



ICSA Labs Firewall Testing An In Depth Analysis

by

**Jack Walsh
Senior Security Engineer
ICSA Labs, a division of TruSecure Corporation**

June 10, 2004

ICSA Labs Firewall Testing - An In Depth Analysis

Executive Summary

The purpose of this paper is to discuss how a firewall that has been tested and certified by ICSA Labs offers a greater degree of security assurance when compared to one that has not been subjected to our rigorous and thorough testing. ICSA Labs testing ensures that firewalls provide the highest level of security and are capable of providing the maximum level of logging when necessary.

Commercial entities and government organizations understand the difference between an ICSA Labs Certified firewall and one that has not been certified. As a result these corporations and agencies insist on ICSA Labs Certified firewall products. Firewall purchasing and corporate security policy decisions should therefore include steps to confirm that a particular firewall has attained and retained the ICSA Labs Certified firewall rating.

The ICSA Labs Firewall Certification Program focuses primarily on providing assurance through robust testing and analysis. The testing is performed against an industry-approved set of criteria that evolves appropriately to encompass current and future firewall functionality. A similarly evolving testing methodology keeps pace with security vulnerability discoveries and ensures the criteria requirements are met. The result is a dynamic testing program that provides ever greater assurance that a deployed ICSA Labs Certified firewall continues to protect your network and safeguard your information assets.

When the ICSA Labs Firewall Certification Criteria is not met the firewall vendor must make all necessary changes in order to attain or retain ICSA Labs Firewall Certification. Without exception, all products tested against criteria version 4.0 – the most current version – have had issues that required the vendor to make changes to the product. A full one half of the identified criteria violations were security, functional, or administrative in nature. Nearly one half were logging related.

Were it not for our testing, both seasoned firewall industry veterans and newcomers alike would be susceptible to attacks ranging from Denial of Service to FTP Bounce to insufficient TCP header inspection. Likewise, they would fail to log events that range from both common and unusual types of service traffic to successful and failed authentication attempts to service traffic arriving at the firewall itself. ICSA Labs has found issues in each of the firewalls we've tested. It is therefore reasonable to conclude that firewall products not tested in our labs are susceptible to the same quantity and kinds of vulnerabilities.

To illustrate the value of ICSA Labs firewall testing, representative details involved in our firewall testing are presented in this paper. Therefore the paper is intended primarily for a technical audience. Personnel within an organization that assist in the procurement of firewalls by performing technical analysis will benefit most from this paper's contents. Additionally, 'C' level management¹ responsible for the security and well being of the organization will find that the paper depicts the benefits and significance of ICSA Labs firewall testing, and why the diligent course of action is to purchase only ICSA Labs Certified security products.

¹ 'C' level management: CEO, CFO, COO, CIO, CISCO, CTO.

Introduction

For almost a decade ICSA Labs has been the security industry's leading firewall certification body. ICSA Labs, in conjunction with firewall vendors, industry experts, and end users, sets the standard by which firewall products are measured. With a list of certified firewall products that represents over 90 percent of the installed base of firewall products globally, ICSA Labs is recognized as the de facto standard for firewall testing and certification requirements.

The purpose of this paper is to discuss how a firewall that has been tested and certified by ICSA Labs offers a greater degree of security assurance as compared to one that has not been tested in our labs. We will show, by discussing criteria violations and the way that we test for and discover those violations, why many corporations and government agencies demand ICSA Labs certification before they consider purchasing a firewall product.

With the exception of Antarctica, vendors from every continent have submitted products to ICSA Labs for testing against the ICSA Labs Firewall Certification Criteria. Through this paper we intend to show the value and robust nature of the ICSA Labs Firewall Testing and Certification Program.

Is The Latest Firewall Version Any Better?

These days it seems that there are endless recommendations to apply a patch or service pack, make a configuration change, or upgrade a firewall. One has to wonder if all of this is really necessary. Hasn't the firewall vendor ironed out all of the security problems yet? Who keeps finding problems with these firewalls?

The firewall testing team at ICSA Labs continuously looks for and frequently finds new ways to circumvent firewalls. When useful new attacks, vulnerabilities, and techniques are discovered or learned they are incorporated into the testing methodology. As a result, there is both an increase in assurance gained and an increase in the rigor of the testing performed.

These testing methodology additions often occur though our criteria remains unchanged. That is not to say that the criteria is static. In fact the opposite is true. Like the testing methodology, the ICSA Labs Firewall Certification Criteria² is dynamic. ICSA Labs updates the criteria – “raises the bar” if you will – for the firewall industry on a regular basis. When new criteria requirements are added, corresponding and appropriate additions are made to the testing methodology.

When issues discovered during ICSA Labs firewall testing are addressed the changes result in improvements to firewall products. Because important issues are being found and resolved through ICSA Labs testing, it is in the best interest of security and network administrators to review the release notes and keep up with patches.

Criteria Violations

When a product does not meet one or more criteria requirements, we say that the product has “violated the criteria”. In order to become ICSA LABS CERTIFIED, or to retain an existing certification, new and previously certified firewall products must meet all the requirements in the

² Version 4.0 of the criteria is available online at:

http://www.icsalabs.com/html/communities/firewalls/certification/criteria/criteria_4.0.shtml

most current set of criteria. All products that have been tested against version 4.0 of the ICSA Labs Firewall Certification Criteria have had one or more criteria violations. If and when criteria violations are found during testing, the firewall vendor must make changes to the product such that it no longer violates the criteria in order to attain or retain certification.

The latest vetted and in use version of the ICSA Labs Firewall Certification Criteria is version 4.0.³ *Figure 1* below depicts the categories of version 4.0 criteria violations that firewall products were required to correct prior to attaining certification. Also included in the figure are the percentages of the occurrences. Note that data on products that either failed or that are currently being tested is not included.

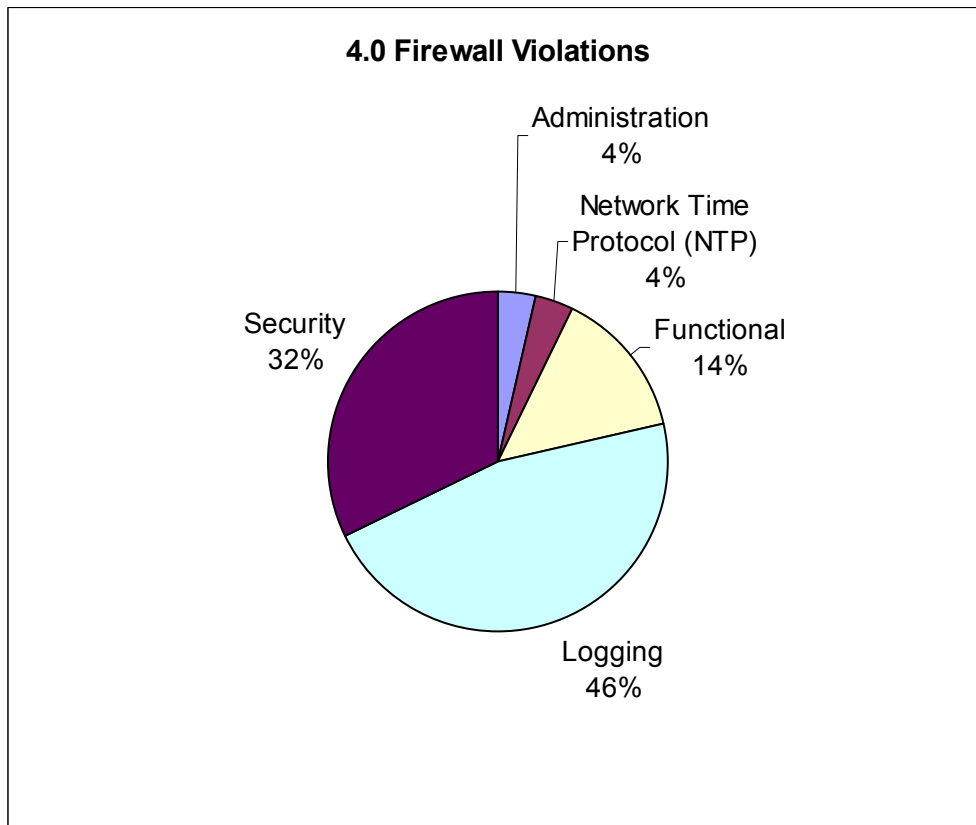


Figure 1.

Logging criteria violations have been most prevalent. They account for 46% of all 4.0 criteria violations. *Figure 2* below depicts the main categories of version 4.0 logging criteria violations. Also depicted is the percentage of firewalls that the testing team initially identified with these problems.

³ Version 4.1 is expected to be released during the second quarter of 2004.

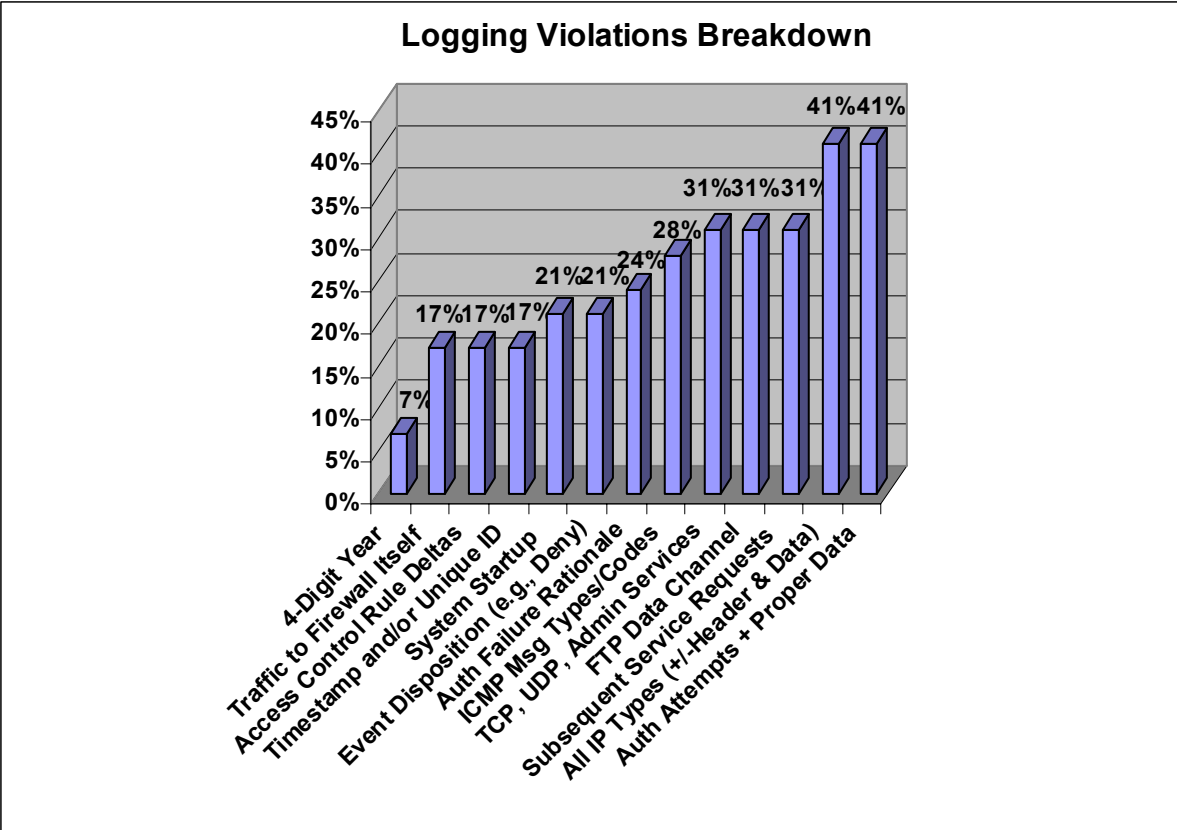


Figure 2.

Security criteria violations that compromise the firewall, violate its security policy, or cause a loss of service account for 32% of all version 4.0 criteria violations. *Figure 3* below depicts the main categories of security vulnerabilities found while testing firewalls at ICSA Labs. Also depicted is the percentage of firewalls that ICSA Labs initially identified with these problems.

Trailing behind both logging and security criteria violations are the number of functional criteria violations found during ICSA Labs firewall testing. They result when the firewall fails to provide the ability to block some class of not-permitted traffic. Logging, security, and functional violations collectively represent over 90% of the criteria violations found during testing.

Comprising less than 10% of the violations were those related to administration and Network Time Protocol (NTP). From this nearly 10%, two examples of the types of things found are as follows. The testing team has been able to interrupt the boot sequence and bypass authentication on some firewalls in order to access administrative functions. Also the testing team has seen products that allow access to administrative functions after accepting a partially correct password.

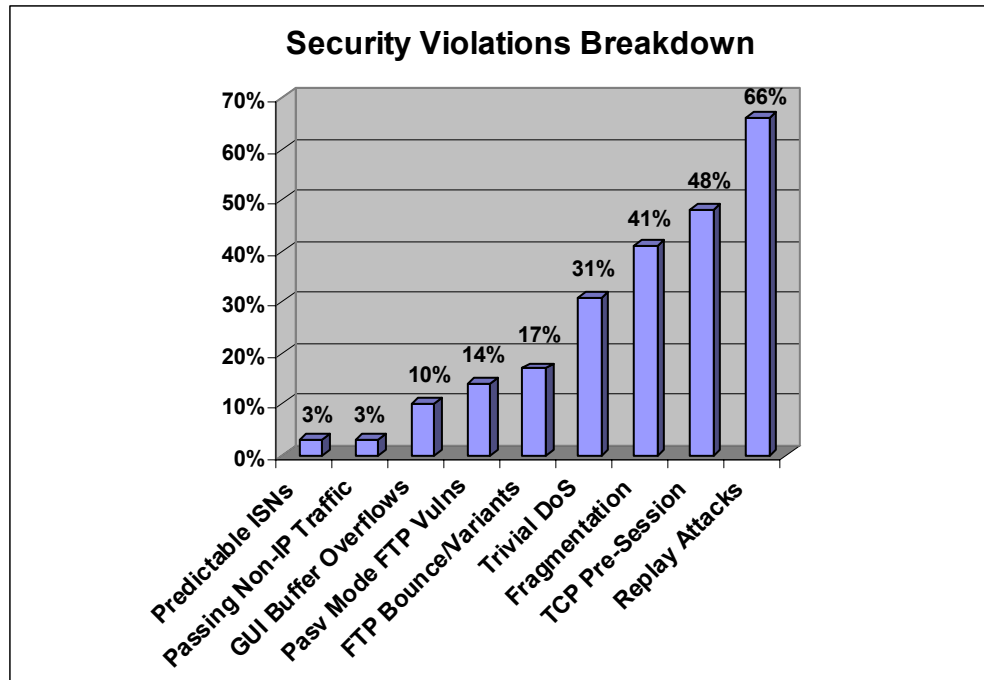


Figure 3.

The sections that follow explore a sample set of the most prevalent logging and security violations found. They include detailed discussions of the testing methodology used for finding some of these violations.

Surprise! Your Firewall Doesn't Log That Stuff

We have all heard that security was an afterthought in the design of many computer operating systems and networking devices. In many network security devices, including firewalls, logging has suffered a similar lack of attention. ICSA Labs has worked over the years to improve the logging capabilities of the firewall products we test.

Historically, logging has not always been a part of the ICSA Labs Firewall Certification Criteria. ICSA Labs began to test logging 6 years ago as it became increasingly clear that accurate, detailed, and reliable logs for both tracking criminal behavior and diagnosing network anomalies were extremely important. At that time the tests for logging were limited to verifying that a sample of permitted TCP or UDP services, when passed through the product, were logged with the correct data. Under criteria version 3.0a, the requirements and testing were expanded to examine a matrix of both permitted and denied TCP and UDP services.

Since the advent of 3.0a testing several years ago, the firewall criteria and log testing methodology have come a long way. The testing done today ensures that firewalls have the capability to log all permitted and denied traffic (not just TCP and UDP), all successful and failed authentication attempts, and other important events.

With regards to the logging criteria requirements, several important aspects bear mentioning. First, not every packet belonging to a permitted session needs to be logged. Only the connection attempt needs to be logged (i.e., the initial packet). Second, *all* traffic dropped or

denied by the firewall where the firewall is either between the source and destination or is itself the destination must be logged. Third, logging does not have to always be enabled in order to meet the criteria. Fourth, the criteria does not demand that logging be enabled by default since administrators have differing logging needs and since enabling the high level of logging required by the criteria may not be productive as a default.

Firewall testing ensures that products include the *capability* to enable the comprehensive level of logging required by the criteria. Thus, network and security administrators using ICSA Labs Certified firewalls have the functionality and level of logging at their disposal in the event that it is necessary. Below we explore some of the required log events that firewalls have frequently not had the ability to log.

All IP Types (+/-Header & Data)

Though the logging of events such as access control, rule changes and administrator authentication are required by the criteria, log testing is largely concerned with permitted and denied traffic. As far as logging traffic permitted by the security policy products do comparatively well. However, few firewalls actually capture all dropped or denied traffic the first time through testing.

A frequent criteria violation occurs where firewalls either fail to log one or more types of traffic destined for the firewall itself or they fail to log blocked attempts to send one or more types of traffic through the firewall. Often firewalls fail to log both of these sets of events. The type of blocked traffic they fail to log varies from TCP and UDP on various high-and-low ports, to many ICMP message types. Firewalls also do not typically log all of the 256 types of IP traffic.

Logging blocked IP datagrams for some IP traffic types but not others is a common occurrence. For example, firewalls that also provide IPsec functionality often will log dropped ESP (IP protocol 50) and AH (IP protocol 51) traffic while ignoring many, if not all, of the other IP protocol types. Others may log dropped TCP, UDP, and ICMP packets unless the headers and data aren't quite right. For example, firewalls receiving a handcrafted IP datagram where the IP protocol is set to 6 (indicating TCP) may fail to log it when various other header parameter values do not make sense.

One of the more unusual IP logging criteria violations that the firewall testing team discovered was that for blocked IP traffic a particular firewall correctly logged the first 128 IP protocol types (0-127), but it then incorrectly logged types 128-255. And when no data was attached to the datagrams, the product logged none of the 256 IP protocol types.

ICSA Labs runs through an extensive battery of logging tests that includes common and unusual protocol types. Though most firewalls initially do not log all dropped traffic belonging to any of the 256 IP types, customers of ICSA Labs Certified firewalls can be confident that their products have the capability to capture this and all blocked service traffic.

Subsequent Service Requests

Certified firewalls usually have not had trouble logging repeated connection attempts for various TCP services. For example, suppose logging is enabled for permitted traffic, and Telnet is a service permitted by the firewall. Each time one "telnets" from the same source to the same destination the connection attempt is logged. Regardless of the number and frequency of

separate Telnet sessions between the same source host and destination Telnet server, the repeated events are logged on firewalls we test.

More frequently we have seen firewalls failing to log second and subsequent UDP requests. We have primarily seen this between the same source and destination DNS servers. Unlike TCP clients that use a different ephemeral source port for each new connection attempt, DNS servers are often configured to always send UDP traffic from *source port 53* to destination port 53 when requesting resolution by other DNS servers. Firewalls have frequently failed to log the second and subsequent DNS requests in such cases in part because the same, non-ephemeral source port is used in the request.

Arguments have been put forth by a few vendors that these subsequent DNS requests between the same two servers belong to the same UDP “session” and should therefore not require distinct log entries. We at ICSA Labs do not accept this rationale. We do not consider subsequent DNS requests and responses between the same two servers to be part of the same “session”. Instead, the ICSA Labs firewall testing team views a query followed by its response as a complete UDP DNS “session”.

Since the information in the DNS query is repeated in the DNS response, it is clearly possible for firewalls to monitor DNS request-response pairings. Additionally helpful is the DNS header’s QR flag, which distinguishes between a query and a response. Firewall products that fail to log second and subsequent DNS requests between the same servers often do not track these request-response pairings. Instead these firewalls may decide what comprises a UDP DNS “session” based on what DNS traffic between the same two servers transpired prior to the expiration of a global UDP timer.

In some cases manually setting such a global UDP session timer to zero (‘0’) has resolved the problem. While in other cases the timer could not be modified or could not be set to ‘0’. Even if the timer could be set to ‘0’, doing so may be less than desirable. Setting it to ‘0’ may cause all permitted UDP traffic – not just DNS – to be logged. For example, if in addition to DNS another UDP service such as IPsec-related IKE traffic was passed through the firewall, then the firewall would capture much more than the desired initial DNS connection attempt. This may unnecessarily fill the logs. Rather than logging each separate DNS resolution request as the criteria intended, the firewall ends up logging all other permitted UDP traffic as well.

Strictly speaking such a solution meets the letter of the criteria. However, the resulting lab report may present the vendor’s solution in a way that alerts end users of these potentially undesirable effects. Solving the problem through the use of DNS header and DNS message inspection is much cleaner. Though it does involve significantly more processing and inspection, it neither adversely impacts logging nor the “session” state table data of other UDP protocols.

ICSA Labs firewall testing has found that the level of logging for particular UDP services such as DNS is often a function of the means by which a firewall vendor tracks a UDP “session”. Unless firewalls perform deeper packet inspection and have knowledge of the UDP services being passed through them, logging permitted UDP service requests will rely primarily on techniques that often include the use of timers. Such reliance may make it difficult if not impossible for a firewall administrator to tune the level of logging appropriately.

FTP Data Channel

FTP is an interesting protocol in that there are two channels. There is a control connection, and a separate data connection established every time information is transferred between a client and server. The FTP client can specify whether the data connection is in “active” or “passive” mode. For “active” mode the data connection originates at the server, while for “passive” mode the connection originates at the client. Some firewalls have a difficult time logging either or both types of FTP data connections.

Some firewalls have logged the data connection but have mixed up the source and destination ports and IP addresses. Still others did not log the connection at all. On several occasions, the firewall log indicated that both active and passive mode data connections were logged when in fact that was not the case.

In one such instance, separate active and passive mode data connections appeared to be logged correctly except that the source port for the passive mode data connection was reported as ‘0’. We found it interesting that only the source port for passive mode was wrong, so we investigated further. We manually sent properly configured FTP PORT and PASV commands through the firewall. It turned out that upon seeing these FTP commands the firewall logged them as if they themselves were data connection attempts. In other words, the product was logging the data connection before it ever occurred.

But how was the data in the logs correct for active mode and only narrowly incorrect for passive mode? For active mode, the firewall knew the source IP would be that of the FTP server and that the source port for active mode is almost always ‘20’. The firewall also used the data in the FTP PORT command that specifies the destination port and destination IP address for the active mode FTP data connection. Because these were all known, the firewall was able to correctly construct the active mode connection in the log even before it happened – or for that matter even if it never happened.

Since the value of the passive mode source port is unknown until the server proposes it the firewall had no means to correctly identify it. Therefore the firewall put in a ‘0’ for the source port after seeing the FTP PASV command. That was what led us to finding that neither the active mode nor passive mode data connections were actually logged.

The ICSA Labs firewall testing team has found that firewalls log permitted traffic reasonably well if the service uses only a single connection. Services like FTP that use separate, secondary connections are often initially overlooked by firewall vendors. ICSA Labs Certified firewalls ensure that all permitted initial and subsequent secondary connection attempts for services like FTP are logged.

Are You Sure Your Firewall Stops These Attacks?

When a firewall is susceptible to attack it is frequently the result of either a failure to properly enforce a particular security policy, or a failure to safely allocate its resources. The firewall testing team at ICSA Labs has seen both. Examples of these include passing formerly valid packets and Denial of Service (DoS) attacks. Each is discussed below.

Replay Attacks

A replay attack occurs when a datagram permitted to pass once through the firewall is captured by an attacker and then repeatedly sent through the firewall. Firewalls fail to enforce the security policy when they permit a stream of once-valid packets to pass continuously through them. The repercussions are that replay attacks consume valuable network bandwidth and may deny access to hosts or services. One of the replay attacks that ICSA Labs tests for involves capturing and replaying once-valid ICMP traffic.

Prior to launching the attack, the firewall product is first configured⁴ to enforce a particular security policy. This security policy allows a handful of common services inbound and outbound through the firewall while dropping or denying all other traffic. While enforcing this security policy no ICMP traffic is explicitly permitted through the firewall.

ICSA Labs realizes that preventing all ICMP traffic from passing through the firewall is not always a useful stance in the real world. For example, suppose a router checks its routing table after having received a packet. Realizing there is a better route available to a particular destination, the router forwards the packet and sends an ICMP redirect back to the sender alerting it of the better route. Unfortunately the sender will never receive the redirect if the intervening firewall is configured to *drop* ICMP datagrams. The sender continues to go to that router even though there is a better route.⁵

Another instance where one may not want their firewall unconditionally dropping all ICMP messages involves hosts using path MTU discovery to avoid the security or efficiency issues associated with fragmentation. An intervening router with a smaller MTU sends an ICMP message back to the sending host telling it that fragmentation is needed but that the “don’t fragment” bit is set. If there is a firewall between the sender and the destination *dropping* this ICMP message, then the sending host may be prevented from communicating with a particular destination.⁶

ICMP error messages are only returned after having been elicited by some other packet. In fact, the use of specific ICMP packets for error message responses to various TCP and UDP packets are explained in sections 3.2.2, 4.1.3.3, and 4.2.3.9 of RFC 1122. A good firewall is intelligent enough to realize that these ICMP error response messages are actually part of an already ongoing TCP or UDP “session”⁷. Because of the network utility lost without these ICMP messages and because they are recommended in RFC 1122, the firewall testing team permits a single ICMP error message response packet when warranted.

It is important to point out that allowing a single ICMP error message response packet when warranted is not the same as explicitly allowing ICMP error messages through the firewall at all times or even just allowing a single one at some arbitrary time. Keep in mind that there is still no ICMP traffic explicitly allowed by the firewall security policy. An ICMP packet that passes

⁴ Unlike firewalls tested against the Small/Medium Business and Corporate criteria modules, firewall products tested against and meeting the Residential module of the criteria do not require configuration of the rulebase. Also services are not allowed inbound in the Residential module of the criteria.

⁵ An additional not-so-nice side effect is that during the course of the session(s) the firewall is basically bombarded with ICMP redirects from the router!

⁶ There are work-arounds for this problem, assuming you can diagnose it, including downgrading the sending host’s MTU to an appropriately lower level.

⁷ Though UDP is a connectionless protocol, a virtual session can be maintained and tracked by a firewall through various means.

through the firewall is only allowed through by virtue of it being recognized as part of an existing TCP or UDP session. ICMP packets being part of an existing TCP or UDP session only make sense in the context of the information presented in the cited sections of RFC 1122.

Having said this, one shouldn't be able to capture an ICMP error message packet and replay it through the firewall even a second time. However, during firewall testing the ICSA Labs firewall testing team has captured and repeatedly replayed all sorts of once-valid ICMP error messages through a variety of firewalls. The easiest solution for some firewall vendors – though not the most useful – has been to simply drop or deny all ICMP traffic at all times.

The ICSA Labs firewall testing team has suggested what it believes to be a more useful alternative to this solution. Every ICMP destination unreachable, redirect, time exceeded, and parameter problem includes the original IP header of the packet that elicited it. Since the IP header's identification field is predominantly unique for every packet with an IP header, the one included in the data portion of the ICMP error response packet uniquely corresponds to that of the single TCP or UDP packet from which it was generated. Therefore, once a firewall sees and decides to pass an ICMP packet whose data includes the same identification field value as that of a formerly valid TCP or UDP packet, the firewall should never allow the same ICMP packet through again.

Some firewall vendors contend that allowing a small handful of these packets through the firewall or allowing them through for a short period of time should be permitted. After all, doing so is far different than passing through the firewall an infinite or significantly large number of replayed packets. Though this is true, ICSA Labs believes it would be a mistake to equate vulnerability to some as yet unknown attack with a certain number of illegal packets. For example, recall that after receiving less than 100 IP datagrams, Cisco IOS routers, switches, and line cards stopped processing IPv4 traffic in July 2003.⁸ Therefore ICSA Labs will continue to require firewall products to block once-valid, replayed packets.

Trivial DoS Attacks

Denial of Service (DoS) attacks – both distributed (DDoS) and otherwise – are essentially attacks on resources. Well-planned and executed DDoS attacks that consume all available network bandwidth or all of a firewall product's resources have caused significant problems to public organizations in the not-so-distant past. There is little a firewall can do when the bandwidth is gone. Unlike many DDoS attacks, firewalls can protect against the class of "trivial" DoS attacks. We have been surprised by how effective older, trivial DoS attacks have been against the current generation of firewall products.

Years ago ICSA Labs relied more on the DoS attacks in vulnerability scanning tools like Nessus, CyberCop, ISS, etc. Over the past several years these tools are no longer the primary source for the DoS attacks used by the firewall testing team. Though we do continue to run the vulnerability scanners to provide additional testing assurance, today, the firewall testing team performs "hands-on" DoS testing. To do this we obtain or create the DoS source code. We then compile the code with any necessary modifications. Modifications help ensure that vendors address the root of what allowed a particular DoS to be successful, rather than

⁸ This worked as long as they were IP protocol types 53, 55, or 77 having TTL values of 0 or 1, or IP protocol type 103 having any TTL value. For more refer to <http://www.cisco.com/warp/public/707/cisco-sa-20030717-blocked.shtml>. Also see CERT advisory CA-2003-15.

engineering the product to be impervious to a specific form of the attack. These DoS attacks are then separately launched in all directions to and through the firewall.

When the testing team began conducting DoS attacks by hand, questions arose as to how we would determine a failure and how we would guarantee that network bandwidth consumption wasn't the cause of the failure. To avoid a criteria violation the testing team determined that during a DoS attack the firewall had to continue passing permitted traffic through the product in both directions, while enforcing the security policy and being administered from the primary administrative interface. Rate limiting the traffic from the attacking machine to 1.54 Mbit/s – the ideal, if not the actual rate one would get with a T1 connection – ensured that bandwidth utilization was not a factor.

ICSA Labs firewall testing of DoS attacks attempts to stress the resources on the firewall under test rather than stressing the network bandwidth. And even with our modest expectations, firewall products – large and small, market-established and new-to-market – struggle to defend against DoS attacks including synflood, jolt2 and others.

Traffic You Thought Your Firewall Could Drop

Firewall vendors can be compared not only by how well they protect networks but also based on other factors including their level of throughput and overall performance. In their efforts to eliminate performance barriers and attain “unparalleled performance” compared to their competitors, some firewall vendors have taken interesting approaches. To that end we have seen firewalls passing packets through without the proper level of inspection. Examples of this include an inability to block both invalid TCP packets prior to session establishment and all fragmented IP datagrams. Both are discussed below.

TCP Pre-Session Establishment

TCP clients and servers such as those sending and receiving HTTP and SMTP require a complete 3-way handshake prior to accepting additional traffic. A TCP connection is not established until after a SYN from the client, followed by a SYN/ACK from the server, and a final ACK from the client. *Figure 4* shows this exchange.

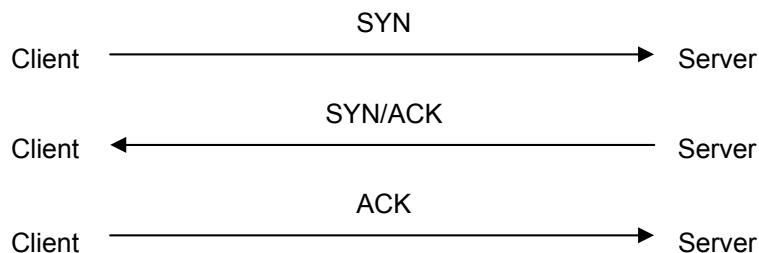


Figure 4.

Prior to session establishment, we have been able to sneak TCP packets past firewalls as long as the source and destination IP addresses, source and destination ports, and service are permitted by the rulebase. What has been permitted through firewalls prior to session establishment has varied greatly depending on just how much attention the firewall is paying to the TCP header during connection establishment.

ICSA Labs first saw this problem several years ago. An initial set of firewalls passed crafted TCP packets through them with a wide variety of flags set even before an initial valid SYN packet. Another group of firewalls passed crafted TCP packets with invalid flags before a completed 3-way handshake, but only after first crafting and sending a TCP SYN packet. These two scenarios are depicted in *Figure 5* below. Note in the second scenario that the SYN/ACK is sent by the server's TCP/IP stack and the RST is sent by the client's TCP/IP stack, while the SYN and PSH/URG are crafted by ICSA Labs.

Passed the offending PSH/URG packet even *before* a SYN packet:



Passed the offending PSH/URG packet only *after* a SYN packet but before any final ACK establishes a connection:

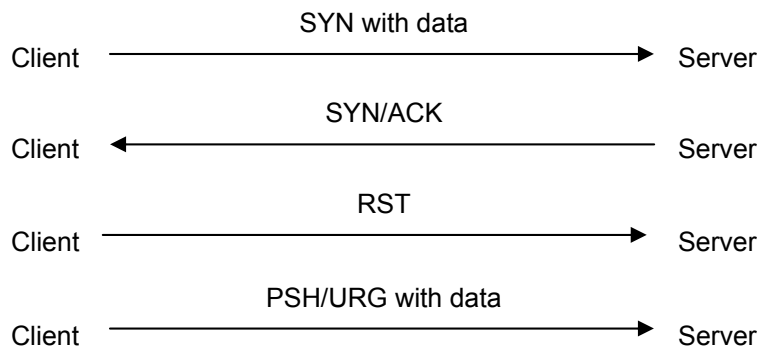


Figure 5.

Following the initial SYN crafted and sent from the client but prior to session establishment, still another group of firewalls were unwilling to pass the second, flag-filled packet crafted and sent by ICSA Labs from the client. This was due to the fact that the client effectively tipped off this group of firewalls before we could craft and send the second packet. The details for this follow below.

The initial SYN, generated artificially with a traffic generation tool, was sent to the server. The SYN/ACK response was sent back to the client from the listening server. The client then, not expecting the server's SYN/ACK, sent a RST to the server. The RST packet was the tip off to the firewall. This resetting of the connection was enough to effectively convince the firewall not to accept the subsequent PSH/URG or similar packets. *Figure 6* depicts this situation where the server is an FTP server.

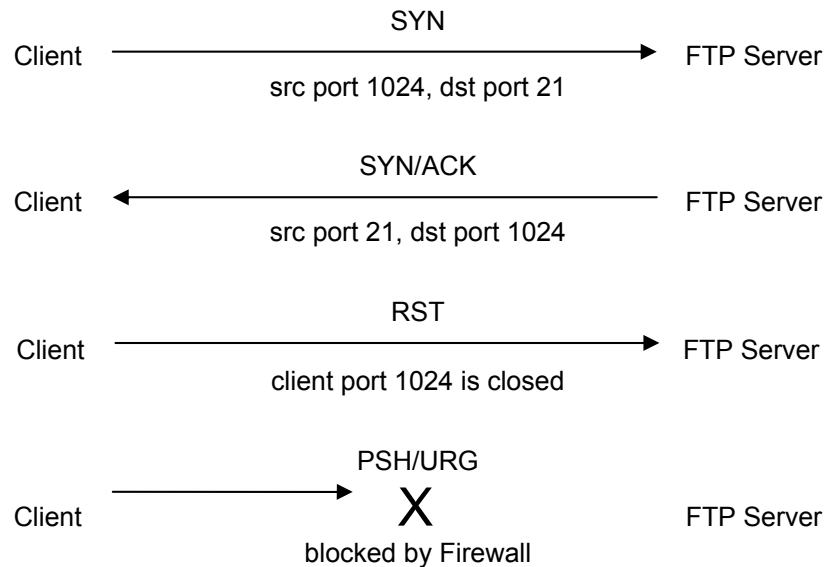


Figure 6.

To prevent the RST packet from interfering, we crafted and sent the TCP SYN from a spoofed source IP address. Because the SYN/ACK returned from the server was sent to an IP address that did not exist (i.e., the spoofed client address) no RST was generated and returned to the server. As a result we were often able to craft and send the subsequent PSH/URG packets (or similar ones) through this group of firewalls from the same spoofed client IP address.

These examples not only show the level of ICISA Labs testing, they also indicate that firewalls submitted for testing vary greatly in the level of TCP header inspection that they perform. When criteria violations like the ones explained in this section are found and eliminated, firewalls certified by ICISA Labs then properly decide what traffic should and should not pass through them. As an added bonus, ICISA Labs tested and certified firewalls also more closely follow the IETF RFCs.

Fragmentation

Not all network administrators permit fragments on their networks due to the inefficiency and associated security risks they pose. Version 4.0 of the ICISA Labs Firewall Certification Criteria requires that firewall products have a means of either blocking all fragments or reassembling them at the firewall. Even fragments of traffic permitted by the security policy have to be dropped by or reassembled at the firewall.

As with logging and other criteria elements, the fragmentation criteria requirement demands that firewalls have the *capability* to behave this way. This behavior need not be the default, nor the only way that a firewall handles fragmentation. Firewall vendors that are attempting to adhere to the RFCs, including RFC 1812, may want to have configuration options to both drop fragments per our criteria and pass fragments per the RFC.

While enforcing a security policy that permitted a handful of TCP and UDP services, detailed testing of a product revealed that it blocked some but not all fragments. For the fragments passed through the firewall, the original datagram before being fragmented could belong to any

of the 256 IP types. However, for the fragments to pass through the firewall they had to have a specific fragment data size and fragment offset. Fragments that we crafted could be passed through the product beginning at fragment offset '256' with a data size equal to the maximum allowed by the MTU. For every 1 byte decrease in fragment data size we had to increase the minimum offset by '256' (up to a maximum offset of '65280') in order to craft and pass other fragments through this product. After the fragment offset reached '65280', the offset wrapped back around to '256' for the next lower data size. This continued for all fragment data sizes in the MTU size range.

Most products that fail to block fragments do so for far less complex reasons. Firewalls primarily have had trouble blocking fragments of IP datagrams for services permitted by the security policy. The data size and fragment offset usually have had nothing to do with it. What has been interesting is that some firewalls were permitting fragments of the initial TCP SYN packet that had data attached.⁹

When the firewall testing team first began testing fragmentation we realized it was going to be a little tricky to determine if a product was passing fragments after first performing reassembly or simply passing fragments without confirming they could be reassembled. The ICSA Labs firewall testing team does not penalize firewall vendor products that send fragments only after first going the extra mile to reassemble IP datagrams. Such a vendor meets both RFC 1812 and the intent of the criteria.

One way that we are able to test for reassembly is to adjust the MTU and IP datagram sizes. We configure the MTU of the source host to be less than that of the destination host and send an IP datagram that is larger than the MTU of both the source and destination. Fragments are automatically created at the source side by virtue of the fact that the data size is larger than the source hosts MTU. If fragments are passed through the firewall and the fragments on both sides are identical, then we can conclude that the firewall is not performing reassembly.

If instead the fragments on the source side are different from those on the receive side then we are assured that reassembly is performed by the firewall. In this case, after the firewall reassembles the IP datagram it has to re-fragment it since the datagram size is greater than the MTU of the destination. When creating these new fragments the firewall considers the destination network MTU which is larger than that of the source side. Thus the fragments at both the source and destination are different in number and size.

These examples illustrate the rigor involved in testing fragmentation by ICSA Labs. The comprehensive fragmentation testing provides assurance that ICSA Labs Certified firewalls are capable of blocking or reassembling all fragments.

Where Can I Find Out How a Firewall Did?

Reports are written for every firewall version that completes testing and attains certification. New reports are written for the most recent version tested and older reports are archived. Vendors are not given an opportunity to influence the report contents. Reports are accessible from the ICSA Labs Certified firewall products Web page at:

⁹ It's arguable as to whether or not the TCP RFC actually permits data on the initial SYN. It seems as though the TCP RFC allows it just as long as the receiving TCP does not push the data up to the application until after the handshake.

<http://www.icsalabs.com/html/communities/firewalls/newsite/cert2.shtml>

When reviewing a report, the “Criteria Violations and Resolutions” section provides what criteria violations were found and repaired during the testing of a particular firewall product. This and the “Miscellaneous Notes” section that provides summary details on our observations during testing contain important product information gained through our testing.

Reports are not written for products that fail or otherwise do not complete testing. Therefore, you may not see a report for your firewall or another with which you are familiar. This means that the product has either not been submitted for testing, is somewhere in the midst of testing, or has failed certification testing. If you do not see your product of interest on the list of ICSA Labs Certified firewall products, you should contact the vendor to determine either why they have not yet submitted their product for ICSA Labs firewall certification testing or to determine the status of their product within testing.

Summary

Beyond the in depth testing described throughout this paper, our “Continuous Deployment” and “Event-based Testing” models contribute value to both end users and firewall vendors. Through “Continuous Deployment” all firewall products remain deployed in an Internet-like environment, protecting a network of their own. Firewalls are then continuously available for periodic testing throughout the year to ensure they continue to meet the criteria. “Event-based Testing” is performed on a moment’s notice against the entire set of firewall products deployed in the lab when new exploits are discovered through testing or from other sources. These two models increase the value of ICSA Labs firewall testing beyond the value provided by our rigorous certification testing.

ICSA Labs recently conducted an Event-based test resulting from a recently published vulnerability. When successful the vulnerability caused ongoing TCP sessions to end after receiving crafted TCP RST packets having a particular range of incorrect sequence numbers. We wrote code to test for the vulnerability and were able to identify several firewalls in the program that were susceptible to the attack. We then alerted and requested product changes from vendors of the affected firewall products.

ICSA Labs works constantly to ensure that the firewall certification program keeps pace with industry advancements and security vulnerability discoveries. This enables us to have a dynamic, evolving testing methodology and certification testing criteria. Because of the program’s dynamic nature, the ICSA Labs firewall testing team continually finds new ways to circumvent firewalls. The improvements to firewall products resulting from ICSA Labs testing ensure that significant value is provided and maintained. What results from ICSA Labs testing are firewalls that far exceed those that have not been through our detailed testing process. It is our hope that this paper demonstrates the value of ICSA Labs firewall testing to both the end user community and firewall vendors alike.

Acknowledgements

I would like to acknowledge the following members of ICSA Labs team that reviewed and provided input to the content of this document:

Firewall Lab Team – Ron Guyer, Dave Archer, Joshua Curtis, Bill James, and Brian Monkman.

Who to Contact At ICSA Labs

For questions or comments about this paper contact Jack Walsh at jwalsh@icsalabs.com. For more information regarding the ICSA Labs Firewall Product Certification Program or Firewall Product Developers Consortium, visit www.icsalabs.com or e-mail the Firewall Program Manager, Brian Monkman, at bmonkman@icsalabs.com.

About ICSA Labs

ICSA Labs, an independent division of TruSecure Corporation, offers vendor-agnostic testing and certification of security products. Hundreds of the world's top security vendors submit their products for testing and certification at ICSA Labs. The end-users of security technologies rely on ICSA Labs to authoritatively set and apply objective testing and certification criteria for measuring product compliance and reliability. The organization tests products in key technology categories such as anti-virus, firewall, IPsec VPN, cryptography, intrusion detection, PC firewall, content security, SSL-VPN, and Wireless LAN.

About TruSecure Corporation

TruSecure is the leading provider of intelligent risk management products and services. TruSecure dramatically improves security and reduces risk by helping organizations make better security decisions and maximize the effectiveness of existing security, people, processes, and products. Leveraging TruSecure's vast security knowledge and intelligence gathering resources—including ICSA Labs, the global leader in information security product certification—as well as innovative technology and time-tested processes, our customers can *predict* which vulnerabilities present real risk, *prioritize* remediation efforts, quickly *adapt* to changes in the security threatscape, *measure* progress in improving their security posture, and *document* compliance with applicable security policies, standards and regulations.

Copyright © 2003-2004 TruSecure Corporation. All Rights Reserved. No part of this report may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording, or any information or storage retrieval system, without the express permission in writing from ICSA Labs. ICSA Labs is a division of TruSecure Corporation and is a registered mark of TruSecure Corporation.