

White Paper

# The Need for Pervasive Application-Level Attack Protection

---

*How Deep Inspection Technology Meets the Requirements for Network and Application Attack Protection*

Sarah Sorensen  
Product Marketing Manager



Juniper Networks, Inc.  
1194 North Mathilda Avenue  
Sunnyvale, CA 94089 USA  
408 745 2000 or 888 JUNIPER  
[www.juniper.net](http://www.juniper.net)

Part Number: 200049-001

---

---

## Contents

Introduction.....	3
Today's Security Solutions .....	4
Stateful Inspection Firewalls Provide Network Security.....	4
Legacy Application-Level Security Solutions .....	5
Intrusion Detection Systems.....	5
Proxies .....	6
New Application-Level Security Solutions .....	6
Application-specific Firewalls (Proxies) .....	6
Intrusion Prevention Systems .....	6
The Need for Pervasive Application Attack Protection.....	7
The New Technology Requirements, Application Attack Protection.....	8
Extracting application information from network traffic .....	8
Identifying Attacks at the Application-Layer .....	11
Other Requirements .....	12
Introducing Deep Inspection Technology .....	13
Summary .....	14

## Introduction

Today’s corporate environment necessitates connectivity on a global scale to promote efficient business practices. All types of users, including employees, partners, vendors, telecommuters, clients, etc. require access to the resources they need, when they need them. Network and IT administrators leverage technologies, such as the Internet, to cost-effectively connect all of these users to the network. However, this connectivity also increases the network’s points of vulnerability. It offers a “way in” for attackers, who are looking to steal, alter or bring down the company’s most critical assets. The challenge facing administrators is how to enable “always on” connectivity for business productivity, but shut it down for unauthorized and malicious use.

Most administrators start securing the network by deploying a firewall. Firewalls were designed to control access and flourished in the mid-to-late 1990’s, ultimately becoming the de facto foundation for an organization’s network security. The recent 2003 CSI/FBI Computer Crime and Security Survey found that 98% of the respondents deployed firewalls, which underscores the pervasiveness of this technology.

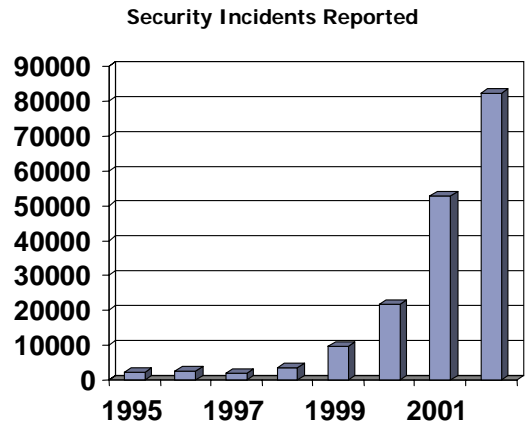


Figure 1 Source: CERT Coordination Center: 2002

So why are security incidents on the rise? Based on the information in Figure 1, it appears that firewalls were successful at protecting the network when they were introduced—delivering on their design. However, over the past few years, attacks have evolved and grown in sophistication, using different mechanisms to exploit and attack network resources.

Since firewalls represent the frontline defense, it is reasonable for organizations to turn to firewall vendors to prevent these attacks from entering. As network devices, firewalls can protect hundreds, even thousands, of hosts behind them. In addition, they are in the perfect position to protect against “Internet facing” attacks, which consistently represent the most frequently exploited entry point for attackers, as evidenced by Figure 2.

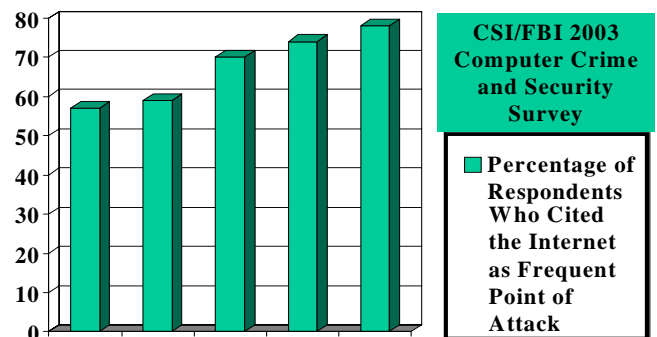


Figure 2

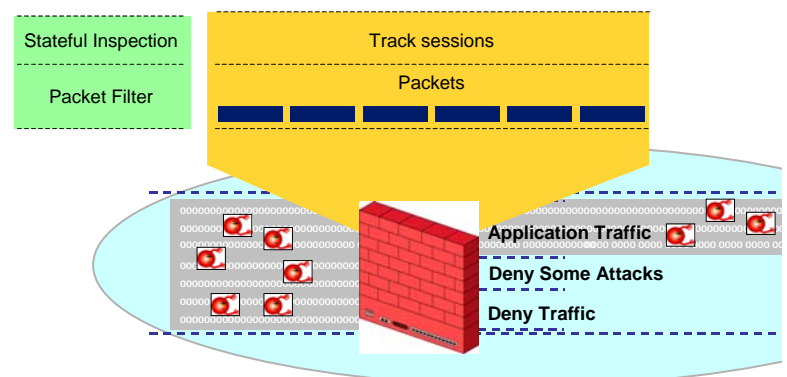
This paper discusses technological advancements in perimeter security that enable corporations to protect against these emerging exploits and attacks. It examines traditional solutions, identifying the strengths and limitations of these technologies. It also describes new technologies, called Deep Inspection, which is designed to secure the network perimeter and protect critical assets from these increasingly sophisticated threats.

## Today's Security Solutions

The foundation for network security is the firewall. A recent study shows that most IT managers are confident in the ability of current solutions to protect their network, but are concerned with the lack of depth of protection that their current security solutions can provide (source: *Vanson Bourne IT Manager Security Survey, 2003*). At first glance, these two data points seem to be in opposition to one another, but on closer examination, it makes sense; administrators believe traditional technologies are doing the job they were designed to do, but recognize that the threats against the network are changing and require new technologies for protection. This section will look at what traditional technologies were designed to do, identifying both their strengths and limitations.

### Stateful Inspection Firewalls Provide Network Security

Stateful inspection firewalls protect against unauthorized users accessing network resources and the unauthorized use of network resources. They do this by making access control decisions, based on a predefined policy, to determine who and what type of application traffic (e.g. Web, e-mail, etc.) is allowed in and out of the network. They also protect against some network-level attacks.



Stateful inspection firewalls make traffic decisions based on session information, instead of packet-level information, to take into consideration the "state" information. A stateful inspection firewall accepts or denies traffic based on the source IP address, destination IP address, source port, destination port and protocol. They track and maintain the state of the session, so they can verify an inbound packet matches a previously allowed session. They also support dynamic protocols, where a server tells the client to switch to another random destination port, by identifying port-change instructions, recording them and comparing future sessions to these records. In this way, stateful inspection firewalls can efficiently consider whether or not a packet is associated with an allowed session.

There was a time when attacks mainly constituted unauthorized access to the network, i.e. finding or accessing a server. As a result, stateful inspection firewalls were designed to provide high performance perimeter access control that determined what traffic should be allowed in and out of the network.

Stateful inspection firewalls were also able to add protection against some attacks aimed at the network itself. These attacks are relatively rudimentary in their methods, due to the layer at which they are operating. The information to be evaluated is relatively simple, “what you see is what you get,” without ambiguity. It doesn’t matter what type of application generated the traffic, it all looks the same at the network level. All the stateful inspection firewall needs to do to detect these attacks is to examine a few fields in each packet and make a decision on whether it should be allowed or denied.

The attacks that are targeted at networks today are often embedded in the application traffic flow that is allowed to run between the client and server. Stateful inspection firewalls were designed to block unwanted application traffic and allow certain application traffic, not look into, or interpret the application traffic itself. Stateful inspection firewalls do not have visibility into the “message” of the traffic, which manifests itself at the application layer. As a result, complementary solutions to stateful inspection firewalls try to protect against application-level attacks.

## Legacy Application-Level Security Solutions

For protection against attacks aimed at the application level, organizations may have intrusion detection and proxy solutions. It is important to be aware of the capabilities and limitations of these solutions, to understand, while potentially filling a niche role in the overall layered security approach, they are not suited for pervasive protection against both network and application level attacks.

### Intrusion Detection Systems

Organizations have adopted intrusion detection technology to monitor their networks and alert them to attacks. Intrusion detection systems (IDSes) are passive devices and are usually used to monitor all outgoing and incoming network traffic. IDSes primarily search for suspicious application activity that might indicate an attack and look for application protocol conformance violations. Some may perform additional analysis, looking for other types of attacks, such as Trojans, that can be identified by using special heuristics and examining statistical and behavioral patterns.

The drawback to intrusion detection systems is that they are passive devices, unable to drop malicious traffic from the network when it is detected. As a result, IDSes place the security burden on IT personnel, who are required to perform attack investigation, analysis and response for application exploits in the network. While offering some insight into the activity on the network, IDSes are unable to protect against application-level attacks.

## Proxies

Some organizations have deployed proxy technology to protect against attacks aimed at the application. Proxies, also known as Security Servers and other similar terms, terminate all connections and broker all communication between the client and the server and then make decisions on what traffic is allowed. The technology is a full client/server implementation of the protocol. As a byproduct to that implementation, proxies understand application logic and can accept or deny traffic based on application information.

The drawback to application proxy is that a full client/service implementation is required. Proxy solutions are limited in the number of application protocols they can support. Protecting against attacks aimed at a particular application protocol requires the proxy understand and implement all of the diminutive details of that protocol. An added complication is that most vendors implement protocols with slight variations. As a result, proxy protection is slow to incorporate new protocol support and hard to scale. In addition, the latency associated with brokering both the client and server connection significantly impacts performance. Another limitation of running in the user/application space is difficulty in supporting fault tolerant, high-availability configurations. Replicating full applications on the network is very difficult to achieve and therefore introduces a single point of failure to the network. Because an application proxy runs in the user/application processing space and, by definition, does not have awareness of the network/kernel information, it is next to impossible for proxies to do analysis on the original traffic to look for attacks, since they never see the individual packets.

## New Application-Level Security Solutions

To overcome the shortcomings of the legacy solutions, new technologies have arisen to provide application-level attack protection, namely application-specific firewalls and intrusion prevention systems.

### Application-specific Firewalls (Proxies)

There are application-specific firewalls that offer very deep application protection for a specific protocol. They are typically implemented as a proxy running on an appliance, and are designed to provide granular control and attack protection within a single protocol, generally Web, e-mail or Instant Messengers. While these still suffer the drawbacks associated with a proxy implementation, they are optimized to run the single application and, therefore, are sometimes appropriate to support certain configurations/applications. Typically, customers that rely on their Web or e-mail servers, such as e-commerce sites, e-mail outsourcing, etc. will implement an application-specific proxy to protect those specific Web and e-mail servers.

### Intrusion Prevention Systems

New technology has come onto the market in the past year designed to protect against a broad array of application-level attacks. Devices that incorporate this technology, namely intrusion prevention systems (IPSeS), have been built from the ground up as network devices that can accept or deny traffic based on source IP address, destination IP address, service/protocol and some application level analysis and verification.

These solutions are able to accurately interpret the intent of the application message, removing ambiguities found at the application-level and then performing application analysis to identify attacks. They are able to look for high impact deviations to protocol specifications, apply pattern matches in relevant service fields that represent attacks, and use special heuristics, statistical information, behavioral patterns, and many more characteristics that are representative of different types of attacks to maximize the attack protection coverage they can offer.

These solutions understand enough of the protocol to make application level decisions, without implementing the full client and server. In this way, they can efficiently process the traffic and deliver a solution capable of meeting network performance requirements. This also allows them to be deployed in a high availability configuration, sharing session information to ensure the connection can persist in the event of a failure.

These solutions are generally managed using a rulebase, making it easy for administrators to control exactly how the application-level attack protection is applied throughout the network. Based on the predefined policy, identified attacks can trigger a variety of controlled responses, from alerting to dropping the malicious packet or connection from the network.

## The Need for Pervasive Application Attack Protection

It is important to deploy application protection throughout the network, since an organization's ability to protect any network resources can be compromised by a single "weak link." The early adoption of intrusion prevention technology has mainly been focused on securing sensitive resources in the corporate headquarters and large regional offices. Generally, customers deploy IPSes behind a stateful inspection firewall and in front of critical servers, where protecting application level data from both internal and external attacks is a primary concern. The need, however, is to make application-level protection pervasive. Small remote and branch offices and telecommuters with home networks also need application-level attack mitigation. Recognizing that these network segments probably do not have the resources or variety of protocols running through them that the regional offices and large central sites do, the need is to add the appropriate application level protection for the resources found at these sites, i.e. a Web server, e-mail server, etc. The logical place to add this protection is at the perimeter or edge, where the firewall sits. Adding deeper protection to the firewall for the types of attacks that threaten these network segments would enable the organization to interdict these threats at the edge and strengthen their overall security. Such a strategy would meet the security goals at minimal cost to management overhead and complexity. This next section identifies what is required to make application-level protection available at the perimeter.

## The New Technology Requirements, Application Attack Protection

It is not a simple task to add application-level attack protection to a network firewall. The fundamental reason it is so difficult to protect against application level attacks is there is so much information that needs to be understood before a decision of the traffic's intent can be discerned. For network attacks, a device only needs to look at specific information (source, destination, port numbers and protocol) to understand the relationship and decide whether to accept or deny the traffic. For application attacks, the solution needs to look at and relate all of the data in all of the packets before making a security decision. As a result, the solution must:

- Accurately interpret the intent of the application message
  - Perform de-fragmentation, reassembly, scrubbing and normalization to remove ambiguities and understand how the host system will see the message
- Do application specific analysis
  - Protocol conformance verification
  - Attack pattern matches applied in relevant service fields - Stateful Signatures
- Offer additional attack identification mechanisms (*Ideal*)
- Meet performance needs of network segment in which it is deployed
- Provide high availability configurations to eliminate single point of failure
- Be easy to manage

There are several challenges to making application-level traffic decisions:

1. How to extract application-level information from the individual packets the solution inspects
2. How to identify attacks in the application-level information

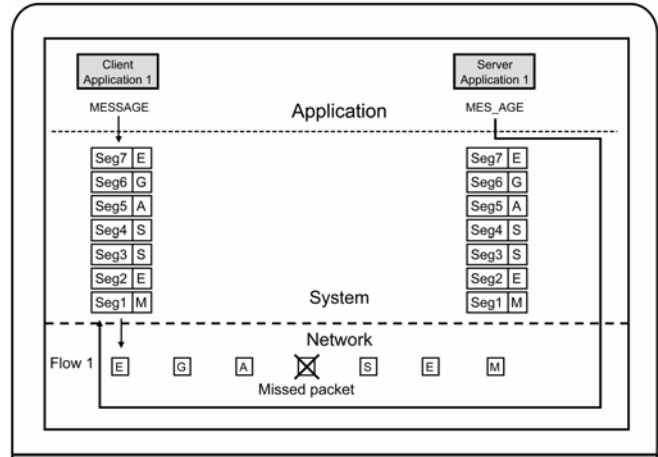
### Extracting application information from network traffic

The basic challenge is the solution must know what the client and server applications are trying to do. As a result, the solution must understand each application before it can even try to make a determination and stay up-to-date on any changes to that application or protocol to ensure it can continue to understand the intent. There are two phases to extracting application level information from the traffic, namely:

- Traffic reconstruction
- Ambiguity elimination

### Traffic Reconstruction (De-fragmentation, Reassembly)

Let's look at how traffic gets transmitted through the network and the potential complications that this transmission can introduce. (See Accompanying Figure for an example.) Application 1 generates information for communication as a stream of data ("MESSAGE"). The network stack within the operating system breaks that stream into segments ("TCP segments", carried by individual packets) and sends them to the network. The receiving operation system's network stack collects the TCP segments and converts them back to a stream of data (MESSAGE), which is then presented to the receiving application.



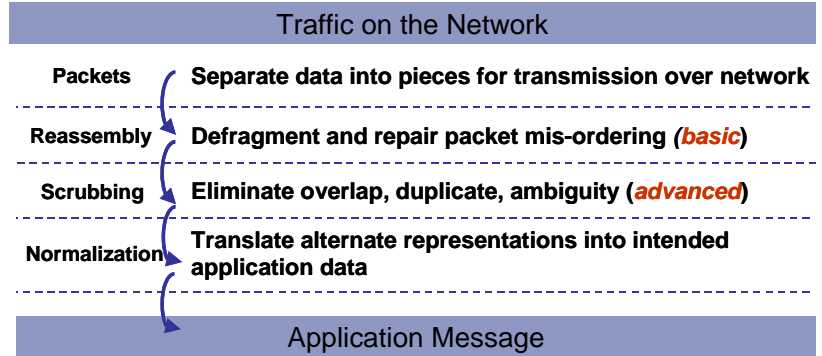
The network is designed to handle the "random" nature of packet, network transmission. When an application message is transmitted on the network, it will make a "best effort" to deliver this message. Messages are split into packets, with no order or predetermined delivery mechanism, resulting in packets transmitted randomly (out of sequence). In addition, networking devices, such as routers, may break the packets down into even smaller pieces of data, called fragments. The solution needs to be able to properly combine fragments of packets into packets. This process is called IP de-fragmentation. The solution must then properly reassemble the packets in the right order, which is called reassembly.

Packets can get "lost," which is depicted in this example. When this happens the receiving network stack is responsible for reconstructing the packets into a stream of information, and noticing that a packet that is part of the message is lost, waits for the retransmission of the missing packets, so that the entire message can be presented to the application, as a whole.

This particular example shows a single flow from the client to the server; the reality is that most client to server communications consist of two flows, one from the client to the server and another from the server back to the client. In addition, most networks will have hundreds of thousands of messages being transmitted back and forth through the network at the same time.

### Ambiguity Elimination

Once the traffic has been reassembled, techniques, such as scrubbing and normalization, must be used to eliminate possible misinterpretations of the data.



A network device must "scrub" the traffic to eliminate ambiguity and properly interpret the message. Ambiguities in the traffic can include overlapping packets with different data, duplicate packets with changing content, different representations of the data, etc. The solution needs to understand how the host is going to handle ambiguities.

Interpretation of data often requires information that is not available to the solution, leading to ambiguities. The example shown to the right is a message that consists of six packets, with a duplicate number six packet. If the receiving network stack was able to process the first number six packet, it would deliver the message "ATTACH" to the application, while ignoring the second number six packet. However, if the first number six packet got lost, or was not processed by the receiving network stack for any reason, the application will receive the message "ATTACK." The intent of these messages, ATTACH vs. ATTACK, is very different, and if the solution uses the wrong interpretation, it may miss an attack. The core problem is there is no practical way for the solution to know what ended up happening at the receiving host. One technique to solve this problem is to simply pick one interpretation, while eliminating the other one from the network, making sure the ambiguous packet never reaches its intended host.

Packet #	Content
1	A
2	T
3	T
4	A
5	C
6	H
6	K

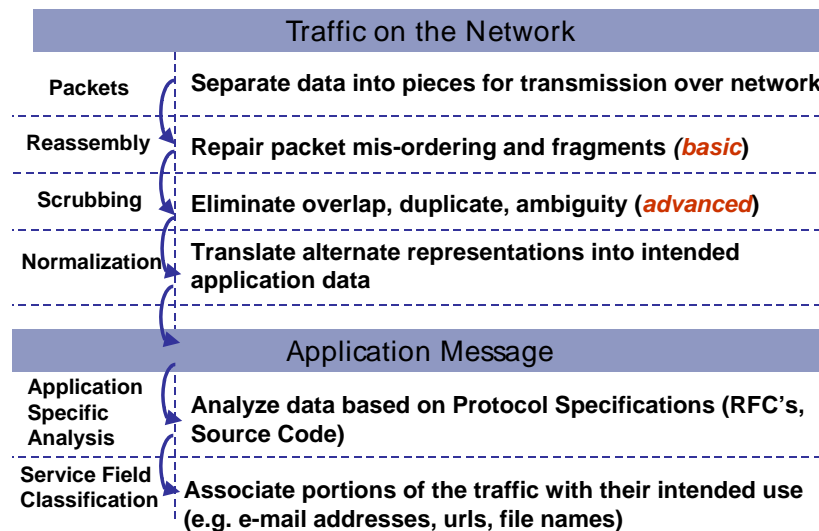
The solution also must be able to normalize the traffic to ensure that alternative representations are translated into a uniform representation. For example, programmers often encode directory names, using characters such as "..\" to support relative path specifications. For example, both "\\data\\confidential\\document" and "\\something\\..\\data\\confidential\\something\\..\\document" refer to the same confidential document that an attack might try to access. In another example, the hexadecimal characters "%61%74%74%61%63%6B" are another representation for the ASCII characters "attack." It is

important for the solution to understand what these representations are intending to do for accurate analysis. This illustrates the importance of being able to normalize the traffic to perform attack identification.

Attackers exploit all of these variables in the network and create their own ambiguities to try and “fool” security devices into thinking the traffic is legitimate. In order to achieve application level security, a solution must be able to remove ambiguities from the traffic to accurately interpret the intent of the application “message.”

### Identifying Attacks at the Application-Layer

Once the application “message” is accurately interpreted, the solution needs to apply application specific analysis to identify whether there are attacks in that traffic. The foundation for this analysis lies in the solution’s ability to efficiently identify non-conformance to protocol specifications and known attack patterns in specific service fields that indicate malicious intent.



### Protocol conformance

First, the solution should be able to analyze the data based on the protocol specifications. If the data deviates from the expected behavior, it represents an anomaly. Anomalies can then be classified based on their potential impact, with high-impact anomalies identified as an attack. An example of a high severity anomaly would be data that exceeds the amount of bytes expected by the receiving application. If this data is allowed to reach the application, get into a memory buffer and a vulnerability exists that enables the excess data to overflow the buffer, then that excess data could be crafted to take complete control over the system that hosts the application. This is called a buffer overflow attack. Anomalous traffic of this nature, such as sending more traffic than is expected or allowed by the application, should be denied access to the network, so that it cannot reach its intended victim.

A solution that can apply protocol conformance to the traffic is able to protect against an entire class of exploits, such as buffer overflow attacks, without needing to know about a specific exploit. This means that the solution can potentially provide “day zero” protection against brand new attacks as they emerge. It is also a mechanism that can protect against some of the more sophisticated attacks that cannot be characterized by a simple pattern.

### Attack Patterns

Attack pattern matching is another component to providing application-level attack protection. In order to properly interpret the application message, it must be extracted in a way that the application message’s communicated intent can be understood. It is only within context that a determination can be made whether the information constitutes an attack or not. Achieving a contextual understanding requires extracting information from the traffic and building the application communication service fields, which are portions of the traffic that relate to specific functions, such as e-mail addresses, URLs, file names, etc. Each protocol will have associated service fields, which imply intended use. These service fields therefore provide accurate insight into what is an attack. For example, in SMTP, the service fields are: command-line (a command from the client to the server), data-line (a line of the e-mail message itself), From: (sender’s e-mail address), etc. These service field classifications enable the solution to apply pattern matches to only the relevant “areas” of traffic where the match represents malicious intent.

To illustrate this point, there is an attack that exploits a vulnerability in Sendmail’s debug mode that could allow an e-mail recipient to achieve all of the privileges of a system administrator (root). The attack is perpetrated when the word debug is sent as an SMTP command from the client to the server. The ability to extract service fields in a granular manner and look for this attack only in an SMTP command-line service field, allows the solution to pinpoint the attack and make a traffic decision (allow/deny). The solution will not look for the debug command in irrelevant portions of SMTP traffic, such as the data-line service field. Attack pattern matching in relevant service fields is therefore optimal for protecting against most known attacks.

### Other Requirements

The requirements for any device deployed in the network certainly apply to solutions providing application-level attack protection. These solutions should be able to efficiently process traffic, be highly available and easy to manage.

It is important for the solution to meet the performance requirements of the network segment in which they are deployed. As application-level security necessitates a lot more processing than network-level security, these solutions should develop their technology to optimize analysis and minimize performance impact. On the flip side, organizations should weigh the performance requirements of a network segment against the risks associated with leaving that segment vulnerable to attack and choose the appropriate solutions which best meet both the level of performance and security required.

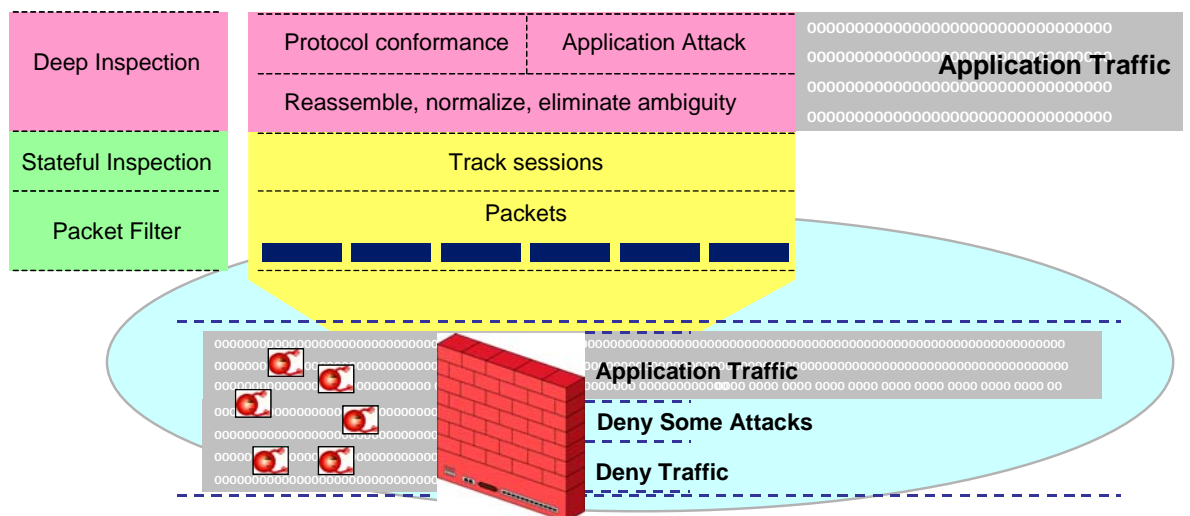
The value of a solution is greatly diminished if it disrupts the flow of network traffic in the event of a failure. The solution should facilitate the connectivity that it is trying to protect. These solutions should not introduce a single point of failure to the network, rather they need to maintain network connectivity in the event of a failure. As a result, the solutions need to provide high availability deployment options to increase the overall reliability of the security.

Finally, if these solutions are not easy to manage, most likely the security that they offer will not be deployed. These solutions need to provide granular control over when, where and how application-level security is deployed throughout the network. The solution needs to offer administrators a way to apply the appropriate levels of security to different traffic. The ideal solution enables administrators to customize the application attack protection to meet the specific needs of their organization.

## Introducing Deep Inspection Technology

Deep Inspection technology builds on the strengths of stateful inspection and integrates the most deterministic intrusion prevention technologies to provide application-level attack protection at the perimeter. Leveraging the efficiencies of both technologies, Deep Inspection can efficiently perform network security functions as well as analysis on the application message to determine whether to accept or deny traffic.

Deep Inspection can apply a deeper level of application understanding to the traffic to make access control decisions based on the intent of that traffic. Deployed at the perimeter, it focuses on preventing application attacks aimed at Internet-facing applications, such as Web, e-mail, FTP and DNS. It eliminates application level ambiguities, performing de-fragmentation, reassembly, scrubbing and normalization, to convert network packets to the application-level message being transferred between the client and the server. It looks for protocol conformance and extracts data from identified application “service fields” where attacks are perpetrated and applies attack pattern matches. It will decide to accept or deny the traffic based on high impact protocol anomalies or any given attack pattern in one of these application service fields, and block these application-level attacks at the gateway, so they never reach their destination.



The technology is optimized to meet the performance requirements for the small remote and branch offices and telecommuter sites for which it was designed. Deep Inspection supports a high availability configuration, synchronizing session state information so connections can persist in the event of a failure.

It is easily managed and controlled by administrators through a rulebase. This approach enables organizations to apply application-level attack protection to the traffic they want to inspect and minimize the potential performance impact associated with the additional processing. For example, instead of all traffic, an administrator can apply application-level protection to the incoming traffic, which represents the largest threat to the Internet-facing resources.

## Summary

Security and network administrators are acutely aware of the multitude of attacks that could target their networks. Driven by a combination of regulations, such as Sarbanes-Oxley, Gramm Leach Biley and HIPAA, and well-publicized attacks, such as the recent Sequel Slammer and Blaster/LovSan worms that made worldwide headlines, security is top of mind throughout most organizations.

Implementing effective security to address the myriad of threats aimed at an organization's network is difficult without the right tools. While stateful inspection firewalls delivered on their design intent, providing access control and network-level attack protection, there are new requirements to protect against the increasing number of attacks aimed at the application-level. It is important to provide the correct level of application protection for each network segment, to both enable the appropriate levels of security and connectivity. Deep Inspection technology brings together the strengths of stateful inspection and IPS technology to address the security issues at the edge of enterprise networks. Focusing on protecting against the Internet facing protocols generally found in small remote, branch and regional offices, Deep Inspection technology makes it easy to achieve and manage application-level attack protection at these locations where resources and expertise are limited. Deep Inspection can secure the smaller locations that have typically represented the "weak links" to the network's overall security stance, enabling IPS to focus on detecting the more sophisticated attacks targeted at the larger, more diverse network segments. Finally, with Deep Inspection, organizations have the ability to secure their network resources and easily achieve the pervasive application-level protection they require.

---

Copyright © 2004 Juniper Networks, Inc. All rights reserved. Juniper Networks, the Juniper Networks logo, NetScreen, NetScreen Technologies, GigaScreen, NetScreen-Global PRO, NetScreen-Remote, NetScreen ScreenOS and the NetScreen logo are trademarks and registered trademarks of Juniper Networks, Inc. in the United States and other countries. All other trademarks and registered trademarks are the property of their respective companies. Information in this document is subject to change without notice. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without receiving written permission from Juniper Networks, Inc.