



Request for Information

Technology Opportunities for Addressing Issues
Associated with Peer-to-Peer File Sharing
on the University and College Campus

White Paper: Passive Network Traffic Blocking Technology

Palisade System's Core Technology

OVERVIEW

The implementation of open Internet standards such as TCP/IP networking and browsers facilitate the ease of access and interoperability so crucial in the E-business world, but in doing so can greatly decrease the security of an organization's network. Many organizations have therefore purchased or built security systems. But those systems focus their security efforts primarily at the organization's perimeter, on firewalls to prevent unauthorized access from outside, content and virus scanning to restrict what enters the organization from outside, and VPN/encryption tools to protect information when it travels outside the perimeter.

Palisade Systems additionally recognizes the importance of monitoring and controlling local network traffic to ensure internal security and optimize productive use of the Internet. Palisade's patented Passive Network Traffic Blocking Technology (protected by U.S. Patent # 6,044,402) is a bundled hardware/software solution used to keep unwanted applications from running on a network, allowing administrators to control local network traffic and detect potential security risks.

This core technology allows Palisade Systems to deliver products with superior performance, security, and stealth. It sits passively on the network, unobtrusively examining traffic for prohibited connections. When detected, the technology impersonates the host end of the connection and issues a reset packet to the requesting (client) end, immediately causing the connection to terminate.

This white paper describes the intended design objectives of the technology, overviews the functions and possible uses of the technology, and closes with a discussion of the technology's theory of operation and means of implementation.

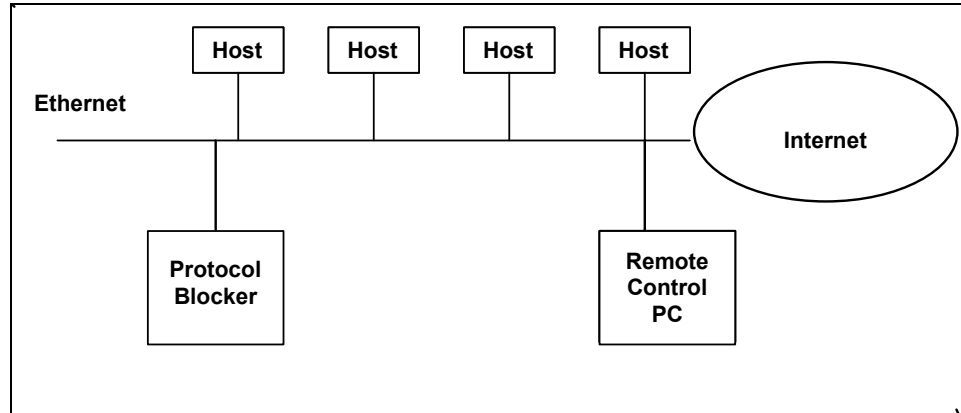
DESIGN OBJECTIVES

The Passive Network Traffic Blocking Technology is designed to control and monitor traffic on a subnet connected to a large internetwork like the global Internet. It allows administrators to use a time-based rule set to block applications they wish to block; for example, they could disable a network game during the day but allow the game to run at night. When a user attempts to use a blocked application, the technology issues a reset packet, causing the application to terminate. In addition, the technology was designed to be

- Low cost,
- Simple to operate,
- Easily integrated into any network running TCP/IP regardless of operating systems, machines, or applications,
- Secure, being virtually impossible to identify, locate, circumvent, or dismantle, even for the most sophisticated and motivated hacker,
- Efficient, contributing absolutely no overhead to an organization's network, thus avoiding network performance degradation and increased resource requirements.

To elaborate this last point, an example network setting for the Protocol Blocker is shown in Figure 1. The technology is a peer on the network, and therefore traffic does not pass through the blocker, affecting network throughput by funneling all Internet traffic through a single point.

Figure 1. Typical Configuration: A remote PC controls the Protocol Blocker over the network.



FUNCTIONS

The primary functions of the Protocol Blocker are as follows:

Shuts down active connections based on

- application type
- network address (source and/or destination)
- time of day
- network load

Limits total number of connections allowed on the network based on

- application type
- network load
- time of day
- network address (source and/or destination)

Prevents the use of unassigned network addresses

Monitors connections based on

- application type
- network load
- time of day
- network address (source and/or destination)

Limits/eliminates WWW pages from being downloaded based on

- network address
- network load
- time of day

POSSIBLE USES

Possible uses for the Protocol Blocker are as follows:

Network security

- blocks unwanted connections
- monitors connections
- protects network address
- restricts access to unused ports
- monitors port usage

Restricted access to inappropriate material

- restricts access to selected sites
- allows access to only selected sites
- monitors traffic
- blocks access to specific applications

Network load protection

- shuts down non essential protocols during high network usage
 - shuts down non-essential machines during high network usage
-

THEORY OF OPERATION

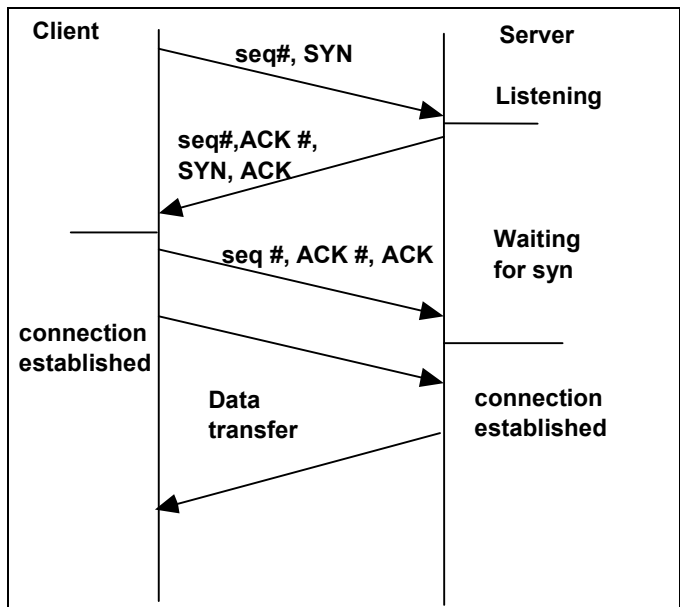
The Protocol Blocker monitors all traffic on the network for connections. Once a connection has been identified, it is compared against the blocker’s database to determine if it is a new connection and if the connection should be blocked. If the connection is new, it is added to the database, and if the connection should be blocked, a RESET packet is sent to both the requesting and the requested machines.

The current implementation is based on TCP (see Figure 2), but the same technique can be applied to any connection oriented protocol. TCP connections are uniquely identified by their two endpoints:

- Connection = {TCP, endpoint, endpoint}
- Endpoint = {IP address, port number}

HTTP selective blocking is achieved by examining the contents of an HTTP request. HTTP requests use this form: GET //host/path/filename.Extension ... Current practice is that the extension indicates the type of requested item. A request always comes in the first segment after the three-way handshake. If the client receives a RESET segment immediately after sending the HTTP request, the connection will be closed, and the client will not receive any data. This approach requires that the Protocol Blocker receives the request segment.

Figure 2. Protocol exchange for a TCP connection.

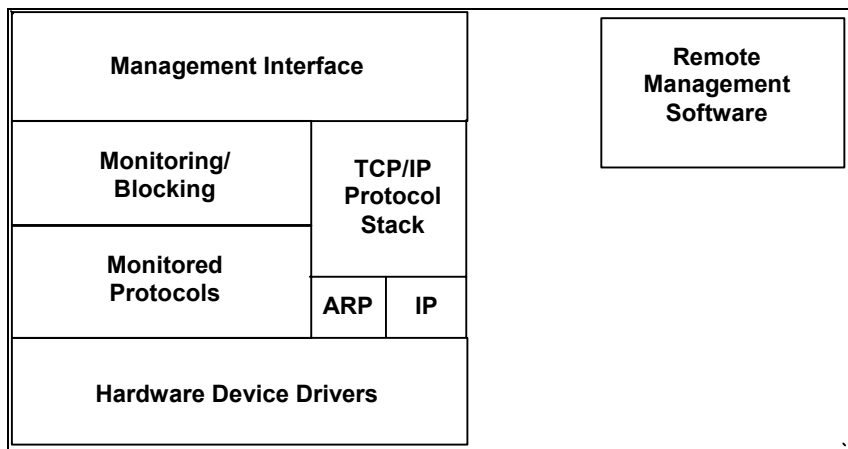


IMPLEMENTATION The Protocol Blocker runs on a standard personal computer motherboard with an Ethernet card and floppy drive. It has been implemented on both DOS and Unix operating systems.

Protocol Blocker Software

Figure 3 shows the basic structure of the Protocol Blocker software. The software is highly modularized, allowing for easy additions of new protocols and functions. Each module is separate and independent of the other modules except for an overlap that exists between the ARP protocol module and the ARP part of the TCP/IP stack.

Figure 3. Software Block Diagram



This modularized design is parallel to the way TCP/IP protocols are organized in independent layers. These modules share common data and functionality, making it possible to implement the core functions and data structures only once, allowing data to be passed between the layers as pointers and allowing

each layer to access only its limited subset of the data. This is a compromise that favors efficiency instead of strict separation of the layers.

The software is organized in two layers: the low-level hardware drivers and the monitor software. The hardware drivers are interrupt driven and talk directly to the hardware devices; the network interface driver, for example, buffers incoming data and puts it on a queue for processing by the main program. The main program then processes the enqueued data and responds to events as signaled by the hardware drivers. This is different from most conventional systems where processing of network input is interrupt driven.

Low-level hardware drivers. The driver for the software version conforms to the standard device driver implementations found on a MS-DOS/Windows PC. The device driver provides a standard interface to the network interface card. This device driver differs for the standard packet drivers in that the data is never moved from the driver to the upper layers of the software. Only a pointer to the data is returned from the driver.

TCP/IP stack overview

The TCP/IP stack interfaces the hardware through a hardware-specific network driver and accesses the application software through the socket layer. The hardware driver is very simple; its function is to copy received and to transmit outgoing Ethernet frames on a first come, first served basis. Received frames are put on queues to be processed by corresponding protocol stacks. The socket layer in the BSD implementations of UNIX is a very complex library that creates a standard interface between processes and almost any hardware device. The Protocol Blocker is a application-specific platform that does not require this kind of generality or complexity, and the application interface of the Protocol Blocker protocol stack has been simplified and optimized by removing system overhead. The optimizations and differences will be pointed out throughout the following presentation of the Protocol Blocker TCP/IP stack.

The Protocol Blocker implementation of the TCP/IP stack is different from the BSD stack in the following areas that are briefly described in this section:

- buffer/memory organization
- signaling/process handling
- source routing support
- ARP implementation

The ARP module. The Address Resolution Protocol (ARP) module takes care of address resolution for the TCP/IP stack. When the IP protocol asks the ARP protocol for the hardware address corresponding to an IP address, it does an ARP lookup. If the local cache does not contain the needed information, a request will be broadcasted on the network. The host that owns the requested IP address (or a proxy server) will then send an ARP response back with the hardware address. The core of the ARP module is the database of ARP entries. In a traditional ARP implementation, these entries are cached for a short time (20 minutes) in the database. A lookup for an address that is not cached causes an ARP request to be broadcasted on the network. An entry is added to the cache in one of two cases; when there is a lookup for an address, and when a

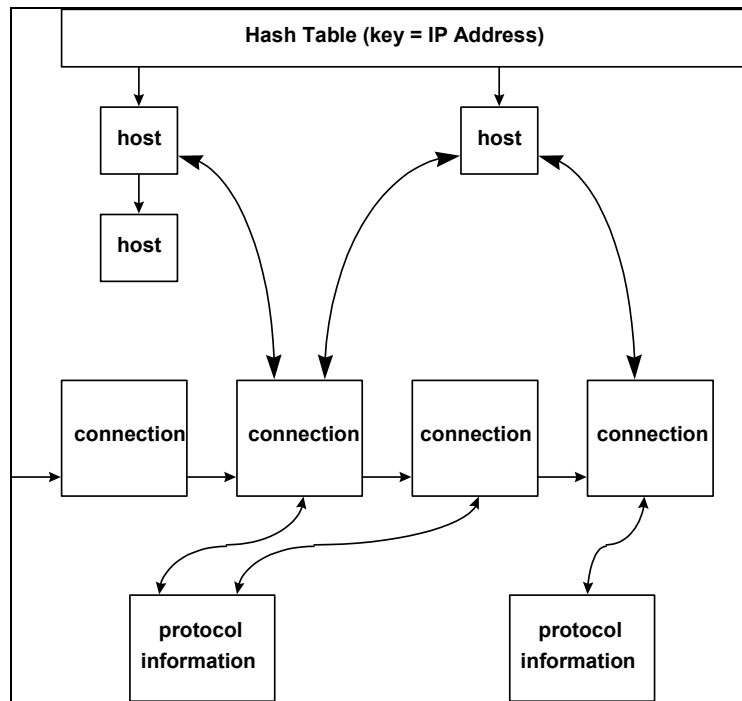
request for a local address is received.

Address resolution in the TCP/IP stack. The ARP implementation of the Protocol Blocker differs a lot from the BSD version. ARP is monitored by the Protocol Blocker as a separate protocol, and the TCP/IP stack uses the data collected by the ARP module to resolve addresses.

The IP module. The IP module does the decoding necessary before packets are passed on to their corresponding transport protocols.

The TCP module. The main part of the Protocol Blocker blocking functions is done in the Transmission Control Protocol (TCP) module. The main components of this module are a protocol database and a connection database. The protocol database contains information about application level protocols, which is Protocol Blocker configuration data. As the Protocol Blocker monitors the network traffic and sees TCP connections being created and disconnected, the connection database is continuously being updated with information about the current connections. The data it contains is used for two purposes: First, information about connections that have been analyzed is stored in the database. This way the Protocol Blocker will know what traffic it has seen before and know to ignore a connection it has decided to allow. Second, historical data is used to detect anomalies (e.g., probing of systems). The structure of the connection database is shown in Figure 5.

Figure 5. Connection database



OPERATION

Determining the state of a new connection

When the Protocol Blocker sees a TCP segment for a connection it does not find in its database, it will first try to determine the state of the connection. The current implementation will only check if the segment is the initial segment of the new connection—the first segment of any TCP connection should have the SYN flag set and the ACK flag not set. A record for the connection is then added to the database with the state either set to initial SYN or established.

If the state of a connection is initial SYN, one more valuable piece of information can be drawn from the TCP segment: who is the client and who is the server. It is the client end of a connection that sends the initial SYN segment that starts the establishment sequence.

The protocol database

The protocol database contains information about all the application-level TCP protocols known to the Protocol Blocker. This is similar to the information in the file `/etc/services` on a UNIX machine, but with more information for each protocol than just a port number and a service name. The extra information tells the Protocol Blocker whether the protocol should be blocked or not and how to handle the blocking. Some protocols are different from the standard Internet services; looking for string fragments instead of port numbers identifies these.

It is possible for a protocol entry to have a special function handler associated with it. An example handler exists for the HTTP protocol, which looks for specific patterns in the requests from the HTTP client. This makes it possible for the client to block requests for World Wide Web objects like graphics to reduce the network load.

Determining connection types

The Protocol Blocker examines the packet header to determine the protocol of a connection. An entry is added to the connection database for each new connection, if a protocol is found to match the new connection, the blocking policy for the protocol is used to mark the entry as blocked or not. If no protocol was found, the entry is stored, but it is never blocked.

The connection database

The connection database contains a list of IP addresses (hosts) and a list of connections. Each host has a pointer to a linked list of connection entries, and each connection entry has a pointer to each of the two hosts that are connected. A connection entry is removed after a fixed interval without any traffic belonging to it, not when the connection is terminated. This ensures the database contains historical data that facilitates detection of anomalies such as a remote system probing a local host.

The host entries in the database are removed as soon as there are no connections associated with them. Whenever the Protocol Blocker system load is low, the list of all the connections is traversed to weed out inactive connections.

Terminating connections

A reset segment is usually sent to block a connection, and it is generated by using a received segment as a template. The TCP header fields are updated as

described above, and the data is removed before the segment is sent to the destination or back to its original source. The use of finish segments to block connections is still at an experimental stage, and the main problem with this method is the sometimes long delay from the time a segment is received by the Protocol Blocker until it is processed.

APPLICATIONS

To date, Palisade has leveraged this core technology in two products:

- **ScreenDoor Internet Management Appliance.** ScreenDoor is a bundled hardware/software appliance that resides on an organization's network and manages access to over 500,000 inappropriate sites divided into some 27 categories. In addition to managing Internet access, ScreenDoor can be used to monitor and manage internal network functions such as various network protocol use, IP address use, and internal system access. ScreenDoor was originally designed specifically for the educational market to be inexpensive, simple to use, and easy to implement. Since its introduction in 1997, ScreenDoor has been installed in hundreds of libraries, schools, and colleges across the country as well as recently in a number of commercial accounts.
- **PacketHound Protocol Management Appliance.** PacketHound is a standalone network appliance designed to address productivity, bandwidth, and liability issues raised by use of various network protocols and file sharing applications including Gnutella, Napster, Imesh, Scour Exchange, Real Audio/Video, Windows Streaming Media, and Shoutcast. Existing firewall and network blocking solutions cannot effectively block these applications because they rely on IP address or TCP port use for application identification. PacketHound effectively and completely eliminates use of these applications on a network because it detects the applications' fundamental characteristics. Future versions, derivations, or completely new applications will be added as they develop.

For more information on these and other Palisade products, visit us at www.palisadesys.com.

Table of Contents

1.0	Technology	1
1.1	Using PacketHound	1
1.1.1	Configure PacketHound.....	1
1.1.2	PacketHound Operations	2
1.2	Audit Tools	3
1.3	Bandwidth Shaping.....	4
1.4	Data/File Sharing Blocking.....	5
1.5	Matching, Screening, and Filtering.....	7
1.6	Network Performance	8
1.7	Network Architecture.....	8
1.8	Scalability	9
1.9	Protocol Identification	10
1.10	Granularity of Protocols.....	10
1.11	Content Identification	11
1.12	Examination of Network Packets or File Content	11
1.13	Distribution Systems.....	11
1.14	Resilience of the Technology to Countermeasures.....	12
1.15	Testing and Installed Base	12
1.16	Competitive Approaches.....	13
1.17	Third-party Components.....	14
1.18	Intellectual Property.....	14
2.0	Corporate Characteristics and Resources.....	15
3.0	Pilot Testing.....	15
4.0	Commercial Terms.....	16
5.0	Additional Information	17
Appendix A – Applications PacketHound Manages		18
File Sharing Applications		18
Instant Messaging		21
Multi-Media		22
HTTP Protocols		22
Ads/Spyware.....		23

Vulnerabilities.....	24
Mail Protocols.....	25
Web-based Mail.....	25
FTP.....	25
HTTP Information.....	26
Remote Login.....	26
Distributed Computing.....	27
Other.....	27
Special Rules.....	27
Appendix B – Traffic Analysis Criteria.....	29
Appendix C – Palisade Core Technology White Paper.....	30

1.0 Technology

Many universities are looking for quick and cost-effective solutions to put out the fires caused by peer-to-peer (P2P) file sharing. They are faced with the growing threat of legal liability from the recording industry because of students downloading copyrighted movies and music and from law enforcement authorities for child pornography. These applications are also a security threat as a source of viruses, worms, and spyware, which can be used by hackers to obtain and alter confidential files without the knowledge of the administrator. In addition, these files also consume large amounts of critical bandwidth, further straining budgets already stretched thin.

Palisade System's PacketHound is an award-winning appliance specifically designed to eliminate the liability, bandwidth, and security issues caused by P2P applications. In addition to managing P2P issues, PacketHound is a key solution for monitoring and managing virtually any internal network application or activity. It also provides flexibility and functionality that allows each college and university to adjust enforcement to meet the needs of its environment and users.

PacketHound is based on the core technology protected by U.S. Patent #6,044,402. This patented technology, developed at Iowa State University, includes a passive network monitoring capability that makes it possible to monitor or block network applications without degrading network performance. PacketHound examines network traffic and looks for data in packets that match criteria defined in the PacketHound rules. When matching data is found, the connection can be blocked, logged, or ignored, as specified by the administrator, for every matched rule. PacketHound passively monitors virtually any application or protocol running on the network and provides a real-time graphical display of the activities.

1.1 Using PacketHound

PacketHound is an appliance and software combination that manages many different file sharing applications, as well as other network protocols and applications. See Appendix A for a complete list of all protocols and applications included with the PacketHound appliance. In addition to these, as new applications or protocols are added, they are available to download from the Palisade Systems web site.

1.1.1 Configure PacketHound

PacketHound can be configured from a Windows-based Administrator or from a Web-based Administrator. The Windows Administrator has all the configuration options and the Web Administrator has a large subset of the configuration options. These Administrators are used to configure the appliance to manage network protocols or applications as you need. PacketHound's features can be distributed among all classes of tools referenced in section 5.1 Technology of the Request for Information (RFI).

A PacketHound configuration consists of Administrator and appliance preferences, alert filters, and a set of rules that define the network protocols or applications to manage. A rule can be set to ignore, log, block and log, or block without logging.

- **Ignore** allows all instances of the protocol to be accessed without logging.
- **Log** keeps track of each time the application is accessed from the network.
- **Block** sends a reset packet, terminates the connection, and logs the attempt to access.
- **Block without Logging** sends a reset packet to terminate the connection, but does not log the attempt to access.

Each rule has its own set of criteria that must be matched when PacketHound examines the packets in the transmission. You can create multiple rules for the same application. The available criteria for a rule includes, but is not limited to:

- Traffic direction to match: Incoming, Outgoing, Internal.
- Days and times the rule is effective.
- Match IPs in a packet.
- Bandwidth limits to shape or throttle the application traffic.
- List of specific web sites to block (only for HTTP rules).
- Defined expressions to match in the packets, such as
 - an exact string
 - a string that begins with or contains specified characters
 - partial or complete URL
 - source or destination IPs or ports
 - TCP sequence and acknowledgement numbers
 - Time To Live (TTL) value

1.1.2 PacketHound Operations

Once the rule is defined, and the rules are updated on the appliance, PacketHound begins to manage those applications you have designated with the rules you defined. You can see exactly how PacketHound is working in real time using the TrendReporter application included with PacketHound. TrendReporter is a graphical display of the traffic PacketHound is managing, monitoring both PacketHound rules and web sites being accessed.

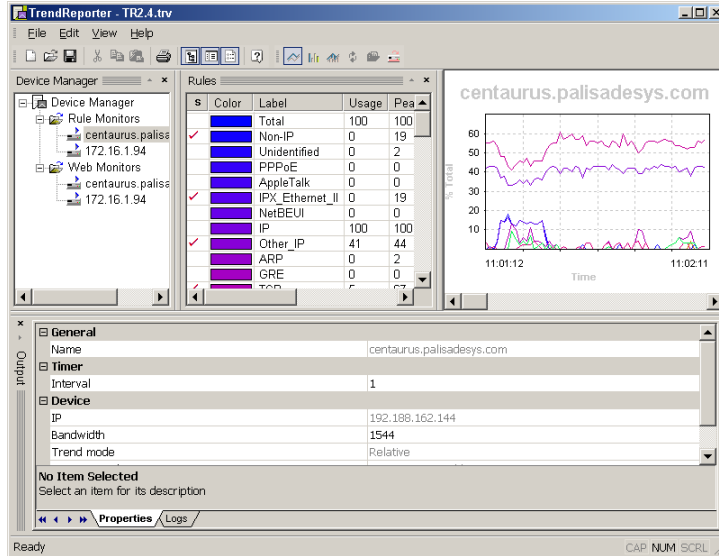


Figure 1 TrendReporter: Real-time Graphical Display

1.2 Audit Tools

PacketHound provides logging capabilities that can be customized to control the size of the database. Data from network connections and the use of network applications is gathered in logs on the PacketHound appliance or on the Administrator machine. The log information is inserted into a database to be available for generating reports.

Reports are generated from the information in the database using the Report Viewer.

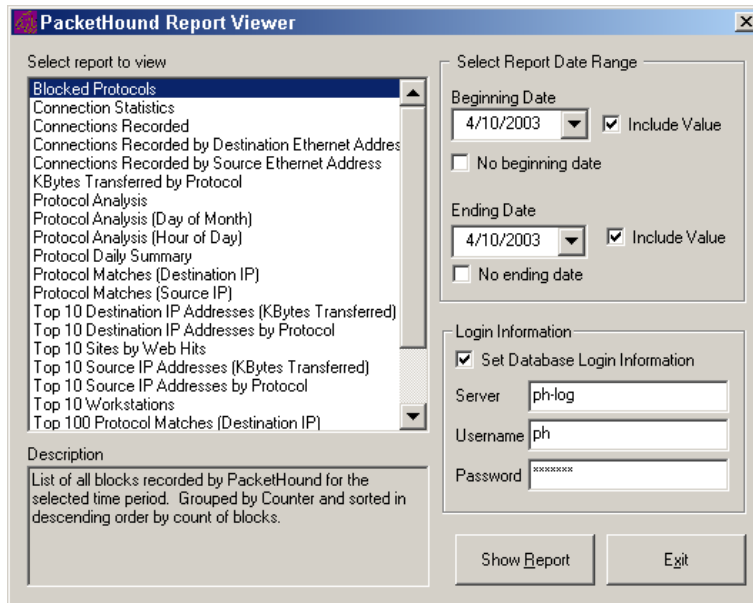


Figure 2 Report Viewer

Types of reports that are available include:

- Connection data
- Protocol analysis

- Destination and source IP addresses sorted by protocol, protocol matches, or amount of traffic.
- Web sites sorted by workstation or by site.

Reports are generated for a range of dates, for up to a month prior to the current date, and are valuable tools for analyzing network traffic and providing information you can use to customize the PacketHound configuration. A report is displayed in a separate window, similar to the following report.

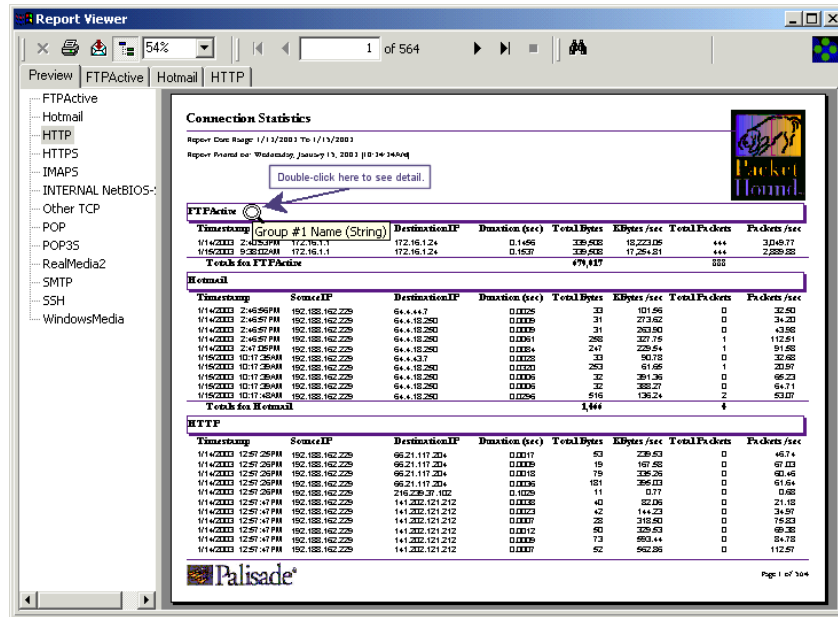


Figure 3 Sample Report

You can view, print, or export the report from this window. The Report Viewer also has several features designed to allow you to “drill down” and view details about data in a report.

- List of sections at the left edge of the window. Click to view the section.
- Capability to view details about a section on a new page. Move the cursor over elements of the report. When you see the magnifying glass symbol, double-click to see details about that element of the report.
- Each time you view details, a new tab is created in the window to allow you to see any page of the report by selecting a tab.

1.3 Bandwidth Shaping

PacketHound can control the amount of bandwidth allocated to each application to limit the effect on network traffic. Shaping is set for individual rules and restricts the amount of bandwidth the application can use, slowing the download speed for protocols matched by the rule. All data is transferred, but is transferred at a slower rate. Bandwidth shaping levels can apply to total network traffic or total traffic for a specified set of applications or rules, designated by a custom bandwidth category name.

Shaping is only effective for rules that are set to Log, but not for rules set to Block. When a rule is set to block, entering a bandwidth threshold causes the rule to be throttled rather than shaped. Throttling is described in the next section.

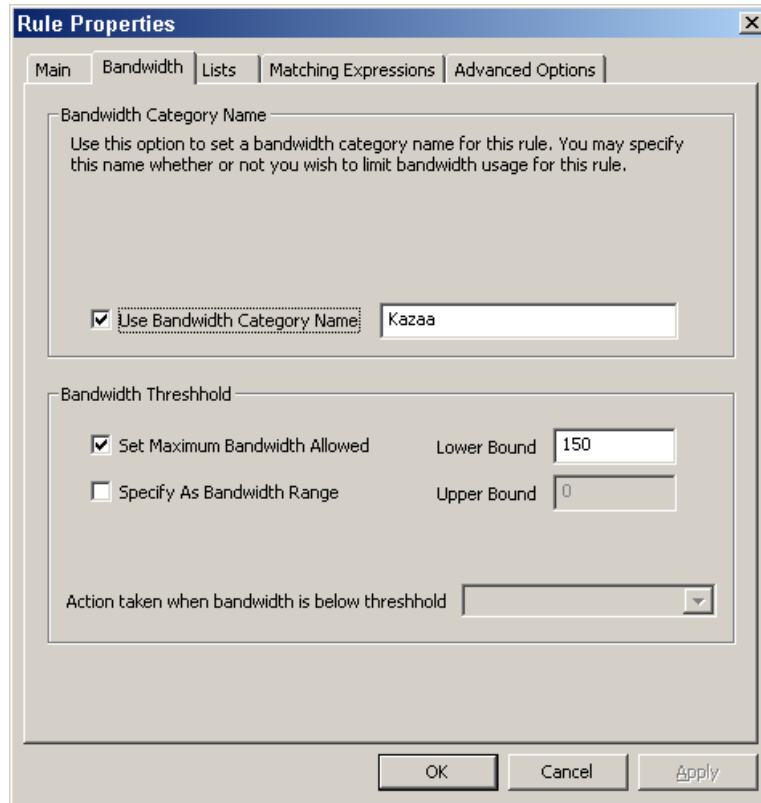


Figure 4 Shaping Kazaa Traffic

Shaping values are set in Kbps (from 1 Kbps to total available bandwidth). A value of zero indicates shaping is off. For example, if you have a T1 line with 1.5 Mbps capability, and you want to limit Kazaa traffic to 150 Kbps:

- For the KazaaXfer rule, enter 150 in the Lower Bound box.
- This sets the targeted throughput to 150 Kbps for Kazaa file transfers, limiting the amount of network bandwidth Kazaa transfers can use.

1.4 Data/File Sharing Blocking

PacketHound allows the user to proactively block virtually any application using a default library of protocols. The blocking abilities are flexible and give network administrators the power to customize policies on how and when P2P file sharing can occur within their organization. For example, PacketHound enables organizations to block applications such as KaZaA and Morpheus from an entire network, a cluster of machines, or for a single user. The appliance also allows administrators to create time-based rules which permit the use of an application only at approved time periods (e.g., block, shape or throttle traffic from 8-5, and all other times monitor).

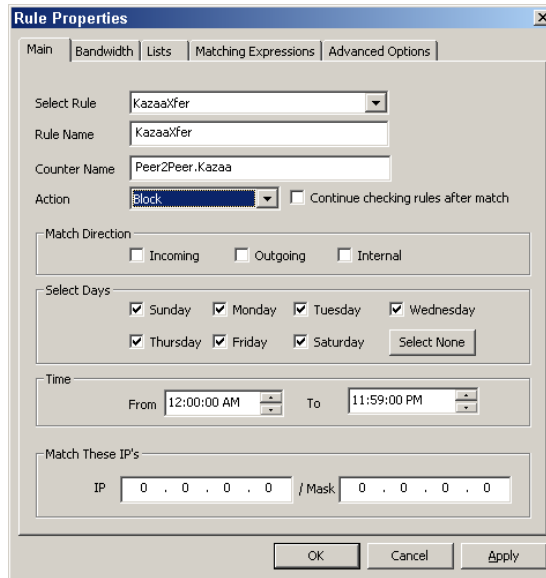


Figure 5 Blocking Kazaa Traffic

PacketHound can also be set to throttle network traffic. Throttling is setting a rule to block only when network traffic exceeds a specified bandwidth threshold. A rule can be logged or ignored when traffic falls below the specified level. Bandwidth throttling levels can apply to total network traffic or total traffic for a specified set of applications or rules, designated by a custom bandwidth category name.

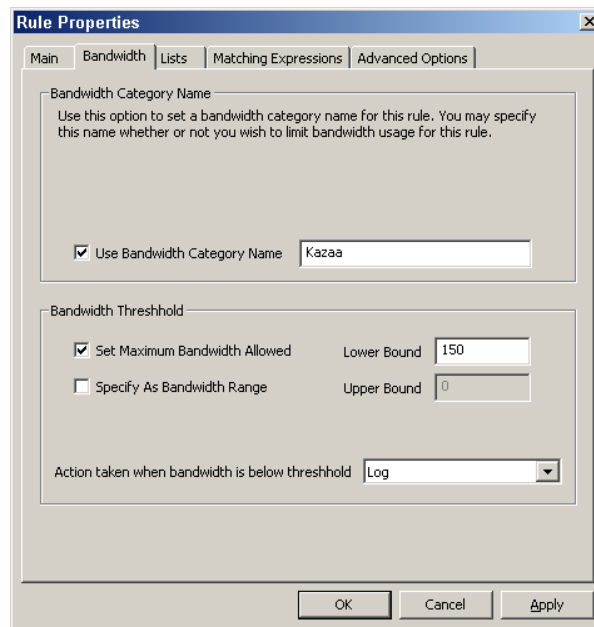


Figure 6 Throttling Kazaa Traffic

Once the specified bandwidth level is reached, connections for that application are terminated with a reset packet. Throttling is only effective for rules that are set to Block, but not for rules set to Log. Blocking and throttling rules can vary per machine, IP, and application.

1.5 Matching, Screening, and Filtering

One or more matching expressions can be defined for a custom rule or added to a default rule. Defining matching expressions narrows the focus of the packet search by making the rule matching more specific and improves PacketHound performance. *All* strings defined for the rule must be matched before the rule becomes effective.

Expression Types include:

- Regular Expression - alpha-numeric characters and symbols used to search for text.
- Exact Match Contains - matches any string in the packet that contains the characters entered.
- Exact Match Starts - matches any string in packet that begins with the characters entered.
- URL Contains - matches any URL that contains the characters entered.
- Port Match - matches the designated port number.
- Source Port - matches the designated source port number.
- Source Port Range - matches any port number in the designated range of source port numbers.
- Destination Port - matches the designated destination port number.
- Destination Port Range - matches any port number in the designated range of destination port numbers.
- Source IP Match - matches the source IP address.
- Destination IP Match - matches the destination IP address.
- TCP Sequence Number Match - matches the designated sequence number. A sequence number is a number assigned to a packet indicating the order in which the packets are arranged.
- TCP Acknowledgement Match - matches the designated acknowledgement number. An acknowledgement number is a number indicating that the packet was received.
- IP Time-to-Live (TTL) Match - matches a field in the Internet Protocol (IP) that specifies how many more hops a packet can travel before being discarded or returned.

These rules can be limited to specific machines or be applied to the entire network. In addition, it can also report traffic that does not contain a specific keyword or expression.

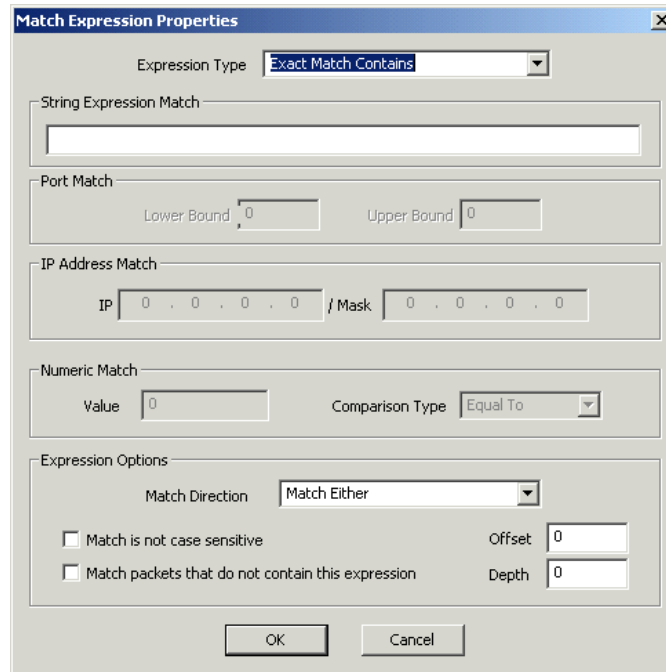


Figure 7 Matching Expressions

1.6 Network Performance

With PacketHound, the network administrator has an immediate picture of all activities flowing on the network through real-time graphical reporting in TrendReporter. Because PacketHound passively monitors the network traffic, it does not interfere with the network performance nor does it interrupt network traffic, even if the appliance breaks down for some reason. Network performance may be easily improved by analyzing traffic data using PacketHound's reporting feature or TrendReporter's real-time display of network activity. Use the information gathered to modify the rules on the appliance to more efficiently manage network applications. Harmful or nonessential applications can be minimized or eliminated by modifying the rules.

1.7 Network Architecture

The PacketHound appliance should be installed on the network segment containing the router or proxy that handles all traffic to the Internet. PacketHound must be able to see all network traffic on the Ethernet segment the organization wants to manage. However, the PacketHound and all the computers it manages do not have to reside on the same subnet. The PacketHound appliance can be connected to any one of these:

- Passive Tap
- Management Port of Switch
- Hub

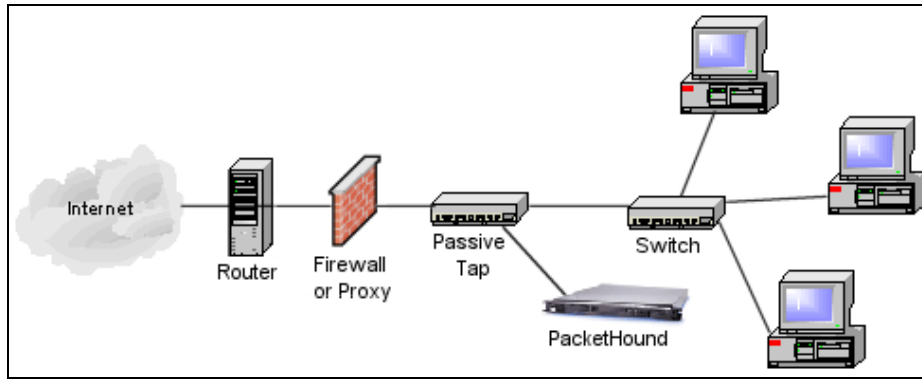


Figure 8 Passive Tap

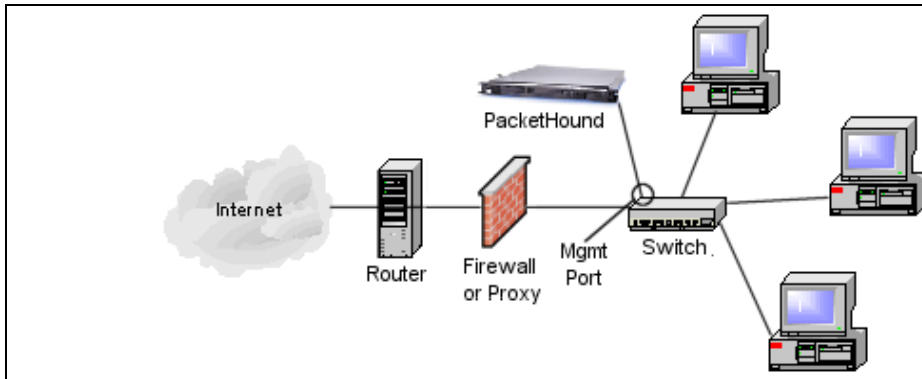


Figure 9 Management Port of Switch

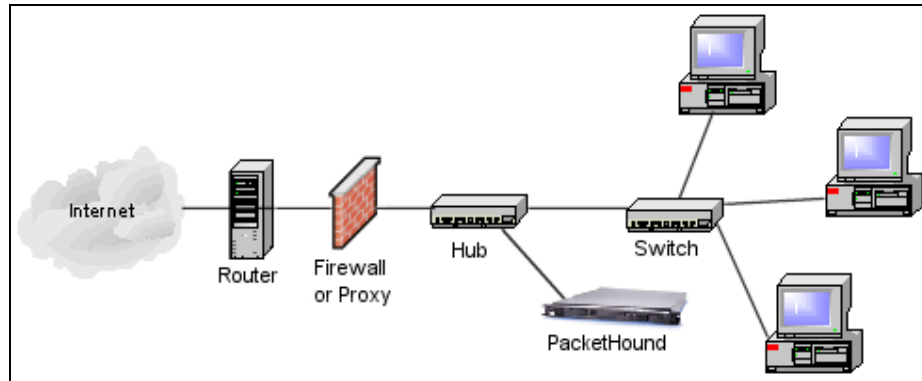


Figure 10 Hub

1.8 Scalability

One of PacketHound's major strengths is the ability to perform reliably on a high-speed network. It uses patented pass-by monitoring and blocking technology, which does not degrade network performance on a high-speed network with heavy traffic. The appliance is used by a variety of universities ranging from a smaller college of 600 students to one of the largest universities in the Big Twelve.

In extreme situations, such as a PacketHound with extensive rulesets on a gigabit network running at fully capacity, the impact of hitting performance limitations may result in a percentage of the network traffic not being examined by PacketHound. This is known as

dropping packets. It is important to note that dropping packets does not affect network or appliance performance. The affect may be that a limited number of connections are allowed to pass by the appliance. In the case of pass-through bandwidth shaping appliances, hitting performance limitations of the machine can degrade or halt the entire network.

The issue of dropped packets may be resolved by reducing the number of rulesets PacketHound monitors or by installing two identically configured appliances. Duplicate appliances can also be used to increase the complexity of the analysis and tracking being performed by each appliance. For example, one unit can be dedicated to particularly high volume streams like HTTP web traffic or P2P file sharing. The other appliance can then monitor and block other less-used protocols. Though this type of configuration is rarely implemented, very broad requirements in high speed environments can make this a necessary option. Because of PacketHound's pass-by operation, this step is very simple to implement.

Single, high performance PacketHound units running version 2.3 have been tested in university environments running 300Mb/s of traffic flow without dropping any packets. Significant performance enhancements have been included in Version 2.4, which is currently being beta tested to verify in-house testing results at 5-700Mb/s.

1.9 Protocol Identification

PacketHound provides a robust set of criteria by which it can identify unique characteristics of network traffic. These criteria range from simple packet header information such as source, destination, and TCP port use to more complex Application Layer information. PacketHound can also identify specific content within the packet.

Palisade employs PhD-level staff who use the above criteria to develop identification signatures based on the unique characteristics of each application or protocol's network traffic flow. Palisade staff closely monitor the Internet to identify new or altered versions of applications that need to be included or updated in PacketHound. Instead of waiting for an updated version of PacketHound, these additions are available to Palisade customers on the Palisade website.

Many suggestions for new signatures come from Palisade's existing customer base through the company's customer support hotline and online discussion group.

1.10 Granularity of Protocols

PacketHound provides varying degrees of granularity for different functions within protocols depending on the individual characteristics and operation of each protocol and the demand from our customers.

Rulesets can be created to manage a single protocol or group of protocols. For an exhaustive list of protocols that are tracked on a default basis, please see Appendix A. At a granular level PacketHound can control the individual functions within P2P protocols. This includes uploads, downloads, and searches on a per application basis. These capabilities may vary per protocol.

In addition, PacketHound's flexibility allows administrators to create their own custom rules for greater granularity within protocols. Palisade can also help an organization to create customized rules for monitoring and blocking applications and protocols.

1.11 Content Identification

Although the standard PacketHound appliance is not focused on content identification relative to copyright infringement issues, the appliance would serve as an excellent platform for allowing content identification companies like Bay TSP, NetPD, and Audible Magic to block strictly copyrighted materials.

Technology from Audible Magic would likely be the first to be integrated with PacketHound. Audible Magic has established the industry standard for audio identification of music and other recorded audio. The company's proprietary content recognition technology, ContentSense, dynamically identifies audio content by utilizing psycho-perceptual measures of the content itself for recognition. Its patented content-based identification system quickly, accurately, and definitively identifies music files simply by "listening" to them. This approach provides an accurate, non-invasive solution to content identification for all forms of digital media. This accuracy can be achieved because the technology analyzes the characteristics perceived by the human ear. These characteristics tend to be preserved even if the content is subjected to compression, equalization, reasonable time scaling, and other signal processing.

This content recognition technology accesses Audible Magic's database containing over 3.7 million copyrighted, the entire North American available catalog of music. This database is updated weekly with 5,000 to 10,000 new releases weekly. All popular audio formats, such as MP3, WAV, WMA are supported with accurate identifications for compressed files to 11Kbps.

For organizations looking to block based upon the actual content being downloaded, the Palisade Systems – Audible Magic solution would be superior to existing tools. It provides a non-invasive, easy-to-install solution, that strictly blocks P2P activity based upon the legality of the content.

1.12 Examination of Network Packets or File Content

PacketHound examines network packets or specific file content as determined by the network administrator. The administrator configures the PacketHound rules to determine what PacketHound looks for in the packets and files.

1.13 Distribution Systems

A complete list of the applications and protocols that PacketHound can manage by default is included in Appendix A, including P2P networks like FastTrack(KaZaa, iMesh, and Grokster), WPNP(WinMX), Gnutella(Bearshare, Limewire, Shareaza, and Morpheus), ED2K(eDonkey, Overnet, and eMule). In addition, PacketHound can monitor and stop alternative means of file sharing such as FTP, instant messaging, Usenet, and Telnet.

1.14 Resilience of the Technology to Countermeasures

One of the fundamental advantages of PacketHound's core technology is its stealthy and secure nature on the network. Because of its passive technology, the PacketHound appliance provides a very high degree of security. It is virtually impossible to detect, locate, circumvent, or attack on the network.

PacketHound can detect and block encrypted applications, although encrypted packets can decrease the appliance's ability to detect content within the packet. Applications that make use of encryption, like Filetopia, have identifiable characteristics in portions of their protocol. Users also use port tunneling and proxy servers to circumvent traditional security appliances like a firewall. In the case of tunneling, PacketHound signatures that track P2P applications can block these attempts. It traces and kills the P2P application connection rather than simply closing the port. In regards to proxies, the administrator can use specifically created rules to monitor and block SOCK proxies and proxies dedicated to the HTTP protocol. PacketHound also has the ability to reassemble fragmented IP packets.

The appliance was built to be resilient in the face of a Denial of Service (DoS) attack. The algorithms that deal with potentially CPU-intensive situations, such as IP fragment reassembly and connection management, are designed to scale well to handle a very serious attack. At the point in which PacketHound might potentially fail, this would be the least of the administrator's worries. In addition, because of its pass-by technology, PacketHound doesn't create any chokepoints on the network under such an attack. Organizations using pass-through tools may risk experiencing a network failure under a DoS attack, due to the tool queuing traffic to the point that it clogs the network.

1.15 Testing and Installed Base

Palisade is arguably the industry leader in this space as the first company to introduce a product specifically designed to help organizations detect and manage P2P and related applications on the network. The PacketHound appliance has been in production release for roughly three years, and has been thoroughly tested and proven in a wide range of internal test and customer environments. PacketHound has matured over its production life with numerous enhancements and improvements driven by market feedback and environmental changes. When updating or improving PacketHound, Palisade follows a rigorous internal testing regimen which includes real world beta testing of significant new product releases.

PacketHound is being used by a wide range of organizations including universities, government, and business. Commercial organizations using PacketHound extend across a variety of sectors, such as financial services, insurance, healthcare, and manufacturing. Nearly 200 organizations use PacketHound including Catholic Health Initiatives, the University of Missouri, the Canadian Ministry of Transportation, Regis Corporation, U.S. Marine Merchant Academy, Grinnell College, APAC Customer Services, and Troy State University.

1.16 Competitive Approaches

Unlike its competitors PacketHound delivers an easy-to-install solution specifically developed to manage and eliminate the security, liability, and bandwidth issues associated with P2P. Organizations relying solely on auditing, bandwidth shaping, or filtering tools may find it difficult to completely control P2P. Other approaches to managing P2P may actually create more problems than they solve.

Audit Tools

- Reactive approach – Although the administrator may know that P2P activities are taking place, the reaction is often after the fact. It is then up to the administrator to take appropriate action. A proactive solution is more effective at resolving P2P issues.
- Resource intensive – Requiring a person to monitor logs for an entire university and punish individuals is very time intensive and takes away from other necessary administrative tasks. Given the resource strain universities are facing and the sheer number of P2P users, this approach is impractical for effectively dealing with the illegal trading of copyrighted movies and music.
- Partially effective – Even if the administrator prevents a user at a specific machine from using P2P applications, there are several ways for the student or employee to circumvent the monitoring tools on the network.

Bandwidth Shaping Tools

- Intrusive pass-through technology – Rather than using a pass-by approach to monitor and block traffic, bandwidth-shaping tools are pass-through appliances. Pass-through appliances create potential network choke points and can degrade network performance, especially on high-speed networks. They can also be more difficult to install requiring the network to be taken offline.
- Blocking vs. Shaping – Queuing down traffic, or shaping, can not guarantee the total elimination of abusive applications. Although these tools are good at controlling bandwidth, they do nothing to eliminate security and liability issues from P2P.
- High Total Cost of Ownership (TCO) – The overall investment for bandwidth shaping appliances is much higher than for PacketHound. In addition to a time consuming installation process, the shaping of specific applications and traffic is very involving. It usually requires either an outside consultant or great time commitment by the network administrator. Traffic shaping is also something that requires constant monitoring and altering. PacketHound offers a simple deployment process that can be completed in a matter of hours. The installation and configuration does not require outside help.

Filtering Tools

- Purpose-built solution vs. Afterthought add-on - Several web filtering tools have only dealt with the issue of P2P and IM applications in the last few months. Filtering products have added the ability to filter a minimal number of popular P2P applications which resulted in an incomplete tool for eliminating P2P.

PacketHound provides a proven, purpose-built solution for eliminating P2P with over three years of testing and development.

- Limited number of signatures – Filtering tools generally filter only a limited list of the most popular applications. PacketHound has the ability to monitor and eliminate virtually every P2P application. Users can create customized signatures for any new or unknown application, which can then be tracked and blocked.
- High cost and complexity – To use a web filtering solution within an academic environment to enforce a P2P policy is a cumbersome and partial solution at best. All features of filtering solutions reflect its purpose of filtering Internet sites rather than network activities. To receive accurate reporting of P2P activities will require the administrator to customize these tools.

Network Performance Tools

- Difficult to install – Network performance hardware may involve taking the network offline to install. There may also be compatibility issues with the university's existing infrastructure that the administrator must consider before installation. PacketHound's core technology allows it to seamlessly integrate with any TCP/IP network regardless of network operating systems, machines, or applications.
- High TCO – Network performance hardware requires a substantial commitment of time and resources from organizations. The tools require a high degree of customization in order to monitor and understand P2P applications. This may require the creation of customized signatures and reports in order to watch such activities. They also require the constant attention of the network administrator to fine-tune enforcement to meet the university's appropriate Internet use policy.
- Intrusive pass-through technology – Network performance tools are usually pass-through appliances, rather than using a pass-by approach to monitor and block traffic. Pass-through appliances create potential network choke points and can degrade network performance, especially on high-speed networks. They can also be more difficult to install requiring the network to be taken offline. PacketHound's pass-by approach does not degrade or disrupt network performance.

1.17 Third-party Components

PacketHound comes complete from Palisade with everything an organization needs. In environments requiring high performance due to high network speeds (>100Mb/s), large amounts of data collection, and/or high degree of signature customization, Palisade may recommend use of a separate database server, which is available from Palisade.

1.18 Intellectual Property

The core technology engine of PacketHound is protected by US Patent #6,044,402 which was issued in March of 2000 to Palisade's founder Dr. Doug Jacobson, Professor of Computer Engineering at Iowa State University. A nationally recognized expert in networking and security, Dr. Jacobson directs ISU's Information Assurance Center (IAC), one of a handful of educational institutions in the country to have secured a

designation from the National Security Agency (NSA) as a Center of Excellence in Information Assurance Education.

See Appendix C for the Palisade Core Technology White Paper.

2.0 Corporate Characteristics and Resources

Palisade Systems was founded in 1996 as a result of technology transfer and entrepreneurial efforts at the Iowa State University Research Park in Ames, IA. Since then, Palisade has introduced four network appliance products addressing network management and security issues.

- **PacketHound** – PacketHound is a network appliance that detects and manages the use of troublesome network applications and protocols such as streaming media, web radio, distributed peer-to-peer file sharing, and others. PacketHound was recently awarded an R&D 100 award as one of the top technologically new products of 2001.
- **FireBlock** – FireBlock is a network appliance providing internal network-level access control. FireBlock allows organizations to monitor and enforce network-level access policies on their internal networks, protecting critical network assets and enabling a new level of network management and reporting capabilities.
- **SmokeDetector** – SmokeDetector is a network appliance that provides electronic decoy capabilities, commonly known as honey pot intrusion detection. SmokeDetector simultaneously emulates a number of interesting or vulnerable network systems in order to attract, detect, confuse, and delay attack attempts. SmokeDetector facilitates a wide range of reporting and response options once an inappropriate event is detected.
- **ScreenDoor** – ScreenDoor is a network appliance that manages Internet access to offensive or inappropriate Internet web sites. In addition, ScreenDoor can be used to block, monitor, and manage internal network functions such as various network protocol use, IP address use, and internal system access.

Palisade serves over 500 customers in almost all 50 States and Canada, including a variety of organizations in general commercial, education, and government sectors.

Though a privately held company and its financial information confidential, Palisade has raised approximately \$3.75 million in outside capital led by prominent investor John Pappajohn and his venture capital firm Equity Dynamics. Palisade currently employs 22 people.

3.0 Pilot Testing

Onsite customer evaluations of PacketHound normally span 15-30 days. In most cases, PacketHound evaluation units are shipped directly to the customer and supported remotely by Palisade technical support staff via telephone, email, and potentially by direct network connection (if allowed by the customer). For high performance or complex network environments, Palisade Field Engineers are available to perform or assist with installation and training. Onsite personnel are provided on a billable basis.

Palisade will consider requests to install PacketHound for pilot testing and evaluation purposes on a case-by-case basis.

4.0 Commercial Terms

Purchase of PacketHound involves two components:

- The physical PacketHound appliance and license for use of its software.
- An annual maintenance and support agreement.

The cost of the physical appliance and software license is a one-time charge based on the number of workstations the appliance will be managing. The annual maintenance fee is 20% of that cost, due for the first and subsequent years.

Since Palisade does not publicly distribute its prices lists, representative pricing for various size networks is supplied below. First year cost is the cost of the physical appliance plus the first year maintenance. Subsequent year's cost is the price of the ongoing annual maintenance agreement.

<u>Number of workstations</u>	<u>First year (\$/ws)</u>	<u>Subsequent years (\$/ws)</u>
500	\$13.20	\$2.20
1000	\$8.10	\$1.35
2500	\$5.52	\$0.92
20,000	\$5.05	\$0.84

The costs above reflect commercial pricing for a standard platform PacketHound and do not include costs for potentially necessary high performance platform options or professional services such as onsite implementation, customization, or training. Palisade offers a standard 15% discount to K-12 and higher education organizations. These costs are provided for informational purposes only and are subject to change at any time by Palisade.

5.0 Additional Information

- Appendix A contains a complete list of the protocols and applications PacketHound manages.
- Appendix B contains a list of traffic analysis criteria.
- Appendix C contains the Palisade Core Technology White Paper.

Appendix A – Applications PacketHound Manages

PacketHound can be set to monitor, manage, and block the following applications.

File Sharing Applications

Applications that allow users to share files across the Internet or Intranet.

AIMXfer

Manages AIM file transfers.

AresXfer

File sharing application.

AresServer

Web server program for the Ares file sharing application.

AudioGalaxyLogin

Real-time file-sharing system for sharing MP3s; it uses a Web-based interface and a local client that controls sharing and downloading.

AudioGalaxySearch

Manages file searches on AudioGalaxy.

AudioGalaxyDownloadReq

Manages download requests on AudioGalaxy.

BitTorrent

BitTorrent is a tool for copying and sharing files. Files clients download are automatically copied onto the client's machine and are made available for peer-to-peer sharing.

BlubsterXfer

Peer-to-peer file sharing application for MP3 files.

DirectConnectXfer

Host-based synthesizer/sampler streaming tool.

EDonkey

Peer-to-peer file sharing application.

FiletopiaXfer

Filetopia is a free communications software that includes: instant messaging, chat, file sharing system with a search engine, online friends list and message boards.

FreeNet

A distributed, encrypted file sharing system used to anonymously share files. FreeNet nodes store encrypted pieces of files that are indexed by keys. As a result of its design, FreeNet can

be used to anonymously and confidentially store and retrieve data, thus avoiding content filters and government limitations.

Gnutella

File-sharing application that allows users to exchange any type of files by connecting them to a “daisy- chain” of other machines. PacketHound also manages these other applications that are based on the Gnutella protocol:

- Bearshare
- Bodetella
- Cooltella
- Furi Launcher
- Furi Updater
- Gnewtella
- Gnewtella 2
- GnOtella
- GnuCache
- Gnucleus
- Gnujatella
- Gnumm
- Gnuspace
- Gnutella for Mac
- Gnut
- Gnute
- Gnutmeg
- Gnutella Crawler
- Gnutella.it
- GnutellaXfer
- Gobobo
- GTK-Gnutella
- Hagelslag
- Limewire
- Mactella
- Morpheus
- MyGnut
- MyTella
- Music City
- N-Tella
- Newtella
- PeaGnut
- Pi
- Pygnut
- Reflector
- SeachLord

Gnutella Web

Allows users to trade files on the Gnutella network through a Web browser.

GnutellaXfer

Matches file transfers between many Gnutella clients, including Bearshare, LimeWire, Gnotella, Gnutella, Gnucleus, and Morpheus.

GnutellaSearch

A log-only rule that records the file search terms used in Gnutella Query messages on the network.

GnutellaSearchHit

A log-only rule records the name of each file, IP address, and port number of the server that provides the file in Gnutella QueryHit messages.

Gnutella2UDP

A UDP-only rule that matches UDP packets with Gnutella2 data that would not be matched by the Gnutella rules.

ICQLogin

Instant messenger and file transfer service. This feature allows you to manage user logins to ICQ.

IRC-DCC-Send

Direct Chat (DCC) file transfers on Internet Relay Chat (IRC).

Important: The IRCLogin rule must have rematch enabled when used with IRC-DCC-Send.

KaZaAXfer

A file-sharing service where users can trade audio, video, images, and other documents through a “daisy chain” connection. Transfers files using the KaZaA, FastTrack, Grokster,

and IMesh protocols.

Madster

File-sharing software used to chat and share any types of files.

MSNMessengerXfer

Manages file transfers through MSN Messenger.

NNTP

Network News Transfer Protocol (NNTP) is a news service that transmits information through port 119.

Napster

Central-server based file-sharing application that allows users to exchange MP3 music files. Though Napster is shut down, there are still many open implementations of the Napster protocol in use. PacketHound also manages these other applications that are based on the Napster protocol:

- Amster
- BeNapster
- Blazter
- Capster
- Console
- Napster CLT
- DeWrapster
- DiaRRIA
- DJnap
- Fanster
- File Navigator
- Gnapster
- GTK-Napster
- Hackster
- iNapster
- JNap
- J Napster
- Jnerve
- KNapster
- Koog Epsilon
- Lopster
- Macstar
- Macster
- MyNapster
- NapAmp
- Napkin
- NapMan
- Napsack
- Napster for Beos.htm
- Napster/2
- Napsterterminator
- Napster - Linux
- Napster Server Manager
- Napster Unban
- Gnome-Napster
- Netstreak iAssimilator
- N-Dream Plug-In for Napster
- OpenNap
- Pakster
- Rapster
- Riscster
- Snap
- Socks2HTTP
- Spyster
- TekNap
- TKNap
- Unwrapper
- Webnap
- Wrapster
- XMNap

Napster XferIn

Manages Napster downloads.

Napster XferOut

Manages Napster uploads.

Napster Xfer

Manages all Napster uploads, downloads, and hotlist transfers.

ScourExchange

File-sharing application that allows users to exchange any types of files.

ScourExchangeXfer

Manages all ScourExchange file transfers.

Twister

A free internet program to find and download MP3 and other music files.

Instant Messaging

AIMLogin

America Online Instant Messenger (AIM) is an instant message and file-transfer application that allows users to chat and share files.

AIMMsg

Manages incoming and outgoing AIM messages.

ICQMsg

Manages incoming and outgoing ICQ messages.

IRCLLogin

Matches logins to the Internet Relay Chat (IRC) protocol. IRC provides chat rooms, but is often used to share files in violation of copyright. Intruders will often set up IRC servers on compromised computers to assist their sharing of files.

MSNMessengerLogin

Instant messaging service provided by the Microsoft Network.

YahooMsgrLogin

Yahoo Messenger is an instant message and file-transfer application that allows users to chat and share files.

YahooMsgrMsg

Manages incoming and outgoing Yahoo messages.

Windows

Microsoft-DS

File sharing application that allows users to share files through port 445.

NetBIOS-SSN

Microsoft file-sharing application that uses port 139.

NetBIOS-SendMulti

Multi-packet NetBIOS user messages.

Note: *_NetBIOS-SendMulti will not work if NetBIOS-SSN is also enabled.*

Multi-Media

Applications that allow users to stream audio and video to their desktops.

RealMedia 1, 2, and Multi Rate

RealAudio and RealVideo are streaming formats that many sites use to transfer audio and video. Real Networks use three different protocols, and we've included all three separately for your convenience. To block *all* RealMedia, you must block all of these protocols.

ShoutCast

Streaming media format that many sites use to transfer audio.

WindowsMedia

Streaming format that many sites use to transfer audio and video.

HTTP Protocols

Protocols that use the HTTP protocol to transfer files.

HTTP

Hyper Text Transfer Protocol (HTTP) is the underlying protocol of the World Wide Web.

HTTP - ACTIVEX

ActiveX controls are objects inserted into a Web page or other applications to reuse packaged programming functionality; because these controls execute on a user's computer, they may be a security and/or virus risk. This setting manages user attempts to transfer ActiveX controls through the HTTP protocol.

HTTP-AVI

Microsoft video format file transfers over HTTP.

HTTP-EXE

Manages user attempts to transfer executable files through the HTTP protocol. Because these files execute on a user's computer, they may be a security and/or virus risk.

HTTP-Audio-MPEG

Manages user attempts to transfer MPEG audio files through the HTTP protocol.

HTTP-Video-MPEG

Manages user attempts to transfer MPEG video files through the HTTP protocol.

IDENT

UNIX-based protocol that looks up real user names when a user attempts to login to a server. IDENT uses port 113.

HTTP-QuickTime

Manages user attempts to transfer Quick Time files through the HTTP protocol.

HTTP-RAR

RAR formatted archive files transferred over HTTP.

HTTPS

Secure connection for the HTTP protocol that uses port 443.

HTTP-SHOCKWAVE-FLASH

Manages user attempts to transfer Shockwave through the HTTP protocol. Shockwave allows users to view interactive Web content like games, business presentations, entertainment and advertisements from a Web browser.

HTTP-Zip

Manages user attempts to transfer Zip files through the HTTP protocol. Because these files execute on a user's computer, they may be a security and/or virus risk.

Ads/Spyware

Spyware is often installed as part of other software. Though it cannot be completely blocked, you can use the spyware rules to identify where spyware has been installed on your network. However, you can block the downloading of programs like BonziBuddy by blocking access to the web site. Create an HTTP rule, add the web site to a List, and set the rule to block.

BonziBuddy

Spyware program that gathers information about user activity and transmits this information to ad services and define which pop-ups appear on your system.

Web site: <http://www.bonzi.com/bonzibuddy/bonzibuddyfreehom.asp>

ClickSpring

Ad service that identifies traffic, tracks browsing habits and creates specialized pop-ups.

DoubleClick

An ad service that supplies a majority of the Web's banner ads and also gathers information from user activity.

Gator

An ad server that displays pop-up ads based on browsing habits.

GroksterPhoneHome

Spyware included with the download of Grokster and other software that reports user information to an ad service.

Hotbar

Hotbar is a browser and email add-on and contains skins to go over your browser toolbar. Hotbar then monitors the websites that you visit and adds these topics to your toolbar. The email toolbar allows users to download graphics from Hotbar to add to

their emails.

RealDownloadAds

Ads associated with the RealDownload program.

Web site: <http://www.real.com/download/>

RVPopup

Ad content from the RV popup program, often included with other downloaded software.

SaveNow

An affiliate program that tracks browsing habits and promotes special offers.

ShopAtHome

Tracks browsing habits and promotes sponsor sites.

SmartPops

Tracks browsing habits and displays pop-up ads based on sites recorded for the user.

Web site: <http://www.smartpops.com/>

WebHancer

Ad service that identifies traffic, tracks browsing habits and creates specialized pop-ups.

Xupiter

A toolbar installed by “drive-by download” that gathers and reports user browsing habits.

Vulnerabilities

Turning on Reassemble IP Fragments and Verify Checksum options in the Preferences window also helps prevent these attacks.

HTTP-CodeRed

Worm that propagates over HTTP and invades Windows systems running Microsoft's ISS web server. After it infects a server, it attempts to infect other servers.

HTTP-CodeRedII

Worm-like CodeRed that propagates over HTTP and invades Windows systems running Microsoft's ISS web server. After it infects a server, it attempts to infect other servers. It is more virulent than CodeRed because it does a better job of selecting target IP addresses. CodeRed II also inserts backdoors into the system to enable future system compromises.

HTTP-IDA

File extension used by the Microsoft Index Server ISAPI Extension. The CodeRed and CodeRed II worms exploited a vulnerability in the code that handles requests for index server lookups. This rule could be used to block future attacks on this vulnerability. Note that order of the rules affects the matches. If HTTP-IDA came before HTTP-CodeRedII or HTTP-

CodeRed in the rule file, HTTP- CodeRedII or HTTP-CodeRed would never be matched because HTTP-IDA would match first.

Mail Protocols

Protocols used to transfer e-mail.

IMAP

Interim Mail Access Protocol (IMAP) is a mail protocol that uses port 143.

IMAPS

Allows users to access their mail through a secure SSL connection and uses port 993.

POP

Post Office Protocol (POP) is a mail protocol that uses port 109/110.

POP3S

Allows remote users to access their mail through a secure SSL connection and uses port 995.

SMTP

Simple Mail Transfer Protocol (SMTP) is a mail protocol that uses port 25.

Web-based Mail

Email programs using standard HTTP.

AOLWebmail

Web access for AOL mail.

Hotmail

Email program that provides email access from any computer connected to the internet.

YahooMail

Email program providing access to email from the internet.

FTP

FTPActive

Manages FTP active file transfers on Port 20.

FTPControl

Manages FTP control port (port 21).

FTPPassive

Manages passive FTP file transfers.

HTTP Information

HTTP_Hosts

This rule matches host names in HTTP requests with host names in a list that you create. If a hostname in the list begins with a period (.), it is considered a domain name and all hosts in that domain will match.

HTTP_Servers

This rule enables the ability to capture data for the Web Monitor capability in TrendReporter and logs URLs and hosts in the URL Log. This rule, like LogUnmatched, does not match connections. It is used for data collection only, not for blocking or allowing. When adding this rule, some options in the Rule Properties window are unavailable. It is recommended that you place this rule near the top of the list so it captures information from all HTTP data.

HTTP_URLs

This rule matches URLs in HTTP requests with URLs in a list that you create.

Remote Login

CitrixICA

Matches Citrix's Independent Computing Architecture (ICA) protocol, which is used for remote access to computers running Microsoft Windows applications.

GotoMyPCShare

Application that allows remote access to a user's computer through the GotoMyPCShare web site.

RLogin

Remote Login Protocol (RLogin) allows users to log into the network remotely using port 513.

REXEC

Allows remote execution of commands on UNIX hosts, or any other system with the REXEC interface. REXEC uses port 512.

RSH

Remote Shell Protocol (RSH) allows users to execute shell commands remotely using port 514.

SSH

Secure shell protocol that provides connectivity, encryption, and authentication for servers. SSH uses port 22.

Telnet

Telnet Remote Terminal Protocol allows users to login to other systems remotely. Telnet uses port 23.

VNC

Virtual Network Computing (VNC) is a remote display system allowing users to remotely view a desktop from anywhere on the Internet. VNC uses port 5190.

WindowsTerminalServer

Remote connections using Windows Terminal Server.

XWINDOWS

Graphical user interface primarily for UNIX similar to Microsoft Windows. XWindows uses port 6000.

Distributed Computing

These programs allow organizations to use your computer when it is idle.

SETIatHome

Downloads radio telescope information, process it, and sends it to the Search for Extraterrestrial Intelligence at Home project.

GoogleCompute

Sponsored by Google Toolbar, uses idle computer time from the Google Compute project.

Other

Standard TCP/IP protocols and others you may need to manage.

Finger

UNIX command used to gather information about other Internet users. Finger uses port 79.

Gopher

System that pre-dates the World Wide Web for organizing and displaying files on Internet servers. Gopher uses port 70.

LPR

A BSD UNIX printing protocol that supports printing to network and local printers. It also acts as a print server. LPR uses port 515.

SOCKS4/5

Protocol that provides access to network services through a SOCKS proxy server. This may be used to hide a user's identity and evade network management systems like PacketHound.

Special Rules

Custom

This setting allows you to enter your own Custom Match String for a protocol that you've identified and need to manage. This is recommended only for advanced users.

EthernetAddresses

This rule matches packets that have a source or destination Ethernet address that is in a user- created list.

Everything

This setting allows you to block, monitor, or ignore any connection by any of the protocols managed by PacketHound.

LogUnmatched

The LogUnmatched rule allows you to log all connections that do not match defined protocols. When adding this rule, some options in the Rule Properties window are unavailable.

Appendix B – Traffic Analysis Criteria

PacketHound uses a layered approach for traffic analysis. First, it verifies the validity of the packets, and can reassemble fragmented IP packets. Next, you can choose to manage traffic by raw packet content or HTTP filtering. Also, PH has built-in categories and rules for high-level matching.

- Packet Validity
 - IP Fragment Reassembly
 - IP Header Checksum Verification
 - TCP Segment Checksum Verification
 - UDP Packet Checksum Verification
 - Sequence and Acknowledgement Number Verification
 - Time To Live (TTL)
- Packet Contents
 - MAC Address
 - TCP/UDP
 - Source / Destination IPs/ Masks
 - Source / Destination Ports
 - Time of Day, Day of Week
 - Incoming/Outgoing/Either
 - Data Offset
 - Data Length
- HTTP / Filtering – block by list
 - URI
 - URL/Host List
- Regular Expressions Strings
- Built-In Categories
 - Ads/Spyware
 - Email Apps
 - Email Protocols
 - File Sharing Apps
 - HTTP Transfers
 - Instant Messaging
 - MultiMedia
 - Vulnerabilities

Appendix C – Palisade Core Technology White Paper

The following document describes Palisade Systems’ core technology, protected by U.S. Patent #6,044,402. See Appendix C attached as AttachmentC_PalisadeCoreTechWP.pdf.