Hardening Windows 2000

Philip Cox (phil.cox@SystemExperts.com)
Version 1.2
5/25/2001

Abstract

Determining what steps need to be taken to secure a system is one of the most frustrating things that system administrators have to do. This white paper provides a method for getting a Win2K server to a "secure" baseline. From this baseline you can work forward to install additional services. The focus of this white paper is on the Win2K server, but much of the information is applicable to Win2K Professional as well.

This paper is excerpted and modified from Chapter 21 of the "Windows 2000 Security Handbook" (ISBN: 0-07-212433-4, copyright Osborne/McGraw-Hill) authored by Phil Cox and Tom Sheldon (www.windows2000securityhandbook.com).

Original material and content © Osborne/McGraw-Hill, 2000-2001. Supplemental material and updates © SystemExperts Corporation, 1995 - 2001.

Table Of Contents

The Requirements

Like anything, you have to start with architecture, especially when hardening systems. There are a few fundamental questions that you need an answer to in order to ensure that the systems you configure are not too "hard" or too "soft" when all is said and done. Here are the questions to ask:

- Can the system be self-contained (that is, a workgroup), or does it need to be part of a group (that is, a domain)?
- If in a domain, can you have native-mode, or will you require mixed-mode? If there will be any WinNT Backup Domain Controllers (BDC's), then you must use mixed mode.
- How many interfaces does the machine require? This is usually more applicable to firewall machines, which require at least two network interface cards (NICs). A third card would be used for an extranet.
- What services will it be providing?
- What protocols will you be using?

Once you have the answers, then you can press on to configuring the system. If you don't have the answers, then get them *before* you start.

Physically Secure It

Win2K security, as with its predecessor WinNT, relies strongly on physical security and proper hardware setup. To maximize security, we need to perform the following tasks and configure the system appropriately: Install case locks on all publicly accessible systems. Put critical or highly sensitive systems in cages. If removable media (that is, floppies, CDs, ZIP drives) are allowed, then you should set the hardware to boot from the hard drive first. Set the EEPROM boot password.

Install the Operating System

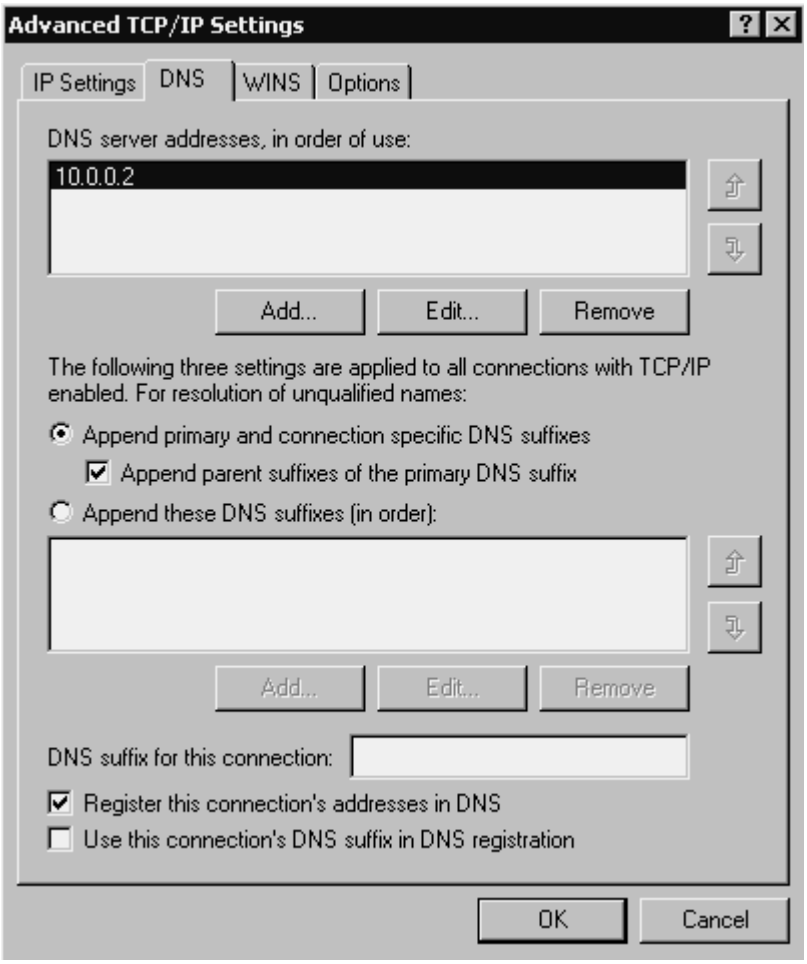
When you are installing for security, you should start from scratch (that is, not upgrade from a previous OS like WinNT), as there are potential unknowns about the software and state of applications on that system. You will save yourself a lot of time and potential frustration if you do a clean install.

VIP: One important item of note is that if you do install from scratch, the default NTFS and registry permissions are adequately secure. This was not the case in WinNT and should be enough of a motivator in and of itself to get you to start clean.

Here are the steps in the install process that have security implications:

1. The install starts with a number of screens about licensing and so on; they are pretty standard and self-explanatory.
2. You will come to the disk configuration screen. You will want to ensure that the disk is partitioned into at least two separate partitions. One for the system and OS files, and the other for data files.
3. The next screen will ask you for the format of the disks; you should choose NTFS (which is the default selection).
4. You will then be prompted to configure proper regional settings (this is especially important for timestamps on logs), name and organization, and license modes.

5. Following that, you will enter the computer name and the administrator password as shown in the following illustration. It is *very* important to select a strong password for the administrator. *This will be the password that is stored in the local SAM database.*
6. The next screen, will allow you to specify which Win2K components are installed. **You should uncheck all the options**. We want a minimum install and will go back later to add anything that we actually need. Remember that we are going for a minimalist system, and then build it up to what we want later.
7. The next option is Networking Settings. Use "Custom Settings" and enter the specifics of your networking setup.
8. Because we choose the Custom Settings option, we will have the opportunity to configure each and every network interface card that is in the system. **For each interface, you should ensure that the only thing that is selected is Internet Protocol (TCP/IP)**. If you need more services or protocols, you can install them later.
9. The general screen on the NIC, allows you to set the IP addresses. Use static IP addresses for highly secure systems. You may choose to use DHCP (that is, assign automatically), but that is a call you must make.
10. Click the Advanced button to configure the DNS and WINS configuration on the NIC. The DNS configuration is set up to have the system attempt to perform dynamic updates to the DNS server. If you do not support this, then you are encouraged to disable the "Register This Connection's Address in DNS" option. Since this is information that we are sending out, and not accepting, it is of minimal risk from this system's perspective[1].



You will also want to disable the Enable LMHOSTS lookup and select the Disable NetBIOS Over TCP/IP option on the WINS tab. This effectively disables a large portion of the NetBIOS that is in Win2K (but not all of it—we'll cover that later on).

[1] This does affect the DNS server that you are flooding though ☹, and is not proper Net Etiquette. You should validate that your DNS servers accept dynamic updates.

11. Next you will determine if the system is to be part of a domain or workgroup. The setting of this is determined by the answers to the questions that you asked earlier. The best bet is to specify that the machine is part of a workgroup, and *not* a domain. If it is self-contained, this means that there is no need for any Microsoft Networking communications between it and any other host. If you put it in a Workgroup and use Microsoft Networking, then you will be using NTLM authentication instead of Kerberos. If you require Microsoft communications, then set up a standalone domain (that is, a new domain in a new forest).
12. Then let the install finish.

What About a Domain?

Note that if you are going to use an isolated domain, here are some guidelines:

- Make sure it is a new domain, in a new forest.
- Validate that there are no trust relationships established.
- Run an internal DNS server on that domain. Use screening routers and DNS configurations to block requests from external networks. Configure DNS to *only accept secure updates* from a host on the isolated network.
- If you require trust, then use the older WinNT method, and establish specific one-way trusts that are not transitive.

What If You Can't Start Anew?

There may be times when there is no option but to go with what you have. In that case, all is not lost, but you have your work cut out for you. In order to go this route, you should configure the system using the Local Security Policy tool. In this tool, use the security policy templates (`setup security.inf` for all systems; then in addition to that, use `DC security.inf` for domain controllers) located in `%systemroot%\security\templates`, to reconfigure the local system to the level of a freshly installed system. (Note that this has a high probability of causing stuff to break, and that is where the work comes in.) Also, ensure that all of the above system requirements are set (that is, with strong passwords and NTFS).

Tightening the System

Once the system reboots and starts up, you need to go to the second phase of "hardening," which consists primarily of disabling services that you did not have a choice to install (and there are a number of them), and setting local security policy. Policy deals with things such as SMB signing, authentication accepted, legal notices, auditing, and more.

Hardening Services

For each service that exists on a Win2K system, you have a number of controlling options. You can also disable the startup of options so they don't run at all. The basic strategy is to run as few services as possible. Another strategy is to run services under less-privileged accounts. All default services runs under the context of the Local System account, which has privileges to the entire system.

Disabling Services

The biggest layer on the security onion is disabling unused and unneeded services. This is probably the hardest thing to get right, mostly because of the horrid lack of documentation from Microsoft as to each service's complete function and dependencies. As a matter of principle, it is easier to define what should be enabled (that is, what's required for basic functionality) for a basic TCP/IP system, and turn off the rest, rather than trying to turn off what is bad.

For a medium- to high-security system, ensure the services listed below are the only ones running. The asterisks (*) indicate the minimal services required to operate the box—all others are optional and represent potential risk.

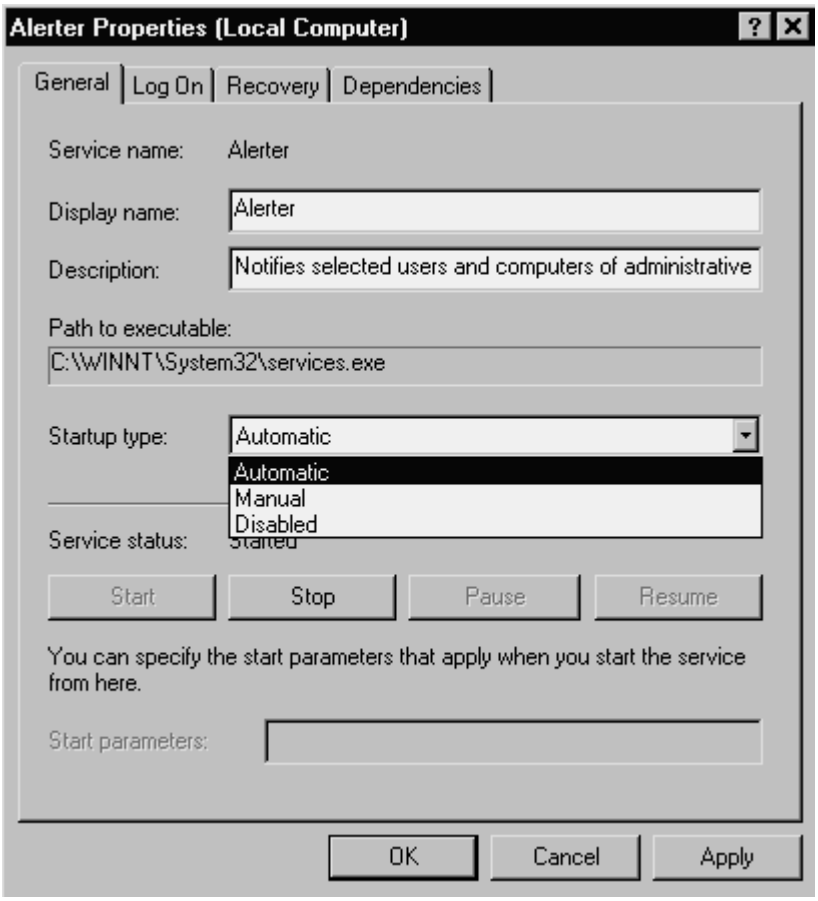
- DNS Client*
- EventLog*
- IPSec Policy Agent
- Logical Disk Manager*
- Network Connections Manager
- Plug & Play*
- Protected Storage*
- Remote Procedure Call
- Remote Registry Service
- RunAs service
- Security Accounts Manager*

For a domain controller you will need

- DNS Server (unless you have a Dynamic DNS server already existing)[2]
- File Replication Service (>1 DC)
- Kerberos Key Distribution Center
- Net Logon
- NT LM Service Provider
- RPC Locator
- Windows Time
- TCP/IP NetBIOS helper
- Server (when sharing resources or running the AD)
- Workstation (when connecting to resources)

[2] Note that the *requirement* to run a Win2K DNS server is argued, the only time a Win2K DNS server is required is when you want to have secure dynamic DNS up dates. DNS architecture is beyond the scope of this document, but suffice to say that you can use static DNS and enter all required Active Directory entries (which are written to a text file when setting up an AD server), or unsecured Dynamic updates with BIND.

To disable the services that are running but are not needed at this time, we will use the Services control panel. To start it up, select Start | Settings | Control Panel | Administrative Tools | Services. Then open (double-click) the appropriate service. This will bring up the service's Properties sheet. From there, you change the Startup Type field to Disabled. Since this will prevent it from starting at the next boot, we will need to stop the service now. To do this, click the Stop button in the Service Status section.



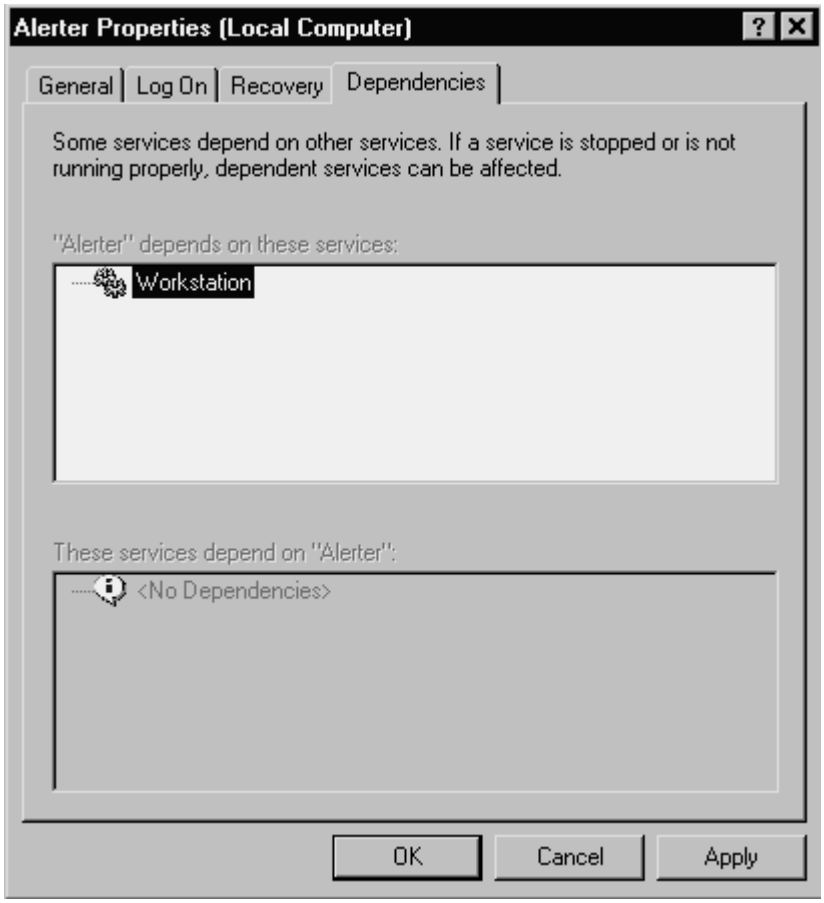
You can also use the command line tool `sc.exe` for the same functions. `SC.exe` is a command line program used for remotely communicating with the Service Controller and services. It can be used to perform many service related functions, including remote service querying and disablement

Tip: When you're not dealing with highly secure hosts, but just want to generally secure systems, this still applies. The general premise of not installing services unless you need them, and then disabling the ones you cannot uninstall is still valid. You will have to take some time and effort to understand the exact setup for any architecture you need, but this should give you a good feel for how to go about it.

Disabling or Deleting?

There is always a question as to whether it is best to *disable* or *delete* a service. The basic premise is that enabling, then starting it can restart a disabled account. This is a simple and easy process *if* you have the right permissions on the system. A deleted service, on the other hand, cannot be started until it is re-installed, thus it is significantly harder to have this happen maliciously, especially for Win2K core services. The answer seems simple, but it's not. The problem is that it is very hard (impossible perhaps) to actually remove some of the services. The best bet for time and efficiency is to disable services that have no easy installation method (that is, are installed at this point in the setup), and delete/uninstall others that you may add later in the process.

Note: Finding Dependencies: You can determine which services rely on other services with the Dependencies tab on the service's Properties sheet.



Deleting "Hidden" Services

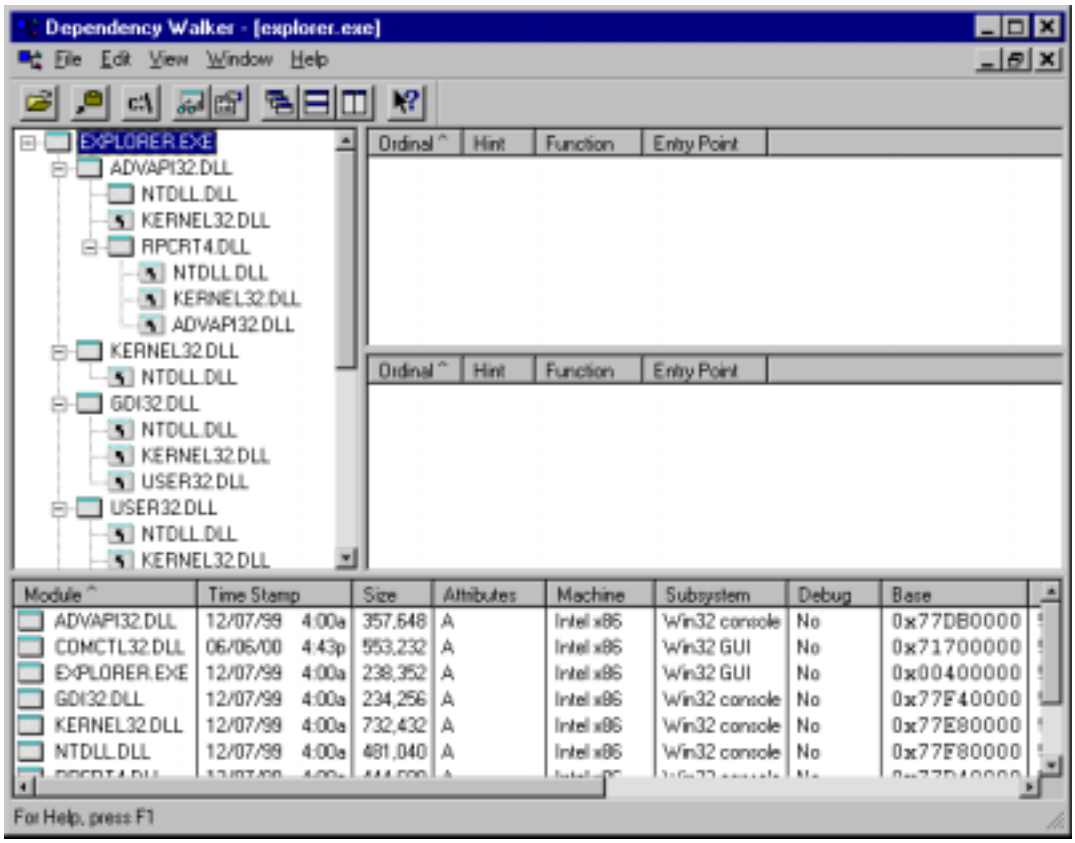
*VIP: These may totally disable your system and require a total rebuild, test this **before** you implement it on a production system.*

You can remove the following programs from a Win2K system: Fax, COM, DTC, Imagevue, Games, Accessory Utilities, Communication Applications, PinBall, Accessibility Options, and WordPad
Open `sysoc.inf` (`%systemroot%\inf`), and remove the "hide" keyword. Now you can use "Add/Remove Programs->Add/Remove Windows Components" to remove them

Another option to look at is removing the service via the registry by deleting its key under `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services`.

Application Dependences

Another helpful tool is `Depends.exe` from Resource Kit. It can be used to determine service dependencies and running processes in exe's. ocx's, and etc. The Dependency Walker recursively scans all dependent modules required by a particular application. See the help associated with the tool form details on its use. A screen shot of the dependencies in explorer.exe is shown below:



Syskey

By default, Win2K strongly encrypts the local Sam database. This is the Syskey option in WinNT. The potential problem is that the key used to decrypt the database is stored in an obfuscated form in the registry. This could be a potential problem. To eliminate the problem, you can reconfigure Syskey to require manual password entry or to read it off a floppy. You open the configuration by running the SYSKEY command from the command line, and then using the Update option. The problem with either of these two options is that they require some level of user intervention (unless you leave the floppy in the drive, which has its own security problems). So you will want to reconfigure this only for highly secure systems that will require manual intervention to start up.

Setting System Policy

Since we have chosen a self-contained (that is, no domain) type system, so we will be using the Local System Policy tool to set our policy.

Password Policies

Make sure that the following options have been configured in the Password Policy section of the local policy. Set the values in accordance with your company's policy.

Enforce Password History	Enabled (recommended value is 5)
Maximum Password Age	Enabled (recommended value is 60)
Minimum Password Age	Enabled (recommended value is 5)
Passwords Must Meet Complexity Requirements	Enabled
Store Password Using Reversible Encryption	Disabled

Account Lockout Policies

Make sure that the following options have been configured in the Account Lockout Policy section of the local policy. Set the values in accordance with your company's policy.

Account Lockout Threshold	Enabled (recommended value is 5)
Account Lockout Duration	Enabled (recommended value is 30)
Reset Account Lockout Threshold After	Disabled (recommended manual reset of accounts)

Audit Policy

Make sure that the following options have been configured (at a minimum) in the Audit Policy section of the local policy. Set the values in accordance with your company's policy, but at a minimum, you should audit success and failure for the following audit categories:

- Audit Account Logon Events
- Audit Account Management
- Audit Logon Events
- Audit Policy Change
- Audit System Events

Audit Log Settings

You need to ensure that there is adequate space in the audit logs for the audits that will be generated. This is especially important if you will halt the system on audit failure (see the next section). You should configure the systems to handle your log capacity (you will have to test to see what that is) plus another 50 percent. Remember to set your rotation policy as well. This should be consistent with whatever policy you have.

User Rights

Make sure that the following options have been configured in the User Rights section of the local policy. Ensure that they have the users and groups you want in them. A good rule of thumb is that the Administrator has most of them. You should be validating the entries that are done during the setup, as they are pretty good. If anything, you should be removing accounts and groups, or adding specific groups that you require.

- Act as Part of the Operating System
- Access This Computer From the Network
- Back Up Files and Directories
- Change the System Time
- Create a Token Object
- Debug Programs
- Force Shutdown From a Remote System
- Increase Scheduling Priority
- Load and Unload Device Drivers
- Log On as a Service
- Log On Locally
- Manage Auditing and Security Log
- Modify Firmware Environment Values
- Profile Single Process
- Profile System Performance
- Replace a Process Level Token
- Restore Files and Directories
- Shut Down the System
- Deny Access to this Computer from the Network
- Deny Logon Locally
- Take Ownership of Files or Other Objects

Additionally, if your systems are part of a domain, you should validate the users and groups that have the following rights:

- Add Workstations to Domain
- Enable Computer and User Accounts to Be Trusted for Delegation
- Synchronize Directory Service Data

Directory Permissions

Default NTFS and registry permissions are adequately secure, but the root (C:\) should be tightened down. The installation of Win2K will give the Everyone group full control of the top level of this directory. This causes problems in that the C:\ directory is the first directory searched for files in many instances. Thus if an attacker can install a trojaned file, then it will be executed before any other. One way to correct this is to ensure that the Everyone group has Read-only access. CAUTION: This has a high likelihood to break some software, so ensure you test it in your environment before propagating it out.

Security Options

Make sure that the following options have been configured in the Security Options section of the local policy. You should be validating the entries that are done during the setup, as they are pretty good, but should be specific to your installation. The settings dealing with signing and encrypting of SMB and the Secure Channel are of critical importance if you are using Microsoft Networking.

Note: These policy settings just create and set registry keys (that is, they are just a pretty interface for the recommended registry keys settings).

Additional Restrictions for Anonymous Connections	No access without explicit anonymous permissions
Allow System to Be Shut Down Without Having to Log On	Disabled
Audit Use of Backup and Restore Privilege	Enabled
Clear Virtual Memory Pagefile When System Shuts Down	Enabled
Digitally Sign Client Communication (Always)	Enabled (for high security)
Digitally Sign Client Communication (When Possible)	Enabled (for medium security)
Digitally Sign Server Communication (Always)	Enabled (for high security)
Digitally Sign Server Communication (When Possible)	Enabled (for medium security)
Disable CTRL-ALT-DEL Requirement for Logon	Disabled
Do Not Display Last User Name in Logon Screen	Enabled (for multiuser systems)
LAN Manager Authentication Level	Send NTLMv2 responses only/refuse LM & NTLM
Message Text for Users Attempting to Log On	Get from your legal department
Message Title for Users Attempting to Log On	Get from your legal department. Something along the lines of “Authorized Users Only”
Number of Previous Logons to Cache (In Case Domain Controller Is Not Available)	0
Prevent Users From Installing Printer Drivers	Enabled
Recovery Console: Allow Automatic Administrative Logon	Disabled
Rename Administrator Account	Rename this to something other than “admin” or “administrator”
Restrict CD-ROM Access to Locally Logged-On User Only	Enabled
Restrict Floppy Access to Locally Logged-On User Only	Enabled
Secure Channel: Digitally Encrypt or Sign Secure Channel Data (Always)	Enabled (for high security)
Secure Channel: Digitally Encrypt Secure Channel Data (When Possible)	Enabled (for medium-high security)
Secure Channel: Digitally Sign Secure Channel Data (When Possible)	Enabled (for medium security)
Secure Channel: Require Strong (Windows 2000 or Later) Session Key	Enabled (for ultra-high security)
Send Unencrypted Password to Connect to Third-Party SMB Servers	Disabled
Shut Down System Immediately If Unable to Log Security Audits	This should be consistent with your policy. It is very drastic and can be used as a DoS attack
Strengthen Default Permissions of Global System Objects (e.g. Symbolic Links)	Enabled
Unsigned Driver Installation Behavior	Do not allow
Unsigned Non-Driver Installation Behavior	Do not allow

Unbinding Services

The best way to secure a system is to use it for one purpose and secure it around that specific purpose. This should be your goal: one service, one system. The reality, though, is that many people will not heed this advice and will run systems that support multiple functions, because either they do not see the problem with it, or they have financial constraints that prohibit them from doing it the right way.

The best way to understand this is with a scenario. Let's say you want to be able to administer all aspects of the Win2K box from your internal network, as well as provide Web and FTP services to Internet users. This model poses a potential problem, in that external users may be able to access the administration services that were meant for internal users only. To set this up securely, you will not only "disable" services, but also you will "unbind" specific services from specific network interfaces.

Networking Services

Consider that there are two basic categories of networking services available on a Win2K computer system:

- Microsoft's File and Print services (a.k.a. Microsoft Networking), that is SMB or CIFS. This is the native method for almost all Microsoft-related API transports. These services are installed by default.
- General TCP/IP and Internet services such as Web servers and FTP servers. They are installed as optional services.

The major problem with Microsoft Networking is that it is more than likely set up for backward compatibility, and not set up to enforce any of its security features. This is not to say that the general TCP/IP services are any more secure, but individually they usually are limited in the scope of what they do. This tends not to be the case with Microsoft Networking. Suffice to say that we need to ensure that "default" Microsoft Networking never reaches any untrusted network (that is, the Internet).

The best way to ensure this is to not install it, or just disable it. This may be easier said than done though; completely disabling Microsoft Networking may not be practical. Assume you set up the Web server to publish information about your company on the Internet. Now suppose the marketing people in your company need to frequently update information on that Web server. If the Web server is isolated and locked up in some closet, they are going to have a tough time making those updates. If the content is complex, transferring it to the Web server via floppy disk, tape, or other media will be difficult and they will probably rely on you, the system administrator, to handle those updates. Here are two possible solutions to this problem:

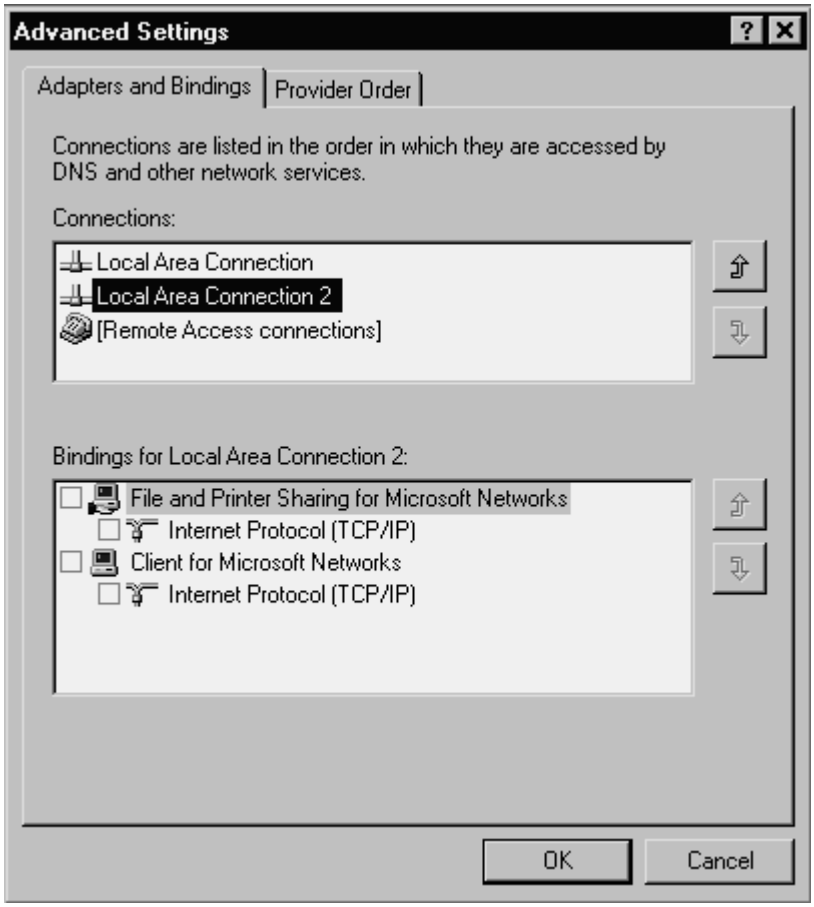
- Use another method of updating the Web server, such as FrontPage extensions.
- Install Microsoft Networking and "disable" it on all non-internal interfaces.

You decide on the second option because it is something you can control and watch very closely.

Note: It would be trivial to set up an IDS system to detect any "anomalous" activity coming from this service on this machine.

Disabling or Unbinding

To unbind Microsoft Networking, select Start | Settings | Network and Dial-Up Connections. Then highlight NIC, go to the menu bar, and select Advanced | Advanced Settings to bring up the Advanced Settings dialog box.



To disable the Microsoft Networking service protocol on the LAN adapter, you would select the LAN adapter in the Connections panel, and clear the checkboxes associated with the Microsoft Networking services.

Note: A reboot is **not** *required to set this feature. Unbinding the File and Printer Sharing for Microsoft Networks option will prevent remote machines from connecting to CIFS/SMB services on this machine and will close port 445 tcp and udp. Tcp 139 will still be listening on this NIC, but will not return any information to the remote machine. If the Client for Microsoft Networks option is still enabled, the host itself will still be able to perform SMB connections to remote hosts even though it won't accept any incoming requests (if the file and printer sharing option is disabled.)*

Once you have completed this, now you will be able to access Microsoft Networking and the Web server on this machine from your internal network, while only allowing Web access from the Internet.

VIP: This setup should not be used on DMZ or firewall systems. Anyone concerned with security should not use any Microsoft Networking on firewall or DMZ systems.

For security reasons, consider requiring that all server administration take place at the console of the server itself, thus eliminating the need for Microsoft Networking connectivity (unless you use a domain architecture). To do this, set the Deny Access to This Computer from the Network local policy for the Administrators group, thus revoking network logon privileges for the Administrators.

Digging Deep

Sometimes you have to dig deep to get the desired results. For example in WinNT, just because you unbound NetBIOS from the interfaces, didn't mean it quit listening; you had to disable the WINS driver to get it to stop listening completely. The question is, "what do you have to do to get your desired effects?" That all depends on your security needs. Just keep in mind, that you may have to "dig deep" to get what you want.

Filtering TCP/IP Connections

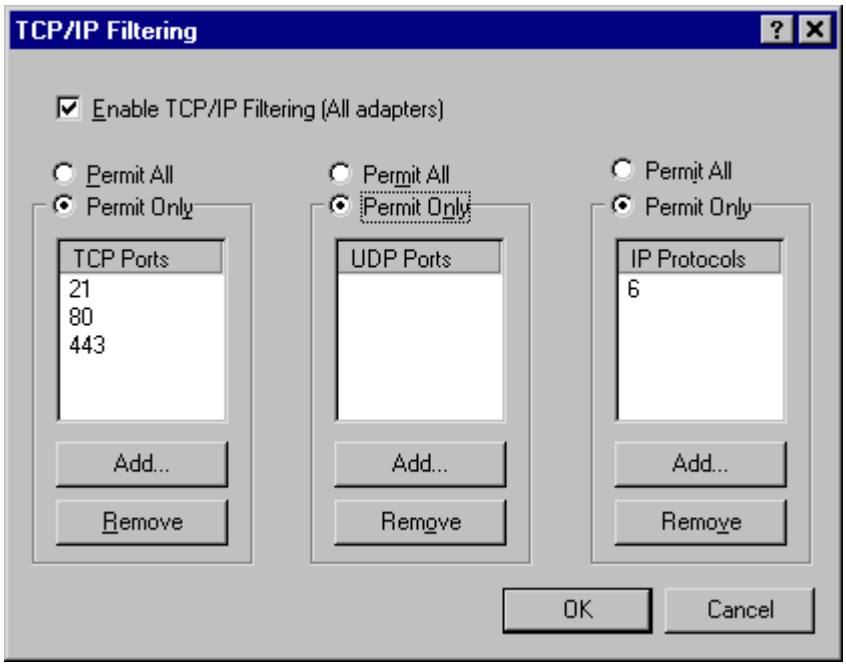
Another option in the hardening process is to set up the TCP/IP filters on a system to help secure it. This should be used on systems that are directly connected to untrusted networks (that is, the Internet) or on systems that you want a higher level of assurance in the security of the system.

There are two methods to accomplish this task: IPSec filters and TCP/IP Filtering. The latter is the same method that WinNT provided. It is not as granular as IPSec filters, but is a bit easier to setup, we will cover that first. IPSec filters are a great way to provide detailed filters, but are more complex to configure. You should use IPSec filters, as they can be implemented in Group Policy, where TCP/IP filtering is only locally configurable.

TCP/IP Filtering

To continue with the Web and FTP server example, we can set up filtering using the Network and Dial-Up Connections control panel. Select the interface you want to set up the filter on, and select Properties | General | Internet Protocol (TCP/IP) | Properties | Advanced | Options | TCP/IP Filtering | Properties (shown below). As shown, we have set up the filter to allow inbound TCP connections to ports 21 (ftp), 80 (http), and 443 (https). Also, we have allowed no UDP (that is, by permitting no ports), and set it to only allow IP protocol 6 (TCP). Rebooting is required to enable this feature.

Note: Even though we have not chosen to allow IP protocol 1 (ICMP), ICMP traffic will still be allowed to and from this host. ICMP can be blocked via the Local Security Policy for IPSec.



You do not want to deploy this on every system for two very good reasons: administration and Denial of Service (DoS). Since the TCP/IP filters must be administered on the local system, you have a major administration issue. But the more pressing issue is that if you don't get the ports right, then you will create a DoS yourself. When you set up filtering, it is important to understand every port that will be open, and when you have RPC-based services, which many of the Microsoft services are, you can't be sure what is running where. A good rule of thumb is to use this on systems that are offering services with well-known ports (that is, Web and Mail).

Important "Features"

TCP/IP Filtering has some very important "features" that you should be aware of:

- It does *not* affect any outbound packets or inbound packets for already established connections. Thus a system with everything blocked in the filter rules can still communicate as normal, as long as the local system in question initiates the packets.
- It does not really understand the IP portion of the stack, in that even if you say to only permit IP protocol 6 (TCP), it will still allow ICMP in. Thus, there can be no assurance that it will block any other IP-based protocol that is apparently not allowed. It is really only useful for filtering TCP and UDP protocols.
- By disallowing UDP, you will block the ability of your client to receive DNS query replies. This is because the filtering is not stateful, and thus the return UDP packet is blocked.

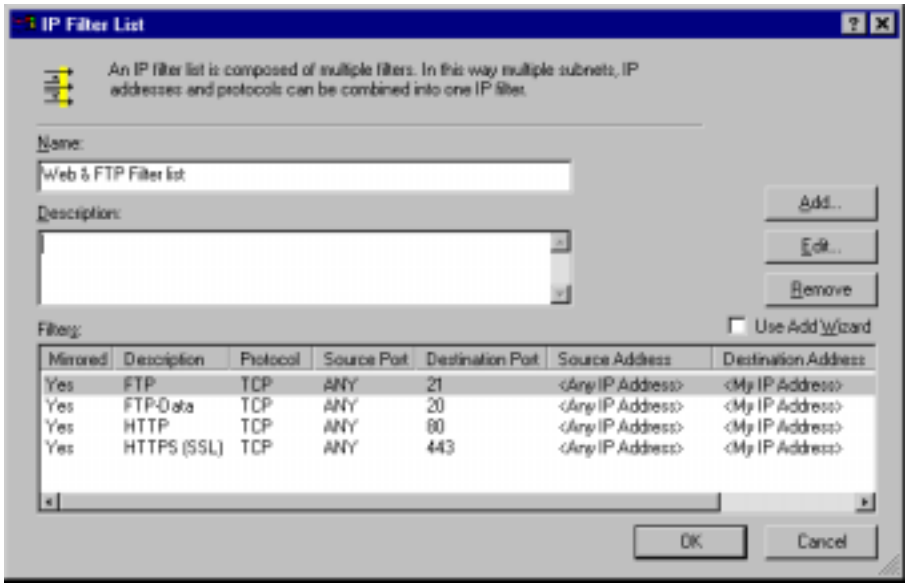
IPSec Filtering

You will manage IPSec policies from the Local Security Policy or the individual IPSec Policy snap-in, and are activated via the Local or Group policy. The three key configurations that we will need to set are:

- IPSec filter lists
- IPSec filter actions
- IPSec policy rules

First you will create an IPSec policy, in our example we will call it "Web & FTP." Then within the new Web & FTP policy, we will add an IPSec filter list that will be use to hold the IPSec filter actions we want applied. In other words, we are creating a filtering *policy* that will contain a *list* of filter *actions* that will be applied to network traffic entering and leaving the system.

The figure shows the IPSec filter actions that are associated with the IPSec filter list we are creating. As you can see the filter allows traffic from any IP address with a destination of the web server with a destination port of HTTP (port 80), HTTPS (443), FTP (port 21), and FTP-DATA (port 20). The mirror rule to the FTP-DATA allows PASV FTP in terms of the FTP server actually initiating the FTP-DATA connection from port 20 to any IP address. By default, all filters are "mirrored," which means that packets with source and destination addresses reversed will also match the filter



After you save all the information in the different configuration panels, the filter actions are stored in the filter list, which is then stored as part of a filter policy. Once you have the filter policy, you can use the IP Security Policies on Local Machine to then assign the policy. To do this you just right-click the policy you want to assign (activate), and select "assign." Once the policy is assigned, it will immediately start filtering packets, no reboot is required.

Note: Interestingly enough, the filtering capabilities that we are using here have nothing to do with IPSec, they just happen to be configured using the IPSec interface. That is to say that you can use the filters without using full IPSec.

Blocking RSVP and Kerberos

By default Win2K allows Kerberos (88) and IKE (500), regardless of the IPSec filters rules established. After SP1, you can change this behavior. You need to create the NoDefualtExempt key in the IPSec service:

Key: HKLM\System\CurrentControlSet\services\ipsec
Data: NoDefaultExempt
Value: 1 (REG_DWORD)

A value of "1" will block RSVP and Kerberos. Thus leaving only IKE, Multicast, and Broadcast exempt. Note: See Microsoft KB article Q254728 for more details.

Tightening TCP/IP

There are a number of TCP/IP settings that can be applied that will increase the robustness of the stack, as well as increase the security level. As with everything we have covered in this chapter, this is not the "total" fix, but another layer on the security onion.

VIP: When you change the registry, you have the potential to make the system unusable. Use a test system to validate your settings before you place them on a production system.

The following parameters are located under HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Tcpip\Parameters:

- **SynAttackProtect** SynAttackProtect is a semi-dynamic way to reduce the time the system will wait for SYN-ACKs, thus protecting itself from a SYN attack. It is a REG_DWORD, with a range of 0–2, (default is 0, recommended is 2). Value of 0 gives no protection; 1 reduces retransmission retries and delays route cache entry; and 2 is just 1 plus a delay indication to Winsock
- **TcpMaxHalfOpen** This determines the number of connections in the SYN-RCVD state allowed before SYN-ATTACK protection begins to operate. It is a REG_DWORD, with a range of 100–0xFFFF

(default is 100 for Win2K Pro and Server and 500 for Advanced Server). You will need to test this in your environment to get a proper value.

- **TcpMaxHalfOpenRetried** This determines the number of connections in the SYN-RCVD state for which there has been at least one retransmission of the SYN sent before SYN-ATTACK protection begins to operate. It is a REG_DWORD, with a range of 80–0xFFFF (default is 80 for Win2K Pro and Server and 400 for Advanced Server). You will need to test this in your environment to get a proper value.
- **PerformRouterDiscovery** This controls whether Win2K will try to perform router discovery (RFC 1256). This is on a per-interface basis. It is located in `Interfaces\<interface>` and is a REG_DWORD, with a range of 0–2, (default is 2 and recommended is 0). Value of 0 is disabled; 1 is enabled; and 2 DHCP controls the setting.
- **EnableICMPRedirect** This controls whether Windows 2000 will alter its route table in response to ICMP redirect message. It is a REG_DWORD, with 0,1 (False, True). Default value is 1, recommended value is 0.
- **KeepAliveTime** This controls how often TCP attempts to verify that an idle connection is still intact by sending a keep-alive packet. If the remote system is still reachable and functioning, it acknowledges the keep-alive transmission. Keep-alive packets are not sent by default. This feature may be enabled on a connection by an application. This is a REG_DWORD with a range of 1–0xFFFFFFFF milliseconds. Default is 7,200,000 (two hours), recommended is 300,000 (5 minutes).

Tidying Up

Now that the majority of work is done, there is still some tidying up to do. This section addresses those issues.

Installing Service Packs and Hotfixes

Once you have installed the OS and installed the applications and services that you will be using on the system, you need to take the time to install any service packs and hotfixes. You should ensure that you install not only the OS-specific service packs and hotfixes, but also those applicable to the applications and services that you have running on the system. But do not install hotfixes for services that you do not have installed! The old adage "If it ain't broke, don't fix it!" is applicable here. You can search for Security Bulletins by particular product at http://www.microsoft.com/technet/security/current.asp.

Removing Unneeded Subsystems

You should remove the OS2 and Posix subsystems from the computer. To do this, you can remove the OS2 and Posix registry values from the `HKLM\System\CurrentControlSet\Control\Session Manager\SubSystems` registry key. Then delete the associated files (`os2*`, `posix*`, and `psx*`) in the DLL cache directory, then from `%systemroot%\System32` (otherwise windows file protection will immediately replace them).

Protecting "Special" Binaries

Many exploits leverage the fact that the LocalSystem account and Local Administrators group have access to basic system utilities. To help reduce the likelihood of a successful exploit, you should create a separate admin group, say ToolsAdmin. Then place the users that you want to use the tools in that group. Change the ACLs on the following tools to "remove" LocalSystem and the Administrators group, and give ToolsAdmin ownership and the ability to Read and Execute. Do this for the following command-line utilities:

```
arp.exe         ipconfig.exe       Nbtstat.exe
at.exe          net.exe            Netstat.exe
atsvc.exe       nslookup.exe       ping.exe
cacls.exe       posix.exe          Qbasic.exe
Cmd.exe         rcp.exe            rdisk.exe
debug.exe       regedit.exe        Regedt32.exe
edit.com        rexec.exe          route.exe
edlin.exe       rsh.exe            Runonce.exe
finger.exe      secfixup.exe       Syskey.exe
ftp.exe         telnet.exe         Tracert.exe
xcopy.exe       tftp.exe           command.com
```

```
clipsrv.exe     dialer.exe          hypertrm.exe
attrib.exe      ping.exe            sysedit.exe
cscript.exe     wscript.exe
```

If you have installed the Windows 2000 Resource Kit, you should perform the same process for the tools in it.

Cleaning Up Anonymous Registry Access

By default, the majority of the registry is secured, but the `Allowed Paths | Machine key` (under `HKLM\SYSTEM\CurrentControlSet\Control\SecurePipeServers\winreg)` has a number of entries that allow the anonymous user too much access. The only real option that you should allow in there by default is the `System\CurrentControlSet\Control\ProductOptions`; all the others should be evaluated and removed unless you *know* that you need them. Don't forget the restrict-anonymous settings.

Other Stuff

Once you have gotten to this point, the system should be fairly secure. However, there are some other settings and precautions that you may want to take:

- Use the encrypting file system (EFS) to encrypt sensitive files.
- Validate permissions on application directories.
- Configure the system to boot immediately into the OS. Select My Computer | Properties | Advanced | Startup and Recovery | System Startup and set the boot time to 0.
- Configure system dumps (on the same screen as previous item).
- Run an integrity-checking software (such as Tripwire) over the final system to get a baseline for later Trojan horse or file corruption detection.
- Event log monitoring software for anomalies

Securing Applications

Now that every path and hotfix that you need is installed, you can tighten down the applications/service the best you can. You should look at the individual chapters for securing the application/service, and if they are not covered in this book, find a book that covers the specific application. A good starting point is the software manufacturer documentation. Other places are things like Usenet news groups and information on the Web.

Testing Security Settings

Once you have the system(s) configured, you will want to test them to see what you can get at from the outside. To do this, you will need some testing tools:

- **Port Scanner** You will need some type of UDP and TCP port scanner. The freeware tool for Win2k and WinNT called Superscan () will work great for tcp port scanning. If you have a Unix system with a compiler close, use NMAP (**http://www.insecure.org**).
- **RPCDump** You will use this tool to help you determine which RPC services have which ports open. This is available from the Microsoft Windows 2000 Resource Kit.
- **Netstat** You will use this tool on the local host to identify its open ports. This comes with Win2K and WinNT.
- **Fport** A great tool from www.foundstone.com to use in the testing

There are literally hundreds of other tools, but these give you a start.

Once you have the tools, then scan the system to see what is open. Hopefully, it is what you think. If not, then use RPCDump and Netstat (on the host) to determine what is running on the ports. As you figure things out, then start turning them off.

Win2K May Still Fall Short

Unless you are hardening a single host, there is a high likelihood that you will be using some type of service that will be relying in some manner on Microsoft Networking (either NetBIOS, SMB/CIFS, or RPC). If this turns out to be the case, then you cannot rely only on Win2K to protect itself; you will have to use other security measures

to isolate those systems from people that you do not intend to access those Microsoft services. In reality, a simple screening router will accomplish the task just fine, but you may choose to have a more full-featured firewall.

Recap

There is really no cookbook answer to the question of how to harden a Win2K system, especially with the vast number of services that can potentially be run on it. This chapter hopefully gave you good insight into the issues surrounding that hardening and made it possible for you to go about hardening your own systems.

There are a few critical things that you should remember:

- Use good user and group security administration practices (that is, no role accounts, administrators have individual admin accounts, and so on).
- Run only minimal services.
- Use a self-contained system if possible.
- Run specific service-related servers, not general multi-service servers.
- Disable Microsoft Networking access from untrusted networks.
- Use strong passwords.
- Do NOT run ANY client based software on the server
- Use the concept of “least privileged” to set security context to accomplish task at hand

If you follow these steps, you will be a long way down the road to security.

Acknowledgements

Thanks to Stephan Norberg, Eric Schultze, and Ken Pfeil for their initial reviews. Osborne/McGraw-Hill for allowing the use of this material.

Change History

1.2 More Updates. Thanks to Derek Seaman for corrections, and Jussi for RSVP and Kerberos filtering pointer

1.1 Updates. Thanks to Alexander Fichman, Jussi Jaakonaho, Steven Cox, Peggy Young, and Nicholas Staff for their corrections, comments, and suggestions.

1.0 Initial release

About SystemExperts Corporation

Founded in 1994, System Experts is the premier provider of network security consulting services. Our consultants are world-renowned authorities who bring to every engagement a unique combination of business experience and technical expertise. We have built an unrivaled reputation by providing practical, effective solutions for securing our clients' enterprise computing infrastructures. Through a full range of consulting services, based on our signature methodologies, we develop network security architectures, perform network penetration testing, develop security policies, and provide emergency response to hacker attacks.

Our consultants are frequent speakers at technical conferences around the world. Our courses on penetration testing, wireless security, secure electronic commerce, intrusion detection, firewalls, sendmail & DNS, and NT/Windows 2000 security at Usenix, SANs, Networld-Interop, CSI, and Internet World are among the most popular and highest rated because our consultants bring years of practical hands-on experience to bear. In addition, our consultants have been technical advisors and on-air guests for CNN, Dateline NBC, and CBS news radio.

Our consultants provide a wide range of services for a diverse customer base in many industries. The list below is a sampling of areas in which we have directed clients.

Security Consulting

Our experts conduct network and host security analyses and penetration tests (sometimes called Tiger Team Attacks) that document the vulnerability of our clients to network penetration. To protect those networks, we develop security architectures, technology plans, policies and procedures, and deploy security systems and software.

Electronic Commerce & WWW Design

Our consultants have been leaders in seminal WWW and electronic commerce initiatives and have helped our clients to use these technologies to gain significant competitive advantage. We help our clients in designing web environments that offer the most value to their customers without putting their internal resources at risk.

Emergency Response

When hackers attack and web sites or critical business resources are compromised, our clients know that we have the expertise and the wherewithal to help them respond to the intrusion in a pragmatic, professional manner. Our emergency response teams quickly assess the situation, properly preserve evidence for use by law enforcement, lock out the attacker, and develop and help implement a plan to quickly regain control of the IT environment. Following the crisis management phase, we work with our clients to help them to understand how the attack happened and the measures necessary to prevent recurrence.

System Management at the "Guru" Level

Sometimes getting the details right is all that counts. Our consultants regularly help our clients to resolve the toughest sendmail, DNS, firewall, networking, and configuration problems in NT, Unix, and heterogeneous environments

Intrusion Detection and Event Management

Few organizations can tell when they are under attack. This gives the attacker the advantage of unlimited time to find and exploit a vulnerability. We have an extensive track record in assisting clients with instrumenting systems to detect anomalous activity and designing log collection, reduction, and event escalation systems that enable our clients to know when they are under attack so they can take appropriate proactive measures.

Technology Strategies and Architectures

Our accomplished architects and engineers with expertise in secure client-server and distributed computing evaluate new technologies, select those best suited to the business needs of our clients, and plan their deployment. We are proud of our outstanding record in developing system management strategies and architectures, benchmarking, and managing software development and deployment projects of all sizes.

To learn more about how SystemExperts can put its expertise to work for you, contact us today at 888 749 9800.

Boston Los Angeles New York San Francisco Tampa Washington DC

www.SystemExperts.com info@SystemExperts.com