breach are much increased. If however the account is locked out after a small number of failed attempts then this will frustrate the attempts to break into the account using these means.

The **Account Lockout Count** specifies how many failed attempts are allowed before the account is locked out. I set this figure to 3. Again, this should be set with whatever is viewed as being appropriate for your site.

The **Lockout Account For** sets the number of minutes an account will be locked out for. This has a range of 1-99999 minutes, or forever. I choose to set this to forever. If an account is locked out on site then I prefer to discuss

with the user what the cause was, and ensure that it was as a result of something they did rather than someone trying to get into their account.

The third setting in this section is the **Reset Count After** that sets the number of minutes until the bad logon count is reset. I set this to 60 minutes. This means that if someone has two failed attempts at guessing a password they have to wait 60 minutes before they can try again in order to avoid locking out the account.

One important thing to note here is that, by default, the Administrators account cannot be locked out. In order to remove the temptation of an online attempt on the administrator's password I utilise a utility called Passprop, available as part of the NT Resource Kit. Passprop allows the Administrators account to be locked out after a number of failed attempts. However, it only locks out remote logons. It is still possible with a locked out administrator's account to log on at the system console using the administrators account.

You should consider renaming the administrators account to something else, then create an account called administrator and watch it closely. I must admit that I have avoided doing this so far because I always felt it was trivial for an attacker to discover what the administrators account had been renamed to. However, after completing the security essentials curriculum I have come around to the concept of 'defence in depth'. Even if it only buys half an hour of time, then hey! It is better for me to have that half an hour rather than the attacker.

Two final settings are available on this screen one of which **Forcibly Disconnect Users** is not required on our site. If you have restrictions placed upon the hours that your users can log on, this setting will disconnect any that are still logged on at the time that the restriction comes into play. The second setting is the **Users Must Log On To Change Password**. If set, this setting ensures that users must log onto to the system before being able to change their passwords. If the password has expired and they are not logged on then an administrator must be contacted for assistance. Even if this is not set, a user stills has to provide the old password before being allowed to change the password.

These settings form the basis of the password policy. However, nothing we have seen so far forces the user to use a strong password. In fact, I could enter 'aaaaaaaaa' as a password and it would be accepted. In order to enforce strong passwords I use a feature that was introduced with service pack 2. A new DLL file was supplied that allows the enforcement of strong passwords. Refer http://support.microsoft.com/support/kb/articles/Q161/9/90 This filter adds the following to our defined password policy.

1. Passwords must be at least 6 characters long (remember ours are set to 9)
2. Passwords must contain characters from at least 3 of the following

    - English upper case letters

- English lower case letters
- Numeric
- Non-Alphanumeric

3. Passwords cannot contain the applicable username
4. Passwords cannot contain any part of the applicable full name (as defined within user manager)

These capabilities cannot be changed. However you can substitute this DLL with one of your own, or that of a third party. The DLL needs to be copied to all domain controllers, although the filter will only ever be used on the domain controller that is acting as the primary domain controller at the time of the change. *The article also discusses an entry in the registry for FPNWCLNT. If, like us, you run only NT then there is no need for this entry to exist, in fact in certain circumstances it could be considered a liability. Refer* http://support.microsoft.com/support/kb/articles/Q99/8/85 *for further details.*

So now, we had the complex password component. Whilst obtaining a copy of L0Pht Crack I came across a very interesting article. Written by Mudge, I believe, who is now the vice president of research and development with @stake after L0pht joined that organisation. The paper described the weaknesses of the NT authentication scheme in respect to LM and NTLM authentication. Unfortunately, I have been unable to rediscover a pointer to the original document on their new web site, but an overview of the issue can be obtained by reading http://support.microsoft.com/support/kb/articles/Q147/7/06. The basic premise is that in order to maintain backward compatibility the password is stored in 7 character chunks (added to this letters are stored in upper case) making password attacks much quicker. This scheme is known as LM (LanManager) challenge/response. There is a second scheme known as NTLM. This uses all 14 characters of the password and represents a much stronger alternative to LM authentication. Clients before SP4 always used both schemes, making network sniffing an attractive option for someone wanting to crack a password. In fact, the crack program has an option to capture passwords in this way. There is an enhancement to NTLM, called surprisingly enough NTLMv2, which significantly increases security of the authentication and session process. I chose to implement this level by following the steps in the document referenced above, being very careful to check all our applications before the site was committed fully to the change.

One final thing I would like to mention that has been done in the area of password protection is in the area of screen savers. Initially we had a system of trust (yeah I know, I can hear you all sniggering) that stopped when I came into work one weekend to find two screens showing information that should not have been available for general consumption (we do have building cleaners etc). I wanted to force everyone to have a screensaver that would start automatically after 15 minutes of idle time and be password protected. To achieve this I used system policies (I already had system policies installed) and set up a screensaver policy as in http://support.microsoft.com/support/kb/articles/Q195/6/55. This worked

extremely well until very recently when I discovered that it was possible to bypass this by using a third party screensaver application. This allows users to use a different screensaver from the one selected through system policies, unfortunately allowing the choice of a screensaver that is not password protected. Changing the permission on the users desktop key in the registry to read only prevents this, but investigating how to do this automatically is on the to do list.

Tightening the Servers

I started by first reading "**Guide to Windows NT Security**" by **Charles B.Rutstein** , which I found to be very good and much of what you see in this section has been learned from this book. I then got to work designing a data directory structure, groups and permissions based upon what I understood of the workflows throughout the organisation. This is another area where smaller sites have an advantage. It is a much easier task for me to look at the whole business process throughout our organisation. I generally try to avoid having permissions change half way down the tree (although its not always possible) or have an odd permission thrown in here and there. It tends to pay dividends when I'm working on a problem at 03:00 in a state that is somewhat less than wide awake and do a permit down the directory subtree. I like to use groups as much as possible as I find things easier to manage this way, and is probably a hang over from my VMS days. I use a local group/Global group structure. Though we are small, and you could possibly argue that this is not needed, it does make things a lot easier to manage if the site starts to expand rapidly.

I then produced a working document for the departmental managers to study. Being a small company documentation is not overly abundant. However some things are very important to document and this certainly qualifies. After considering the feedback and small changes made, everyone was happy to start. The first thing was to create groups to make administration of the users easier. I started by creating global groups and adding the users into the relevant groups.
Note that before I actually started adding users to groups I created a matrix using a spreadsheet and mapped all of the groups and users onto this spreadsheet. This is because I also like to create groups containing users that should have no access . This group is then granted no access to the disk resource. This can, if not done carefully, lead to problems. Groups that are explicitly assigned no access are processed before groups that have other types of access granted. If a user is in a group that allows access and one that denies access then the result is access denied.

Once I had all of the groups that were going to be used initially local groups were created and the global groups were added as members into the appropriate local groups. The local groups were the ones that had permissions assigned to them. I should say that all of the volumes on our site are NTFS allowing a full granularity of permissions to all files contained on the volume.

The following lists permissions that can be applied to files and directories. NT is structured in such a way that individual access rights can be granted to a file and directory, or more commonly, generic rights which are individual rights grouped in ways that are generally more useful.

Directories

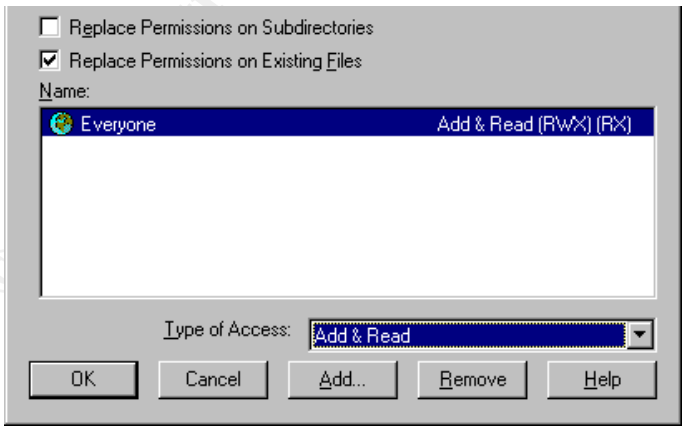
Generic Rights

No Access - As it says, no access
List - Change default to, list files and sub-directory names.
Read - Read and execute files
Add - Can create new files within the directory, but not read them
Add & Read - Create new files, read and execute files
Change - Create new files, read and execute files, modify existing files
Full Control - Read files, change files, execute files and take ownership

The last two options allow you to set specific rights.

Special Directory Access
Special File Access

For example selecting Add & Read groups the following permissions together.
Read, Write, Execute for the directory, Read and Execute for files within the directory. The specific rights granted by each generic group are shown in brackets following the right, the first bracket shows directory access rights, the second shows file access rights.



Overall I find the generic right groupings sufficient and are quick and easy to

use. I have however on a few occasions had to resort to setting specific permissions dependent upon requirements.

Files
Generic rights

No Access - The user or group has no access to the file
Read - The user or group may read or execute the file
Change - The user or group may read, write, execute and delete the file
Full Control - The user or group can do anything with the file

The last option Special Access lets you choose individual rights to build up a custom access scheme.

There are a number of things to remember when assigning permissions. Unless a permission is specified then NT will not grant access, although personally I always create a No Access group. It allows me to specifically restrict users without worrying if they are inadvertently in another group that does have permission as NT always looks at a No Access permission first. Another thing to be aware of is that file permissions are additive rather than minimalist. By this I mean that if a user happens to be in two groups, one of which has read access and one of, which has change access then the users, effective permissions is change access.

The data structure is very much a corporate specific implementation. However, some areas of the file system are common to all systems. In order to be able to tighten access to the default directories I again referred to the **Windows NT Security guidelines** by **Steve Sutton**. Pages 38 – 40 give some guidelines for permissions to be set on these default directories. Note again that these are only guidelines. The text warns that some applications may fail if these settings are utilised. This I discovered for myself. First I read the table (brief example given below) and marked off those that made sense for me to follow, at least in part. I then made each change and watched what happened. I did have some applications fail particularly when I reset the permissions on the \winnt\system32 directory. I had a couple of services that were trying to write temporary files into the directory. In order to be able to gain some visibility of what was happening, as I changed the permissions I turned on file auditing for that directory and sat back and watched. This did catch most of the problems for me, but not all. Some jobs do not run all of the time and I was caught out by at least one job. If you decide to lock down these directories, take your time, be careful, and consider using auditing as a means of troubleshooting if you find out an application has stopped working. This will tell you what problems the application was having and what access it was looking for. Do not change too much at once.

Brief Example

	Guidelines	*Std NT Install*

C:\Winnt\	Installers:Change Everyone:Read SvrOps:Change Creater/Own:Full Admins:Full System:Full	Everyone:Change SvrOps:Change Creater/Own:Full Admins:Full System:Full
C:\winnt\repair	Creater/Own:Full Admins:Full System:Full	Everyone:Read SvrOps:Full Creater/Own:Full Admins:Full System:Full
C:\winnt\system32\config	Everyone:List Creater/Own:Full Admins:Full System:Full	Everyone:Read SvrOps:Change Creater/Own:Full Admins:Full System:Full

I went through a similar process with the system registry. I found a few keys that had recommendations for changes for example

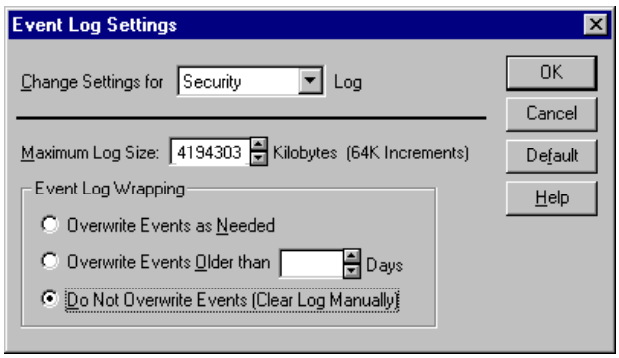
How to restrict remote access to event viewer
http://support.microsoft.com/support/kb/articles/Q245/1/28

Restricting information available to anonymous users
http://support.microsoft.com/support/kb/articles/Q143/4/74

If you do decide to make changes to any registry settings, do pay heed to the warnings and ensure that you have backups of the registry. However the biggest changes I made to the registry were related to auditing and lead us comfortably to the next area, which was auditing.

Auditing
I must admit I love auditing, I have always found it to be immensely useful in my role as problem troubleshooter. I audit probably more than I really need to, but I would rather have too much information than too little. It just means that we have to be smart how we search that data. More on this later, but first lets take a look at what auditing facilities are available for us.

NT has three primary logs encompassed by the umbrella term of “Event Logs” The three logs are the “System Log”, “Application Log” and “Security Log” The log we are primarily interested in for auditing file and registry access is the “Security Log”. The first task with the security log is to ensure that it is large enough so that we do not lose any events. By default the size is set to 512k and set to overwrite events as needed.



The maximum size can be set to around 4Gb. As can be seen from the figure above there are three available settings for event log wrapping. The first, overwrite events as needed , will cause the first (oldest) events to be overwritten once the log has reached its maximum size. I wouldn’t consider this option. The second option will cause events to be overwritten after a certain number of days. Personally, I do not like this option because I can see where someone could flood the logs and then once full the system would fail to audit events because there were no events in the log that matched the criteria for overwriting. The third option prevents any overwriting of logs. Again, this can cause a failure to audit if the maximum size is reached. This is my preference. I set the log size up very high, and then ensure that I get a copy of events from the log at least once per day to a second system. There is one further option that can be set by utilising the following registry key.

Hkey_Local_Machine\System\CurrentControlSet\Control\Lsa\CrashOn AuditFail

If this key is set to 1 then once a failure to audit is detected (possible using option 2 or option 3) then the system will crash. I have come across similar facilities within other operating systems and it is generally recommended. I could not obtain management acceptance to use this feature, so instead I have used option 3, set up maximum log sizes and copy events off automatically to a second system on a daily basis. So, we have our log set up, what now? Well auditing is not enabled by default; it requires a few changes to enable it.

Open User Manager, select Policies and select the Audit option. This displays the Audit Policy dialog box.

In order to turn auditing on select the radio button “Audit these events”. From here auditing can be enabled for success, or failure, of a number of events.

Events

Logon/Logoff

Use this option to audit logon/logoff activity on the system. I select both success and failure. Failures are an obvious choice, but I log success as well. Here I am looking for logons occurring at inappropriate times. If someone had obtained a user password via social engineering, it is a good chance that they would attempt to use it while nobody is around.

File and Object Access

This allows auditing of access to standard objects, for example files. This needs to be turned on in two places in order to be successful. Here, and on the file in question (remember NTFS is required for this). I audit for failures and have various sensitive files, and directories audited permanently. However, I will also use the successful audit sometimes when troubleshooting. Use this with care as it can generate large numbers of entries in the event logs. If I do this I usually try to do it outside of normal working hours when I can close extraneous processes down, and I try to keep the logging specific.

Use of User Rights

All user rights except the ones logged by the logon/logoff audit setting. I have this set for failure

User and group Management

Audits change to users and their environment (i.e. group membership, Passwords etc.) I set this to monitor for failure and success. As I am the only person who should be able to manipulate things such as group membership, I want to keep a close eye to ensure that no other changes do occur.

Security Policy Changes

Auditing of changes to audit policies, user rights, and trust relationships. I select for both success and failure. I am the only person on site who should be able to manipulate any of these settings so I want to keep a very close eye on this one. If someone manages to change my audit

policies without my knowledge, I could effectively be blind to any unauthorised activity that occurs on the system.

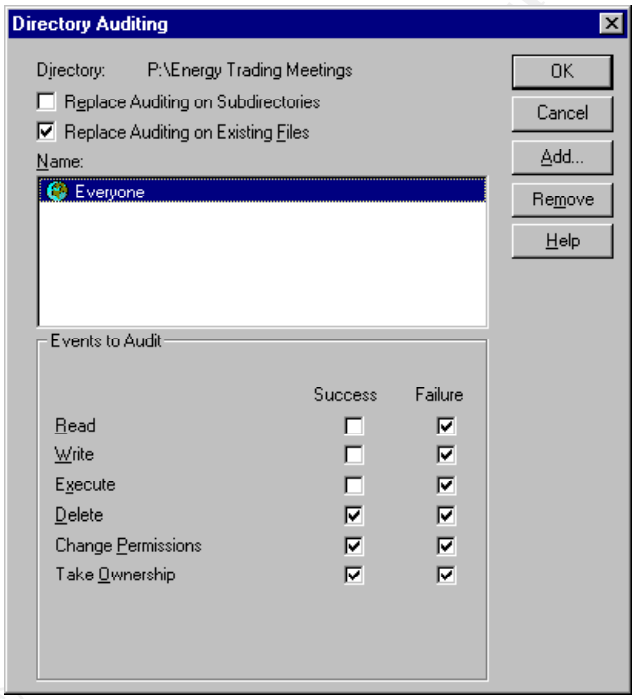
Restart, Shutdown, and System

This option audits restarts and shutdowns of the system. I used to have this set for success and fail, but one of the service packs (SP 4 I think) added an option whereby an event is generated in the event logs so now I only audit for failures.

Process Tracking

This option audits indirect use of system rights. I normally have this set to audit for failures only, but I have on occasion I have used success auditing as an aid to troubleshooting. If you do, beware that an awful lot of entries can be generated by this option.

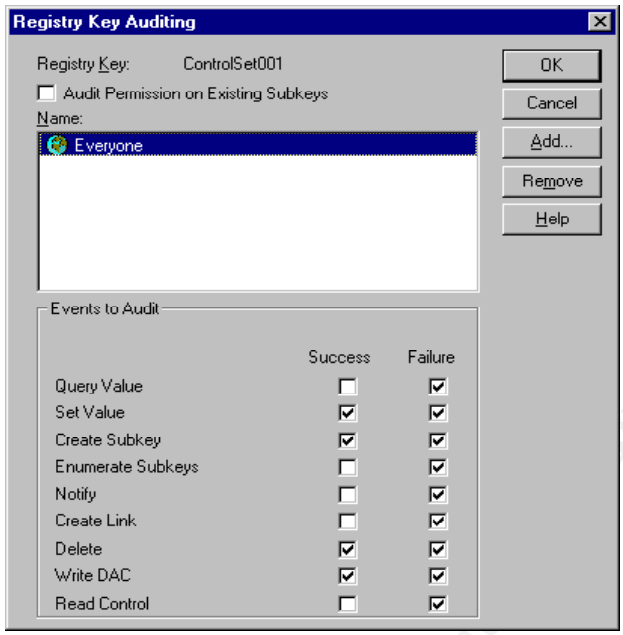
How do we set up auditing on a file, or a number of files? First ensure that you have set up the level of auditing you require on the File and Object Access option within the audit policy. Select the file or folder for which you are interested and right click with the mouse this presents the object property sheet. Choose the tab labelled Security.



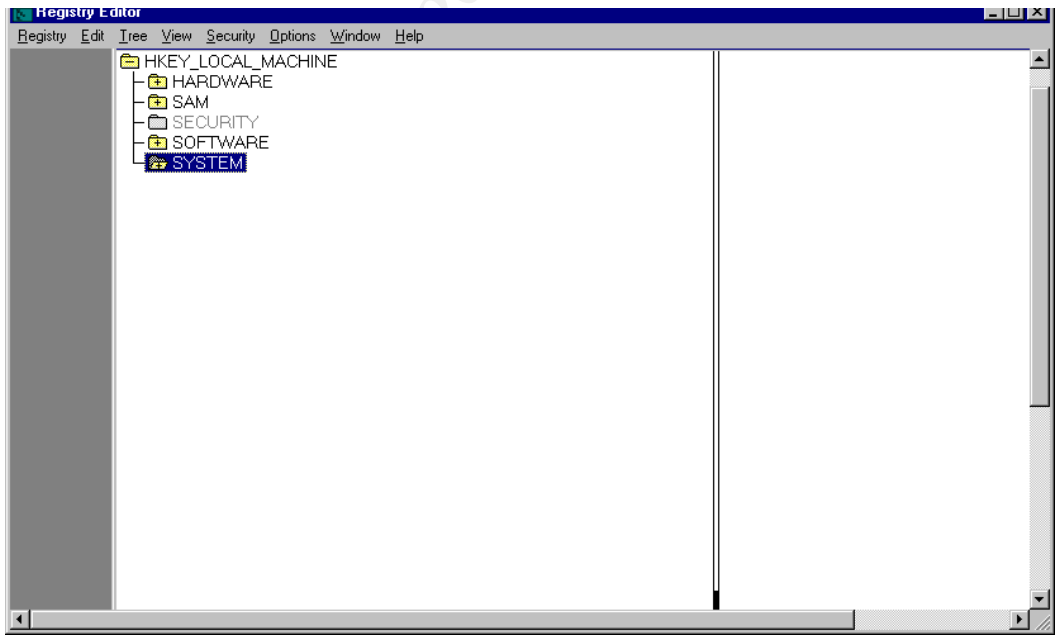
Generally, I would have the options set as shown above. However for some files/folders I am interested in a little more information and may possibly select Write and execute.

Now its time to look at how we audit the registry. Enabling auditing of the registry is accomplished by using the same setting within the audit policy as is used for files and folders etc. Once this has been set then run regedt32

(note that although regedit is very useful for some things it not of any use when it comes to tailoring auditing of the registry) and highlight the tree of interest. Choose security followed by Auditing

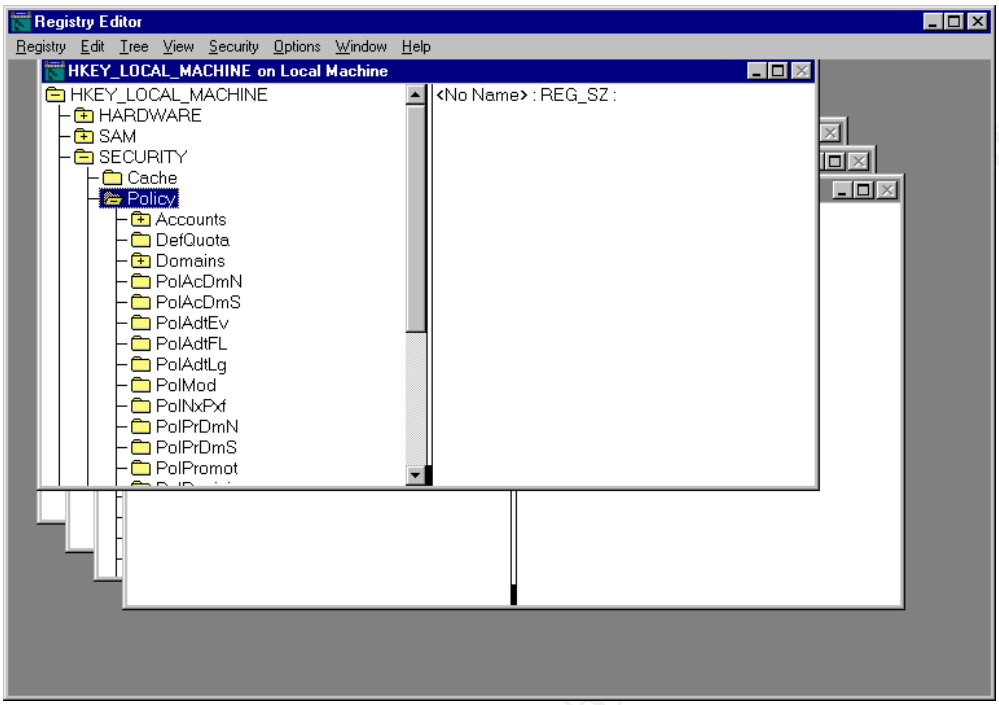


This is how I set the auditing for the majority of registry keys. Note that if you wish to set some of the success settings such as query value then be prepared for searching through large numbers of events. By default, the administrator is unable to access the security hive. It is available only within a system context.



I like to audit the Security hive. In order to gain access I started the schedule

service under the system account (note if you don't use the scheduler service, disable it, or at the very least consider running it under a different user context) and submitted a job to run regedt32 interactively. As you can see I now have access to go in and set auditing up on this tree.



Well now we are producing all of this auditing, what do we do with it? At the moment I check all of the auditing every morning by hand. With us being a small site it's not too bad, it generally takes me about one hour. The logs are all backed up once per week and written to CD to be kept offsite. Just in case anything has been missed that we need to verify at a later date. After completing the Gsec course I have now decided to move all of the audit logs to a central 'Audit server' (Just an old PC) on a daily basis. In order to make analysis quicker and more efficient I am currently writing a VB program to parse the event log and mail me with entries I am particularly interested in. I have also been 'playing a little with snort' Http://www.snort.org both in its Linux form and windows form in order to produce an audit of what is happening on the network.

Although this document has been necessarily brief, and has only been able to cover some of the areas that have been looked at, I hope this has given you a brief glimpse of how the site has progressed from being somewhat skeptical of implementing security measures to gradually accepting increased security as a part of life. Every week, with each new published cracking of a site, it gets easier to convince management of the need to improve security. Whilst we are a long way from where I would like us to be, both in terms of site hardening and of security budget, I do feel that we will continue to improve our overall level of security.

References

Books

National Computer Security Association guide to
Windows NT Security

By Charles B. Rutstein ISBN 0-07-057833-8

Online references

http://www.atstake.com
Suppliers of security tools, advisories and consultancy services.

http://www.trustedsystems.com/tss_nsa_guide
Guide on hardening NT commissioned by the NSA

Microsoft's online technical service

http://support.microsoft.com/support/kb/articles/Q161/9/90
Passfilt.dll and its use in ensuring strong passwords

http://support.microsoft.com/support/kb/articles/Q99/8/85
Potential issues with *FPNWCLNT*

http://support.microsoft.com/support/kb/articles/Q195/6/55
Create a screensaver policy

http://support.microsoft.com/support/kb/articles/Q245/1/28
How to restrict remote access to event viewer

http://support.microsoft.com/support/kb/articles/Q143/4/74
Restricting information available to anonymous users

http://www.snort.org
Snort – An intrusion detection system

© SANS Institute 2000 - 2005 Author retains full rights.