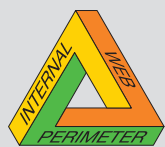


Check Point Application Intelligence



In this Document

- 2 Introduction—Application-driven Attacks
- 2 Application Intelligence—Defending Against the Next Generation of Threats
- 5 Network and Transport Layers: Necessary Foundation for Application Intelligence
- 6 Conclusion
- 7 Addendum





We Secure the Internet.

INTRODUCTION—APPLICATION-DRIVEN ATTACKS

Over the past several years, enterprise firewalls have become the staple of network security architectures. Designed primarily to provide access control to network resources, firewalls have been successfully deployed in the large majority of networks. A major reason for firewall success is that when used to enforce a properly defined security policy, firewalls commonly defeat more than 90% of network attacks — a crucial element in providing networks with the reliability required in today's competitive environment. However, while most firewalls provide effective access control, many are not designed to detect and thwart attacks at the application level.

Recognizing this reality, hackers have devised sophisticated attacks that are designed to circumvent the traditional access control policies enforced by perimeter firewalls. Today's knowledgeable hackers have advanced well past scanning for open ports on firewalls and are now directly targeting applications.

Some of the most serious threats in today's Internet environment come from attacks that attempt to exploit known application vulnerabilities. Of particular interest to hackers are services such as HTTP (TCP port 80) and HTTPS (TCP port 443), which are commonly open in many networks. Access control devices cannot easily detect malicious exploits aimed at these services.

By targeting applications directly, hackers attempt to achieve at least one of several nefarious goals, including:

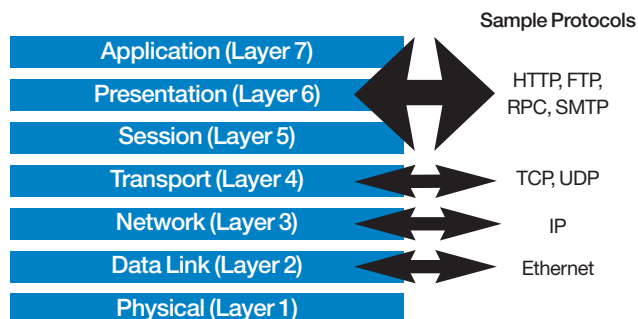
- Denying service to legitimate users (DoS attacks)
- Gaining administrator access to servers or clients
- Gaining access to back-end information databases
- Installing Trojan horse software that bypasses security and enables access to applications
- Installing software on a server that runs in "sniffer" mode and captures user IDs and passwords

Since application-driven attacks are sophisticated in nature, effective defenses must be equally sophisticated and intelligent. Enterprise firewalls, in order to address the increasing threat from application-driven attacks, must provide comprehensive security on multiple levels. These levels of security should protect against both network and application attacks, while providing robust access control to IT resources.

Check Point Application Intelligence™ is a set of advanced capabilities, integrated into Check Point's FireWall-1® and SmartDefense™, which detects and prevents application-level attacks.

Application Intelligence—Defending Against the Next Generation of Threats

Many firewalls, particularly those based on Stateful Inspection technology, have maintained successful defense arsenals against network assaults. As a result, a growing number of attacks attempt to exploit vulnerabilities in network applications rather than target the firewall directly. This important shift in attack methodology requires that firewalls provide not only access control and network-level attack protection,



The OSI Reference Model is a framework, or guideline, for describing how data is transmitted between devices on a network. NOTE: The Application Layer is not the actual end-user software application, but a set of services that allows the software application to communicate via the network. Distinctions between layers 5, 6, and 7 are not always clear, and some competing models combine these layers, as does this paper.

OSI (Open Systems Interconnection) Reference Model



Intelligent Security



but also understand application behavior to protect against application attacks and hazards. Based on INSPECT, the most adaptive and intelligent inspection technology, Check Point Application Intelligence provides this expansive view of network security solutions.

APPLICATION LAYER SECURITY

The application layer attracts numerous attacks for several reasons. First, it is the layer that contains a hacker's ultimate goal—actual user data. Second, the application layer supports many protocols (HTTP, CIFS, VoIP, SNMP, SMTP, SQL, FTP, DNS, etc.), so it houses numerous potential attack methods. And third, detecting and defending against attacks at the application layer is more difficult than at lower layers because more vulnerabilities arise in this layer.

In order to successfully provide application-layer security, a security solution must address the following four defense strategies.

1) Validate Compliance to Standards

Firewalls must be able to determine whether communications adhere to relevant protocol standards. Violation of standards may be indicative of malicious traffic. Any traffic not adhering to strict protocol or application standards must be closely scrutinized before it is permitted into the network, otherwise business-critical applications may be put at risk. Examples include:

- **Voice Over IP (VoIP)** VoIP traffic is typically supported using H.323 and SIP protocols. The operation of these protocols can be complex, resulting in numerous communication ports supporting the establishment and maintenance of VoIP calls. Improper enforcement of these protocols can leave the VoIP deployment vulnerable to the following hazards:
 - Call redirection—calls intended for the receiver are redirected
 - Stealing calls—the caller pretends to be someone else
 - Denial of Service (DoS)—preventing legitimate usage of VoIP

Security gateways must ensure that H.323 and SIP commands fully conform to appropriate standards and RFCs, and that packets are structurally valid and arrive in a valid sequence. In addition, firewalls should examine the contents of packets passing through every allowed port to ensure that they contain proper information.

- **Binary Data in HTTP Headers** While the official HTTP standard prohibits binary characters in HTTP headers, the rule is ambiguous and not checked by most firewalls. As a result, many hackers launch attacks by including executable code in HTTP headers. All security gateways should allow for the blocking or flagging of binary characters in HTTP headers and requests.

2) Validate Expected Usage of Protocols (Protocol Anomaly Detection)

Testing for protocol compliance is important, but of equal importance is the capability to determine whether data within protocols adheres to expected usage. In other words, even if a communication stream complies with a protocol standard, the way in which the protocol is being used may be incongruous with what is expected. Examples include:

- **Use of HTTP for Peer-to-Peer (P2P) Communications** P2P is a communication model in which each party has the same capabilities and either party can initiate a communication session. P2P applications can be divided into two major categories:
 - Instant messaging (IM)—whose main goal is to enable direct online communication between people
 - File sharing networks—whose main goal is to share resources such as storage

P2P communications often utilize TCP port 80, which is usually designated for HTTP traffic and is thus open for outgoing connections. While many proprietary P2P protocols exist, P2P communications often embed themselves within HTTP traffic. In these situations,





firewalls that check only for protocol compliance will allow the P2P session (since the session is using standard HTTP). Since commonly expected usage of HTTP is for web traffic, P2P communications embedded within HTTP traffic should be blocked or flagged by the firewall.

Many organizations want to block or limit P2P traffic for security, bandwidth, and legal reasons. Security issues arise because P2P communications are designed to allow file transfers, chat, games, voice and e-mail, while bypassing firewalls, virus checking, logging and tracking. As a result, hackers can use P2P as an attack vector into networks. Security gateways should block unauthorized P2P traffic, or conversely, selectively allow authorized P2P traffic.

- **Directory Traversal** Directory traversal attacks allow a hacker to access files and directories that should be out of reach, and can result in running undesired executable code on the web server by trying to access unauthorized resources. Most of these attacks are based on the “..” notation within a file system. Firewalls should block requests in which the URL contains a directory request that complies with syntax, but does not comply with expected usage. For example, `http://www.server.com/first/second/../../..` should be blocked because it attempts to go deeper than the root directory.
- **Excessive HTTP Header Length** The HTTP standard does not limit header length. However, excessive header length falls outside of normal or expected HTTP usage. Headers of excessive length should be blocked or flagged to reduce the chance of buffer overflows, and to limit the size of code that can be inserted using the overflow.

3) Limit Applications' Ability to Carry Malicious Data

Even if application-layer communications adhere to protocols, they may still carry data that can potentially harm the system. Therefore, a security gateway must provide mechanisms to limit or control an application's ability to introduce potentially dangerous data or commands into the internal network. Examples include:

- **Cross Site Scripting Attacks** Scripts provide a common mechanism for launching attacks against an application. While most scripts are harmless, unsuspecting users can easily and inadvertently execute malicious scripts. These scripts can often be hidden in innocuous looking links or, for instance, disguised as an email card. A common example of malicious scripts appears in Cross Site Scripting attacks (XSS). Cross Site Scripting attacks exploit the trust relationship between a user and a website by employing specially crafted URLs. The intention of the attack is to steal cookies that contain user identities and credentials, or to trick users into supplying their credentials to the attacker. Typically, a cross-site scripting attack is launched by embedding scripts in an HTTP request that the user unwittingly sends to a trusted site. To protect web servers, the security gateway should provide the capability to detect and block HTTP requests that contain threatening scripting code.
- **Limit or Block Potentially Malicious URLs** Malicious data can also enter the internal network by embedding itself in URLs. For example, an application such as an email client could automatically execute an HTML-embedded URL. If the URL was malicious, damage to the network or the user's system may occur. Access to potentially malicious URLs should be blocked or limited.
- **Detect and Block Attack Signatures** Security gateways should perform content filtering on all data streams to detect and block data patterns that are indicative of attacks, worms, etc.

4) Control Application-Layer Operations

Not only can application-layer communications introduce malicious data to a network, the application itself might perform unauthorized operations. A network security solution must have the ability to identify and control such operations by performing “access control” and “legitimate usage” checks. This level of security requires the capability to distinguish, at a granular level, application operations. Examples include:



Check Point

SOFTWARE TECHNOLOGIES LTD.



We Secure the Internet.

- **Microsoft Networking Services** A network security solution can implement security policy using many parameters from CIFS, the Microsoft-based Common Internet File System. CIFS supports, among other capabilities, file- and print-sharing operations. Using these operations as examples, a security gateway should have the capability to differentiate and block file-sharing operations originating from a user or system that does not have appropriate authorization. Conversely, print-sharing operations originating from the same user may be allowed and accepted. Providing a level of security with this granularity requires a thorough understanding of CIFS, as well as the ability to control application-layer protocol components.
- **FTP** A firewall should place connection restrictions on particular file names and control potentially hazardous FTP commands like PUT, GET, SITE, REST, and MACB. For example, a security policy may require operational restrictions on all files containing the word “payroll.”

Network and Transport Layers: Necessary Foundation for Application Intelligence

Application Intelligence, in its purest form, associates itself with application level defenses. However, in practice many attacks aimed at network applications actually target the network and transport layers. Hackers target these lower layers as a means to access the application layer, and ultimately the application and data itself. Also, by targeting lower layers, attacks can interrupt or deny service to legitimate users and applications (e.g., DoS attacks). For these reasons, Application Intelligence and other network security solutions must address not only the application layer, but also network and transport layers.

NETWORK LAYER SECURITY

Preventing malicious manipulation of network-layer protocols (e.g., IP, ICMP) is a crucial requirement for multi-level security gateways. The most common vehicle for attacks against the network layer is the Internet Protocol (IP), whose set of services resides within this layer. While many network-layer hazards and attacks exist, some examples include:

- **IP Fragmentation** IP fragmentation can be used to deliver and disguise attacks in order to avoid detection. This technique utilizes the resilience mechanisms inherent in the IP protocol itself (RFC 791 and RFC 815) to intentionally fragment attacks into multiple IP packets so they can circumvent firewalls that do not perform IP fragment reassembly. IP fragmentation can also be used to launch a DoS attack by inundating IP fragment reassembly devices with incomplete fragment sequences.
- **Smurfing (smurf Attack)** ICMP allows one network node to ping, or send an echo request to, other network nodes to determine their operational status. This capability can be used to perpetrate a “smurf” DoS attack. The smurf attack is possible because standard ICMP does not match requests to responses. Therefore, an attacker can send a ping with a spoofed source IP address to an IP broadcast address. The IP broadcast address reaches all IP addresses in a given network. All machines within the pinged network send echo replies to the spoofed, and innocent, source IP. Too many pings and responses can flood the spoofed network and deny access for legitimate traffic. This type of attack can be blocked by dropping replies that don’t match requests, as performed by Check Point’s Stateful ICMP.

TRANSPORT LAYER SECURITY

As with the network layer, the transport layer and its common protocols (TCP, UDP) provide popular access points for attacks on applications and their data. Examples of transport layer attacks and threats include:

- **Non-TCP DoS** Non-TCP (e.g., UDP and ICMP) DoS attacks can completely overwhelm mission-critical applications—such as SMTP, HTTP, FTP, etc., which use TCP traffic. Firewalls can protect against these threats by reserving a dedicated portion of the state table for TCP connections. If non-TCP connections attempt to utilize too many resources, TCP connections will be unaffected because they are handled by reserved or dedicated system resources.



Intelligent Security

Check Point
SOFTWARE TECHNOLOGIES LTD.

We Secure the Internet.

- **Port Scan** A port scan does what the name implies: a hacker scans a range of ports on a target host in hopes of identifying and exploiting weaknesses on running applications. The reconnaissance that a port scan performs is a hazard that can lead to an attack. A security gateway must be able to raise alerts, and block or shutdown communications from the source of the scan.

Conclusion

Firewalls have established themselves as the staple of network security infrastructures based on their ability to block attacks at the network level. As a result of firewall success, hackers have developed more sophisticated attack methodologies. The new breed of attacks directly targets applications, often attempting to exploit vulnerabilities inherent in the applications themselves or in the underlying communication protocols. Providing security on multiple levels is required to safeguard corporate networks from these threats. Additionally, multi-level security solutions must protect against both network and application-layer attacks, while providing access control to IT resources.

Check Point Application Intelligence, based on INSPECT, is a set of advanced capabilities, integrated into Check Point FireWall-1 NG and SmartDefense, which detects and prevents application-level attacks. Check Point solutions provide the industry's most proven and comprehensive answer to the increasing number of attacks directed at critical applications.

About Check Point Software Technologies

Check Point Software Technologies is the worldwide leader in securing the Internet. It is the confirmed market leader of both the worldwide VPN and firewall markets. Check Point provides Intelligent Security Solutions for Perimeter, Internal and Web Security. Based on INSPECT, the most adaptive and intelligent inspection technology and, SMART Management, which provides the lowest TCO for managing a security infrastructure, Check Point's solutions are the most reliable and widely deployed worldwide. Check Point solutions are sold, integrated and serviced by a network of 1,900 certified partners in 86 countries. For more information, please call us at (800) 429-4391 or (650) 628-2000 or visit us on the Web at <http://www.checkpoint.com> or at <http://www.opsec.com>.

CHECK POINT OFFICES:

International Headquarters:

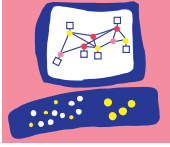
3A Jabotinsky Street, 24th Floor
Ramat Gan 52520, Israel
Tel: 972-3-753 4555
Fax: 972-3-575 9256
e-mail: info@CheckPoint.com

U.S. Headquarters:

800 Bridge Parkway
Redwood City, CA 94065
Tel: 800-429-4391 ; 650-628-2000
Fax: 650-654-4233
URL: <http://www.checkpoint.com>

© 2004 Check Point Software Technologies Ltd. All rights reserved. Check Point, Check Point Express, the Check Point logo, ClusterXL, ConnectControl, FireWall-1, FireWall-1 GX, FireWall-1 SecureServer, FireWall-1 VSX, FireWall-1 XL, FloodGate-1, INSPECT, INSPECT XL, IQ Engine, Open Security Extension, OPSEC, Provider-1, Safe@Office, SecureKnowledge, SecurePlatform, SecureXL, SiteManager-1, SmartCenter, SmartCenter Pro, SmartDashboard, SmartDefense, SmartLSM, SmartMap, SmartUpdate, SmartView, SmartView Monitor, SmartView Reporter, SmartView Status, SmartViewTracker, UAM, User-to-Address Mapping, UserAuthority, VPN-1, VPN-1 Accelerator Card, VPN-1 Pro, VPN-1 SecureClient, VPN-1 SecuRemote, VPN-1 SecureServer, , and VPN-1 VSX are trademarks or registered trademarks of Check Point Software Technologies Ltd. or its affiliates. All other product names mentioned herein are trademarks or registered trademarks of their respective owners. The products described in this document are protected by U.S. Patent No. 5,606,668 and 5,835,726 and may be protected by other U.S. Patents, foreign patents, or pending applications.





Check Point Multi-Layer Security: Attack Prevention Safeguards and Attacks Blocked

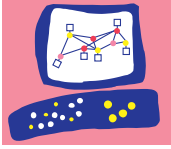
FireWall-1 NG with Application Intelligence blocks many attacks and provides numerous attack prevention safeguards. This table lists some of these defenses and organizes them by protocol and OSI Model layer.

Note: Check Point continually expands the breadth of defenses provided. This table is a snapshot not an exhaustive list. List as of July 16, 2003.

Application Layer/ Presentation Layer	Session Layer	Transport Layer	Network Layer
---------------------------------------	---------------	-----------------	---------------

Application Layer/Presentation Layer

	ATTACK PREVENTION SAFEGUARDS	ATTACKS BLOCKED
HTTP Client	<ul style="list-style-type: none"> • Block Java code • Strip script tags • Strip applet tags • Strip FTP links • Strip port strings • Strip ActiveX tags • Camouflage default banner • URL filtering • Limit maximum URL length • Limit maximum number of response headers allowed • Limit maximum request header length • Limit maximum response header length • Prohibit binary characters in HTTP response headers • Prohibit binary characters in HTTP requests • Validate HTTP response protocol compliance • Block user-defined URLs • Enforce maximum GET and POST length • Restrict download of user-defined files 	<ul style="list-style-type: none"> • Code Red Worm & Mutations • Nimda Worm & Mutations • HTR Overflow Worm & Mutations • Directory Traversal Attacks • MDAC Buffer Overflow & Mutations • Cross-Site Scripting Attacks • Malicious URLs • User-Defined Worms & Mutations
HTTP Server	<ul style="list-style-type: none"> • Limit maximum URL length • Distinguish between different HTTP v1.1 requests over same connection • Limit maximum number of response headers • Limit maximum request header length • Limit maximum response header length • Prohibit binary characters in HTTP response headers • Prohibit binary characters in HTTP requests • Block user-defined URLs • Restrict non-RFC HTTP methods • Enforce HTTP security on non-standard ports (ports other than 80) • Compare transmission to user-approved SOAP scheme/template • Restrict unsafe HTTP commands • Restrict download of user-defined files 	<ul style="list-style-type: none"> • Encoding Attacks • Cross-Site Scripting Attacks • HTTP-based attacks spanning multiple packets • WebDAV Attacks • User-Defined Worms & Mutations • Chunked Transfer Encoding Attacks
SMTP	<ul style="list-style-type: none"> • Block multiple "content-type" headers • Block multiple "encoding headers" • Camouflage default banner • Restrict unsafe SMTP commands • Header forwarding verification • Restrict unknown encoding • Restrict mail messages not containing sender/recipient domain name • Restrict MIME attachments of specified type • Strip file attachments with specified names • Strict enforcement of RFC 821 & 822 • Monitor and enforce restrictions on ESMTP commands • Hide internal mail user names and addresses • Perform reverse DNS lookup • Strict enforcement of MAIL and RCPT syntax • Restrict mail from user-defined sender or domains 	<ul style="list-style-type: none"> • SMTP Mail Flooding • SMTP Worm & Mutations • Extended Relay Attacks • Message/ Partial MIME Attack • SPAM Attack (large number of emails) • Command Verification Attack • SMTP Worm Payload & Mutations • Worm Encoding • Firewall Traversal Attack • SMTP Error Denial-of-Service Attack • Mailbox Denial-of-Service Attack (excessive email size) • Address Spoofing • SMTP Buffer Overflow Attacks



Check Point Multi-Layer Security: Attack Prevention Safeguards and Attacks Blocked

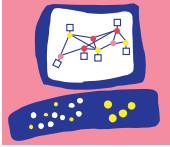
FireWall-1 NG with Application Intelligence blocks many attacks and provides numerous attack prevention safeguards. This table lists some of these defenses and organizes them by protocol and OSI Model layer.

Note: Check Point continually expands the breadth of defenses provided. This table is a snapshot not an exhaustive list. List as of July 16, 2003.

Application Layer/ Presentation Layer	Session Layer	Transport Layer	Network Layer
---------------------------------------	---------------	-----------------	---------------

Application Layer/Presentation Layer

	ATTACK PREVENTION SAFEGUARDS	ATTACKS BLOCKED
SMTP	<ul style="list-style-type: none"> Restrict mail to user-defined recipients Restrict mail to unknown domains Enforce limits on the number of RCPT commands allowed per transaction Restrict mail relay usage 	
RSH	<ul style="list-style-type: none"> Auxiliary port monitoring Restrict reverse injection 	
RTSP	<ul style="list-style-type: none"> Auxiliary port monitoring 	
IIOP	<ul style="list-style-type: none"> Auxiliary port monitoring 	
FTP	<ul style="list-style-type: none"> Analyze and restrict hazardous FTP commands Block custom file types Camouflage default banner Strip FTP references 	<ul style="list-style-type: none"> Passive FTP Attacks FTP Bounce Attack Client and Server Bounce Attacks FTP Port Injection Attacks Directory Traversal Attack Firewall Traversal Attack TCP Segmentation Attack
DNS	<ul style="list-style-type: none"> Restrict DNS zone transfers Restrict usage of DNS server as a public server Provide separate DNS service for private vs. public domains 	<ul style="list-style-type: none"> DNS Query Malformed Packet Attacks DNS Answer Malformed Packet Attacks DNS Query Buffer Overflow - Unknown Request/Response Man-in-the-Middle Attack
Microsoft Networking	<ul style="list-style-type: none"> CIFS filename filtering (protect against worms utilizing CIFS protocol) Restrict remote access to registry Restrict remote null sessions 	<ul style="list-style-type: none"> Bugbear Worm Nimda Worm Liotan Worm Opaserv Worm
SSH	<ul style="list-style-type: none"> Enforce SSH v2 protocol 	<ul style="list-style-type: none"> SSH v1 Buffer Overflow Attack
SNMP	<ul style="list-style-type: none"> Restrict SNMP get/put commands 	<ul style="list-style-type: none"> SNMP Flooding Attack Default Community Attacks Brute Force Attacks SNMP Put Attack
MS SQL		<ul style="list-style-type: none"> SQL Resolver Buffer Overflow SQL Slammer Worm
Oracle SQL	<ul style="list-style-type: none"> Verify dynamic port allocation and initiation 	<ul style="list-style-type: none"> SQLNet v2 Man-in-the-Middle Attack
SSL	<ul style="list-style-type: none"> Enforce SSL V3 protocol 	<ul style="list-style-type: none"> SSL V2 Buffer Overflow
VoIP	<ul style="list-style-type: none"> Verify protocol fields and values Identification and restriction of the PORT command Enforce existence of mandatory fields Enforce user registration Prevent VoIP firewall holes Disable H.323 audio and video transmissions Enforce H.323 call duration limits For H.323, allow only traffic associated with a specific call 	<ul style="list-style-type: none"> Buffer Overflow Attacks Man-in-the-Middle Attack
X11	<ul style="list-style-type: none"> Restrict reverse injection Block special clients 	



We Secure the Internet.

Check Point Multi-Layer Security: Attack Prevention Safeguards and Attacks Blocked

FireWall-1 NG with Application Intelligence blocks many attacks and provides numerous attack prevention safeguards. This table lists some of these defenses and organizes them by protocol and OSI Model layer.

Note: Check Point continually expands the breadth of defenses provided. This table is a snapshot not an exhaustive list. List as of July 16, 2003.

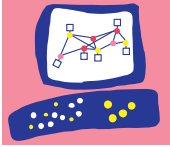
Application Layer/ Presentation Layer	Session Layer	Transport Layer	Network Layer
---------------------------------------	---------------	-----------------	---------------

Session Layer

	ATTACK PREVENTION SAFEGUARDS	ATTACKS BLOCKED
RPC	<ul style="list-style-type: none"> Block RPC portmapper exploits 	<ul style="list-style-type: none"> ToolTalk Attacks snmpXdmid Attack rstat Attack mountd Attack cmsd Attack cachefs Attack
DEC-RPC	<ul style="list-style-type: none"> Block DCE-RPC portmapper exploits 	
HTTP Proxy	<ul style="list-style-type: none"> HTTP Proxy enforcement: Enforce HTTP session logic in proxy mode 	
VPN	<ul style="list-style-type: none"> Validate digital certificates used against Certificate Revocation List Monitor for pre-shared secrets vulnerability 	<ul style="list-style-type: none"> IKE Brute Force Attack Hub-and-Spoke Topology Attack IKE UDP DoS Attack Windows 2000 IKE DoS Attack VPN IP Spoofing Attack VPN Man-in-the-Middle Attacks

Transport Layer

	ATTACK PREVENTION SAFEGUARDS	ATTACKS BLOCKED
TCP	<ul style="list-style-type: none"> Enforce correct usage of TCP flags Limit per-source sessions Enforce minimum TCP header length Block unknown protocols Restrict FIN packets with no ACK Enforce that TCP header length as indicated in header is not longer than packet size indicated by header Block out state packets Verify that first connection packet is SYN Enforce 3-way handshake: Between SYN and SYN-ACK, client can send only RST Enforce 3-way handshake enforcement: Between SYN and connection establishment, server can send only SYN-ACK or RST Block SYN on established connection before FIN or RST packet is encountered Restrict server-to-client packets belonging to old connections Drop server-to-client packets belonging to old connections if packets contain SYN or RST Enforce minimum TCP header length Block TCP fragments Block SYN fragments Scramble OS fingerprint Verify TCP packet sequence number for packets belonging to an existing session 	<ul style="list-style-type: none"> ACK Denial-of-Service Attack SYN Attack Land Attack Tear Drop Attack Session Hijacking Attack Jolt Attack Bloop Attack Cpd Attack Targa Attack Twinge Attack Small PMTU Attack Session Hijacking Attacks (TCP sequence number manipulation) TCP-Based Attacks Spanning Multiple Packets XMAS Attacks Port Scan
UDP	<ul style="list-style-type: none"> Verify UDP length field Match UDP requests and responses 	<ul style="list-style-type: none"> UDP Flood Attacks Port Scan



We Secure the Internet.

Check Point Multi-Layer Security: Attack Prevention Safeguards and Attacks Blocked

FireWall-1 NG with Application Intelligence blocks many attacks and provides numerous attack prevention safeguards. This table lists some of these defenses and organizes them by protocol and OSI Model layer.

Note: Check Point continually expands the breadth of defenses provided. This table is a snapshot not an exhaustive list. List as of July 16, 2003.

Application Layer/ Presentation Layer	Session Layer	Transport Layer	Network Layer
---------------------------------------	---------------	-----------------	---------------

Network Layer

	ATTACK PREVENTION SAFEGUARDS	ATTACKS BLOCKED
IP	<ul style="list-style-type: none"> • Enforce minimum header length • Restrict IP-UDP fragmentation • Enforce that header length indicated in IP header is not longer than packet size indicated by header • Enforce that packet size indicated in IP header is not longer than actual packet size • Scramble OS fingerprint • Control IP options 	<ul style="list-style-type: none"> • IP Address Sweep Scan • IP Timestamp Attack • IP Record Route Attack • IP Source Route Attack • IP Fragment Denial-of-Service Attack • Loose Source Route Attack • Strict Source Route Attack • IP Spoofing Attack
ICMP	<ul style="list-style-type: none"> • Block large ICMP packets • Restrict ICMP fragments • Match ICMP requests and responses 	<ul style="list-style-type: none"> • Ping-of-Death Attack • ICMP Flood