

Protocol and Application Awareness: A New Trend or an Established Tradition?

Paul A. Henry, CISSP

Despite the recent marketing blitz by several high-profile security vendors, protocol anomaly detection (also called protocol awareness) and deep packet inspection (also called application awareness) are *not* new technologies these vendors conceived to deal with the growing threat of application-layer network attacks on the public Internet. Strong application proxy-based firewalls have incorporated sophisticated protocol *and* application awareness technologies since first introduced in the 1980s. When compared to offerings from providers of intrusion detection systems (IDSs), intrusion prevention devices (IPDs), and — for that matter — stateful inspection (SI) firewall vendors, it quickly becomes evident that strong application proxy technology is more mature, powerful, and proven in the most demanding network environments.

While nearly all stateful inspection firewall vendors claim to have the ability to inspect all seven layers of the OSI model, relatively few ever actually inspect, nevermind *act* upon, the packet above layer 4 (transports) (Figure 1).

Thinking back just a year or so ago, this author can recall stateful inspection firewall

vendors asserting that proxy firewalls (using techniques such as protocol and application awareness) were somehow “old outdated technology” (Figure 2) and that their products could provide better, faster network security without using these methods. As a security professional who has worked with leading government and commercial entities around the globe, this author is pleased that the “stateful inspection” firewall vendors have reversed that stance and now acknowledge the critical importance of these technologies to mitigate risk.

A vendor’s approach to protocol anomaly detection reveals a great deal about its basic design philosophy and the resulting capabilities of its network security products. The tried and true practice with strong application proxy firewalls is to allow only the packets that are known to be “good” and to deny everything else. Because most protocols used on the Internet are standards based, the best approach is to design the application proxy to be fully protocol-aware, and to use the standards as the basis for deciding whether to admit or deny a packet. Only packets that demonstrably

PAUL A. HENRY, MCP+I, MCSE, CCSA, CFSA, CFSO, CISSP, is vice president at the CyberGuard Corp. (www.cyberguard.com).



FIGURE 1 Firewalls without Protocol and Application Awareness

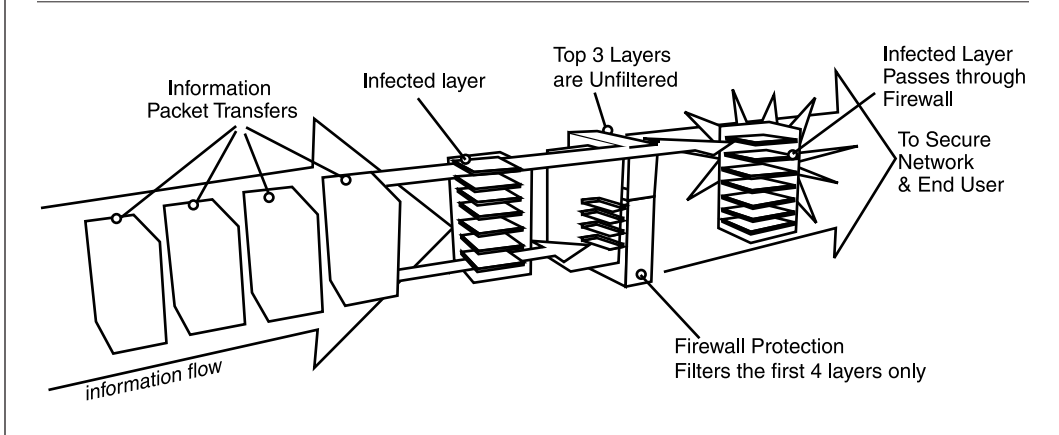
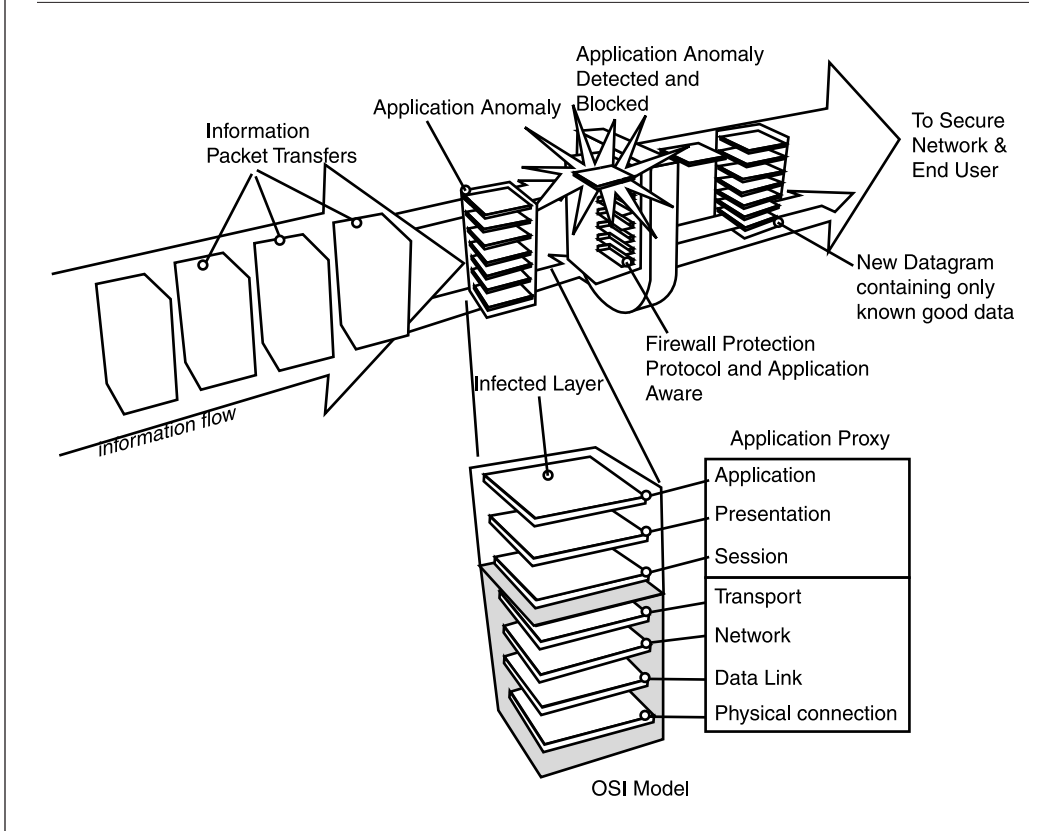


FIGURE 2 Firewall with Protocol and Application Awareness

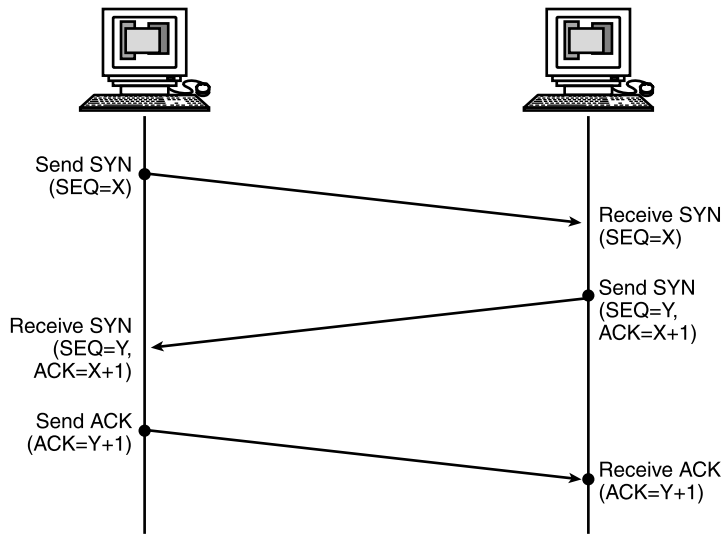


conform to the standard are admitted; all others are denied.

Most stateful inspection firewalls — as well as many IDS and IDP products — take the opposite approach. Rather than focusing on recognizing and accepting only good

packets, they try to find — and then deny — only the “bad” packets. Such devices are very vulnerable because they require updates whenever a new and more creative form of “bad” is unleashed on the Internet. Sometimes, especially with ASIC (Application

FIGURE 3 Three-Way Handshake



Specific \pm C) vendors that implement these packet rules in silicon, it is impossible to make these changes without replacing the ASIC itself.

Another problem with the “find and deny the bad” methodology is its intrinsic inefficiency. The list of potentially “bad” things to test for will always be much larger than the predefined and standardized list of “good” things. It is very much like getting into heaven: virtue should be its own reward.

One can argue that the “find and deny the bad” approach provides additional information about the nature of the attack, and the opportunity to trigger a specific rule and associated alert. However, it is unclear how this really benefits the network administrator. If the attack is denied because it falls outside the realm of “the good,” does the administrator really care which attack methodology was being employed? As many have seen with IDSs, an administrator in a busy network may be distracted or overwhelmed by useless noise generated by failed attacks.

A strong application proxy elevates the art of protocol and application awareness to the highest possible level. The simplified path of a packet traversing a strong application proxy is as follows:

1. The new packet arrives at the external interface. Layer 4 data is tested to validate that the IP source and destination, as well as service ports, are acceptable to the security policy of the firewall. Up to this point, the operation of the application proxy is similar to that of stateful packet filtering. For the most part, the similarities end here.

The RFC-mandated TCP three-way handshake (<http://www.faqs.org/rfcs/rfc793.html>) is fully validated for each and every connection (Figure 3). If the three-way handshake is not properly completed, the connection is immediately closed before any attempt is made to establish a connection to the protected server. Among other benefits, this approach effectively eliminates any possibility of SYN flooding a protected server.

This is where vital differences become apparent. Many stateful inspection firewalls do not validate the three-way handshake in order to achieve higher performance and packet throughput. In the author’s opinion, this approach is dangerous and ill-conceived because it could allow malicious packets with a forged IP address to sneak right past the stateful firewall.

More troubling is the “fast path” mode of operation employed by some stateful

Stateful inspection firewalls allow attackers to make a direct connection to the server, which is supposedly being protected behind the firewall.

inspection firewall vendors. When “fast path” is engaged, the firewall inspects only those packets in which the SYN flag is set. This is extremely dangerous. Given the availability of sophisticated and easy-to-use online hacking tools, any 13-year-old with a modem and a little spare time can exploit this weakness and penetrate the “fast path” mode firewall by simply avoiding the use of SYN flagged packets. The result: malicious packets pass directly through the firewall without ever being inspected. An informed network administrator is unlikely to open this gaping hole in his security infrastructure to gain the marginal increase in throughput provided by fast path.

2. For each “good” packet, a new empty datagram is created on the internal side of the firewall. Creating a brand-new datagram completely eliminates the possibility that an attacker could hide malicious data in any unused protocol headers or, for that matter, in any unused flags or other datagram fields. This methodology — part of the core application proxy functionality found within strong application proxy firewalls — effectively eliminates an entire class of covert channel attacks.

Unfortunately, this capability is not available in any stateful inspection firewall. Instead, stateful inspection firewalls allow attackers to make a direct connection to the server, which is supposedly being protected behind the firewall.

3. Protocol anomaly testing is performed on the packet to validate that all protocol headers are within clearly defined protocol specifications. This is not rocket science, although there is some elegant engineering needed to do this quickly and efficiently. Because Internet protocols are based on published standards, the application proxy uses these as the basis for defining what is acceptable and denies the rest.

Stateful inspection firewall vendors have tried to address this requirement by adding limited filtering capabilities that are intended to identify attack-related protocol anomalies and then deny these “bad” packets. Unfortunately, this approach is inherently flawed.

Most stateful inspection firewalls employ a keyword-like filtering methodology. Rather than using the RFC-defined standards to validate and accept good packets (our “virtue is its own reward” approach), stateful inspection firewalls typically filter for “bad” keywords in the application payload. By now, the problem with this approach should be evident. There will always be new “bad” things created by malicious users. Detecting and accepting only those packets that adhere to RFC standards is a more efficient and — in this author’s opinion — far more elegant solution.

Take a look at the Simple Mail Transfer Protocol (SMTP) as an example. A strong application proxy applies the RFC 821 Standard for the format of ARPA Internet text messages (www.faqs.org/rfcs/rfc2821.html) and RFC 822 Simple Mail Transfer Protocol (www.faqs.org/rfcs/rfc822.html) standards to validate protocol adherence. It also lets one define “goodness” using another dozen or so protocol and application-related data points within the SMTP packet exchange.

This enables an administrator to minimize or eliminate the risk of many security issues that commonly plague SMTP applications on the Internet today, such as:

- Worms and virus attacks
- Mail relay attacks
- Mime attacks
- SPAM attacks
- Buffer overflow attacks
- Address spoofing attacks
- Covert channel attacks

In contrast, a stateful inspection firewall must compare each packet to the predefined signatures of hundreds of known SMTP exploits — a list that is constantly growing and changing. This places the security professional in a virtual “arms race” with the entire hacker community. One will never be able completely filter one’s way to a secure network; it is an insurmountable task.

Another element of risk with filter-based approaches is their vulnerability. Attackers frequently “fool” the filter simply by adding white space between the malicious commands. Not recognizing the command, the firewall passes the packet to the “protected” application, which will then disregard the white spaces and process the commands. As with any filter, if the signature does not explicitly match the packet, the packet will be allowed. No network administrator can confidently rely on such a vulnerable technology.

With the strong application proxy approach, virtually all SMTP-related attacks can be mitigated more effectively and efficiently than is possible with the filtering approach used by the stateful inspection vendors.

4. The application proxy applies the (very granular) command-level controls and validates these against the permission level of the user. The application proxy approach provides the ultimate level of application awareness and control. Administrators have the granularity of control needed to determine exactly what kind of access is available to each user. This capability is nonexistent in the implementation of most stateful inspection firewalls.

It is difficult or impossible to validate the claims made by many stateful inspection firewall vendors that they provide meaningful application-level security. As we have seen, the “find and deny the bad” filter-

based approaches are inefficient and vulnerable. They simply do not provide the same level of security as a strong application proxy firewall.

5. Once the packet has been recognized as protocol compliant and the application-level commands validated against the security policy for that user, the permitted content is copied to the new datagram on the internal side of the firewall. The application proxy breaks the client/server connection, effectively removing any direct link between the attacker and the protected server. By copying and forwarding only the “good” contents, the application proxy firewall can eliminate virtually all protocol-level and covert channel attacks (Figure 4).

Stateful inspection firewalls do not break the client/server connection; hence, the attacker can establish a direct connection to the protected server if an attack is successful. And because all protection requires the administrator to update the list of “bad” keywords and signatures, there is no integral protection to new protocol-level attacks. At best, protection is only afforded to known attacks through inefficient filtering techniques.

In conclusion, this author applauds the desire of stateful inspection firewall vendors to incorporate protocol and application awareness capabilities into their products. The new generation of network threats requires nothing less. However, the author also believes that the attempt by vendors to “filter” their way to higher levels of security is a fundamentally flawed approach. Upon closer inspection, stateful inspection firewalls clearly lack the depth of protocol awareness, application awareness, efficiency, risk mitigation, and high performance found in a strong application proxy firewall. ■

Attackers frequently “fool” filters simply by adding white space between the malicious commands.

FIGURE 4 Forwarding Only the "Good" Packets

