Global Information Assurance Certification Paper

Assessing the security of the Windows XP Internet Connection Firewall

GSEC Practical Version 1.3
David Collins

Abstract

In public statements surrounding the release of Windows XP, Microsoft has touted it as a secure operating system for use in homes and small offices, as well as in large corporations. A key security feature of Windows XP is the *Internet Connection Firewall* (ICF), which provides a protective boundary between a computer and the Internet. Since ICF is targeted for home and small office computers, it is unlikely to be installed or configured by security professionals. Thus it is important for the security community to provide guidance to users as to the level of security provided by ICF, and how it compares to alternative "personal firewall" products.

This paper describes an empirical evaluation comparing ICF with ZoneAlarm Pro, a popular workstation firewall. The evaluation consisted of running several simulated attacks against both products installed on a test network. The results of the tests lead to the conclusion that ICF, while not as effective as ZoneAlarm, is easy to use and has value as part of a "defense in depth" security strategy.

Introduction

"A key reason for the early success of Windows XP is its incredibly strong security -- it has literally hundreds of security improvements over Windows 98, making it the most secure Windows operating system ever."[1] This quote from Microsoft chairman Bill Gates emphasizes two things: Windows XP security is a selling point, and (by the reference to Windows 98) Windows XP security is specifically intended to benefit home and small office users. The second point is reinforced by this statement from the XP online help: "Internet Connection Firewall (ICF) is firewall software that is used to set restrictions on what information is communicated from your home or small office network to and from the Internet . . ."[2]

It is now common for home and small office users to have an "always on" broadband connection to the Internet, through DSL or cable modem. In order to provide access to the network connection point from multiple computers, a router is needed. Broadband routers from vendors such as LinkSys may provide built-in firewall capabilities, which can be used in place of, or in addition to, a

[1] Quoted in Microsoft 2001c.
[2] "Internet Connection Firewall overview," in Microsoft 2001a.

personal firewall installed on each workstation. I did not test this option.[3] The assumption in the tests here is that there is no additional network security besides the personal firewall (ICF or ZoneAlarm).

Internet Connection Firewall (ICF)

The Windows XP Internet Connection Firewall (ICF) is a component of XP that provides a barrier between a computer running XP, and the Internet. By default, it is disabled; it must be enabled for each connection accessing the Internet. So, for example, if several dial connections are defined for Internet access, ICF must be enabled, and possibly configured, separately for each connection. Though Microsoft does not recommend using ICF for local LANs (because it blocks file and printer sharing), it seems to function identically for local and Internet traffic.

The essential function of ICF is to block all unsolicited inbound traffic from the Internet. ICF is a stateful firewall, i.e., it maintains state in the form of a table for all Internet access originating from the XP computer. It then examines incoming packets, allowing them only if they are part of a session originating in the XP computer.[4] For example, an HTTP packet arriving at the computer will be checked against the table to see if it matches a browser request previously sent—if so, it is allowed through; if not, it is dropped (and optionally logged).

Though the default is to drop all unsolicited inbound traffic, the user can specify the existence of services on the XP computer which modify the default. For example, by default an unsolicited packet destined for TCP port 80 (HTTP) will be dropped; but the user can specify that the computer is running a web server, in which case it is accepted. Services are defined in the settings dialog for ICF, as shown in figure 1. On the left side of the figure, the user can simply check off common services being run on the XP computer. On the right is a dialog popped up from the “Add…” button, where the user can define additional services beyond those in the standard list.

The user can also specify that certain kinds of unsolicited ICMP packets should be allowed through the firewall. This is also defined in the ICF settings dialog, as shown in figure 2. For example, by default, incoming ICMP echo (ping) packets will be blocked (dropped by the firewall). If “Allow incoming echo request” is checked, the firewall allows the packets through, and the system can respond to a ping.

ICF differs from ZoneAlarm in this selective permission by service type: ICF allows any program to receive traffic on a permitted port. ZoneAlarm

[3] For more information on this option, as well as why one might want to use both a router/firewall and a personal firewall, see McCabe.

[4] For connectionless protocols, such as UDP, ICF uses heuristic methods to match outgoing and incoming packets.

permissions are based on a combination of program identity, and a permitted (or denied) port list for each program.

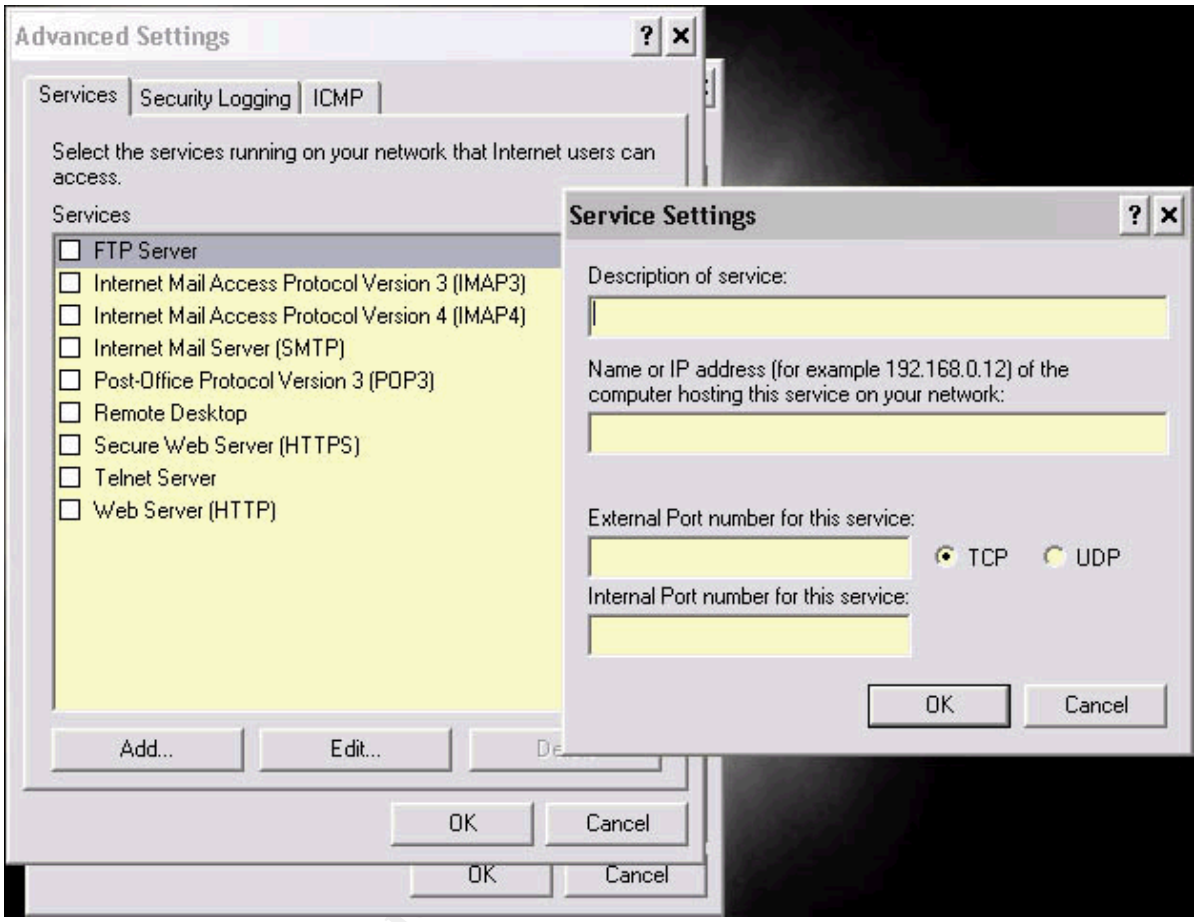

Figure 1 ICF “Services“ settings

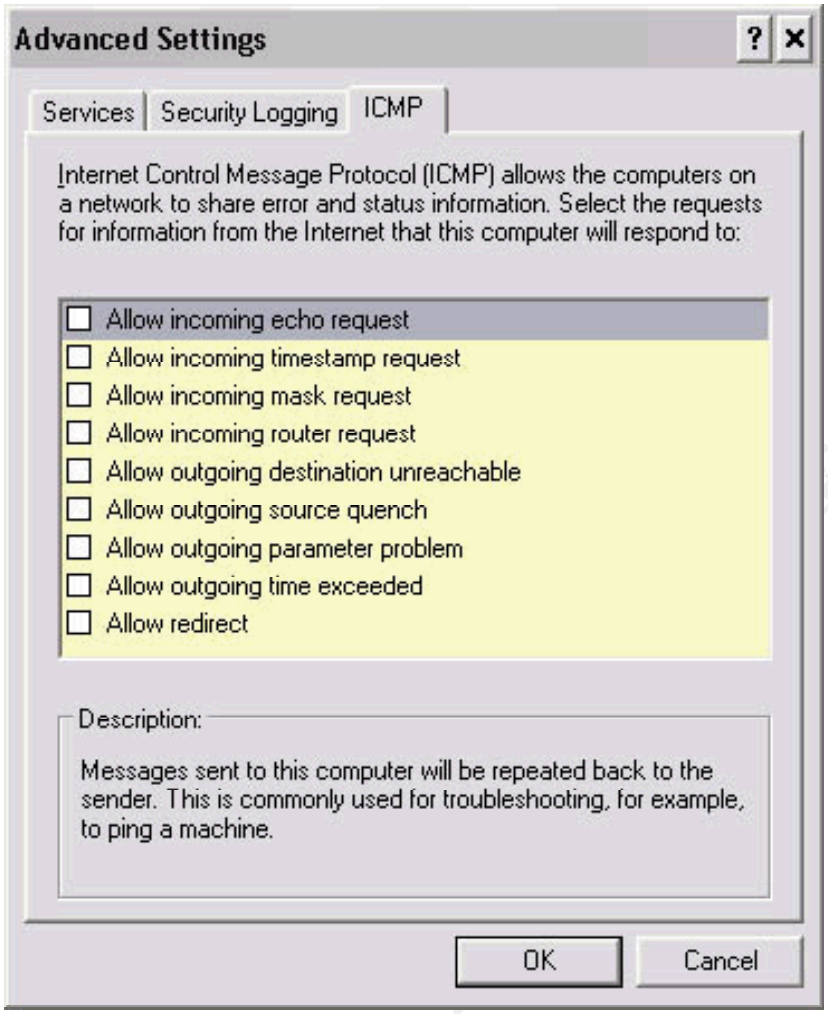

Figure 2 ICF "ICMP" settings

Though the user documentation shipped with XP does not cover this, ICF will also drop obviously malformed packets (e.g., packets with invalid flag combinations), or outbound packets with spoofed (forged) source IP addresses.[5] This capability is not foolproof, however, as the "outbound spoofed ping" test revealed (see below under **Test Results**).

Another topic not covered in the user documentation is the existence of an API that allows programs to modify ICF's port configuration.[6] The rationale for this, apparently, is to allow applications such as networked games and file-sharing programs to circumvent ICF restrictions that would prevent them from operating. It was beyond the scope of this paper to investigate the API, but it seems to be a fruitful target for exploits.

[5] Morgan, pp. 3-4.
[6] Morgan, p. 8.

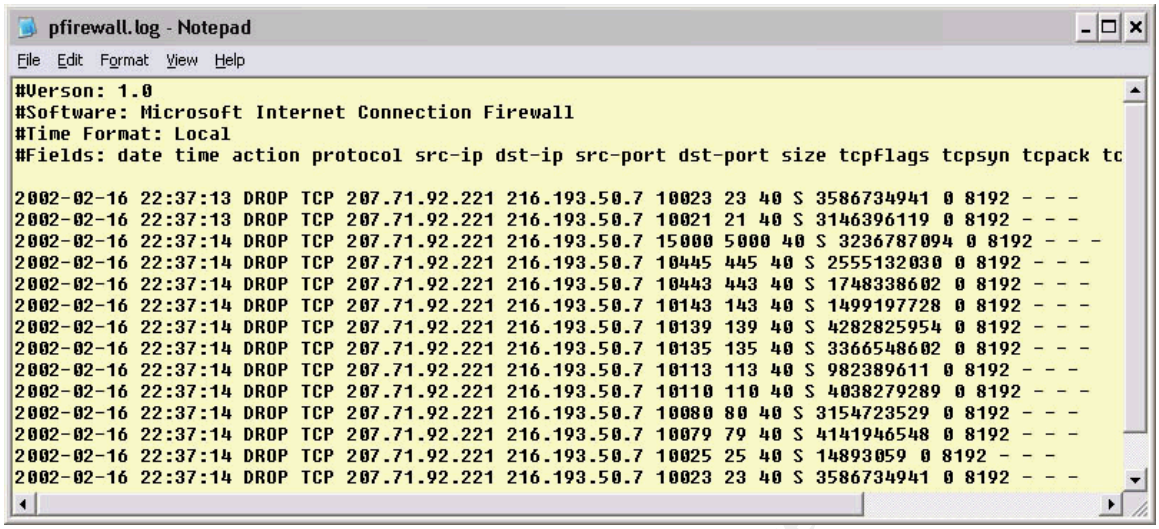

Figure 3 ICF log entries

In addition to blocking prohibited traffic, ICF logs certain events if the appropriate options are checked on the “Security logging” tab of the ICF settings dialog. (The default is no logging.)

- Dropped packets: A “DROP” line is created in the log file with information on each incoming packet that was dropped by the firewall. Outbound packets may also be dropped, if they contain a spoofed origin IP address.
- Successful outbound connections: “OPEN” and “CLOSE” lines are logged when a TCP or UDP socket for outbound traffic successfully connects or is closed. (A socket opened to listen for incoming connections, as in the “Listener” test described below, is not logged.)

No user interface is provided for the log file, other than browsing the entries (see figure 3). The file is written in the W3C extended log file Format[7], and thus could be processed by existing tools designed for that format.

ZoneAlarm “personal firewall”

As a benchmark for comparison to ICF, I chose ZoneAlarm[8], a popular "personal firewall”.[9] A personal firewall protects an individual workstation by

[7] Hallam-Baker.
[8] CNET 2000c, Zone Labs 2001. For Windows XP, I used the latest version of ZoneAlarm Pro (2.6, at the time this paper was written). I found that it was important to *completely* uninstall an older version of ZoneAlarm before installing 2.6 on XP, according to the instructions at http://www.zonelabs.com/services/support_install_XP.htm.
[9] See McDougall for an overview of personal firewalls.

examining inbound (and sometimes outbound) TCP and UDP packets, blocking those that appear to be malicious. Blocking is accomplished based on default rules and user configuration.

ZoneAlarm is primarily a stateful packet-filtering firewall, but it has some awareness of the application level. It maintains a program list and then filters traffic based on the configuration defined in the list for the program attempting to open a TCP socket.

Program list elements contain the name and location of the program, and rules defining what the program is allowed to do: send packets on the “local zone” or “Internet zone”, and listen for packets on the local or Internet zone. (Whenever the computer connects to a network that has not been seen before, ZoneAlarm will query the user as to whether it is “trusted”; if the answer is yes, the network becomes part of the local zone.)

The default for a program allowed to act as a server (listen for incoming traffic) is that it can accept traffic on any port; but the user can alternatively define a list of permitted ports, or a list of denied ports, for each program.

The list also stores an MD5 hash for each program; this defeats attacks in which an identically named malicious program is substituted for a trusted one. ZoneAlarm issues a “Changed program” alert if a program attempting to access the network has changed (has a different hash) since its last access.

The list may be configured explicitly, by adding programs and defining rules. Typically, the list is dynamically configured over time: As programs attempt to access the network, ZoneAlarm suspends the access while asking the user for permission (see figure 4). If the user checks “Remember this answer”, the program is added to the program list.

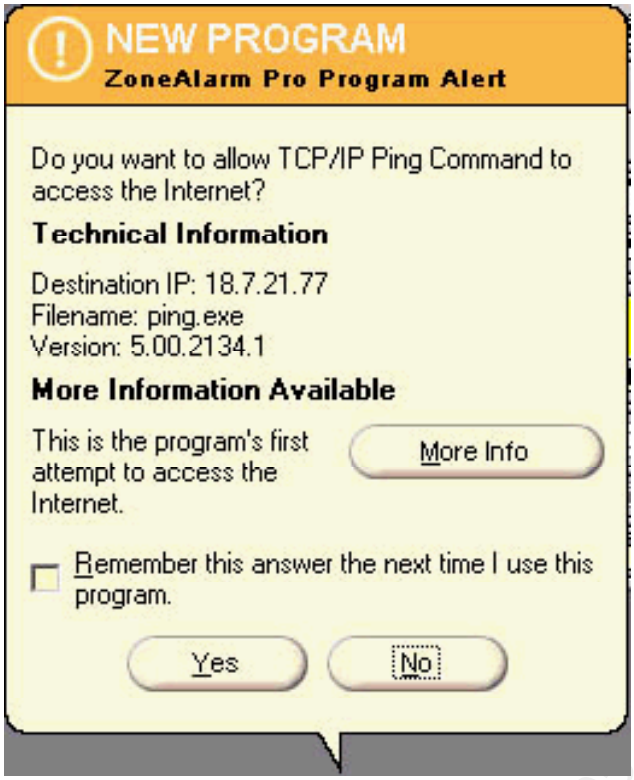

Figure 4 ZoneAlarm alert for a new program connecting to the Internet

Though ZoneAlarm's dynamic configuration via program alerts is easier than coding rules, it is not foolproof. In the course of ordinary activities such as web browsing, the user will see alerts for Windows programs with names like "Generic Host Process for Win32 Services" and "LSA Executable and Server DLL". Without doing some technical research, it is impossible to tell whether these programs are legitimate or are malicious modules named to appear innocent. There is a temptation for the user to simply allow access by any program that requests it.

Beyond the program list, ZoneAlarm has additional configurability: The user can specify whether fragmented packets should be blocked, whether to accept incoming ping packets, etc. Most of these options can be set in blocks by selecting from high, medium, or low security for a given "Zone" (see below for a description of zones). On high security (the recommended setting for Internet connections), ZoneAlarm assumes that programs in the list are trusted to the extent specified by the user and blocks everything else, both inbound and outbound.

Specific IP addresses (or ranges of addresses) and host names can be added to "Zones." All traffic from hosts in the "Restricted Zone" will be unconditionally blocked. Hosts in the "Local Zone" (normally other computers on a LAN) by

default operate in “minimal security“ mode.

ZoneAlarm also has an application-level feature called “MailSafe”, which quarantines (by changing the file extension) certain file types sent as e-mail attachments.

ZoneAlarm logs events in a file, similar to the log file used by ICF. Three types of events are logged:

- PE: A program attempted to access the network and was blocked.
- FWIN: An incoming packet was blocked.
- FWOUT: An outgoing packet was blocked.

Logging is on by default, but may be turned off. Whether the user looks at the log is less important with ZoneAlarm, since it produces popup alerts for loggable events. (See figures 4, 6 and 7 for examples.) The user could turn off both the log and the popups, however.

Testing setup and tools

To test ICF and ZoneAlarm under “simulated battlefield conditions,” the test setup shown below (figure 5) was used. The test driver machine was running Windows 2000, along with various tools. It could “attack” the Windows XP machine either via the Internet (using a separate connection for each machine, through different ISPs) or over an Ethernet LAN. The LAN mode simulated either an insider attack, or an attack from a compromised computer infected with malicious code as a result of a prior attack.

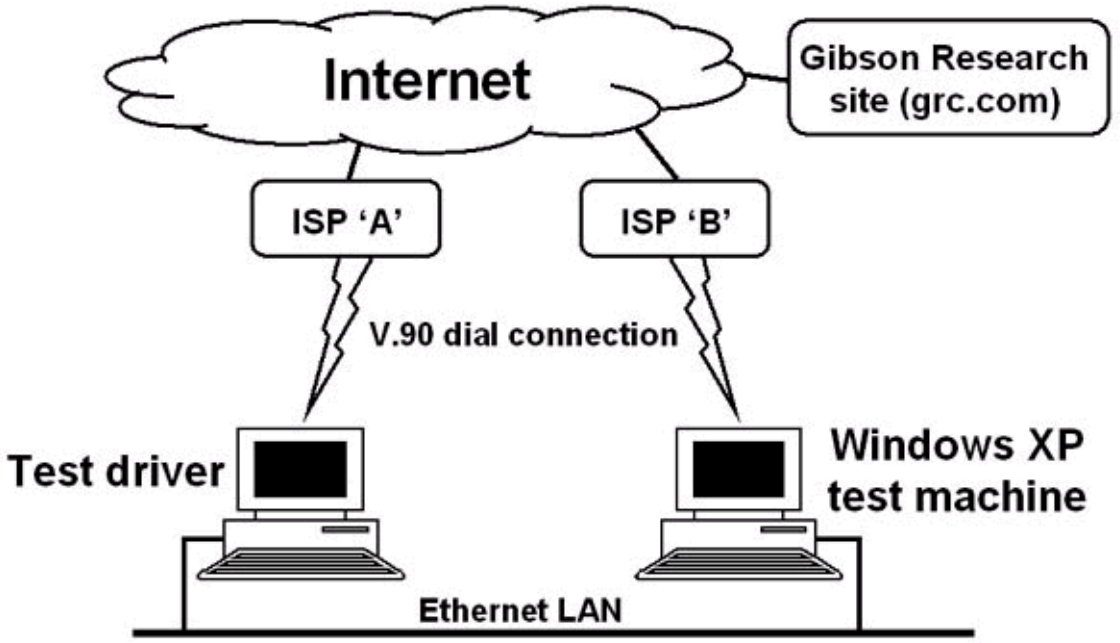

Figure 5 Network topology used for testing

Each machine had two IP addresses: an Internet IP address associated with its RAS (remote access) adapter and a LAN IP address associated with its Ethernet adapter.[10]

Three general types of test were run, to assess firewall response to behavior simulating common attack scenarios:[11]

1. *Tests simulating an attack based on a port scan or malicious port access:* Port scans are a common first step in hacker attacks, either to assess vulnerabilities in the target system, or to locate computers infected with malicious code used to take control and launch further attacks. Accessing an open TCP/IP port is the vector for many attacks; in Windows systems, attempting to access the NetBIOS ports (137-139) is common.[12]

2. *Tests simulating malicious outbound traffic originating on the test machine:* This would be symptomatic, for example, of a computer

[10] At any given time, the addresses for a given machine can be determined using the Windows 'ipconfig' command.
[11] For information on attack types see Houle, Honeynet 2000, Honeynet 2001.
[12] In Windows XP Microsoft is discouraging the use of the older Netbuei protocol for LAN file and printer sharing. It is not an option in the default XP install, though it can be added later, using a semi-documented procedure. This forces the use of "NetBIOS over TCP" for LAN file sharing, increasing the likelihood of exposing ports 137-139 on the Internet.

infected with a malicious program being used in a distributed denial-of-service attack. The origin address in the sent packets might be forged, and the sending program might be a Trojan masquerading as (or concealed in) an innocent program.

3. *Tests simulating the opening of a port to accept inbound connections:* This would be symptomatic of a computer infected with a malicious program as in (2), where the program listens on a TCP or UDP port in order to receive instructions to activate an attack on a third party.

Both types of tests were run with traffic on the local LAN and the Internet.

The following tools were used to execute these tests:

- GRC (Gibson Research Corporation), at http://www.grc.com, offers tools for executing the first two of the general scenarios described above.[13]

  ShieldsUP!! Attempts to connect from GRC's site to common TCP/IP ports on the machine being tested[14] and reports the results.

  LeakTest is a small program, downloaded from grc.com, that attempts to connect to port 80 (HTTP) at grc.com from the machine being tested, then reports whether it was successful.

- *CommView* is a commercially available packet capture utility ("packet sniffer") for Windows.[15] It also has a packet generator function, which can be used to send packets; the sent packets may be completely handcrafted, or modifications of captured packets. The packet generator can be used for IP spoofing, i.e., sending packets with forged source IP addresses.

- *MingSweeper* is a "network reconnaissance tool"[16] which does various types of port scans, ping sweeps (locating active hosts within a range of network addresses), and operating system fingerprinting. For these tests, I only used the port scanning capability. This is similar to what is done by ShieldsUP!!, but is more flexible: arbitrary port ranges can be scanned and various combinations of TCP flags may be set. If the target of the scan responds to a ping, this test is a superset of the ShieldsUP!! test. However, if the target does not respond to ping, MingSweeper does no further scanning. ShieldsUP!! scans all the ports on its list regardless of

[13] Gibson.
[14] Ports tested are 21 (FTP), 23 (Telnet), 25 (SMTP), 79 (Finger), 80 (HTTP), 110 (POP3), 113 (IDENT), 135 (RPC), 139 (NetBIOS), 143 (IMAP), 443 (HTTPS), 445 (MSFT DS), and 5000 (UPnP).
[15] TamoSoft.
[16] Jones.

ping response.

- *Legion*[17] is a scanning program that searches for open NetBIOS shares. This is a common type of attack against computers running Windows. If an open share is found, Legion will attempt to connect to it; if the connection is successful, files on the target machine can be accessed directly.

- *Listener* is a small program I wrote in C specifically to test scenario (3) above.[18] This program, run from a command prompt, simply opens a TCP socket to listen on a specified port. E.g.,

  ```
  listener 12345
  ```

  listens on port 12345. Listening on a high-numbered port is typical of the behavior of "Trojan horse" programs used in denial-of-service and other types of attack.

The following specific series of tests uses these tools to implement the general test scenarios described above. There is some redundancy in the tests, which was deliberate, in order to insure that each firewall was thoroughly exercised. Each individual test was run twice, once with ZoneAlarm enabled and ICF disabled, then again with ICF enabled and ZoneAlarm disabled. Numbers in parentheses after each test identify the general scenario being tested:

1. *GRC ShieldsUP!!:* The Windows XP test machine accessed the grc.com web site, and ran ShieldsUP!! (1)
2. *GRC LeakTest:* After downloading the LeakTest program from grc.com, the test machine ran LeakTest while connected to the Internet. (2)
3. *Inbound Internet port scan:* Using MingSweeper, the test machine scanned ports 1-65,535 on the Internet address of the Windows XP machine (i.e., the scan was done via the Internet). The scan type used was TCP SYN. (1)
4. *Inbound Internet NetBIOS scan:* Using Legion, the test machine scanned for open NetBIOS shares on the Internet address of the Windows XP machine. (1)
5. *Inbound LAN port scan:* Same as the inbound Internet port scan, but using the LAN IP address of the Windows XP machine (i.e., the scan was done on the local area network). (1)
6. *Inbound LAN NetBIOS scan:* Same as the inbound Internet NetBios scan, but done via the LAN. (1)
7. *Outbound Internet spoofed ping:* CommView was used to send a spoofed ping (ping with a forged source IP address) from the Windows XP machine to the test machine's Internet IP address. (2)
8. *Outbound LAN spoofed ping:* Same as the Internet spoofed ping, but the

[17] Rhino9.
[18] For detailed information on socket programming in C, see Hall.

target was the test machine via its IP address on the LAN. (2)

9. *Outbound Internet port scan:* MingSweeper was used to run a port scan from the Windows XP machine, directed at the test machine via its Internet IP address. (2)
10. *Outbound Internet NetBIOS scan:* Legion was used to run a NetBios scan from the Windows XP machine, directed at the test machine via its Internet IP address. (2)
11. *Outbound LAN port scan:* Same as the outbound Internet port scan, but directed at the test machine's LAN IP address.
12. *Outbound LAN NetBIOS scan:* same as the outbound Internet NetBIOS scan, but directed at the test machine's LAN IP address.
13. *Listener:* The Listener program attempted to open a socket to listen on a high-numbered TCP port.

Test results

The following table (table 1) shows the results of the tests. Notes below the table provide additional explanation of certain test results.

Test	ZoneAlarm Pro	Microsoft ICF
1.GRC ShieldsUp!	Blocked (all tested ports in "stealth mode," i.e., invisible to the network).	Blocked (all tested ports in stealth mode).
2. GRC Leak Test	Outbound connection attempt detected and blocked pending user approval (note 1).	Outbound connection attempt logged, but not blocked (note 6).
3. Inbound Internet port scan	Blocked (no response to ping of target).	Blocked (no response to ping of target). If incoming ping is allowed, all ports are still blocked.
4. Inbound Internet NetBIOS scan	Blocked (no open shares found).	Blocked (no open shares found).
5. Inbound LAN port scan	Ports 135, 139, 445, and 6789 open with default LAN security setting. All ports in stealth mode on high security setting (note 2).	All ports in stealth mode with ICF enabled.
6. Inbound LAN NetBIOS scan	All open shares can be connected with default LAN security setting. No shares are visible with high security setting.	No shares are visible with ICF enabled.

7. Outbound Internet spoofed ping	Not detected (note 3).	Not detected (note 7).
8. Outbound LAN spoofed ping	Not detected (note 3).	Not detected (note 7).
9. Outbound Internet port scan	Blocked (note 4).	Logged, but not blocked (note 6).
10. Outbound Internet NetBIOS scan	Outbound connection attempt detected and blocked pending user approval (notes 1, 4).	Logged, but not blocked (note 6).
11. Outbound LAN port scan	Not detected with default LAN security setting. Same as test 9 result with high security setting (note 4).	Logged, but not blocked (note 6).
12. Outbound LAN NetBIOS scan	Outbound connection attempt detected and blocked pending user approval (notes 1, 4).	Logged, but not blocked (note 6).
13. Listener	Blocked pending user approval (note 5).	Not detected.

Table 1 Test results

Note 1: Figure 4 shows an example of the alert displayed by ZoneAlarm when a new program (LeakTest, in this case) attempts to connect to the Internet. Unless the user clicks "OK", it is blocked.

Note 2: Port 135 is used for communication with Microsoft Exchange servers;[19] apparently it is opened by default even if it is not needed. Port 139 is the well-known NetBIOS port. Port 445 is used by Windows for SMB (server message block) traffic related to file and printer sharing if "NetBIOS over TCP/IP" is enabled. Use of this port, in addition to 139, started in Windows 2000.[20] Port 6789 is used by many different software packages; in my case, by using the netstat command and selectively killing processes, I established that it was being used by the IBM DB2 database manager[21] to provide remote access to JDBC (Java database connection) clients.

Any of these ports could be the target of an exploit, of course. The fact that they are open (and not blocked by ZoneAlarm) on the LAN is normal, given the default ZoneAlarm setting of "low" for LAN security. If LAN security is set to "high", these and all other ports are blocked and do not respond in any way; inbound ping is also blocked, so the computer is in "stealth" mode, i.e., a port

[19] Microsoft 1999.
[20] Microsoft 2001D.
[21] DB2 was installed on the Windows XP machine for reasons unrelated to my firewall testing.

scan cannot detect its existence.

Note 3: Not only was the spoofed ping not detected, ZoneAlarm did not detect CommView sending a *valid* ping (this is done by using the CommView packet capture to capture the outgoing ping packet and resend it). My conclusion is that this is because ZoneAlarm is monitoring "standard" packets sent using Winsock, the normal Windows socket interface. CommView (like other packet capture programs) installs its own Windows packet driver in order to be able to operate in promiscuous mode (capturing all packets, regardless of MAC address). Apparently, ZoneAlarm cannot monitor the activity of custom packet drivers.[22] Note 4 has more information on ZoneAlarm's ability to detect outgoing packets.

Note 4: The outbound port scan resulted in an alert like figure 6 for each port that MingSweeper attempted to connect with. This was unexpected; if I used the Telnet program from a command line, for example, I received an alert like the one shown in figure 4, which identified the *program* attempting to connect. I expected a similar alert identifying the MingSweeper program. The alert in figure 6 indicated that ZoneAlarm recognized the packet going out and blocked it because it could not identify what program it was sent by.

[22] This raises a topic for further research: whether ZoneAlarm, or other personal firewalls, will detect spoofed packets sent through the "raw sockets" capability introduced in Windows XP. See Gibson, "Windows XP Home Edition Must be Made More Secure," for a discussion of the problem. See also Cyrano de Bergerac, where he points out correctly that tools already exist to do raw sockects on *any* Windows version (CommView is an example). ICF apparently will detect spoofed packets (see note 7), but only if they are generated using the Winsock interface with the IP_HDRINCL option.

Figure 6 ZoneAlarm firewall alert

Zone Labs provides no technical details on TrueVector, an automatically started Windows service that performs the actual monitoring ZoneAlarm is based on. The patent covering TrueVector[23] states only that it operates by "intercepting process loading and unloading," "intercepting certain file activity," and "intercepting and interpreting all TCP/IP communication". It also provides a brief description of a possible implementation: "In the instance of Windows Winsock communication driver, for example, a process can hook into the driver using Winsock VxD extensions."

Apparently, ZoneAlarm can detect that a program is about to access the Internet by monitoring its loading and then its access to the Winsock DLL. It also seems to monitor packet activity through the Winsock TCP/IP stack. It is clear from my testing that the monitoring is not 100% successful: in tests 10 and 12, it detected the program (Legion); in tests 9 and 11, it detected the packets but not the associated program initialization; in tests 7 and 8 it detected nothing, presumably because CommView does not use Winsock.

Note 5: Figure 7 shows the server program alert displayed when Listener was started. This is similar to figure 4, but indicates the program is accepting inbound connections (acting as a server), versus sending outbound traffic.

[23] Zone Labs 1999.

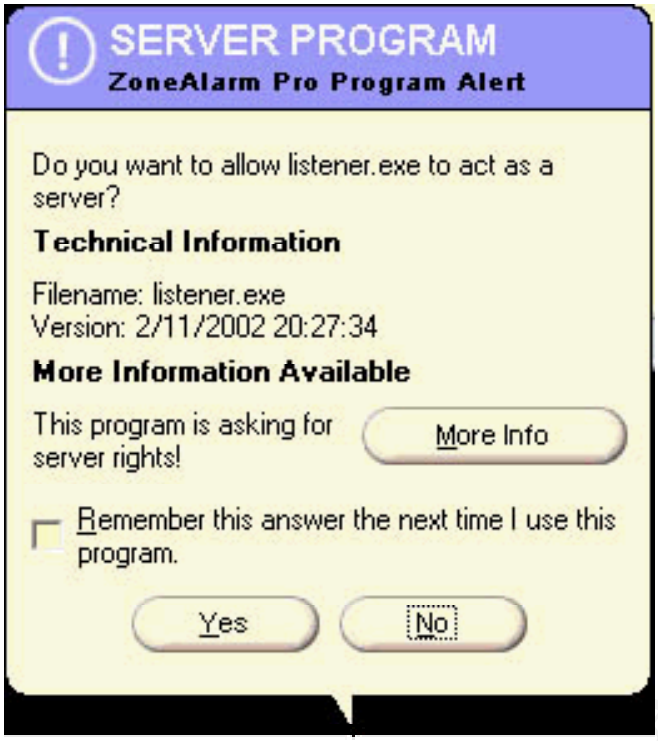

Figure 7 ZoneAlarm server program alert

Note 6: In these cases, the ICF log (if enabled) shows OPEN and CLOSE events for each port being scanned. It does not block the traffic, nor does it provide any information that would enable the scanning program to be identified. Some scanners might be blocked, if they send packets with invalid flag combinations.

Note 7: As with ZoneAlarm, there are no events logged for CommView, since it is operating "under the radar" by using its own packet driver. Thus the capability of ICF to detect spoofed packets works only if the packets are sent through Winsock using the IP_HDRINCL option.[24]

Summary and conclusion

My testing indicates that ICF provides less security than ZoneAlarm. However, this is not inconsistent with Microsoft's objective: "With Windows XP, Microsoft's goal is to provide a simple and unobtrusive security experience."[25] In order to achieve simplicity, they have provided a facility that will, in most environments, not have any effect on normal operations and not require any configuration.

[24] Morgan, p. 4.
[25] Morgan. P. 1.

ZoneAlarm, in providing a higher and more configurable level of security, places a greater burden on the user. There are many tailorable options, though reasonable defaults are provided. In the initial period after ZoneAlarm is installed, the user must decide whether each program that generates an alert should be allowed to access the network.

ICF has value in two contexts:

- *To provide a basic level of security for users who would otherwise have no protection:* Many, if not most, home and small office users will not purchase and install a firewall until they are attacked and suffer damage. ICF, by virtue of being free and simple, will provide significant protection against many common threats.
- *To provide an additional security layer at no cost:* ICF can be run along with a firewall such as ZoneAlarm, and provides additional protection (for example, against malicious code that might disable ZoneAlarm[26]).

Of course, neither ICF nor ZoneAlarm is enough by itself. The most common cause of damage to home and small office systems is attack by viruses or worms carried in e-mail or on diskettes; no firewall will protect against this threat. One might hope that Microsoft would enhance future releases with a basic-level antivirus capability to go along with ICF.

Ideally, users will recognize the value of defense in depth:

- If a broadband Internet connection is used for multiple computers, purchase a router with firewall capabilities. Insure that the router is configured to remove obvious security holes, such as default remote-access administrator passwords.
- Periodically scan all workstations for known security problems and vulnerabilities. Microsoft's free, easy to use "Personal Security Advisor,"[27] accessed via the Internet, will scan a user's workstation and report on such items as missing security patches, weak passwords, and security options set inappropriately.
- Install and use a competent antivirus software package such as Norton or McAfee.
- Enable XP's Internet Connection Firewall with the default security settings.
- If the user is willing to invest additional time and money in a higher level of security, consider a personal firewall such as ZoneAlarm for all

[26] Landesman.

[27] Microsoft 2002a. More thorough vulnerability scanners, such as the Cerberus Internet Scanner (Cerberus Information Security, Ltd.), are less appropriate for home or small office users, because interpreting their results requires more skill.

computers accessible from the Internet.

This list of steps will not “bulletproof“ a networked workstation, but it will result in a significant improvement in security, in return for a relatively small investment of time and knowledge.

Motivating users to provide themselves with better security would be easier if vendors would do more to make it simple. ZoneAlarm, for example, would be much easier to use if it came with installable default configurations tailored to various environments. Both ZoneAlarm and ICF produce logs that could be the basis for intrusion detection analysis, but no tools are provided to help the user—as a result, most logs will never be looked at.

ICF represents a small step in the right direction. Let us hope that more such steps will help keep the average user secure from the bad guys.

References

Cerberus Information Security, Ltd. “Cerberus Internet Scanner.” No date. http://www.cerberus-infosec.co.uk/cis.shtml (February 4,2002).

CNET 2000a. “BlackICE Defender 2.1 – CNET Review.” CNET Software. August 1, 2000. http://www.cnet.com/software/0-806183-7-2342113.html (February 4, 2002).

CNET 2000b. “Norton Personal Firewall 2000 – CNET Review.” CNET Software. August 1, 2000. http://www.cnet.com/software/0-352108-7-2342100.html?st.sw.352108-7-2342097.txt.352108-7-2342100 (February 4, 2002).

CNET 2000c. “Zone Labs ZoneAlarm 2.1: Win9X/NT4/2K – CNET Review.” CNET Software. August 1, 2000. http://www.cnet.com/software/0-352108-1205-5045346.html?tag=st.sw.352108-1204-5045346.rev-rev.352108-1205-5045346 (February 4, 2002).

Coburn, Justin. “XP - The Future of Secure Operating Systems?” SANS Institute Information Security Reading Room. November 20, 2001. http://rr.sans.org/win/XP.php (February 2, 2002).

Cyrano de Bergerac. “Dissecting Steve Gibson GRC DoS Page.” GRCSucks.com. No date. http://grcsucks.com/grcdos.htm (February 16, 2002).

Gibson, Steve. “LeakTest.” Gibson Research Corporation. 2002.

http://grc.com/lt/leaktest.htm (February 8, 2002).

Gibson, Steve. "ShieldsUP!!" Gibson Research Corporation. 2001. https://grc.com/x/ne.dll?bh0bkyd2 (February 8, 2002).

Gibson, Steve. "Windows XP Home Edition Must be Made More Secure." Gibson Research Corporation. August 31, 2001. http://grc.com/dos/sockettome.htm (February 16, 2002).

Hall, Brian "Beej". *Beej's Guide to Network Programming Using Internet Sockets* (Version 2.3.1). October 8, 2001. Download in PDF format from http://www.ecst.csuchico.edu/~beej/guide/net/ (February 11, 2002).

Honeynet 2000. "Know Your Enemy: The Tools and Methodologies of the Script Kiddie." The Honeynet Project. July 21, 2000. http://project.honeynet.org/papers/enemy/ (February 7, 2002).

Honeynet 2001. "Know Your Enemy: Statistics." The Honeynet Project. July 22, 2001. http://project.honeynet.org/papers/stats/ (February 7, 2002).

Houle, Kevin J., and George M. Weaver. *Trends in Denial of Service Attack Technology* (v1.0). October 2001. CERT Coordination Center. Download in PDF format from http://www.cert.org/archive/pdf/DoS_trends.pdf (February 7, 2002).

Jones, Greg. "MingSweeper." HoobieNet. August 23, 2001. http://www.hoobie.net/mingsweeper/index.html (February 11, 2002).

Landesman, Mary. "How Vulnerable is Your Security? New breed of worms targeting scanners, firewalls." December 13, 2001. http://antivirus.about.com/library/weekly/aa121301d.htm (February 17, 2002).

McCabe, Mike. "Cable/DSL Router and Personal Firewall: Belt and Suspenders?" SANS Institute Information Security Reading Room. February 14, 2001. http://rr.sans.org/homeoffice/cable.php (February 5,2002).

McDougall, Bonnie. "Personal Firewalls - Protecting the Home Internet User." SANS Institute Information Security Reading Room. August 17, 2001. http://rr.sans.org/win/XP_firewall.php (February 4,2002).

Microsoft 1999. "XGEN: TCP Ports and Microsoft Exchange: In-depth Discussion (Q176466)." Microsoft Product Support Services. September 2, 1999. http://support.microsoft.com/default.aspx?scid=kb;EN-US;q176466 (February 12, 2002).

Microsoft 2001a. Windows XP Professional (online help, ICF topics). Redmond: Microsoft Corporation, 2001.

Microsoft 2001b. “Windows XP to Take the PC to New Heights.” Microsoft PressPass. August 24, 2001. http://www.microsoft.com/presspass/press/2001/Aug01/08-24WinXPRTMPR.asp (February 2, 2002).

Microsoft 2001c. “Gates Showcases Tablet PC, Xbox at COMDEX; Says New ‘Digital Decade’ Technologies Will Transform How We Live.” Microsoft PressPass. November 11,2001. http://www.microsoft.com/PressPass/press/2001/Nov01/11-11Comdex2001KeynotePR.asp (February 2, 2002).

Microsoft 2001d. “Direct Hosting of SMB Over TCP/IP (Q204279).” Microsoft Product Support Services. December 22, 2001. http://support.microsoft.com/default.aspx?scid=kb;EN-US;q204279 (February 12, 2002).

Microsoft 2002a. “Microsoft Personal Security Advisor.” Microsoft Corporation web site. January 28, 2002. http://www.microsoft.com/technet/mpsa/start.asp (February 4, 2002).

Microsoft 2002b. MCSE Training Kit—Windows XP Professional. Redmond: Microsoft Press, 2002. 635-642.

Morgan, Dennis. “Internet Connection Firewall Feature Overview.” Redmond: Microsoft Corporation, 2001. Download in MS Word format from http://www.microsoft.com/windowsxp/pro/techinfo/planning/firewall/icf.doc (February 17, 2002).

Rhino9. “Legion.” 1998. http://rhino9.ml.org. (As of February 11, 2002, this site is no longer operational. The Legion program can still be downloaded from http://packetstorm.widexs.nl/groups/rhino9/.)

Smith, Gary. “A Brief Taxonomy of Firewalls - Great Walls of Fire.” SANS Institute Information Security Reading Room. May 18, 2001. http://rr.sans.org/firewall/taxonomy.php (February 5,2002).

Snitchler, Matt. “Introduction to the Microsoft Windows XP Firewall.” SANS Institute Information Security Reading Room. August 13, 2001. http://rr.sans.org/win/XP_firewall.php (February 2,2002).

TamoSoft. “CommView: See what’s inside.” TamoSoft, Inc., 2001. http://www.tamos.com/products/commview/ (February 8, 2002).

Hallam-Baker, Phillip M., and Brian Behlendorf. “Extended Log File Format - W3C Working Draft WD-logfile-960323.” Worldwide Web Consortium. March 3,

1996. http://www.w3.org/TR/WD-logfile.html (February 17, 2002).

Zone Labs 1999. “United States Patent 5,987,611: System and methodology for managing internet access on a per application basis for client computers connected to the internet.” Assigned to Zone Labs, Inc. by Gregor Freund. November 16, 1999. Available online from the U. S. Patent office at http://www.uspto.gov/main/patents.htm (February 16, 2002).

Zone Labs 2001. *Zone Alarm User Manual*. December 29, 2001. Download in PDF format from http://www.zonelabs.com/services/support_za_zap.htm (February 5, 2002).