Raptor Firewall 6.0
White Paper

Additional copies of this document or any other AXENT publications may be ordered from your authorized distributor or directly from AXENT.

AXENT Technologies, Inc.
2400 Research Blvd. Suite 200
Rockville, MD 20850
1-888-44-AXENT
(301) 258-5043 (outside USA)
FAX: (301) 227-3745
Internet: www.axent.com

Contents

Figures

Section 1

Raptor Firewall 6.0: Raising the Standard

Defining The Need

In a survey undertaken by the FBI in cooperation with the Computer Security Institute, 73% of the respondents said their sites had been penetrated by hackers. Extrapolating from this, the survey takers estimate that the average corporate network is hacked approximately 12 to 15 times each year. Many times, these hacks occur without the knowledge of the corporation being attacked. Statistics such as these are a sobering reminder that no site is immune from hack attacks.

Preventing assaults upon the data that forms the lifeblood of your corporation takes a coordinated effort from system/network administrators and users. Prevention can take many forms, but a critical component in any protection scheme should be a sophisticated full-featured firewall such as Raptor® Firewall 6.0. This paper explains the technical aspects of the Raptor® Firewall, and describes the features that make it a cornerstone in any network security architecture.

Leading The Way

Raptor Firewall Version 6.0 by AXENT Technologies, Inc. carries forward the reputation of the Raptor Firewall as the most feature-rich and secure firewall available today. With the release of version 6.0, AXENT™ introduces the Raptor Management Console (RMC), a new graphical user interface for the Raptor Firewall. Designed as a snap-in module for the Microsoft® Management Console, RMC brings Microsoft Windows® native look and feel to the firewall user interface. The standard GUI has been updated and carried forward as the Raptor Console for UNIX® (RCU) for users who wish to manage their firewalls from the UNIX environment. Along with the new interface, AXENT's Raptor Division has designed and optimized the firewall to run seamlessly on the Microsoft Windows NT® 4.0, and Sun™ Solaris® 2.6 operating systems.

Raptor Firewall 6.0 is a proxy firewall that works at the application level. Using a set of application-specific (for example, http, ftp, or H.323) security proxies, the firewall evaluates each attempt to pass data through it for possible security risks. In addition, built-in support for Virtual Private Network (VPN) tunnels allows the creation of secure network level sessions. Packet filtering on a per-interface and per-tunnel basis provides increased flexibility to system administrators and end users.

This paper provides a detailed description of the Raptor Firewall 6.0 architecture and features, and is organized as follows:

Section 1	Provides an overview of features and strengths of the Raptor Firewall architecture, and a discussion of other security approaches currently in use
Sections 2 and 3	Describe the Raptor Firewall architecture and key features in depth
Section 4	Discusses internal and external deployment and management of the Raptor Firewall in an enterprise security framework
Section 5	Explains how VPN Technology enables you to create secure IP tunnels with fine-grained access control across public networks

Raptor Firewall: Certified Strong

The Raptor Firewall has been cited repeatedly for the strength of its architecture, ease-of-use, and unrivaled richness of features. The first firewall to receive ICSA™ certification on both UNIX and Microsoft Windows NT platforms, the Raptor Firewall has won both the coveted *Best of LAN Times* award (1996), and the Ziff-Davis Internet Magazine award for Best Internet Firewall. From security and ease-of-use standpoints, the Raptor Firewall application proxy architecture has distinct advantages over other designs. With the release of version 6.0, these include:

- Enhanced list of built-in application-proxies that include the most popular protocols, including support for Microsoft LAN Manager file and print sharing services (SMB/CIFS proxy) allowing AXENT to integrate Microsoft networking into the Raptor Firewall 6.0 environment.
- High availability support using Microsoft's Cluster Server for Windows NT and Veritas FirstWatch® for Solaris.
- Automatic port blocking to protect services running on the firewall system.
- Anti-SPAM functionality built into the SMTP proxy to prevent internal mail servers from being used as SPAM relays.
- Reliance on *best fit, non-order-dependent, explicit rules* rather than filtering criteria to evaluate connection attempts.
- Support for symmetric multiprocessing.
- Integrated anti-spoof and IP routing protection.
- Built-in port scan detection.
- Ability to create packet filters on a per-interface basis.
- Automatic and continuous system hardening of the firewall host to protect against intrusions into the firewall system.
- Ability to block content of Java® applets and objectionable Web sites and newsgroups using third party products (for example: Finjan® SurfinGate, WebNOT®, NewsNOT™).

- Built-in support for Virtual Private Network connections with features such as exportable 56-bit DES, triple DES (3DES), ISAKMP/Oakley (IKE) key management, IPsec and swIPe.
- Security protocols, cascaded and nested connections, and the ability to filter packets into and out of trusted IP tunnels.
- Automatic enforcement of *address hiding*, to conceal your system and network information from the outside world.
- Ability to set-up *transparent access* between clients and servers through the Raptor Firewall for both TCP- and UDP-based applications.
- Integrated support for secure mobile computing, including the use of Network Address Translation (NAT) to simplify routing tables.
- Extensive logging capabilities, including database export.
- ICSA certified and Year 2000 compliant.

Protection Against a Broad Range of Attacks

The Raptor Firewall is ICSA certified for UNIX and Windows NT. It has been tested in many scenarios involving such tools as SATAN, COPS, Tripwire and Internet Security Scanner. Results prove that the Raptor Firewall protects your networks from a wide array of attacks, including the following:

- IP Address spoofing attacks
- IP Source Route attacks
- SMTP backdoor command attacks
- Snooping of network traffic
- Attacks via download of Java applets
- Information leakage by means of finger, echo, ping, SMTP and traceroute commands
- Modification of network traffic
- HTTP cgi-bin wildcard attacks and buffer overrun attacks
- TCP SYN Flood attacks
- IP Fragmentation attacks
- SMTP Buffer overrun attacks
- TCP Session Hijacking
- TCP Sequence Number Prediction Attacks
- Random port scanning of internal systems
- Large packet PING attacks
- Password replay attacks

You can find additional information on this subject, as well other topics in the AXENT online network security library at *www.axent.com*.

Proxy Equals Security

The Raptor Firewall's use of *application-level access controls*, its *fail-safe architecture*, and its *ease of configuration and management* set it apart from firewalls that use other approaches. While other firewall vendors claim to support vast numbers of services right out of the box, there is no guarantee that the services are secure. The Raptor Firewall offers full protection from hackers or snoops for all supported services. Important firewall strengths are discussed in the sections that follow.

Application Level Access Controls

While attackers may try a variety of ways to invade a targeted system, most attacks seek to exploit *application services* and their data streams. For example, hackers often use well-known SMTP (email) application holes to break into internal mail systems. Other application-level attacks exploit services such as FTP (the service used for file transfer), HTTP (used for Web browsing), ping (a service commonly used to check connectivity), and gopher (a document search and retrieval tool).

The Raptor Firewall's application-level access controls prevent these attacks by *scanning* for and *filtering* them within the data stream of the network connection. These scanning and filtering operations are performed most efficiently at the application-level - the most likely zone of attack.

Working at this level allows the Raptor Firewall to use *dedicated security proxies* to examine the *entire data stream for every connection attempt*. This provides the Raptor Firewall with a considerable advantage over approaches such as packet filtering that work at lower levels in the protocol stack. These approaches typically verify network connections on a packet-by-packet basis, rather than as a whole.

In contrast, the Raptor Firewall evaluates each connection in its totality, and only allows packets to pass if they are part of an established and authorized network connection. This is one reason why the application-level approach is widely considered to be the most secure method in use today.

The Raptor Firewall 6.0 application proxy architecture provides other benefits to users, including the ability to control the type of operations allowed for specific protocols. For example, if you use the Microsoft file and print sharing protocol, CIFS/SMB, you can control specific operations such as read/write. This capability extends across the supported protocols.

Easy to Configure, Easy to Manage

New in version 6.0, the Raptor Management Console provides a clear and consistent means for maintaining and monitoring the firewall. As a snap-in module to the Microsoft Management Console, RMC offers powerful firewall configuration and management capabilities with native NT look and feel. If you can navigate the NT desktop, you can comfortably manage the Raptor Firewall for NT.

In the UNIX environment, AXENT has updated and enhanced the standard firewall toolkit interface and renamed it the Raptor Console for UNIX (RCU). This interface retains the time-tested GUI familiar to current firewall users, and updates it to support the new functionality shipping with version 6.0.

In the world of security, ease-of-use is a vitally important consideration. The reason is simple: improperly configuring a firewall can leave holes in your security framework. What's more, you may not even become aware of the holes until your site has been compromised.

The possibility of inadvertent misconfiguration is a primary reason that security experts do not recommend packet filtering systems. The basis for their concern is clear: rules written for packet filtering systems are *highly order dependent*. If rules are ordered incorrectly, such systems may actually allow unwanted connections. This point of complexity makes such systems comparatively easy to misconfigure. Moreover, as rules are added, the likelihood of misconfiguration increases.

In contrast, Raptor Firewall rules are not *order dependent*, eliminating the chance of one of its rules superseding and nullifying another. Moreover, the Raptor Firewall GUI (see Section 3)

makes it simple to create powerful rules for specific hosts or whole networks. Once rules are in force, the Raptor Firewall monitoring interface and comprehensive logging take the guesswork out of security management. These features make it easy to verify that the rules you have created are working as intended.

A "Fail-Safe" Architecture

Some of the most common, and damaging, attacks are routing based. To protect against them, the Raptor Firewall simply *does not route any IP traffic*. Instead, the firewall acts as a virtual brick wall that provides both a physical and logical separation of *internal* (secured) from *external* (public) networks.

The Raptor Firewall's refusal to perform routing operations ensures that it is impossible for packets to pass through the firewall at the routing layer *even if a failure occurs in the firewall host.* The Raptor Firewall provides no network-level route, or path, for packets to take into your networks. This fail-safe design is a fundamental feature of the Raptor Firewall: one that is conspicuously absent in systems that filter and route packets at the network level.

Shortcomings of Other Approaches

The strengths of the Raptor Firewall architecture are most obvious when contrasted with other approaches, such as:

- Simple packet filtering
- "Stateful" packet filtering
- Circuit-level gateways

In the following sections, the various alternatives to the Raptor application-proxy approach are discussed. While reading about these technologies, it is important to understand that our design has been proven time and again to be the most secure method available to protect your network traffic. In fact, when other vendors claim that they offer alternatives that provide similar protection, what they do provide are application proxies to close the holes in their product, whether it be pure packet filtering or "stateful" filtering. By doing this, these vendors validate our technology. Why settle for a patchwork of protection, when AXENT offers the Raptor Firewall 6.0, a true application proxy firewall with fully integrated VPN tunneling.

Simple Packet Filtering

Packet filtering has long been a basic feature of routers, and still serves as the foundation of many firewalls on the market today. As discussed earlier, packet filters have numerous drawbacks, which include the following:

- They are inherently complex making them *difficult to set up* and administer
- They are by nature *less-secure* than application-level proxy firewalls
- They *do not automatically hide network and system addresses* from public view

Packet filters work by distinguishing packets based on IP addresses or specific bit patterns. Because only limited information in the data packet is accessed, these filters are unable to protect against application level attacks, making them susceptible to sophisticated IP fragmentation and IP source routing attacks.

"Stateful" Packet Filtering

Several companies have introduced "stateful" security devices that purport to be a significant technical advance over usual packet filtering techniques. In fact, the advance consists of extracting certain well-known bit patterns in the protocol header fields of TCP and UDP connections, creating and maintaining a table of established (or open) TCP and UDP connections, and then examining and comparing header information on each packet that passes through the firewall.

This state information is used to track open, valid connections without having to process the rule set for each packet. This architecture has a number of weaknesses, which include:

- Most importantly, IP-level controls offer no protection against application level attacks. There is no application-level security. Nor do they allow for control of application-specific operations, such as read/write or put/get.
- Packet filter rules *must be implemented in the proper order* to work as intended. If they are not, the filtering process may actually allow unwanted packets into the "protected" network.
- Stateful packet-filtering engines do not automatically perform address hiding. This requires the user to manually configure all of the addresses to be "translated" or hidden.
- Packet filtering systems perform routing operations; application-level proxies like those used by the Raptor Firewall do not. This is a critical point of difference: in principle, it makes pure packet filtering firewalls *open to routing-based attacks,* and to "failing-open" if problems occur with the firewall system.
- Even when packaged with a GUI, stateful packet filtering firewalls are very difficult to configure. Because they operate at the network level, configuring them typically requires detailed knowledge of IP and IP-based protocols. Also required is expertise in network security in order to understand the risks posed by specific networking protocols.

Circuit-Level Gateways

Unlike the Raptor Firewall, which looks for application level data before allowing a connection, this type of firewall operates at the session level. It typically relies on a state table containing a list of valid connections. Subsequent TCP- or UDP-based connection attempts are allowed or disallowed based on a comparison with this information.

The down-side to this approach is that it works at the session-level only. Once a session has been established, the firewall may allow any kind of traffic to pass through. This is inherently less secure than proxying connections at the application level, and may leave the protected network open to attacks that exploit the firewall’s lack of contextual information. Moreover, lack of this information makes it difficult to distinguish different traffic types, such as FTP put and FTP get.

Feature	Raptor Firewall	Stateful Firewalls
Authorize and pass UDP traffic securely	Yes - using generic application security proxies.	Yes - by holding a connection at both ends for UDP packets.
Examination and filtering of application level data streams	Yes - via security proxies, including FTP, HTTP and SMTP, allowing control over the type of data transferred, and operations performed.	No - not at the stateful layer. Access based on IP addresses; no control over type of data transferred.
Simple to configure and administer	Yes - RMC or RCU, also authorization rules not order dependent. Rules are easy to create due to application level controls.	No - rules are highly order dependent and hard to create due to IP-level filtering.
Internal addresses hiding	Yes - automatically as part of the architecture of a proxy firewall. Also, address transparency is supported if needed.	Not automatically. Requires intervention by the administrator
Examination of bit patterns in packets	No - Not required by an application level firewall.	Yes - typically used to emulate the features found in a proxy level firewall.
Logging of statistical data such as URL's, byte counts, port scans, and connection attempts.	Yes - In addition, the Raptor Firewall logs all connection attempts, successful as well as failed, automatically.	Yes - But not automatic

Figure 1. Comparing Raptor Firewall with "Stateful" Firewalls

Performance

As we have shown in this section, the Raptor Firewall 6.0 is a powerful tool for maintaining a secure network environment. But what is the cost in performance imposed by the Raptor Firewall 6.0? For all it’s superior protection, the Raptor Firewall adds little in the way of network throughput degradation. With "stateful" or other packet filter firewalls, you sacrifice protection for performance. They achieve their throughput numbers at your expense. The Raptor Firewall 6.0 makes no such compromise, and still performs efficiently. In fact with the Raptor Firewall 6.0, your Internet connection is the most likely bottleneck you will have.

Now capable of handling a saturated T3 connection with a throughput of 45 Mbps or greater, the Raptor Firewall is fast enough to handle all of your site firewall traffic. A series of performance tests were run against the Raptor Firewall at the National Software Testing Laboratories (NSTL). The results of these tests are posted on the AXENT Technologies® web site, www.axent.com. Please visit this site for specific test details.

Overview of Raptor Firewall 6.0 Features

Use of Rules at the Application Level

As shown in Figure 2, the Raptor Firewall forms a virtual brick wall against all forms of attack, from internal as well as external sources. The Raptor Firewall authorizes all connection attempts into and out of the networks it secures using a set of explicit rules, defined by the administrator through the RMC or RCU. The Raptor Firewall applies these rules at the *application level*, giving it access to all contextual information needed to make correct authorization decisions. Moreover, the Raptor Firewall is an especially suspicious guardian. By default, it denies any connections not explicitly allowed by a rule.

Each rule the Raptor Firewall applies can incorporate a *range of criteria*, including source and destination addresses, type of service, one or more forms of strong authentication, user or group, and the time of day (or date) of the access attempt. The ability to create rules that incorporate all of this information, and to remotely manage multiple firewalls, gives you the power to build a *comprehensive security policy* tailored to the specific needs of your business.

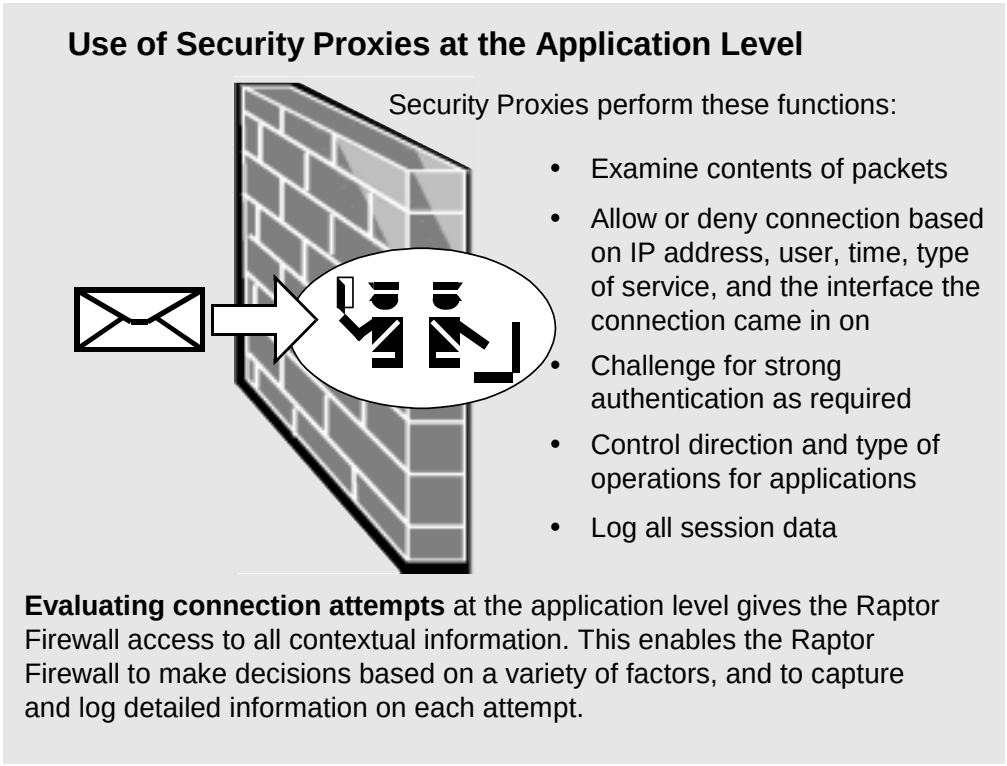


Figure 2. Use of Application Proxy Services

The rules you create can be customized to meet your specific needs. A rule can control the type of operations allowed, as well as define the type of control enforced on data and content. This provides network administrators with a powerful tool for protecting system content from unwanted manipulation.

Support for Standard Services

The Raptor Firewall 6.0 is an application proxy firewall that uses a set of security proxies to initiate connection attempts through the firewall. Each security proxy is designed to listen for specific types of connection attempts. As part of the Raptor Firewall 6.0 release, all application proxies are multithreaded. Raptor provides out-of-the-box support for the following services and service restrictions:

- Support for Microsoft LAN Manager file and printer sharing services
- SQL*Net
- Telnet
- FTP
- SMTP - Secure e-mail
- HTTP-Enhanced with support for ratings, and HTTP v1.1 features including connection persistency and request pipelining.
- HTTP-FTP
- HTTP-Gopher
- HTTP-HTTPs
- H.323
- Ping
- NNTP-News service with ratings support
- Java filtering-Using optional third-party application
- Gopher+ - with URL ratings support
- RealAudio® and RealVideo®
- DNS - graphically configurable name server with built-in support for split-DNS
- NTP (Network Time Protocol)

  Additional enhancements to individual proxies include the following:

  - FTP - control of put and get file operations.
  - NNTP - added functionality over the Generic Service Passer including the ability to filter objectionable newsgroups using either the NewsNOT subscription service or wildcards, protection against known NNTP-based attacks, and control of such NNTP operations as feeds and posts.
  - The SMB/CIFS proxy provides support for Microsoft file and print sharing. This proxy allows SMB data to pass through the firewall, and allows for the control of such file and print operations as Read, Write, Print, Delete, Rename, Directory, and Pipes.
  - Java applet filtering protects your systems against the latest form of attack: destructive Java applets. Raptor Firewall 6.0 provides optional third-party support for robust Java filtering.

Support for Generic Services

In addition to handling commonly used services, you can configure the Raptor Firewall to handle connections based on other protocols. The Raptor Firewall, through the RMC or RCU GUI, makes it easy to do this using a configurable service daemon called the Generic Service Passer (GSP). For example, you can write a rule to allow connections to a specific host using a custom database protocol.

Support for Multiple Authentication Types

The Raptor Firewall supports a number of popular strong and weak authentication methods, including the following:

- Security Dynamics ACE®
- S/Key
- Defender™ by AXENT Technologies, Inc.
- CRYPTOCard™
- Gateway password
- NT Domain Authentication

Administrators can create *custom templates* that apply one or several authentication methods to access attempts in a definable order. In addition to these methods, the Raptor Firewall supports any authentication server that supports the TACACs+ or RADIUS protocol.

Graphically-Configurable DNS Implementation

The Raptor Firewall's graphically-configurable split-level DNS implementation protects the identity of internal hosts, and conceals the overall topology of secured networks from the outside world. DNS is the most difficult aspect of any framework to configure properly. Through the RMC on NT and the RCU for Solaris GUIs, Raptor makes DNS configuration a simple point-and-click task.

Secure SMTP Proxy With Anti-SPAM Support

The Raptor Firewall's secure, graphically-configurable SMTP security proxy makes it easy to configure mail delivery into and out of your network. It also prevents SMTP-related attacks, such as buffer overrun attacks and command backdoor attacks. To set up mail, all you have to do is indicate what internal and external mail systems you want to use for mail servers, and whether you want to allow direct delivery of mail from internal to external systems. The SMTP security proxy configuration wizard does all the background work of generating rules and redirecting services.

Anti-SPAM functionality built into the SMTP proxy protects your internal mail servers from being used as SPAM relays. You can specify explicit domains from which messages can be received, and you can specify maximum recipient counts to protect against widescale spamming of internal users.

Automatic Suspicious Activity Monitoring

The Raptor Firewall performs suspicious activity monitoring (SAM) on all connections through the firewall. SAM works by keying on *thresholds* you establish for connection rates when writing authorization rules.

The Raptor Firewall applies these thresholds on a rule-by-rule basis. In creating rules, you specify thresholds for them based on anticipated levels of access for each of them. When a threshold is exceeded, the firewall generates an alarm. How alarms are handled depends on individual site practices.

Transparent Access Through the Firewall

For rules that do not require authentication at the firewall, the Raptor Firewall can support *transparent connections* between internal and external hosts or subnets. *Transparency* refers to a user's awareness of the firewall. You can configure the Raptor Firewall so that users can connect through it to a destination system, still subject to existing authorization rules, without being aware of the intervening presence of the firewall.

The administrator can specify transparent access to specific systems, or sets of systems, on a per-interface basis. Systems designated as transparently accessible are included on the Raptor Firewall's list of transparent servers. The Raptor Firewall consults this list for every connection attempt that specifies an internal host as its destination address.

With the introduction of Raptor Firewall 6.0, you are able to configure the firewall so that it no longer hides source addresses on connections through the firewall. This feature allows you to use the actual client address as the source address when communicating through the firewall to other systems. One benefit to this feature is that it allows Web servers protected by the Raptor Firewall to track activity based on client source addresses. Also, the firewall administrator can configure both server and client transparency on a per interface or per address basis.

Raptor Firewall 6.0 supports transparency for both TCP- and UDP-based connections through the firewall. With the addition of UDP, a number of commonly used protocols, including SNMP, can pass through the Raptor Firewall.

Service Redirection Capability

As part of your overall security plan, you may want to conceal the identity of certain inside hosts, yet still allow outside users to view or obtain data that reside on these hosts. Using service redirection, you can provide access to information on any internal system without exposing that system's identity to external users. This capability complements the firewall's transparency feature by allowing you to provide the *illusion* of transparency to outside, accessing hosts.

With the release of Raptor Firewall 6.0, UDP service redirection is supported. As with TCP, this capability allows UDP traffic to pass through the firewall without exposing the addresses of hosts to the outside world. This feature enhances the Raptor Firewall feature set and provides symmetry with supported TCP features.

Virtual Clients

New with version 6.0 is the concept of Virtual Clients. Using this feature, the firewall administrator can assign addresses from a defined pool to clients on one side of the firewall as they pass traffic through the firewall. These addresses are then recognized by the destination system as the source addresses of the incoming data packets. This feature is particularly useful when dealing with source-side transparency and the networks on either side of the firewall use private addresses. With the Virtual Clients feature implemented, there is no risk of address conflicts when an entity on one network sends data through the firewall to an entity on a second network.

Spoof-Checking for Specified Systems or Subnets

The Raptor Firewall 6.0 allows you to associate network entities with specific network interfaces. Doing so enables you to assure that data packets from network entities arrive at the Raptor Firewall by an expected pathway. Accesses from that entity on any other firewall interface are dropped as a possible spoof attack. It is also possible to list specific subnet IDs that the firewall should check for anti-spoofing. For sites with many internal subnets, this feature provides a convenient way to implement complete spoofing protection.

Built-In Port Scan Detection

Port scanning, either automatic using a tool such as SATAN or manually, is a method of attack in which an intruder attempts to break into a network by repeatedly querying all well-known TCP or UDP ports. To minimize the risk from this form of attack, the Raptor Firewall 6.0 has built-in port scan detection support. With port scanning activated, the Raptor Firewall generates an alert when suspicious activity is detected. The alert contains the source address from which the attack was initiated, allowing the network administrator to trace the source of the attack.

Automatic Port Blocking

Automatic port blocking protects the services that are running on the firewall system from access by other systems. You must specifically enable services and ports to allow access. This feature works for both static and dynamically allocated ports. This feature can be very effective for protecting the firewall system from hostile attacks.

Packet Filters for Specific Interfaces

To provide fine control over the data that passes through the firewall, the Raptor Firewall 6.0 allows the administrator to create packet filters on a per interface basis. For each Raptor Firewall interface, you can now specify what is filtered through to the secure network or out to the unprotected Internet.

Content Filters to Block Unwanted Sites and Newsgroups

With the Internet becoming glutted with information ranging from trenchant to trivial to profane, network administrators can be tasked to limit access to those sites that are not relevant to day-to-day business. The Raptor Firewall 6.0 supports WebNOT, a web site blocking filter, and NewsNOT, a newsgroup blocking filter. These tools allow the firewall administrator to control access to sites deemed to be inappropriate by organizations.

High Availability Support

High Availability support maximizes the availability of secure business-critical internet connections by minimizing downtime due to system failure, scheduled maintenance, software updates, backup procedures, and system upgrades. When one system goes down, a second system automatically takes over and new connections can be created immediately. The Raptor Firewall 6.0 supports high availability on Windows NT systems using Microsoft Cluster Server.

Year 2000 Compliance

With the introduction of Raptor Firewall 6.0, all time-related configuration options such as time limit on rules, and notifications, will operate correctly into the twenty-first century. As we reach the end of the current millennium, this feature becomes more and more of a mission critical requirement, especially in the world of finance and in other time-critical industries. Buying Raptor eliminates this worry for the firewall administrator.

User Database Support

As part of the Raptor Firewall 6.0 enhanced feature set, AXENT adds the ability to make use of information from external user databases. This feature allows you to connect directly to an external database such as NT Domain. You can also import data either as a one-shot option for initializing the Raptor Firewall, or as part of a regularly scheduled program for adding, deleting, or modifying user information based on changes to some external database such as NIS or NT Domains. Combined with the capability to import VPN tunnel data, this feature greatly simplifies the support of large user populations, as well as simplifying support for large-scale RaptorMobile deployment.

Enhanced Activity Logging

AXENT has enhanced the already extensive logging capabilities of the Raptor Firewall as part of the 6.0 release. One new feature provides a greater degree of granularity in the recording of normal firewall activities. The logging subsystem allows the administrator to control, on a per rule basis, whether normal activity is logged or not. This can greatly reduce the amount of data logged on a daily basis by eliminating the recording of activities that are of no interest at a given site.

Other new features include support for a trace facility in most daemons as a debugging aid; and the logging of rule IDs, allowing administrators to associate connections back to the matching rule. Also, statistical log data can be exported to an external database.

As part of the enhanced activity logging, any invalid packet that arrives at the firewall can now be logged. The logfile will contain pertinent information concerning the packet, and the action taken when it was received. This feature can be useful for troubleshooting internal network problems or spotting network attacks.

Secure Virtual Private Network Tunneling

Already a world class product, AXENT's Virtual Private Network (VPN) technology has been significantly enhanced with the introduction of Raptor Firewall 6.0. The Raptor Firewall enables administrators to configure secure VPN tunnels across public networks, using either the swIPe or IPsec protocols. In Raptor Firewall 6.0, IPsec support is further enhanced to track the latest updates to the standards, including support for the new transforms defined in the IPsec RFCs.

Raptor Firewall VPN is fully integrated with AXENT's RaptorMobile product. With the release of version 6.0, the following features have been added or enhanced:

- ISAKMP/Oakley (IKE) dynamic keying has been added to support key management. This standard enhances security by eliminating long-term encryption keys. Adherence to a common standard makes administration of large numbers of RaptorMobile clients a straight-forward task.
- Proxying of VPN traffic allows you to maintain tighter access control and content checking. This feature, which can be applied on a per tunnel basis, allows the

administrator to force VPN traffic up through the application proxies. Support for transparency is also provided.

- Network Address Translation (NAT) support allows the mapping of RaptorMobile addresses to a pool of addresses that can be routed back to the Raptor Firewall. This greatly simplifies the task of setting up routing tables for RaptorMobile systems.
- Exportable 56-bit DES is in the international version of Raptor Firewall 6.0.
- Importing of user and tunnel information simplifies the administration of multiple RaptorMobiles. As with user data imported at the Raptor Firewall, this feature can be used to initialize a VPN tunnel as well as make routine updates to add, delete, or modify users and tunnels.
- With support for cascaded and nested tunnels, Raptor Firewall 6.0 VPN capability is compliant with industry standards. See Section 5 for further details.
- Tunnel time and data volume limits can now be set by the administrator. Tunnels using dynamic keying are forced to rekey on expiration of these configurable tunnel limits. Additionally, RaptorMobile requires that users be reauthenticated and tunnels reenabled after limits have expired.
- In Raptor Firewall 6.0, overall VPN performance has been enhanced to increase encryption speed.

Packet Filtering Within Secure Tunnels

Packet filtering provides an additional level of security within secure tunnels by limiting the *types of services* allowed through tunnels, and restricting the *direction* of allowed services.

Implementing filtering in tunnels does not change the essential nature of the Raptor Firewall: it remains a true, application-level proxy firewall. Packet filters are only implemented in secure tunnels as a means of giving you fine-grained control of the type and direction of allowed traffic. You can increase security and exercise even greater control of data flow by forcing tunnel traffic through application proxies with Raptor Firewall 6.0.

Support for Illegal Addresses (RFC 1918)

Within many companies, use of Internet addresses that have not been assigned by the NIC is a common practice. While use of these addresses internally presents no difficulties, trouble occurs when they are used to send or receive information on public networks. Problems arise when a company unknowingly adopts an address that has been assigned to another organization.

The Raptor Firewall (UNIX only) supports RFC 1918, a standard that provides companies with a scheme for mapping their internal (*illegal*) addresses to a set of reserved, legitimate addresses. Doing this mapping eliminates the address conflicts that can otherwise occur.

Illegal address support is provided for Telnet and FTP only. Use this feature only if you are using an illegal address to connect to the same, but legal, address on the Internet.

Performance Features

With the release of Raptor Firewall 6.0, AXENT continues to offer support for up to four processors in a single system. A fast data path has been added to the Raptor Firewall 6.0 design that, when enabled, allows data that does not have to be scanned or analyzed to flow through a connection. This feature is for user environments in which access control based on IP address and application protocol is sufficient, and maximum throughput is the primary consideration. This fast data path is not a local tunnel, but a feature that is integrated into the firewall and, as with AXENT-provided proxies, provides automatic address hiding, and statistical session logging.

Summary: Advantage Raptor Firewall

With the introduction of the Raptor Firewall 6.0, AXENT Technologies continues to provide the most secure, easy-to-use firewall available in today's marketplace. Features that make version 6.0 of the Raptor Firewall stand out in an otherwise crowded field include the following:

- Application level access controls and protection against application level attacks.
- Transparent access through the firewall for authorized users and applications
- Integrated Virtual Private Networking
- Automatic address translation and hiding of all internal addresses
- Ease of use and configuration
- Support for strong authentication mechanisms
- "Fail-Safe" Architecture
- Highest Performing throughput of network traffic
- High Availability support to minimize system downtime
- Extensibility with world-class third-party products tested and supported by AXENT
- Extensibility with other AXENT products to support:
  - *Remote firewall administration and deployment (Raptor Remote)*
  - *Remote user network protection with PC to firewall encryption (RaptorMobile)*

Section 2

Architectural Overview

Introduction

The Raptor Firewall 6.0 serves as an application-proxy between systems that physically connect to different network interfaces on the firewall server. Raptor Firewall 6.0 is an application-proxy because it acts as an agent or substitute at the application level for entities that reside on one side of the firewall when dealing with entities on another side of the firewall. By maintaining this separation between interfaces and providing a simple management GUI, the Raptor Firewall 6.0 provides a secure, easy to use environment for data exchanges.

As discussed previously, the Raptor Firewall 6.0 architecture is by its very nature more secure than that used by other firewalls on the market today. While based on the use of secure application level proxies, the Raptor Firewall incorporates Virtual Private Networking (VPN) tunneling capabilities with packet filtering to secure session-level traffic as an integral part of its design, not as a costly add-on. With Raptor Firewall 6.0, you can force VPN tunnel traffic through application proxies for even greater data security. These design features give the Raptor Firewall the proven strength and fine-grained access control associated with secure proxies, and the flexibility and extensibility of packet filtering approaches.

With Raptor Firewall 6.0, you get first-tier features such as: easily configurable topology; high-performance throughput; flexibility; expandability; and powerful security support. These come standard with the Raptor Firewall 6.0.

Key Architectural Characteristics

The Raptor Firewall 6.0 presents formidable obstacles to unauthorized entry. Unless configured otherwise, the Raptor Firewall automatically hides *all* source IP addresses of the systems it protects. This is a primary benefit of the Raptor Firewall's reliance on *secure proxy applications* to handle all connections into and out of the networks it secures. The Raptor Firewall's architecture, however, is as flexible as it is formidable. It uses a combination of techniques to *conceal* and *selectively reveal* network and system addresses. It also enables you to *redirect connections* from specified systems to systems protected by the firewall. Key features of the Raptor Firewall architecture include:

- **Address Hiding:** Fundamental to the Raptor Firewall design, this feature hides addresses by placing the Raptor Firewall as a go-between, or proxy, between outside systems and the secured systems on your network.
- **Address Transparency:** Opposite of address hiding, this feature allows the addresses of specified entities on one firewall interface to be visible to systems on another firewall interface. When enabled, transparency controls whether the address of a system on the network accessible through one interface is made visible to networks on other interfaces or

not. Transparency is limited by the access rules defined for a specific system. With Raptor Firewall 6.0, transparency is supported for TCP and UDP.

- **Address Redirection:** This design component allows you to create, and make public, *aliases* to protect systems inside your network. The Raptor Firewall automatically redirects valid connection attempts made to publicly visible aliases to the protected system. You can use this feature to give users seemingly unrestricted access to data on a system, while hiding the true source of that data from public view.
- **Virtual Clients:** Configured through the Redirected Services feature, creating a virtual client allows you to use a virtual address in place of the real address of the system initiating a connection. This way, the client can receive responses from any address you specify. Ordinarily, unless transparency is in place, clients receive all responses from the firewall's outside interface address.
- **Authorization rules:** This feature allows you to control the degree of access permitted to specific addresses. Firewall rules are independent of whether the connection is attempted transparently (specifying the destination in the connection request), or by the client connecting to the firewall and specifying the real destination, or through an address mapping entry. Authorization rule checks are always made with respect to the real source address and the real destination address, whether address mapping is in effect or not.

In addition to these key design elements, Raptor Firewall 6.0 incorporates other important features, including the following, which are discussed in greater detail in this white paper:

- Automatic disabling of insecure network services
- Enhanced support for strong and weak authentication methods on a rule-by-rule basis
- Automatic suspicious activity monitoring and alerting
- Comprehensive logging for all connection attempts, enhanced with per rule control of log messages
- Sophisticated VPN tunnel architecture integrated into the Raptor Firewall 6.0, with support for ISAKMP/Oakley (IKE) dynamic keying, VPN proxies, time and data volume limits, and integrated packet filtering among other features
- Expanded Transparency support to include UDP-based applications and client-side support.
- Year 2000 compliance tested to ensure that the year 2000 doesn't leave you unprotected.

Address Hiding

Spoofing attacks involve outside users acquiring knowledge about your inside networks, and using this knowledge to gain unauthorized access in the guise of a trusted host. This process is similar to using a false ID: the host "recognizes" the spoofed source address as being that of a trusted host, and allows the packets to pass into the network. In this case, the intruder gains entrance by supplying an internal, or known safe, IP address to the host securing the network.

The Raptor Firewall's application proxying architecture protects against this class of attack in two ways, as follows:

- Proxying hides the structure of internal networks from the outside world by forcing all outside-to-inside requests to connect to the Raptor Firewall, rather than to an internal host. Doing this makes it difficult for would-be intruders to gain information on protected systems.

- The Raptor Firewall drops packets that contain the address of any of its protected systems, and logs a security alert. This prevents outsiders from exploiting knowledge of an IP address behind a firewall interface.

How Address Hiding Works

The Raptor Firewall hides addresses by forcing all connection attempts to one of its proxy services (SMB/CIFS, FTP, HTTP, and so forth), or a generic service configured by the user, up the protocol stack to the application level. This requires that the original IP packets be disassembled and then reassembled into a data stream. Figure 3 shows this process.

Performed as a function of the Raptor Firewall software, this reassembly operation protects networks from IP fragmentation and "ping of death" attacks. This operation *removes the original source and destination IP addresses* from the data stream, and replaces them with the firewall, or gateway, address as source and the target system as destination. The specified Raptor Firewall proxy application receives a reassembled data stream and a connection from a client computer.

Using a separate internal mechanism, the Raptor Firewall retrieves the original source and destination IP addresses from the data stream. It then creates *a second network connection* to the destination computer. This connection generates new IP packets with the firewall, rather than the sending client, as the source address. The destination of the connection changes to the target specified by the originating system. In this example, transparency, discussed in the next two sections, is not in effect. The user is aware that the firewall is part of the data transfer process.

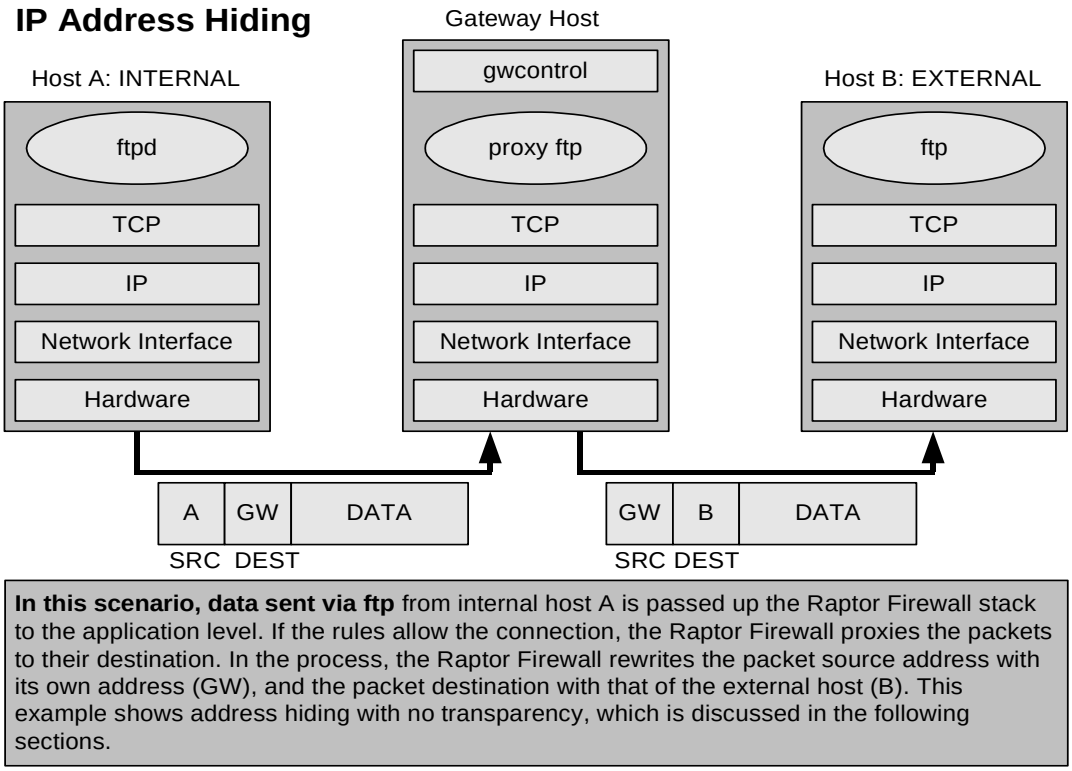


Figure 3. IP Address Hiding

Address Transparency

The Raptor Firewall 6.0 hides your system and network addresses by default and the firewall is the target destination of data flowing through it. By invoking address transparency, however, you can eliminate the firewall from the view of the internal system. This feature allows hosts to connect through the firewall as if it were not present, making it appear that there is no firewall between the user and the destination system. With transparency on, you can expose addresses and networks on a per-interface basis to external hosts. Figure 4 graphically illustrates this concept. As depicted in the figure, with transparency activated, and the appropriate rules in place, data pass through the Raptor Firewall with no apparent effect.

On these connections, the firewall continues to exercise all applicable authentication rules, and it continues to allow or disallow connections based on its authorization rules database. However, the connecting user is usually not aware of these operations.

With Raptor Firewall 6.0, you can configure transparency for all TCP- and UDP-dependent applications. In addition, transparency has been extended to include client-side as well as server-side support. These features are discussed in the following sections.

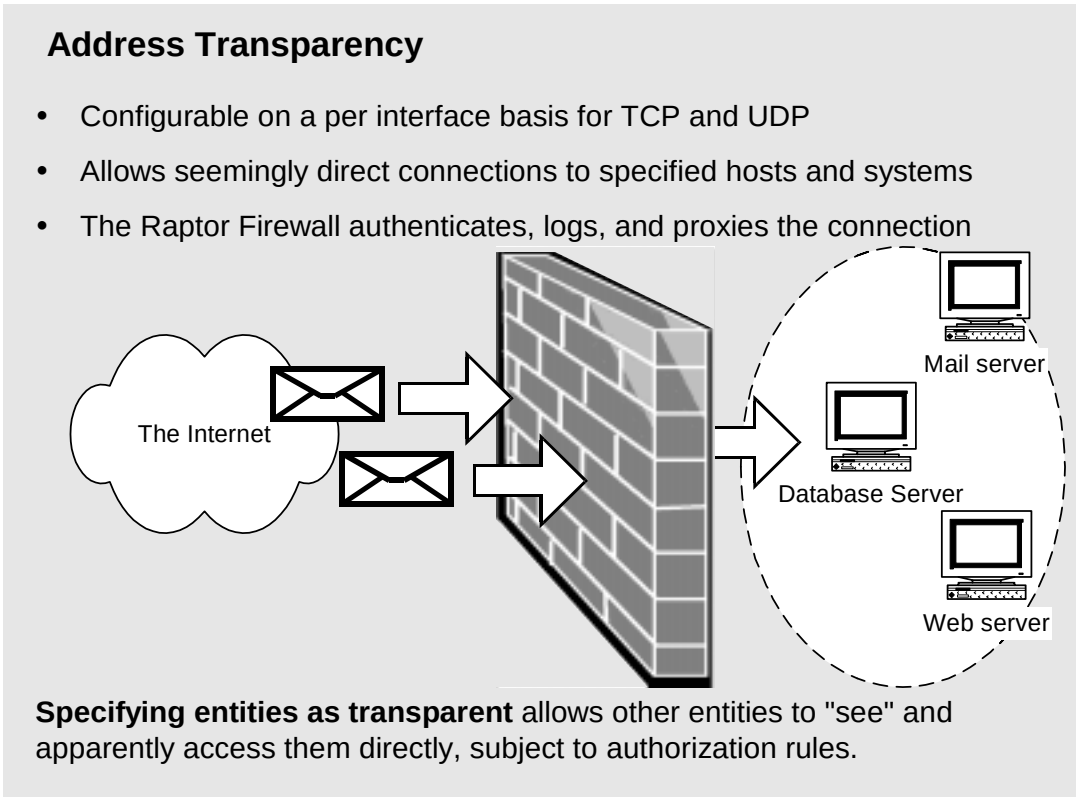


Figure 4. Address Transparency

Transparent Servers

The term "transparent server" defines the process of making the address of the actual target system the destination address in the originating IP packet. By allowing the source system to do this, the firewall is no longer visible to it, and to all appearances there is no firewall in place.

As with no transparency, the source address seen by the outside world is the firewall gateway address. The Raptor Firewall hides source addresses by forcing all connection attempts to one of its proxy services (such as FTP, HTTP, or a user-configured Generic Service) up the protocol stack to the application level. Doing this requires the original IP packets be reassembled into a data stream with the firewall gateway address substituting for the original source address. Figure 5 shows how packets are transmitted when server transparency is enabled. Using the Raptor Management Console, you can specify which addresses are to be made visible with this feature.

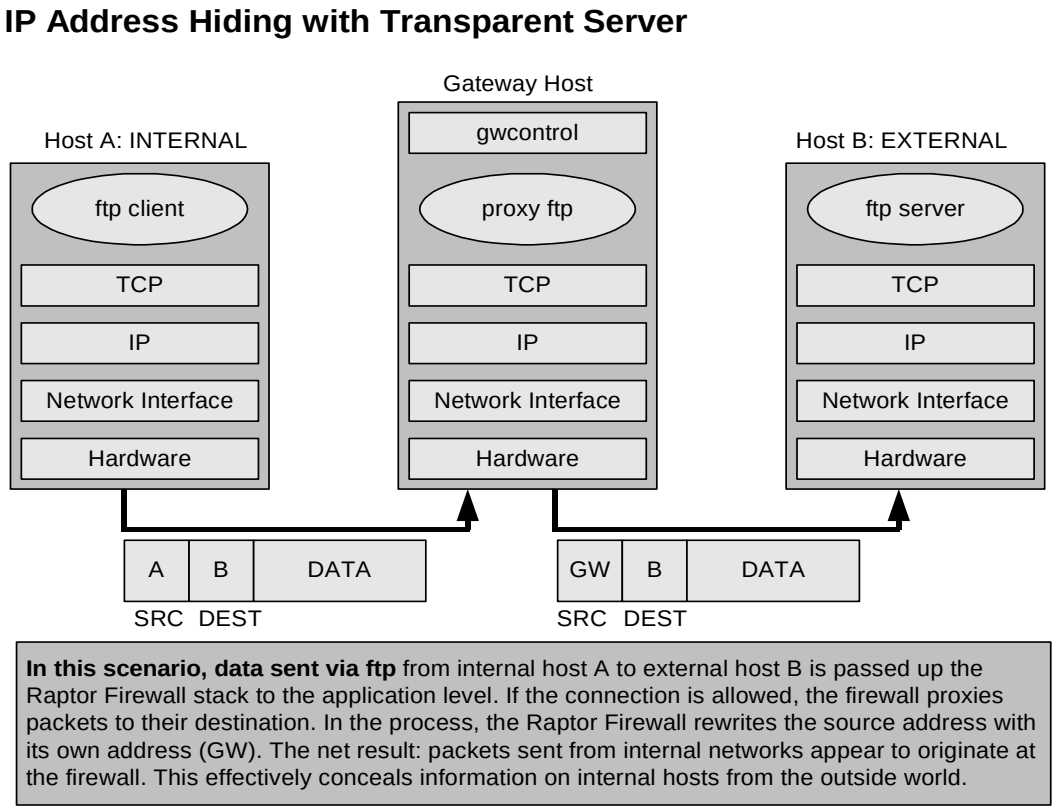


Figure 5. Packet Flow with Transparent Servers

Transparent Clients

As part of the Raptor Firewall 6.0 enhancements, you can configure the firewall so that the IP addresses of hosts on one side of the firewall can be exposed to systems on the other side. This feature allows the actual source address to penetrate the firewall and be available to the destination system or network. You can configure each Raptor Firewall 6.0 interface to use the actual client address as the source address in the connection to the server on the other side of the firewall. Figure 6 shows this concept.

As with transparent servers, the firewall is transparent to the source entity, but it is also invisible to the destination system. Data packets carry the actual source and destination addresses with the firewall making no attempt to hide either side from the other. As with transparent servers, you can configure the specific client addresses to be made visible.

Bear in mind though, that all applicable rules are still strictly enforced by the firewall. Only if a connection is expressly permitted, is it established. Client-side transparency does not eliminate the safeguards provided by the firewall. Client-side transparency is useful to those sites wishing to use their web servers to track user activity based on client source addresses.

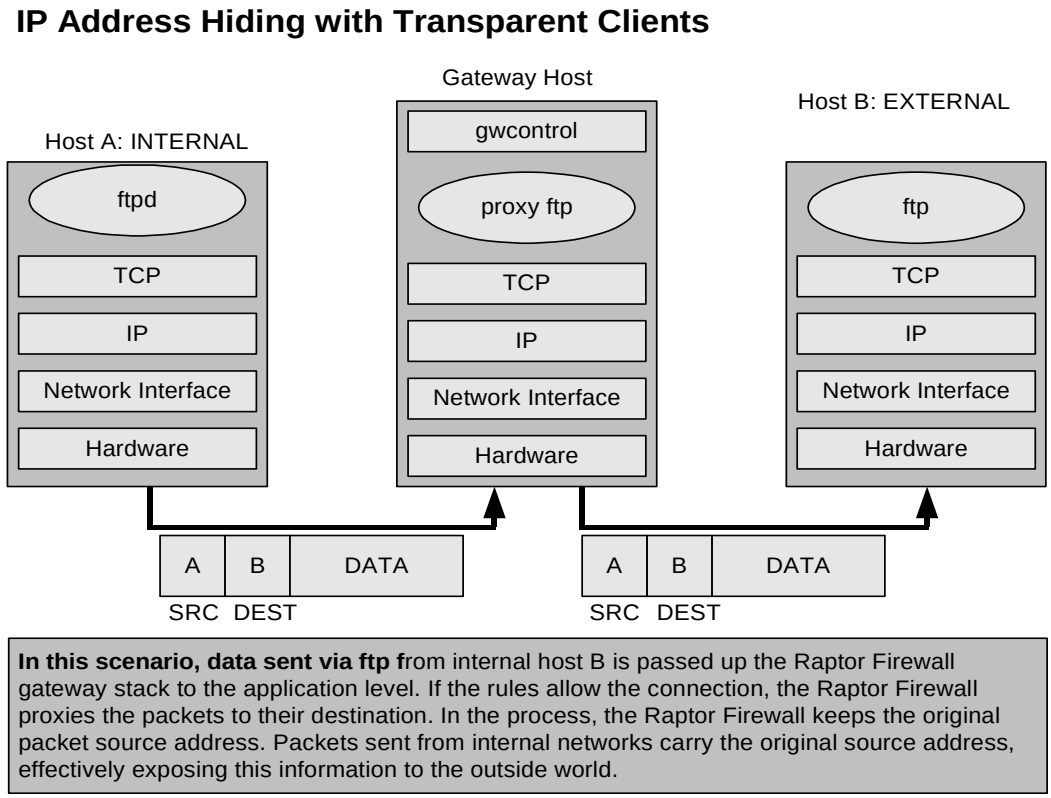


Figure 6. Packet Flow with Transparent Clients

Transparency Increases Flexibility

The ability to allow transparent access on a per-system, per-interface basis increases the flexibility of the Raptor Firewall 6.0. With transparency in effect, administrators do not have to train *internal users* on how to connect through the firewall. Another benefit is that *external users* and applications can (seemingly) access internal machines directly. For example, you can make your site's internal mail server "visible" (that is, transparently accessible) to machines accessing it from an outside (unprotected) network. This allows outside users to connect transparently to the mail server (if authorization rules allow the connection). At the same time, external machines would still have to connect to the Raptor Firewall to communicate with any other internal hosts. Figure 7 shows the Raptor Management Console (RMC) property page for transparency. Service, or address redirection, discussed later in this section, provides another means for protecting systems from unwanted exposure to outside networks.

Support for TCP and UDP Connections

As part of the Raptor Firewall 6.0 release, you can specify transparent access for both TCP and UDP-based connections. Earlier versions of the Raptor Firewall provided limited UDP support. With Raptor Firewall 6.0, full UDP support, equivalent to TCP-based application proxies, is provided.

Configuring Transparency

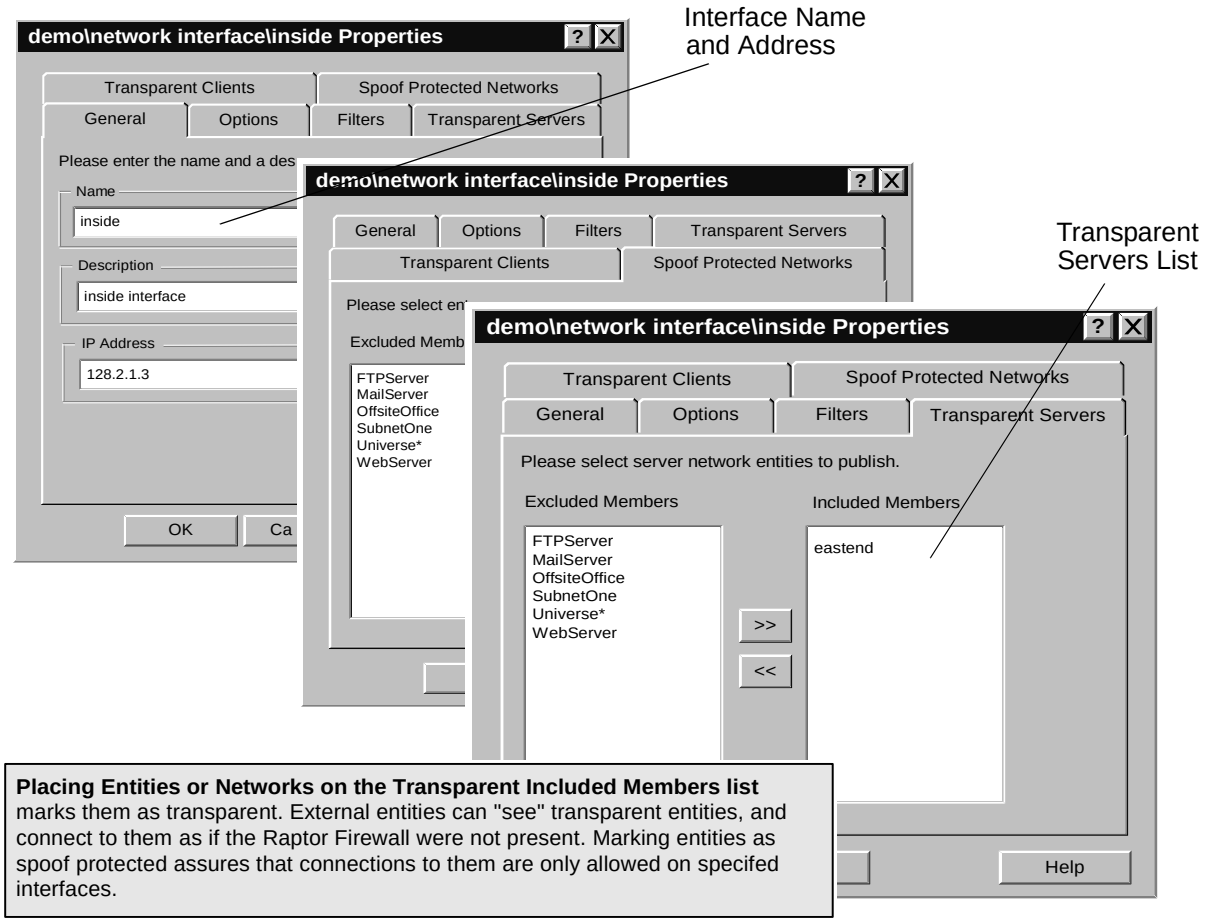


Figure 7. Using RMC to Configure Transparency

Transparency and Event Logging

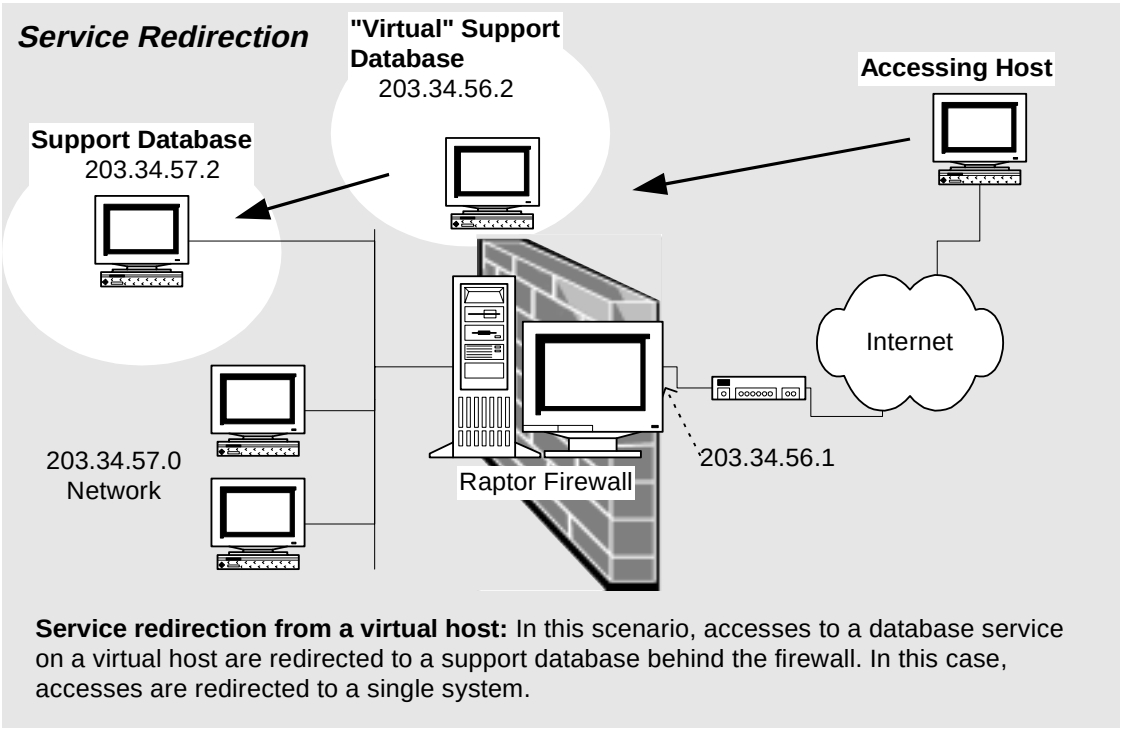
The Raptor Firewall logs the same type of information for transparent as for non-transparent access. This means Raptor Firewall administrators can still monitor the sources of access attempts and their duration, and receive alerts about suspicious activity, whether these attempts come from inside or outside of the secured network and regardless of whether they are transparent or non-transparent.

Transparency and Authorization Rules

The Raptor Firewall authorization rules apply to transparent access methods exactly as they do to non-transparent access. Transparent access is allowed only if an applicable rule permits it. If the rule selected for a connection attempt specifies no authentication, *and transparency is in effect*, the user directly connects to the remote machine as if the Raptor Firewall were not present. However, if the rule specifies an authentication type, the Raptor Firewall prompts the user for information in a manner similar to non-transparent connections.

Address Redirection

The Raptor Firewall's address redirection feature allows you to provide the *illusion of transparent access* to certain hosts. This feature provides a way to create an alias between a non-existent, or virtual, external server and a host within your network. See Figure 8 for a conceptual view of address redirection. All specified TCP- and UDP-based connections (FTP puts, for example) addressed to the virtual system are redirected to a real host residing in the protected network.



Service redirection from a virtual host: In this scenario, accesses to a database service on a virtual host are redirected to a support database behind the firewall. In this case, accesses are redirected to a single system.

Figure 8. Address Redirection

The ability to do address redirection fulfills two important objectives:

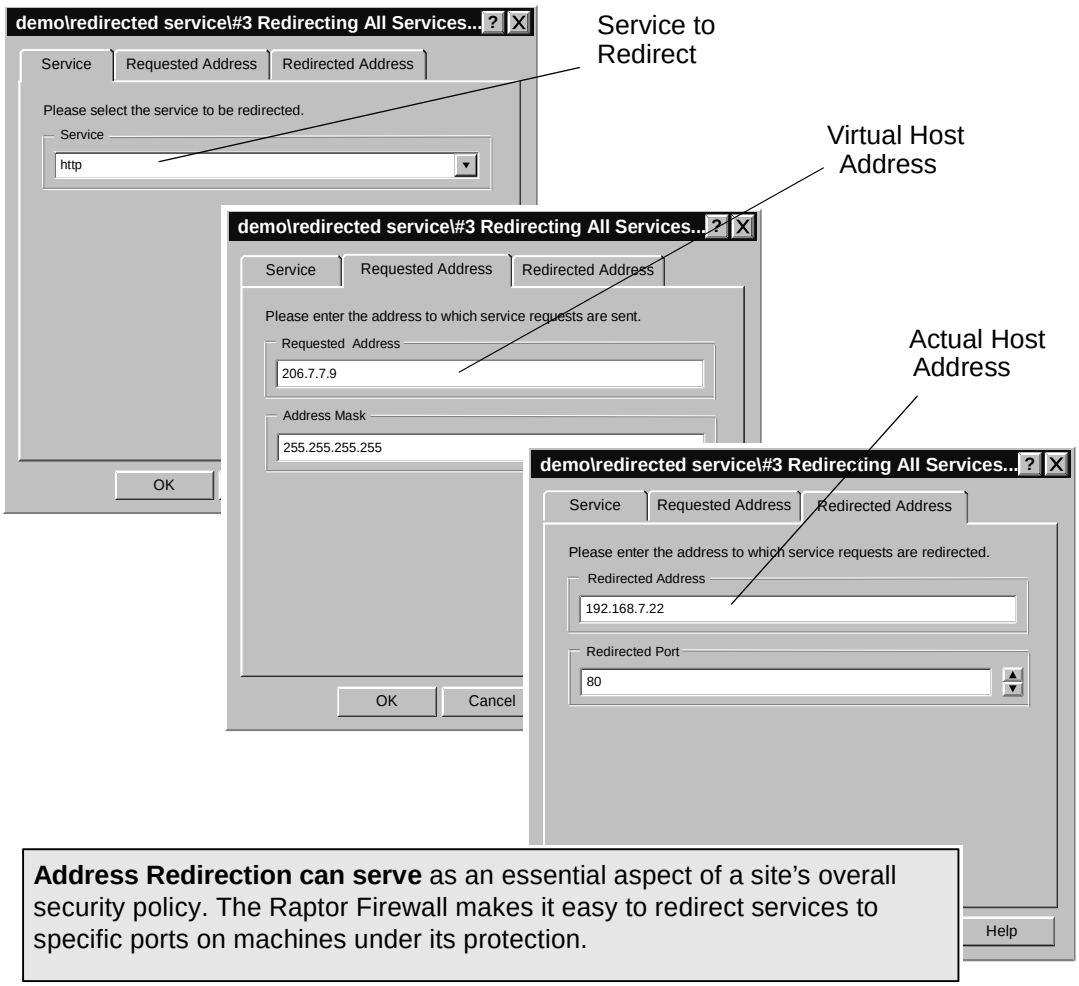
- The IP address of the host on your network is hidden from public view.
- Simultaneously, since the data they retrieve appears to originate at the virtual system they have accessed, the illusion of transparency to accessing hosts is maintained.

Specific rules must be in place to authorize each redirected service. The Raptor Firewall makes all authorization decisions based on the real destination address (and port). The virtual system never appears in the Raptor Firewall's rule database. This greatly simplifies the process of configuring rules.

Load Balancing

Another benefit service redirection provides is the ability to perform load balancing among multiple servers while presenting a single server interface to accessing clients. You can accomplish this by redirecting accesses to a DNS name that maps to multiple IP addresses. This capability is especially useful for large sites, or in smaller sites that must handle large volumes of accesses to multiple internal web servers or databases.

Redirecting Service Requests



Address Redirection can serve as an essential aspect of a site's overall security policy. The Raptor Firewall makes it easy to redirect services to specific ports on machines under its protection.

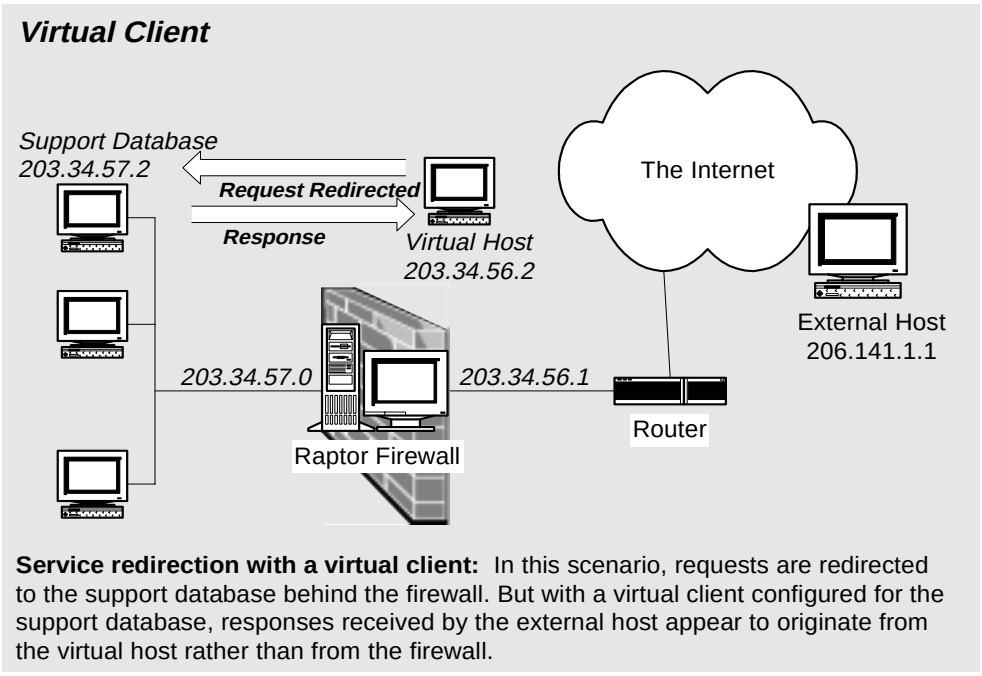
Figure 9. Address Redirection from RMC

Virtual Clients

New to Raptor Firewall 6.0, through the Redirected Services feature, are Virtual Clients. Creating a Virtual Client allows you to use a virtual address in place of the real address of the host initiating a connection. This is particularly useful if you have a redirected service configured on your network. By using the address of the virtual host as the source for a particular connection originating from behind the firewall (even with a redirected service involved in the connection) an external host can receive a reply from the same address it originally communicated with (see Figure 10). Without virtual clients or transparency in place, the external host normally sees the firewall address on any replies it receives.

For example, in Figure 10, the External Host only sees the Virtual Host address (203.34.56.2) when it connects to the Support Database. With service redirection configured, the packet is redirected to the Support Database (203.34.57.2). If the Support Database now initiates a connection back to the External Host, the External Host expects to see the address of the Virtual Host on the incoming packet. However, if the Virtual Clients feature is not enabled, the External Host will see Support's actual address (if transparency is configured) or the firewall address on any communication it receives back.

Creating a Virtual Client allows you to use the address of the Virtual Host as the source for any connection originating from the Support Database. Therefore, even with the redirected service involved in the connection, the External Host can receive a reply from the same address it originally communicated with.



Service redirection with a virtual client: In this scenario, requests are redirected to the support database behind the firewall. But with a virtual client configured for the support database, responses received by the external host appear to originate from the virtual host rather than from the firewall.

Figure 10. Virtual Clients

Services Disabled at Installation

The Raptor Firewall hardens the underlying UNIX or Windows NT™ host by disabling unneeded services at the operating system and application layers. Doing this ensures these services cannot be exploited by a knowledgeable hacker searching for unattended back-doors into your network.

This is another key feature of the Raptor Firewall's overall design. It sets the Raptor Firewall apart from more porous architectures based solely on filtering and state capturing techniques.

IP Packet Forwarding/Routing

The Raptor Firewall product does not directly forward IP packets. This prevents the firewall from acting as a router. For example, it will not allow ping requests to pass through to your internal networks.

By not providing IP packet forwarding services, the Raptor Firewall ensures a logical disconnect between protected and unprotected networks. This design guarantees that the firewall cannot "fail open" if system problems occur. The only way for data to pass through the firewall is by means of application proxies, and not through the operating system layer.

Source Routing

Disabling source routing prevents untrusted hosts on the Internet from specifying the exact route that the packets should travel. All source routed connection attempts are summarily rejected by the Raptor Firewall. As a further anti-spoofing measure, the Raptor Firewall automatically rejects packets received on its *external* interfaces that contain an *internal* IP address.

NFS

Though it is one of the most widely used UNIX network services, Network File System (NFS) is inherently insecure, and is disabled by the Raptor Firewall at installation time.

Other Disabled Services

At the application layer, the Raptor Firewall disables all services except the Raptor Firewall proxies themselves. Doing this serves to prevent attacks on unsecured network servers that are constantly accepting connections on TCP and UDP ports.

Strong and Weak Authentication Mechanisms

The Raptor Firewall supports strong and weak authentication methods. These methods use single-use and multi-use passwords, respectively. These methods are supported for FTP put and get, Telnet, and HTTP connections for both inbound and outbound access attempts.

Strong Authentication Methods

The Raptor Firewall supports several single-use password schemes for authentication of connections: Secure Dynamics SecureID® (ACE) authentication, AXENT's Defender, CRYPTOCard™, and Bellcore S/Key™. Since passwords generated by each of these schemes change with each login, using any of them eliminates the threat of password replay attacks.

SecureID (ACE) Authentication

ACE is a time-based authentication scheme that uses smart card technology. The ACE card produces a new six-digit password at sixty-second intervals.

To use this type of authentication, users must have installed the ACE/Server® software on a separate system behind the Raptor Firewall. The Raptor Firewall is then able to send and receive ACE authentication requests to that system for validation.

CRYPTOCard Authentication

CRYPTOCard provides another strong form of authentication for connections to the Raptor Firewall. Unlike ACE, CRYPTOCard uses a challenge/response method based on cryptographically generated passwords.

Authentication via this method requires the user to enter a response based on a numeric challenge issued by the Raptor Firewall. Upon receiving the challenge from the Raptor Firewall, the user enters it into the CRYPTOCard hardware device. The device computes a one-time password which the user must enter for authentication by the Raptor Firewall. The Raptor Firewall then sends the password to the CRYPTOCard authentication server for validation.

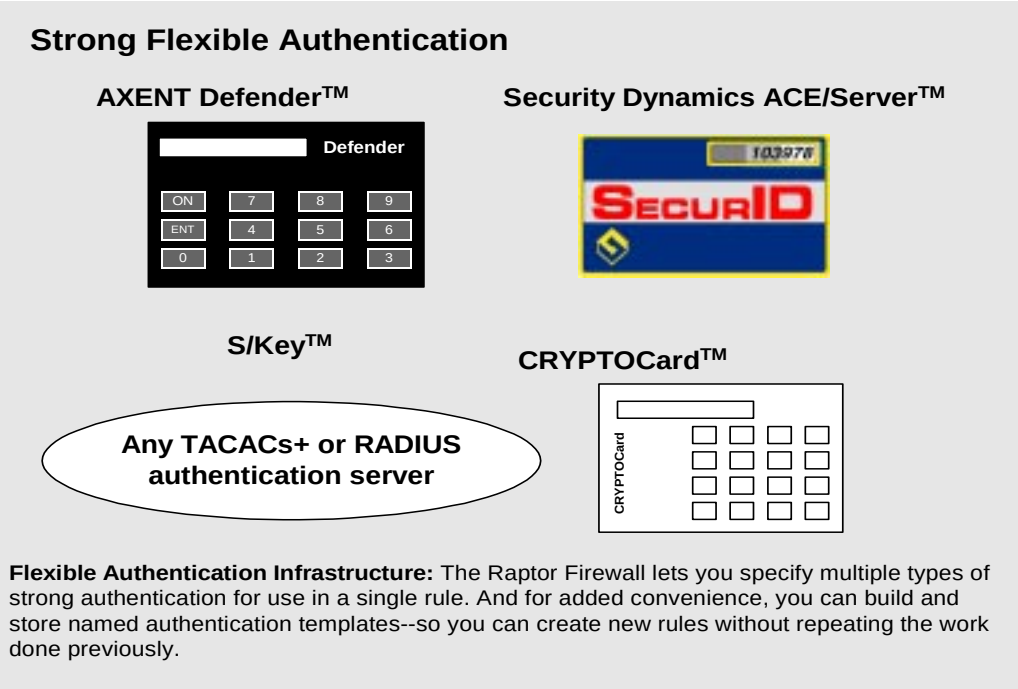


Figure 11. Strong Authentication Methods

AXENT's Defender Authentication

Defender by AXENT Technologies is a strong, cryptographically-based authentication method similar to CRYPTOcard. The user generates a password with a dedicated hardware device, then provides the password to the firewall when prompted. The firewall authenticates the password by sending it to the Defender authentication server.

S/Key Authentication

S/Key authentication is software based. The S/Key server is integrated within the Raptor Firewall. S/Key generators for PC, UNIX, and MAC clients are included in the Raptor Firewall kit.

To connect to the Raptor Firewall, S/Key users must provide the correct password and seed value to a local S/Key password generator. Upon supplying these, the S/Key software on the user's client system generates a one time password string in the form of six four-letter words.

The user enters this string when prompted by the Raptor Firewall. With each subsequent connection, the S/Key software generates a new password string and generates the user's iteration count. When the user's count decrements to zero, no further connections are permitted.

Weak Authentication Methods

The Raptor Firewall 6.0 supports two types of weak authentication: gateway password (called *gwpasswd* on the Raptor Firewall interface), and NT Domain authentication (for Raptor Firewall NT only). Although the Raptor Firewall 6.0 supports weak authentication, AXENT Technologies strongly recommends that users avoid implementing this type of authentication. An intruder could capture and re-use a given password, creating a critical security breach.

Suspicious Activity Monitoring and Rule Thresholds

The Raptor Firewall performs suspicious activity monitoring on all connections through the firewall. Suspicious activity monitoring works by keying off of the *Rule thresholds* you establish for connections when writing authorization rules. These thresholds, which are *off* by default, specify the level of usage you expect for each rule.

For example, you can write a rule that translates as follows:

Allow certain users on some external host to Telnet to some system in my network between noon and 3pm each day

The thresholds you establish for this rule can specify some number of accesses per 5 minutes, 15 minutes, hour, day and week. The Raptor Firewall regards each access that exceeds this level as suspicious. It automatically generates an *alert* message in the log file, and performs a traceroute to the client(s) that caused the threshold to be exceeded. You can also configure the firewall to issue one or more notifications in response to any alert message.

Accesses that exceed established rule thresholds do not necessarily constitute a security risk. In some cases, you may find it necessary to increase certain thresholds to accommodate higher levels of activity than you had anticipated when writing a rule. For this reason, the Raptor Firewall does not deny accesses that meet or exceed rule thresholds, but responds by logging an alert message.

Disabling SAM Thresholds

The Raptor Firewall 6.0 allows you to disable suspicious activity monitoring thresholds for any rule. HTTP rules provide an example of a case where disabling suspicious activity monitoring may make sense. High volumes of access to a web server probably does not indicate a security threat. In this case, turning off the thresholds can spare you the trouble of monitoring large numbers of HTTP-generated alert messages. All SAM thresholds are *off* by default. The firewall administrator must configure these thresholds as needed to support site requirements.

Automatic Alert Generation for Specific Events

The Raptor Firewall monitors all connections and connection attempts, and records these actions in its log files. The significance of these actions varies. The Raptor Firewall logs invariably include many benign events, like the startup and shut down of expected FTP sessions. However, they may also record more suspicious actions, like a user trying (and failing) to log in repeatedly to the network

The Raptor Firewall classifies all actions into one of seven message categories ranging in significance from *informational* through *emergency.*

Classes of messages and their associated meanings are as follows:

Message Severity	Implication
Emergency	Raptor Firewall (gwcontrol) has failed, and network traffic through the gateway has been shut down.
Critical	A Raptor Firewall ancillary service has failed
Alert	A suspicious activity threshold has been met or exceeded.
Error	Normal gateway activities cannot complete successfully.
Warning	Recoverable errors exist.
Notice	Attempted connection denied by the Raptor Firewall.
Informational	Connection attempt allowed.

Notifications are the Raptor Firewall's programmed reactions to these events. Each notification you establish directs the Raptor Firewall to perform a specific action (such as sending out an email message, or paging someone) in response to a specific event (such as a Warning message).

Notification Types

The following table shows the types of notifications the Raptor Firewall can generate in response to connection attempts.

Action Type	System Response
Audio	Plays an audio file on the gateway
Mail	Emails the text of the message to a designated recipient
Page	Transmits the text of the message to a designated alphanumeric paging device
Client	Launches a client program
SNMPv1	Sends an SNMPv1 trap to a designated system
SNMPv2	Sends an SNMPv2 trap to a designated system

The Raptor Firewall allows different (or multiple) actions to occur according to the severity of network activity it detects.

Event Logging for All Connections

The Raptor Firewall logs all network activity, making logfile messages available for viewing via the RMC Logfile window in Figure 13. Log messages include date, time, severity, computer names and IP addresses, and a description. Firewall administrators can examine the logfiles to identify potential or actual attacks, and underlying access patterns. The Raptor Firewall automatically backs up all log information on a daily basis.

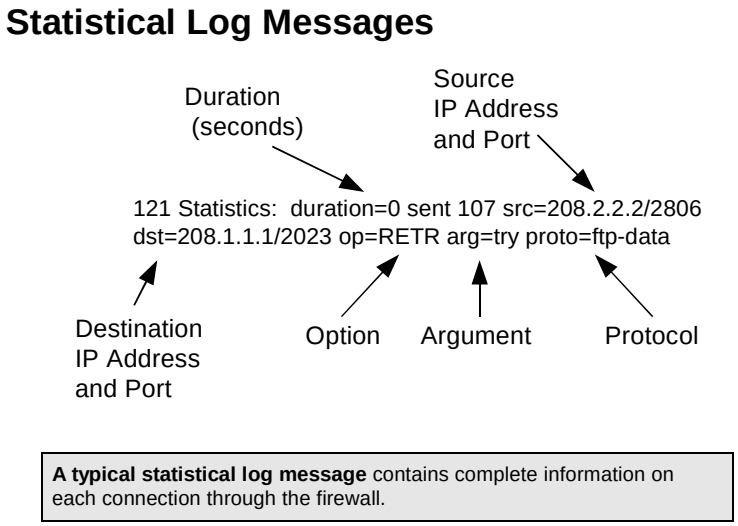


Figure 12. Example Statistical Logfile Message

Enhanced Logging Support

As part of the Raptor Firewall 6.0 release, event logging has been enhanced to allow for activity logging on a per rule basis. This provides the administrator with much greater control over what is logged and how often it is recorded. As part of the configuration process, you can define the level of normal activity that is logged to reduce the disk space requirements for log storage. Additionally, statistical data can be exported to an external database by means of the *flatten* command.

Also part of the Raptor Firewall 6.0 release, support for tracing is provided as a problem solving tool. This feature can be enabled for various processes through a simple edit of the config.cf file. Additionally, rule IDs are now included in the log, giving administrators the ability to associate connections with the matching rule.

Enhanced Logfile Viewing

With the new Raptor Management Console for Raptor Firewall 6.0, the firewall's logfile viewing capabilities have been enhanced. A filtering option allows you to view only designated logfile events. You can filter your logfiles using one or more of the parameters displayed in Figure 13.

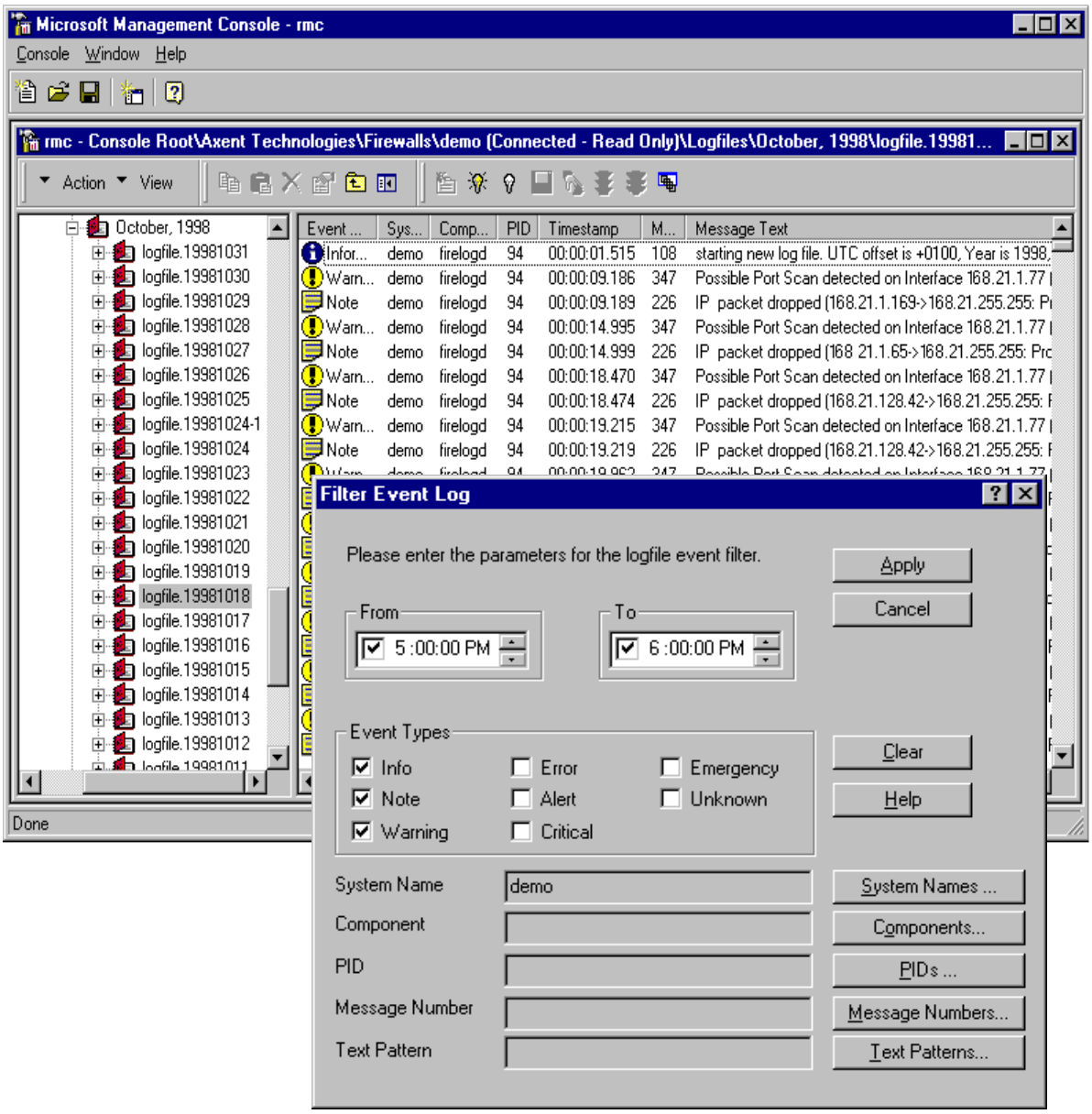


Figure 13. RMC Logfile Filter

Automatic Detection of Unauthorized Processes

On the Raptor Firewall, the Vulture program runs in the background, unobtrusively monitoring for suspicious processes running on the gateway. Specifically, Vulture hunts for and kills any processes that meet any of the following criteria:

- Are not part of the underlying operating system
- Are not part of the Raptor Firewall software
- Are not specifically allowed by the Raptor Firewall administrator

Administrators can customize Vulture by specifying services and users that are expressly allowed to run on the Raptor Firewall host. Because allowing unnecessary services to run on the firewall can compromise the security of the system, however, this practice is not recommended.

Summary of Features

Following is a partial list of key Raptor Firewall architectural features:

- The Raptor Firewall is an *application-level proxy* engine that examines both TCP- and UDP-based connection attempts into and out of the networks it secures. Packets from non-VPN connection attempts travel up the protocol stack to the application level for processing. Here they are manipulated according to firewall rules and forwarded to their intended destination address.
- You can configure the Raptor Firewall to conceal as much of the *topology of your inside networks* and your *network addresses* from public view as you require. Network administrators can use transparency, service redirection, and virtual clients to meet specific site needs, making the Raptor Firewall as flexible as it is powerful.
- Support for multiple network interfaces makes the Raptor Firewall an extremely flexible tool for creating secure connections among multiple disparate networks.
- The Raptor Firewall allows or denies a connection attempt based on defined *authorization rules* and associated *authentication methods*. Once a connection is authorized and the source is authenticated, application specific proxies (SMB/CIFS, HTTP, SMTP mail, and so forth) examine the application data flowing over the connection for security threats.
- Support for multiple types of strong authentication is built into the firewall.
- Strong anti-spoof checking, port scan detection, and fine-grained filtering are basic components of the firewall that allow administrators to create strategic defenses against unwanted network intrusion.
- VPN technology is fully integrated into the Raptor Firewall 6.0. The ability to create IPsec- or swIPe -compliant, secure tunnels between designated hosts or subnets inside and outside of your network is designed into the firewall, not a separate add-on product. Support for ISAKMP/Oakley (IKE) dynamic keying is also included.
- To fine-tune tunnel security, the Raptor Firewall allows you to create packet filters and filter groups for VPN tunnels, and force tunnel traffic up the protocol stack through the firewall application proxies. This allows you to limit the *type*, *direction*, and *content* of traffic sent through the tunnel.
- On a *selective* and *readily configurable* basis, the Raptor Firewall allows transparent connections between certain hosts in the protected network and clients or servers on the unprotected (public) network.

- On a selective and transparent basis, the Raptor Firewall can *redirect* certain service connections destined for specified hosts to different host/port combinations. This feature can also be used to balance traffic loads among several network hosts.
- The Raptor Firewall automatically alerts system administrators (or other designated personnel) if the volume of access attempts to given hosts are suspiciously high.
- The Raptor Firewall provides extensive logging support, including collecting statistics for connections and connection attempts, traffic volume, and other characteristics. This data can then be exported to a database for storage purposes or further interpretation.

Section 3

Managing the Raptor Firewall

Using the Raptor Management Console

The Raptor Management Console (RMC) ships as an integral part of version 6.0 of the Raptor Firewall for NT. RMC is the means by which you configure and manage the firewall from an NT system. As shown in Figure 14, RMC's Windows look and feel translates into a straightforward design which makes the firewall easy to configure and manage. This section describes the RMC GUI, and highlights certain key features of the GUI to demonstrate its ease of use. For a tour of RMC, go to our web site at `www.axent.com/product/`, select the Raptor Firewall 6.0 and pull down the demonstration of our Raptor Management Console.

The main RMC windows used for firewall management and configuration can be broken down into three categories: the RMC Root Directory, RMC Property Pages, and RMC Property Page tabs. The root directory, displayed in Figure 14, represents each aspect of the firewall as an icon. These icons allow you to easily access the RMC Property Pages (*see* Figure 15) which provide fields for firewall configuration. Each Property Page contains tabs which help you navigate through the window and configure all the necessary components.

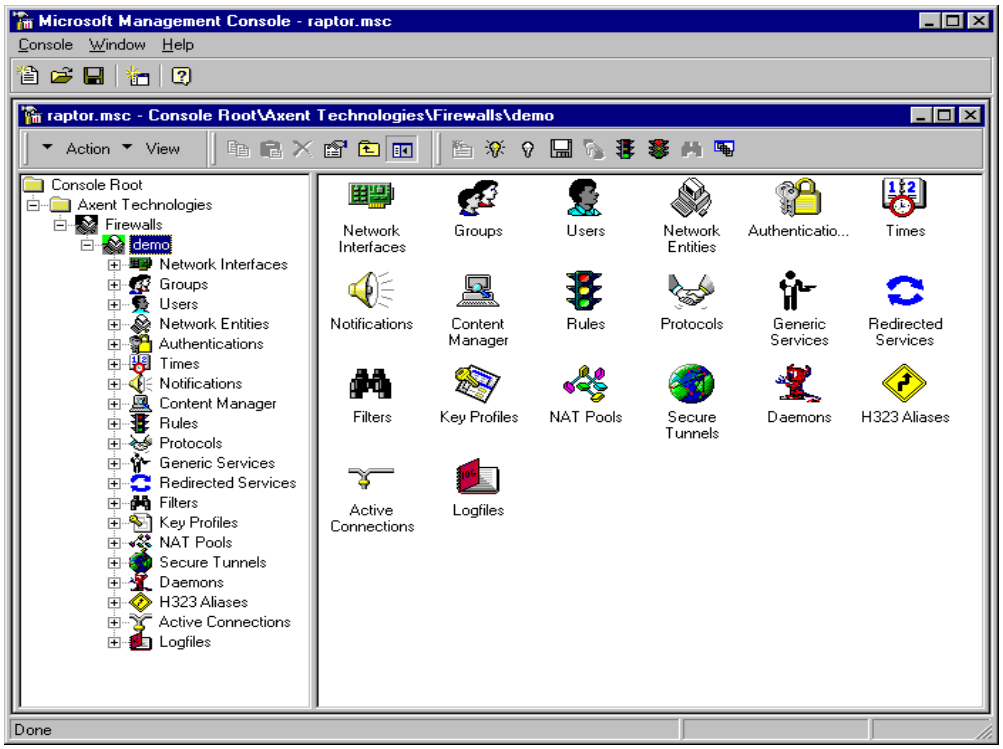


Figure 14. RMC Root Directory Window

Creating Rules

By default, the Raptor Firewall 6.0 denies any connection that is not explicitly allowed by an authorization rule. Unlike packet filtering firewalls, which use a "first fit" method, the Raptor Firewall 6.0 evaluates rules on a "best fit" basis. This assures use of the most conservative and specific rule for each connection attempt. The Raptor Firewall 6.0 rules are also non-order-dependent, simplifying the task of establishing a security policy for administrators. The ability to weigh each rule for best fit, as opposed to simply using the first rule that fits, sets the Raptor Firewall 6.0 apart from filtering firewalls, routers, and more complex filtering schemes that require the capture of the "state" of each packet.

The Raptor Firewall 6.0's authorization rules evaluate each connection based on a wide range of criteria, including:

- Source and destination address of the connection
- Type of service
- Network interface of the incoming connection
- Time of day and date restrictions
- Group and user restrictions
- Restrictions based on strong authentication methods

Because the Raptor Firewall 6.0 is an application proxy firewall, each connection attempt is passed up the protocol stack, where a secure proxy scans for all information required to complete the connection. The Raptor Firewall 6.0 then searches its database for the rule that best matches the connection attempt.

The Raptor Management Console makes specifying rules easy. Figure 15 shows the RMC Rules Property Page and indicates where to add information to create a rule.

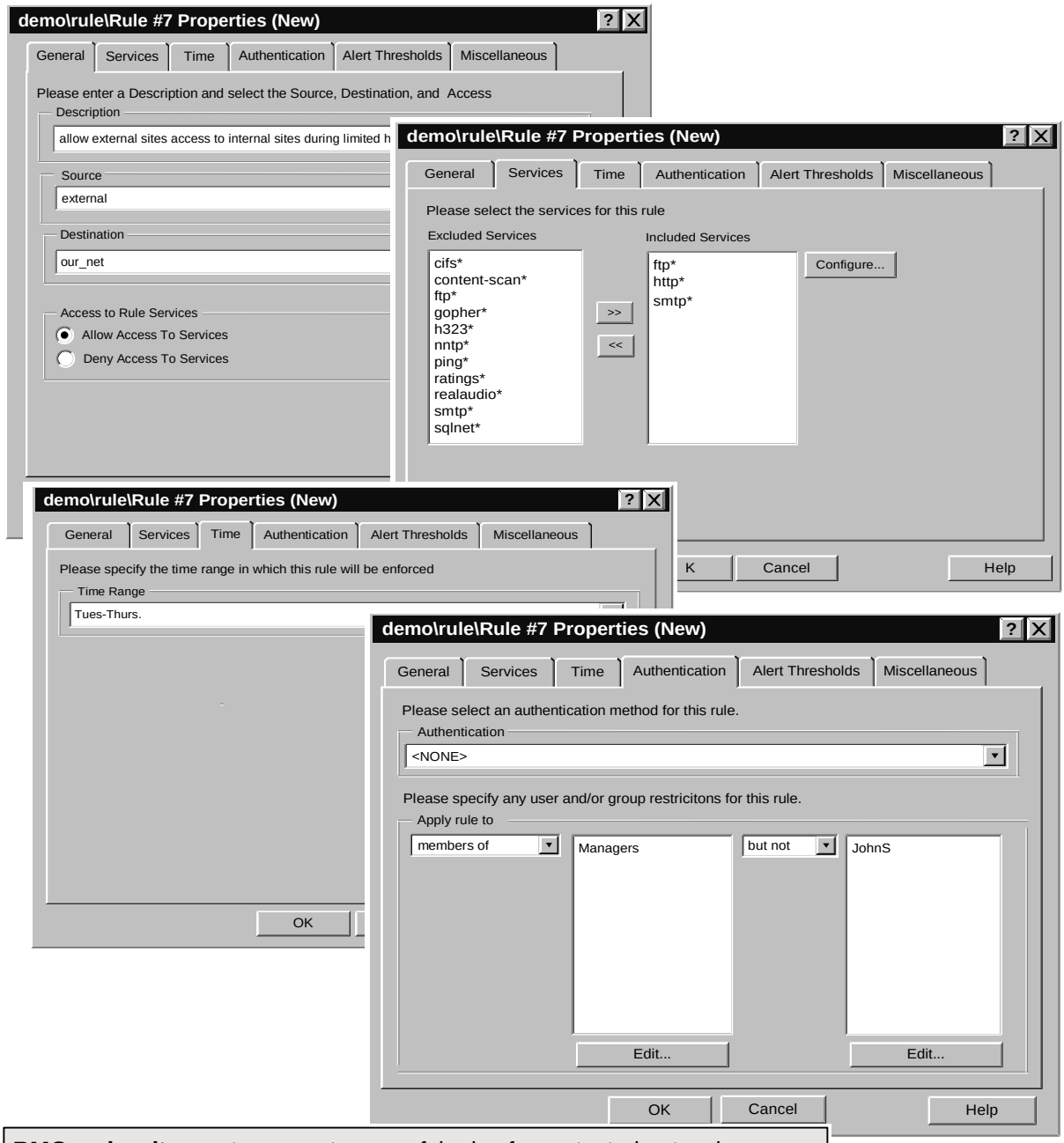
Overview of Rule Processing

The Raptor Firewall 6.0 performs rule scanning in two stages. On the first pass, it determines which subset of rules can apply to a particular connection attempt. On the second, it determines which rule provides the *best fit*. If no rule matches its best-fit criteria, the Raptor Firewall 6.0 disallows the connection attempt and logs a denial.

The Raptor Firewall 6.0 begins by selecting all rules that apply to a requested service, and that match the source and destination entities equally well. Source and destination IP address are given the highest priority in determining the match. Also, the source address can be further qualified by specifying the interface on which the connection arrives. Following this initial selection, the gwcontrol process applies additional criteria to determine which rule best matches the connection attempt, as described in the next section.

After selecting all available rules that apply, the Raptor Firewall 6.0 determines if any rule permits the requested connection. If one or more rules do, the firewall decides if user authentication is required based on the authentication methods chosen for the rule. If authentication is not required, the Raptor Firewall 6.0 gwcontrol process selects a rule based on a different set of criteria.

Fast Graphical Rules Generation



RMC makes it easy to generate powerful rules for protected networks. RMC's Property Page tabs give you quick access to the building blocks you need to create your network policies. Simply point and click on the appropriate tab to access the rule element you want to configure. To view the information used in any rule, select the Rules icon in the RMC Root Directory. All your rules appear in the right pane.

Figure 15. Creating Rules with RMC

How it Works

Figure 16 is a conceptual view of the way the Raptor Firewall 6.0 handles connection attempts. A connection attempt triggers the following chain of events:

1. When the security proxy receives a connection attempt, it spawns an instance of itself (through process or thread creation) and prompts the sender for a destination address.
2. Upon receiving the destination address, the security proxy has the following information: source and destination address of the connection, type of service, firewall interface on which the connection was received, whether or not it came through a VPN tunnel, and the interface on which the server connection will go out.
3. The security proxy delivers this information to the authorization process, *gwcontrol*, which verifies the source and destination address, and checks to see whether a rule exists that authorizes the intended connection.

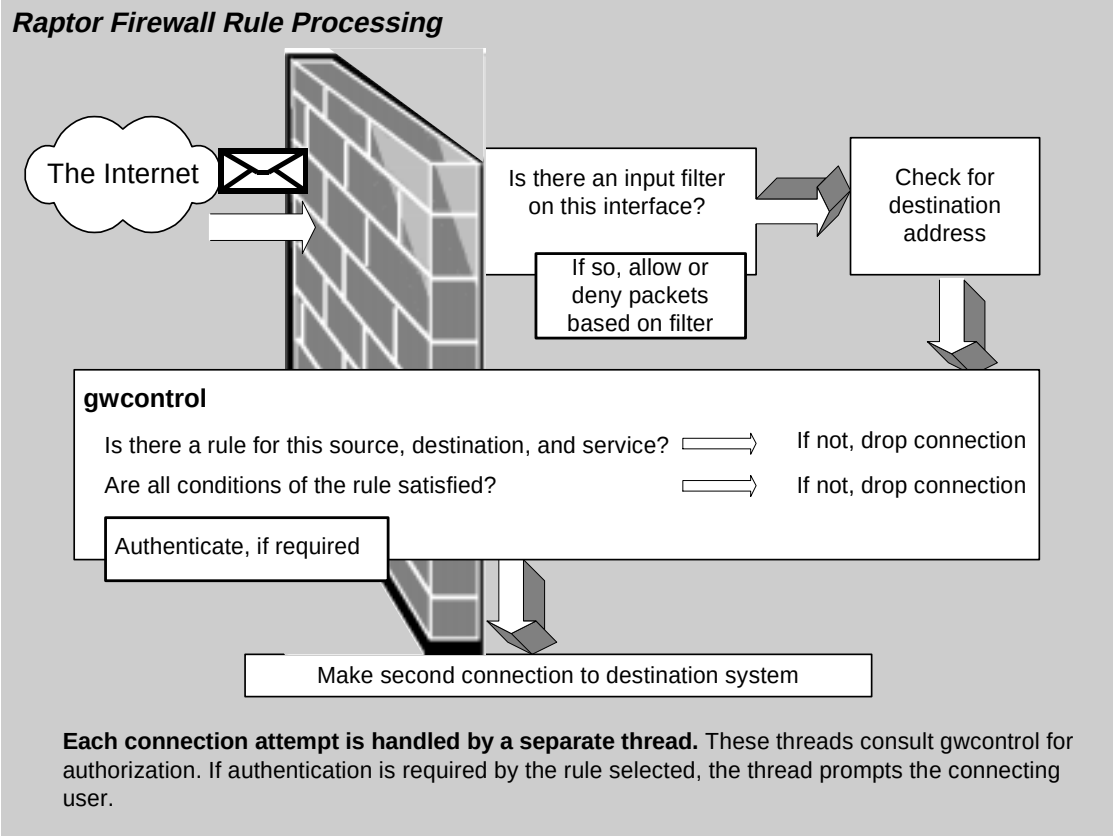


Figure 16. Raptor Firewall Rule Processing

4. After selecting an authorization rule, gwcontrol checks connection limits, which derive from the rule chosen. They can specify a user, a group, time, and/or authentication mode for the connection.

   Authentication can be multi-use password, or a single-use password scheme such as Bellcore S/Key, Security Dynamics SecureID (ACE), CRYPTOCard, or AXENT Technologies Defender. Because the Raptor Firewall 6.0 firewall supports TACACS+ and RADIUS, authentication can be made extensible.

5. The security proxy then authenticates the connection if this is required by the rule selected, and either allows or disallows the connection.
6. If the connection is allowed, the security proxy creates a second connection from the firewall to the destination system, and begins proxying the data stream to it.
7. Because the second connection is created by the firewall, outbound packets carry the firewall's IP address as their source address, unless client-side transparency is enabled on the destination interface. In which case, the source address is that of the originator. The Raptor Firewall 6.0 associates each session with a unique ID and the logical port of the security proxy.

Security Policies and the Raptor Firewall 6.0

As the focal point for your site's security framework, the Raptor Firewall 6.0 is designed to implement and enforce *a security policy* across a range of key areas. These areas include, but are not limited to, the following:

- Delivery of electronic mail
- Control of web browsing activities
- Configuration of your name server

The way in which you manage these and other features is central to your site's overall security stance. The reasons for this are obvious. Electronic mail is perhaps the most basic and defining feature of the networked world, enabling people and groups in disparate locations to meet, share ideas and coordinate activities as if under the same roof. For similar reasons, use of Internet browsers has grown explosively, placing a broad range of information and services within keystrokes of any user. Browsers themselves are evolving rapidly, incorporating file transfer, secure commerce, and other capabilities once isolated to LAN-based applications.

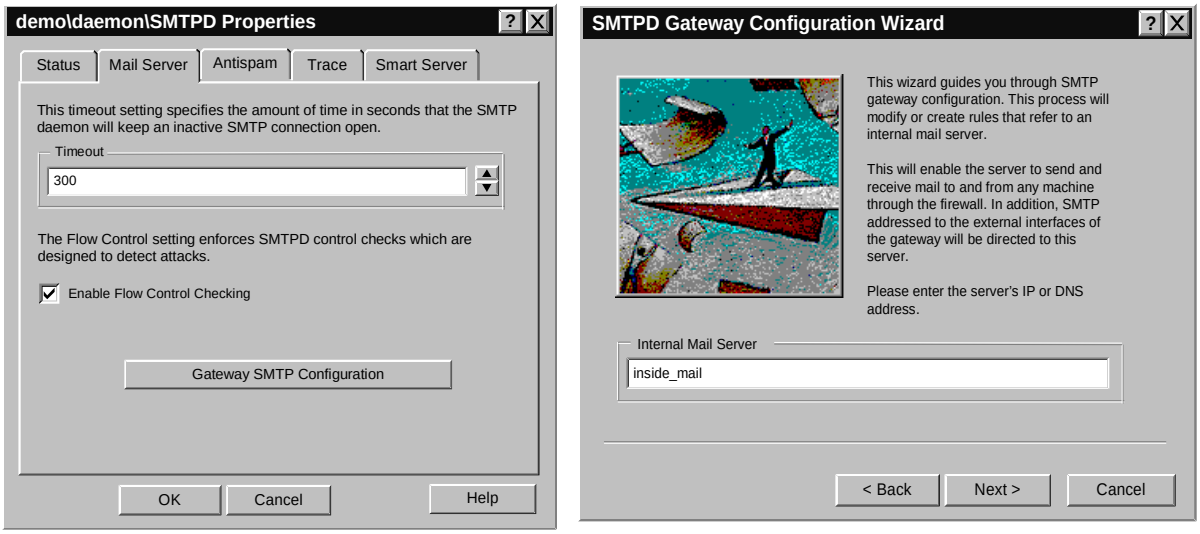
Secured Email: The SMTP Security Proxy

The Raptor Firewall 6.0 SMTP security proxy sits at the application level and supports transparent, bi-directional access for email connections through the firewall. Like the other Raptor Firewall 6.0 security proxies, the SMTP proxy accepts or rejects delivery of email on a connection-by-connection basis, subject to the existence of authorization rules.

Unlike packet filters or *stateful* firewalls that simply allow SMTP packets into your network, the Raptor Firewall 6.0 SMTP security proxy performs syntactical and other checking on each email connection, and scanning for known mail-based forms of attack. Moreover, the SMTP security proxy does not send, receive, or store electronic mail. This ensures that traffic throughput is not negatively affected by the proxy's operations, and that the firewall system itself is not open to email-based attacks.

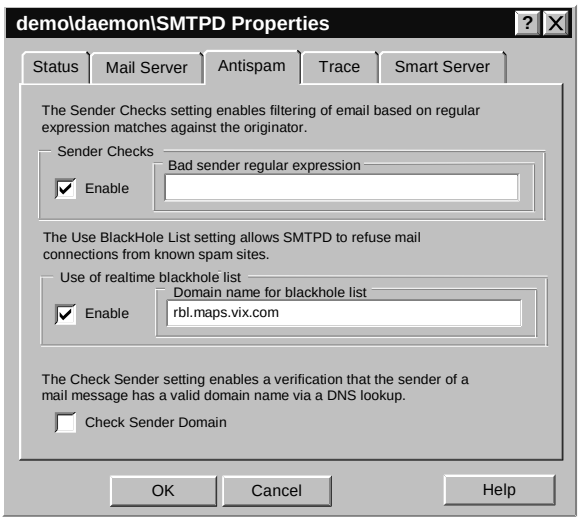
Furthermore, in order to prevent the Raptor Firewall from being used as a spam relay, version 6.0 includes SMTP mail enhancements which allow you to specify limits on the number of recipients allowed in a mail message. You can also specify which domains the firewall will accept mail for. Additionally, the SMTP proxy can be used in conjunction with the Realtime Blackhole List (RBL) which maintains a database of known spammers (see Figure 17). This way, the firewall will drop any incoming mail connection from an IP address listed in the RBL.

Delivering the Mail



The Raptor Firewall SMTP Wizard greatly simplifies mail policy configuration. You provide the information on internal and external mail servers, and indicate whether to allow direct mail to end systems. The Raptor Firewall then automatically creates all necessary mail delivery rules.

AntiSpam Measures



The Raptor Firewall's AntiSpam measures allow you to protect your internal mail server from being used as a spam relay. By specifying domains for internal users, only messages directed at those domains are accepted. The firewall also uses the Realtime Blackhole List to identify and stop known spammers.

Figure 17. Email Setup

Secure Web Browsing: The HTTP Security Proxy

Many security gateways or firewalls allow a special pass-through for HTTP packets. Not so with the Raptor Firewall 6.0. To protect your resources, the Raptor Firewall 6.0 implements a customized HTTP security proxy to proxy each connection attempt to its destination.

As in the case of our other security proxies, this assures that only connections explicitly allowed by a well-defined rule are permitted. You can also configure the Raptor Firewall 6.0 to direct all out-bound web accesses to a web caching server. This action is transparent to end users, and can greatly reduce the volume of internet accesses required by your user community.

Raptor Firewall 6.0 supports HTTP v1.1 features such as *connection persistency* and *request pipelining*. This combination significantly enhances throughput and latency when used with browsers that support HTTP v1.1. You can also configure external proxies on a per rule basis for HTTP in the event that you wish to assign proxies to a particular subset of HTTP connections.

Additional HTTP-based Services

As the use of browsers has grown, so has the list of supported services. For instance, many browsers now support file transfer capability (using FTP) and access to Gopher URLs. Some support encrypted HTTP or HTTPs, and SHTTP. The Raptor Firewall 6.0 supports these additional user requirements with an array of HTTP-based services. You can use these services to achieve fine-grained control over browser traffic into and out of your network. In addition to plain HTTP, the Raptor Firewall 6.0 allows you to write specific rules for the following services:

- HTTPs
- FTP (HTTP between client and firewall; FTP from firewall)
- Gopher (HTTP between client and firewall; Gopher from firewall)

Service Restrictions

In addition to limiting web browsing to those with business needs, the Raptor Firewall 6.0 makes it easy to restrict the web sites that users can access, and the types of files/MIME types they can bring into your network. The latter capability is particularly important. Certain file/MIME types, such as Java applets and other executables, can do severe harm if introduced into your network. In addition, certain sites contain information that makes any access to them problematic. Examples of this include pornographic, violent, or other materials that are not appropriate in a business environment.

The service restrictions the Raptor Firewall 6.0 supports give administrators the power to limit browsing activities in specific, carefully defined ways. This ensures that organizations get the full benefit of the Internet's resources, while avoiding unnecessary risks and performance degradation.

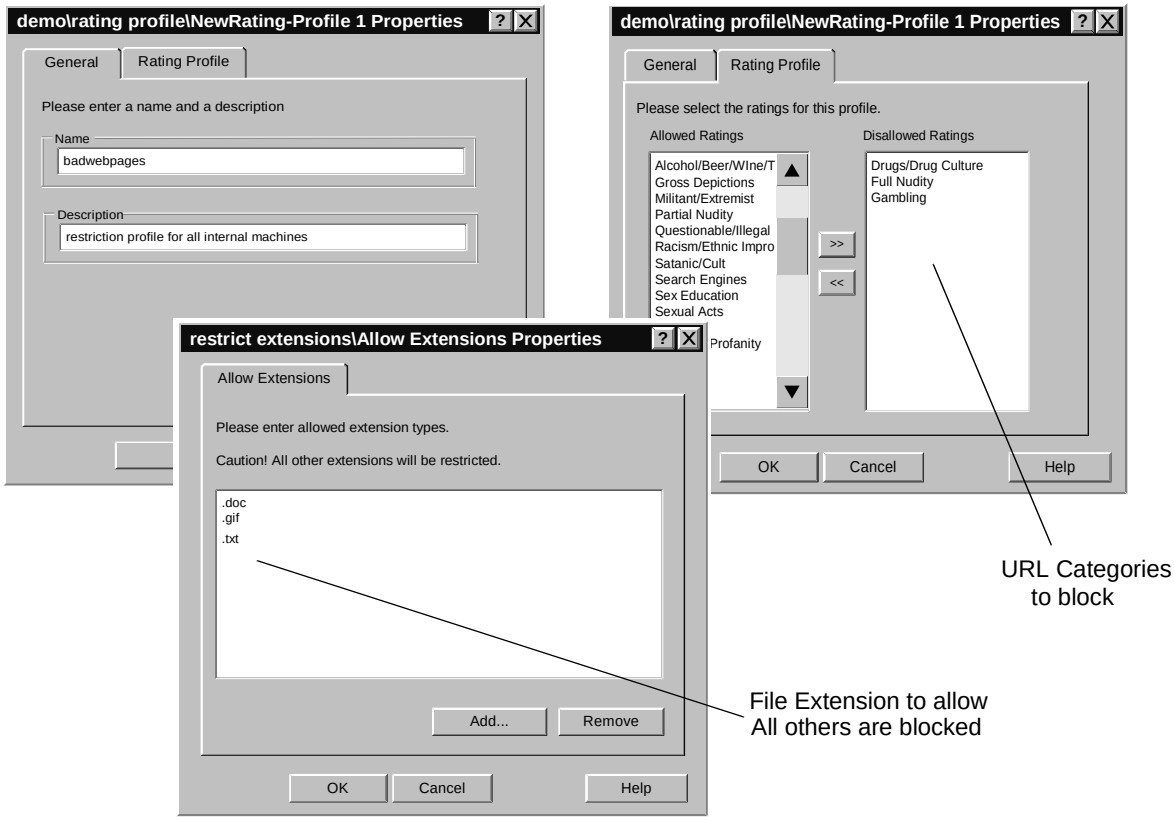
Selective Site Blocking with WebNOT and NewsNOT

WebNOT is an Internet access management utility that provides a convenient way to block access to inappropriate locations on the web. WebNOT draws upon on a continuously updated list of sites that are organized in the following categories:

Violence/Profanity	Partial Nudity
Full Nudity	Sexual Acts
Gross Depictions	Intolerance

Satanic/Cult	Sex education
Militance/Extremism	Drugs/Drug culture
Questionable/Illegal & Gambling	Alcohol and Tobacco

Secure Web Browsing, Caching, and URL Restrictions



Setting up policies for secure Web access is one of the most important aspects of your security framework. The Raptor Firewall places all of the functionality you need at your fingertips. The WebNOT URL blocker enables you to create custom profiles for later application in rules. You can also designate an external caching proxy, and identify the types of files (such as executables, graphics files, and Java applets) not to allow into your site.

Figure 18. Configuring Web Access and Restrictions

By applying these profiles to the HTTP rules you create for the Raptor Firewall 6.0, you can enforce your organization's policies on information access. Moreover, WebNOT lets you restrict access at the file directory or page level, so you can restrict objectionable locations without denying access to an entire site.

The Raptor Firewall 6.0 also supports the ability to block access to NNTP-based news servers through the use of NewsNOT. Since its inception, the Internet has been home to an increasing number of newsgroups of various kinds. With NewsNOT, you can now control access to news sites containing offensive or objectionable material. NewsNOT functions similar to WebNOT. The firewall administrator can filter access based on IP address, user groups, groups, newsgroups names, or ratings available through a subscription service. You can also apply these blocking services to gopher, as well as HTTP.

Graphically Configurable Domain Name Server

The task of installing and properly configuring the Domain Name System (DNS) can be one of the most difficult and time-consuming aspects of firewall installation. A true distributed name service, DNS functions as a kind of global directory that maps internet names (such as *my.college.edu*) to specific addresses (128.234.56.7, for example) in much the same way as a telephone directory matches names and numbers. The DNS makes it possible to send and receive electronic mail, browse to specific locations, and use other services requiring the exchange of data across the public Internet.

As part of the Raptor Firewall 6.0, AXENT ships an implementation of DNS called *DNSd*. AXENT has designed this application to simplify DNS configuration and management for the firewall administrator.

How it Works

The Raptor Firewall 6.0 DNS security server makes it easy for administrators unfamiliar with the concepts behind DNS to set up and administer this important part of their security framework.

The Raptor Firewall 6.0 DNS security server works by maintaining separate public and private databases of DNS entries on the firewall. This makes it unnecessary to configure a split-level DNS system, which requires two separate DNS servers, each of which must be maintained by a network administrator. In addition, the server uses two files (called *hosts* and *hosts.pub*) to store all DNS information, including information on forward and reverse lookups. This simple mechanism simplifies both initial setup and ongoing maintenance of the security server. Figure 19 shows a sample topology with Raptor Firewall 6.0 DNS incorporated into the network.

The Raptor Firewall 6.0 is flexible enough to accommodate most any configuration you require. Network administrators are not limited by DNSd. Through RMC on NT or RCU on UNIX, they can create a dual-level DNS proxy as well as other configurations, including default name server override, public and private mail servers, and recursion for outside systems. A discussion of these advanced features is beyond the scope of this document. For details, consult the AXENT Technologies Web site (www.axent.com).

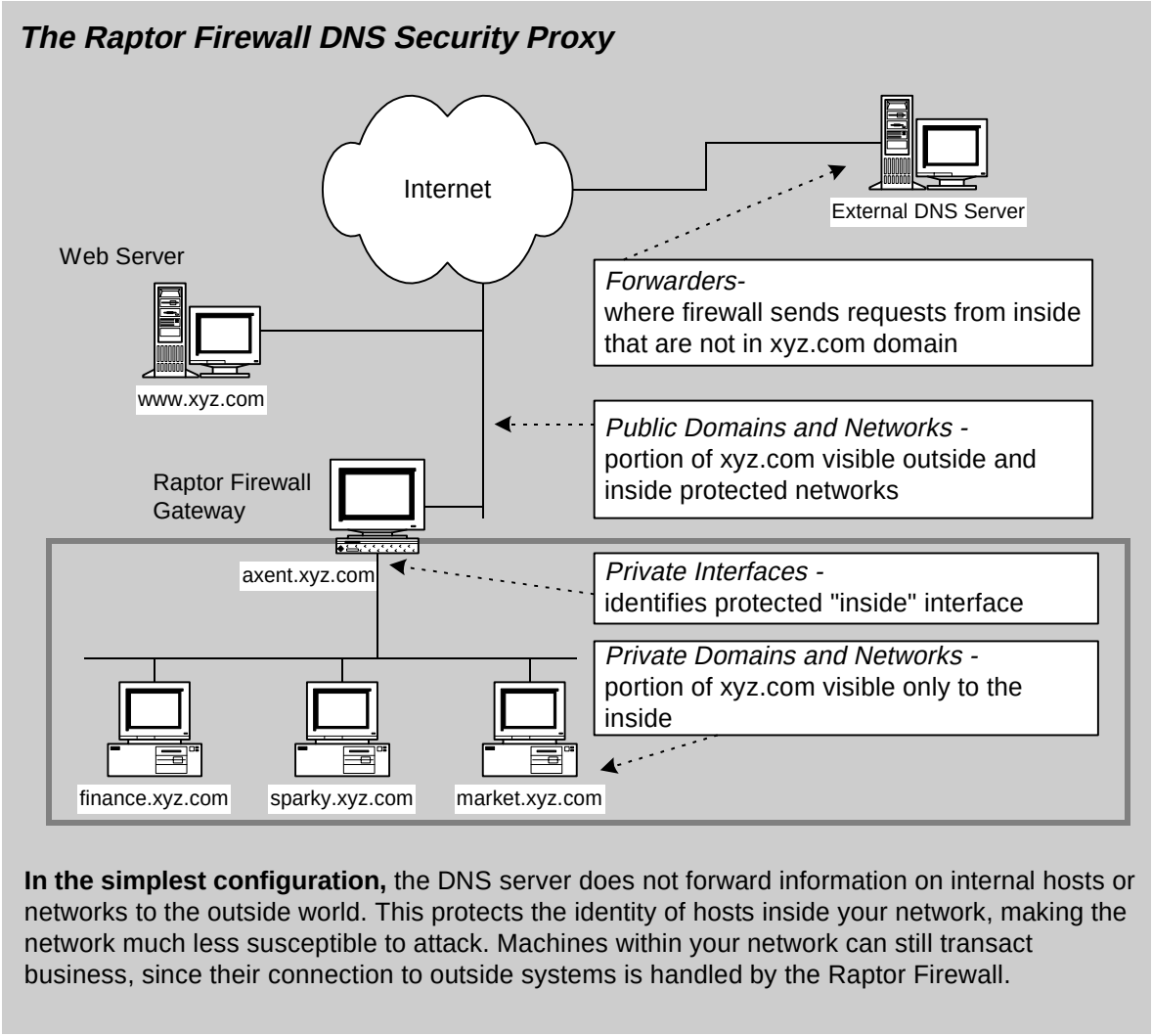
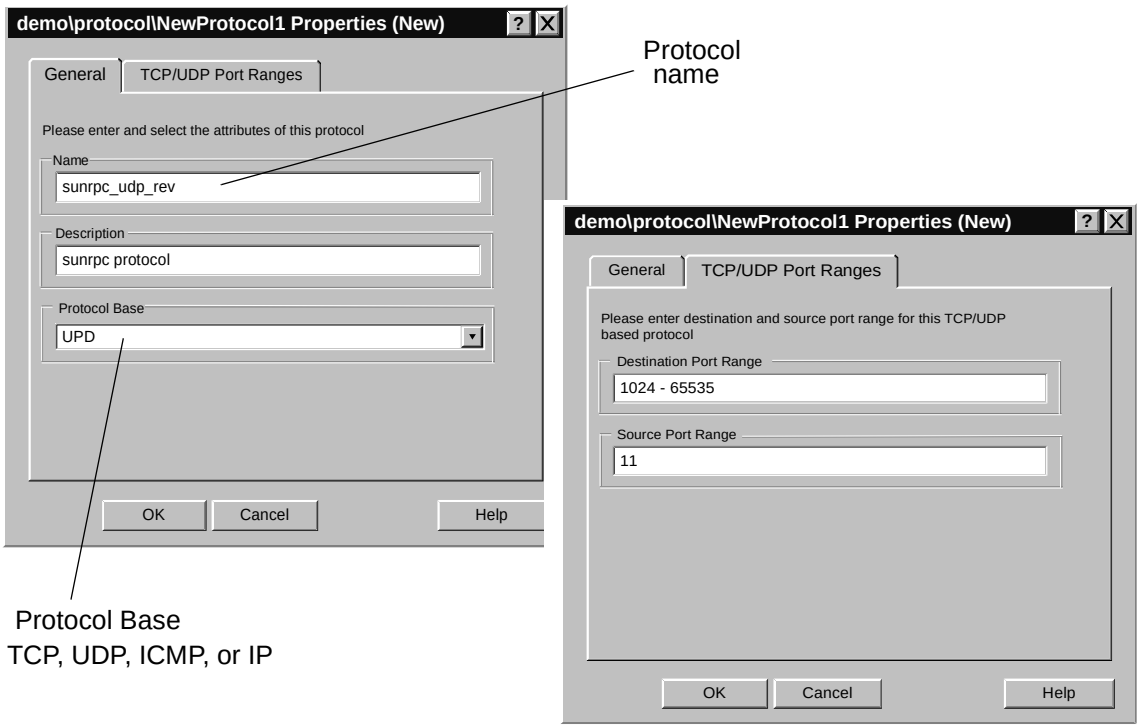


Figure 19. Raptor Firewall 6.0 DNS Setup

Support for Additional Protocols

In addition to those supported out-of-the-box, the Raptor Firewall 6.0 supports other well-known protocols or services through a generic service proxy (GSP). If necessary, you can specify new services by providing information on the underlying protocol (TCP and UDP are supported) and other information such as the source and destination port range for the service. Once you've defined these services, the Raptor Firewall 6.0 proxies them in same way as it does Telnet, FTP, or SMTP. Figure 20 shows the RMC Property Page for creating a GSP.

Specifying New Protocols



Defining a new protocol is a snap with RMC. All you do is provide the information the Raptor Firewall needs to recognize the protocol. You can then write rules that use the service, or filter for or against the service within VPN tunnels.

Figure 20. Defining New Protocols with RMC

Section 4

Raptor Remote and Network Topologies

Enterprise Security and Remote Management

Providing a strong perimeter defense is only one aspect of securing your enterprise domains. With Raptor Remote, a full-featured, enterprise-level firewall, you have the following capabilities:

- Enables a company to protect remote resources connected to the corporate network from security breaches by outside users.
- Enables a company to protect resources on internal subnets from attack by users on other internal subnets.
- Adds *remote management capabilities* to the Raptor Firewall's proven strengths as a perimeter firewall.

The Raptor Remote firewall is functionally equivalent to the Raptor Firewall Enterprise product, *minus* the GUI management layer. The Raptor Remote firewall in an Internet configuration supports sites that require multiple internet firewalls managed from a central location. The Raptor Remote firewall provides the following functionality:

- Rules and databases are stored locally on the system on which the firewalls reside.
- Raptor Remote can be remotely managed by multiple management tools, RMC or RCU GUI, or both.
- Management sessions are encrypted to protect your network configuration.
- Remote access requires a Gateway name and password.
- The management tool connects to the Raptor Remote in the same manner as any Raptor Firewall.
- Previously saved Raptor Firewall 6.0 or Raptor Remote configurations can be recalled for quick connection.
- Full-featured Raptor Firewall 6.0 functionality for internal and enterprise-wide applications.

Secure Remote Management Capability

The remote administration feature allows you to use the Raptor management tool of your choice to edit gateway configuration information and control the operation of multiple remote Raptor Firewalls. Configuration files, rules, and user databases are stored on the remote system. A highly secure, encrypted connection is used for all data communications between the Raptor management interface and Raptor Remote. Figure 21 illustrates the flexibility and security of Raptor Remote in an extended enterprise configuration.

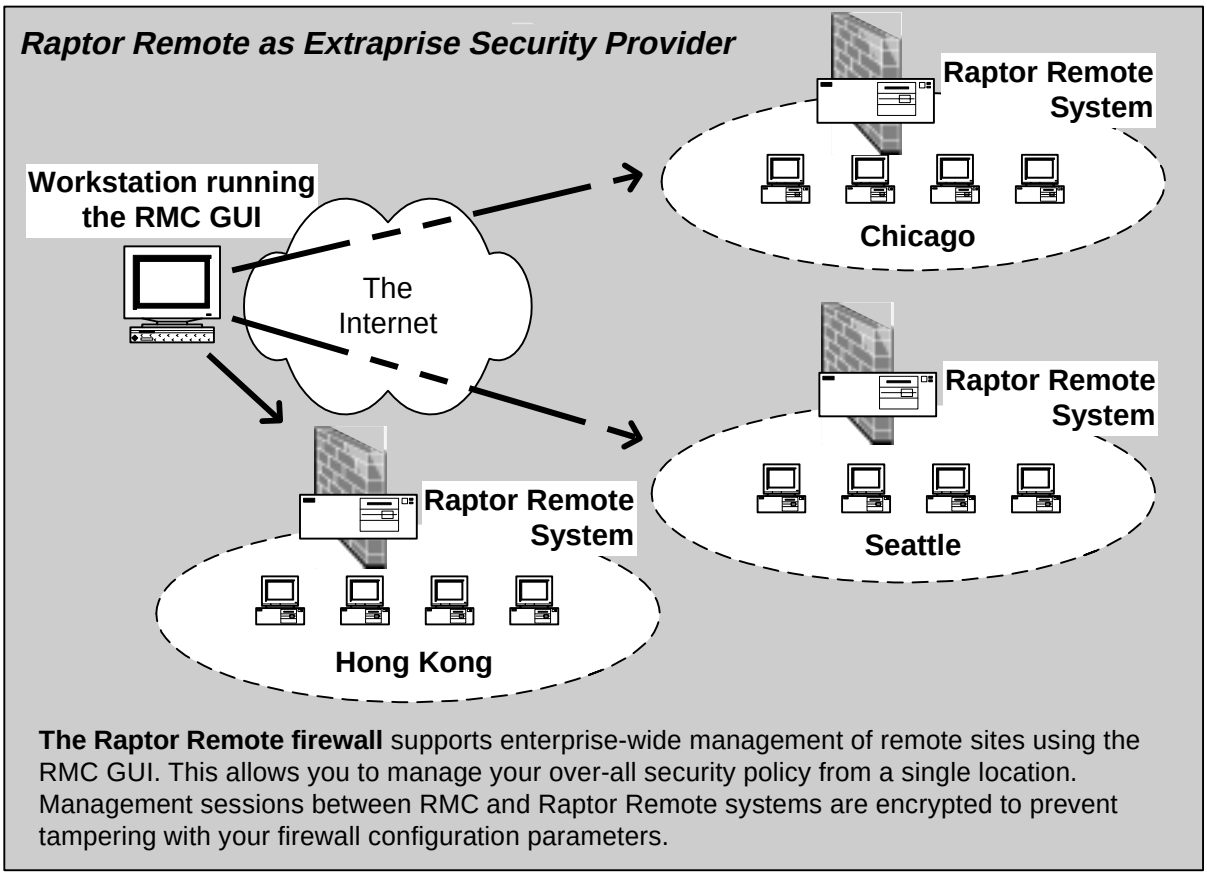


Figure 21. Remote Management with Raptor Remote

Management of Raptor Remote is supported through three components: Rempass, Readhawk, and Gwproxy.

- Rempass is a *password* program on the Raptor Firewall 6.0 for registering allowable remote management stations, remote log event submission, or remote content scanning.
- The Readhawk process runs on the Raptor Firewall 6.0. It provides the management interface with a secure network interface for retrieving Raptor Firewall 6.0 configuration files and returning modified rules and other configuration information to the Raptor Firewall 6.0.
- Gwproxy runs on the Raptor Firewall 6.0. It provides the management interface with a secure network front-end to gwcontrol.

Rempass

New with release 6.0, Rempass is a password program on the Raptor Remote for registering allowable remote RMC management stations. Rempass must be executed at install time. The program prompts the system manager for the DNS resolvable names or the IP addresses of the machines on which the managing RMC GUI will be run.

The system manager is also required to provide a password or "key" which will be associated with each RMC for use in the encrypted data communication between RMC and this Raptor Firewall 6.0 during remote management.

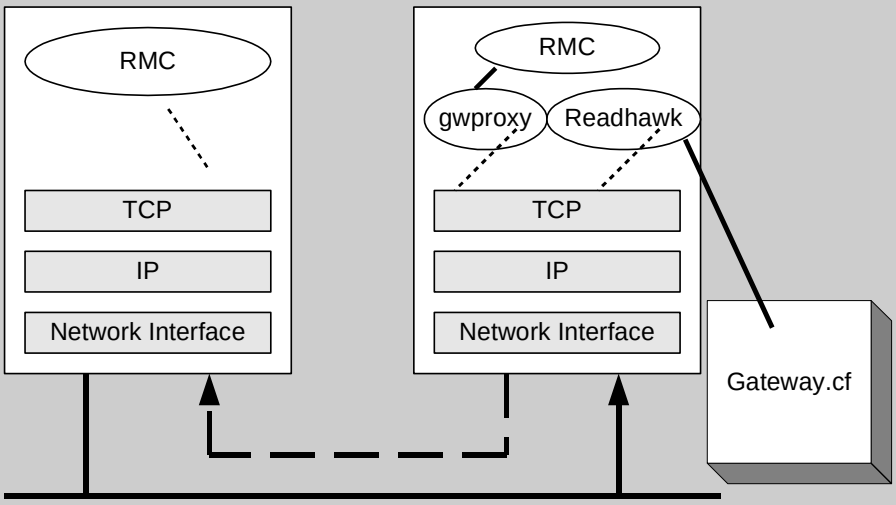
Rempass may be subsequently run from the Raptor Firewall console, to add new entries or modify existing RMC entries.

Readhawk

Readhawk provides RMC with a secure network interface for retrieving configuration files and returning modified rules to the Raptor Remote firewall. Readhawk accepts one connection at a time. When a connection from a remote RMC is accepted, Readhawk attempts to locate the RMC's corresponding key. If no key can be located, the connection with the remote RMC is terminated.

To defend against replay attacks, Readhawk sends RMC a randomly generated challenge for use in all messages passed in both directions on the connection. Additionally, each message sent across this connection includes an MD5 Digest computed across both the challenge and the data contained in the message. Use of the MD5 digest guards the data against tampering in transit. Readhawk then enters a command processing loop, waiting for and then acting upon commands received from the remote RMC.

RMC to Raptor Remote Secure Data Path



All communication between RMC and the Raptor Remote it manages is passed over a secure socket interface. The Raptor Firewall first authorizes (or refuses) connection by RMC. If a connection is allowed, the Raptor Firewall includes an MD5 digest to guard against tampering with data in transit.

Figure 22. Raptor Remote and RMC Communications Path

Gwproxy

The Gwproxy program provides RMC with a secure network front-end to the gwcontrol daemon. Gwproxy will accept only one connection at any given time. When a connection from a remote RMC is accepted, based upon the IP address and/or name of the attached peer, gwproxy attempts to locate the RMC's corresponding key. If no key can be located, the connection with the remote RMC is terminated.

Gwproxy then attempts to connect to gwcontrol. If gwproxy is unable to connect to gwcontrol, the connection is terminated. Gwproxy then sends the RMC a randomly generated challenge which is used in all messages passed in both directions on this connection. Additionally, each message sent across this connection will include an MD5 Digest computed across both the challenge and data contained in this message. Gwproxy then enters a command processing loop, waiting for and acting upon commands received from the remote management interface.

Raptor Remote as Internal Firewall

The majority of network and system attacks occur from the inside of corporate networks, launched by "inside" or trusted people. The sensitive nature of information stored in certain internal areas makes it vitally important to protect these areas from unauthorized access. When using Raptor Remote in this application, you create a firewall *behind* the firewall. Each Raptor Remote system has the same robust, secure proxy architecture as the Raptor Firewall 6.0 enterprise firewall. Each Raptor Remote system can be remotely managed by a single, designated Raptor Firewall 6.0 enterprise firewall. This management strategy makes it easy for administrators to deploy multiple internal firewalls, and manage them all from a single location. Figure 23 is a conceptual view of a typical network topology for a Raptor Remote internal application.

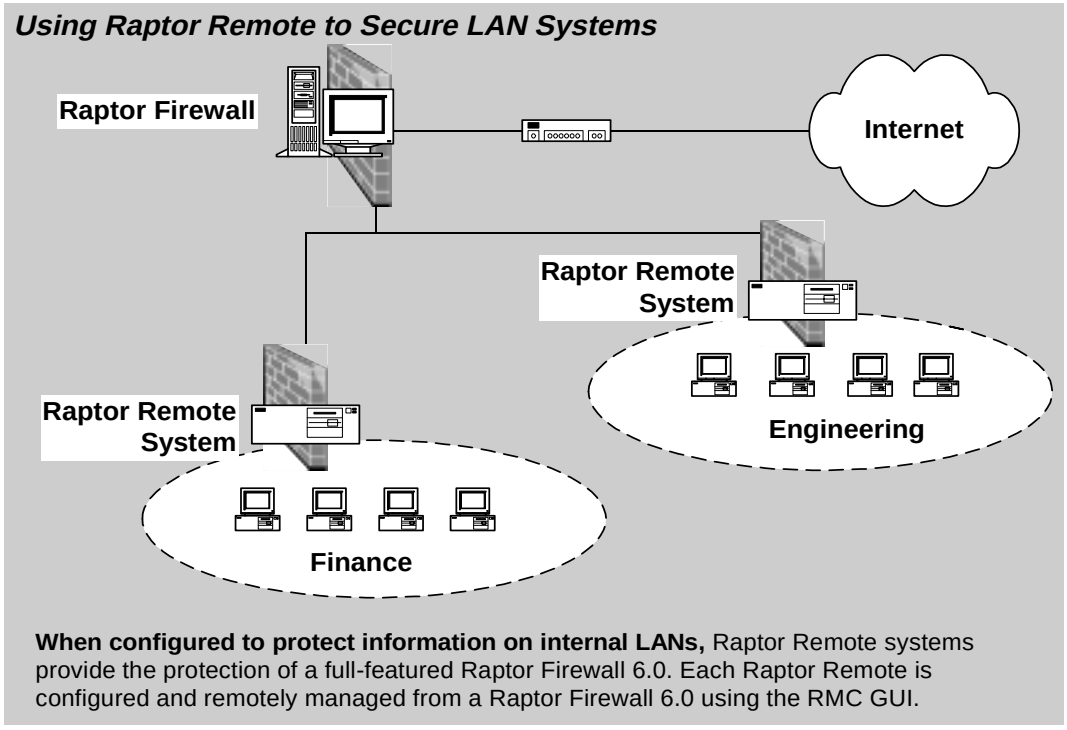


Figure 23. Protecting Your Intranet with Raptor Remote Systems

Protecting Sensitive Intranets

Used as an intranet firewall, the Raptor Remote protects sensitive company information on internal LANs. When configured with appropriate rules, Raptor Remote is extremely effective in isolating information and resources on defined subnets from users in different functional areas. There is no need, for example, for someone outside of the human resources department to access administrative data on other employees or company practices unless authorized to do so. The same is true of materials relating to trade or developmental secrets, financial and medical records, and other areas.

This level of network isolation is as important to a well-planned security policy as a perimeter defense against Internet-based attacks.

Managing an Internal Configuration

As with Internet application, internal Raptor Remote configurations can be remotely managed using the RMC or the RCU management interface. A remote management interface can perform monitoring operations, including killing active connections, examining the logfile, and starting or stopping the firewall. Rules and configuration files are stored locally on the system running the Raptor Remote.

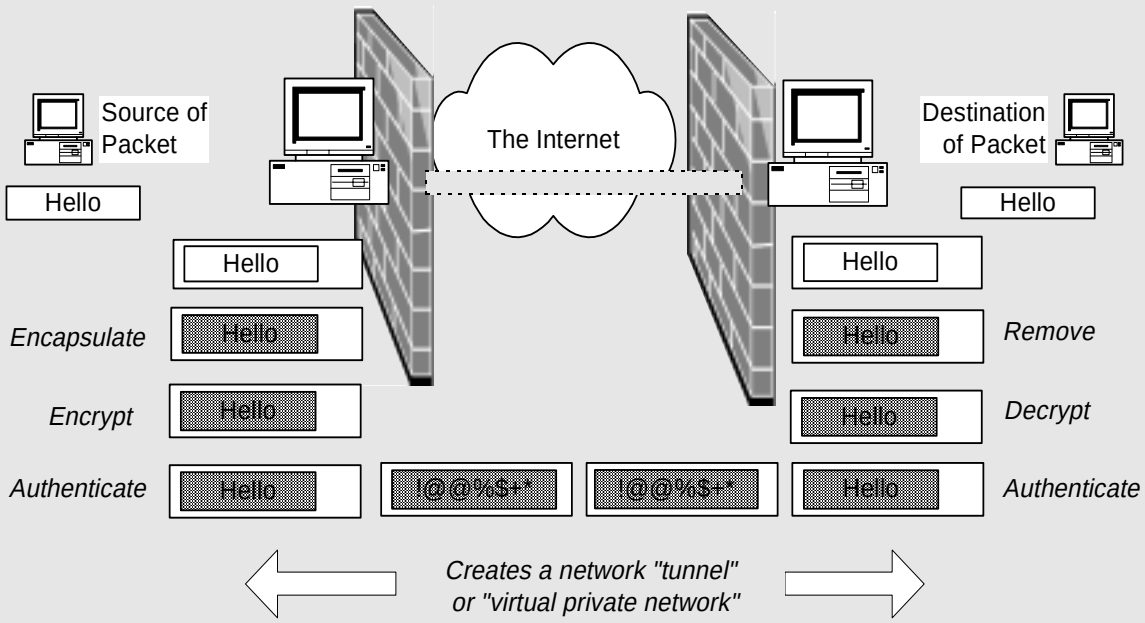
Section 5

Virtual Private Networking

Secure Connections for Remote Users

The Raptor Firewall 6.0 supports industry standard Virtual Private Networking (VPN) as part of its feature set. This feature allows seamless and transparent communication between systems on the Internet while maintaining both the privacy and integrity of the communicated data. VPN does this by creating a secure path, or tunnel, between two network entities. Data transmitted through this tunnel is encrypted and authenticated. Figure 24 is a conceptual view of a VPN session.

Encapsulation and Encryption of VPN Packets



Raptor Firewall VPN Technology makes it easy to build secure tunnels across public networks. All packets passed through the tunnel are encrypted and authenticated by a Raptor Firewall (or other IPsec-compliant security gateway) on either end of the tunnel. To further restrict the flow of traffic within tunnels, the Raptor Firewall supports filtering based on source and direction of tunneled packets, protocol used, and applications

Figure 24. VPN Tunnel Between Cooperating Networks

Use of encryption ensures that privacy is maintained for the data being communicated, and also for vital networking information, such as addresses and port numbers. Use of authentication protects against tampering with data in transit by including a message authentication code which can only be generated by authorized VPN tunnel endpoints. The Raptor Firewall implements these capabilities without requiring modification of software on the systems at either end of the VPN tunnel.

Defining VPN Tunnels

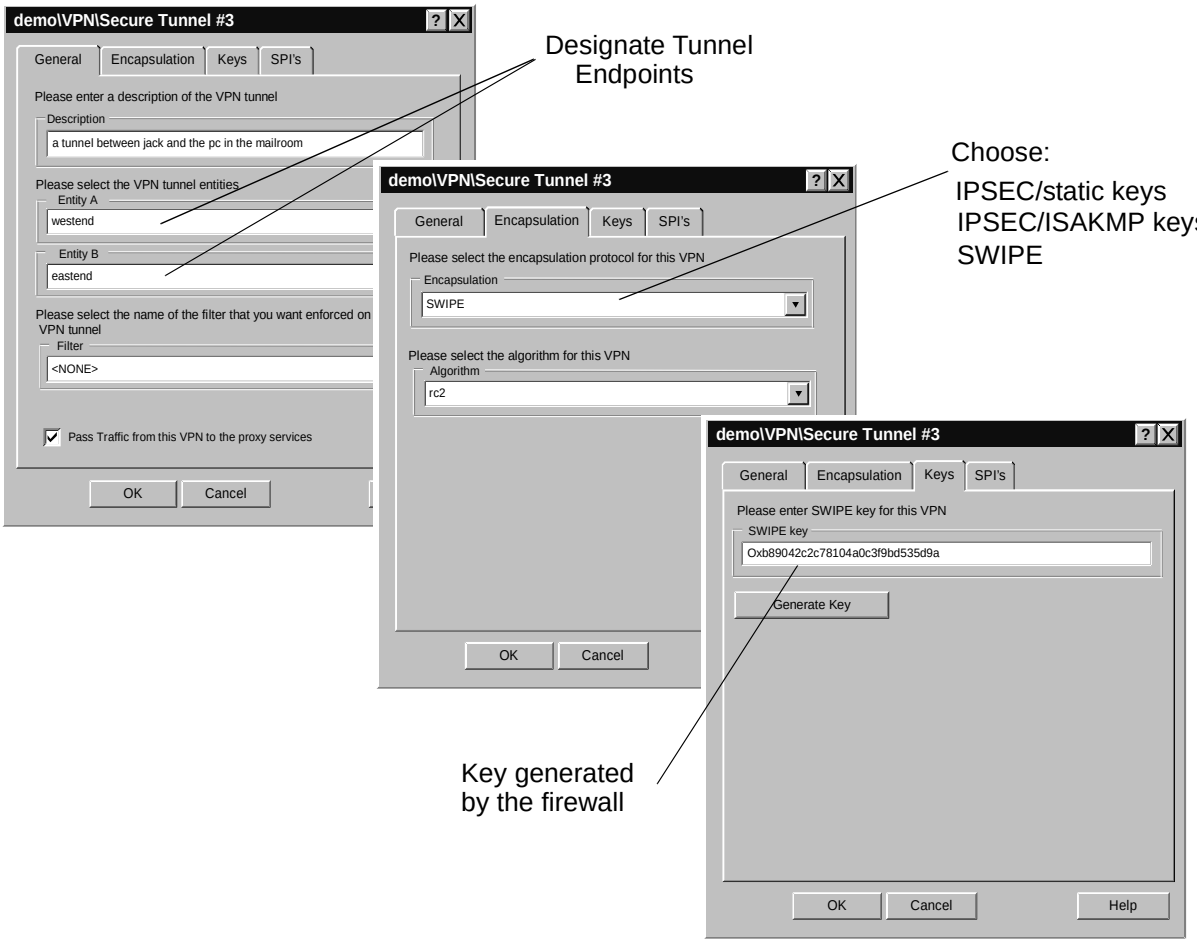
As an example of how VPN works, let's suppose two networks are connected to each other through an insecure network such as the Internet. Each network is protected from unwanted intrusion by a Raptor Firewall 6.0 with standard VPN support. Alternatively, you can set up an IPsec tunnel between a Raptor Firewall and another IPsec-compliant security firewall. The two firewalls are configured with a VPN tunnel defined between two sets of hosts. This tunnel definition includes the following information:

- The type of tunnel used. Both sides must agree on type.
  The Raptor Firewall supports swIPe and IPsec tunnels
- The type of encryption used in the tunnel
  exportable DES, triple DES, and RC2 are supported
- The type of secret key used for authentication and encryption
  This key can be static or dynamic. In either case it is a shared secret, used by receiving systems to forward tunnel traffic. For static tunnels, you must also specify the Security Parameters Index (SPI) and key.
- The type of integrity algorithm used
  The Raptor Firewall 6.0 supports MD5 and SHA 1
- The network entities allowed to use the tunnel on each network
  You can define tunnels between individual hosts or entire subnets

Information used to define each tunnel is shared by the two firewalls controlling each end of the tunnel. You enter this information using the VPN Tunnels Property Page shown in Figure 25. Sharing tunnel information is essential, as it enables each firewall to perform the correct decryption operations, and identify the correct destination system for the VPN packets it receives.

You can also create a VPN tunnel between an Raptor Firewall 6.0 and a remote system running RaptorMobile. This is an excellent option for sales people, telecommuters, and personnel who travel frequently and require secure access to corporate databases or other confidential data. A secure VPN tunnel established over the Internet provides a low-cost means to freely exchange data with corporate systems without worrying about tampering or theft of confidential data.

Defining VPN Tunnels



To Define IPsec or swIPe tunnels enter the appropriate information in the Secure Tunnels/VPN window. You can also add a filter to VPN tunnels to limit the type and direction of traffic allowed to flow between the controlling firewalls.

Figure 25. Defining a VPN Tunnel

How Raptor Firewall Handles Tunneled Data

In Figure 26, a virtual tunnel connects two networks (labeled Finance and Administration) residing between a Raptor Firewall 6.0 Enterprise and a Raptor Remote system, respectively. The VPN application is shipped as a fully integrated part of the Raptor Firewall 6.0 product suite.

All traffic passed between these networks is encrypted by one firewall, and decrypted by the other. In addition, one of the firewalls computes an authentication hash on each VPN packet, to ensure the packet has not been tampered with. The hash is recomputed by the receiving firewall. If the hash matches, this firewall forwards the packets through to their intended destination host.

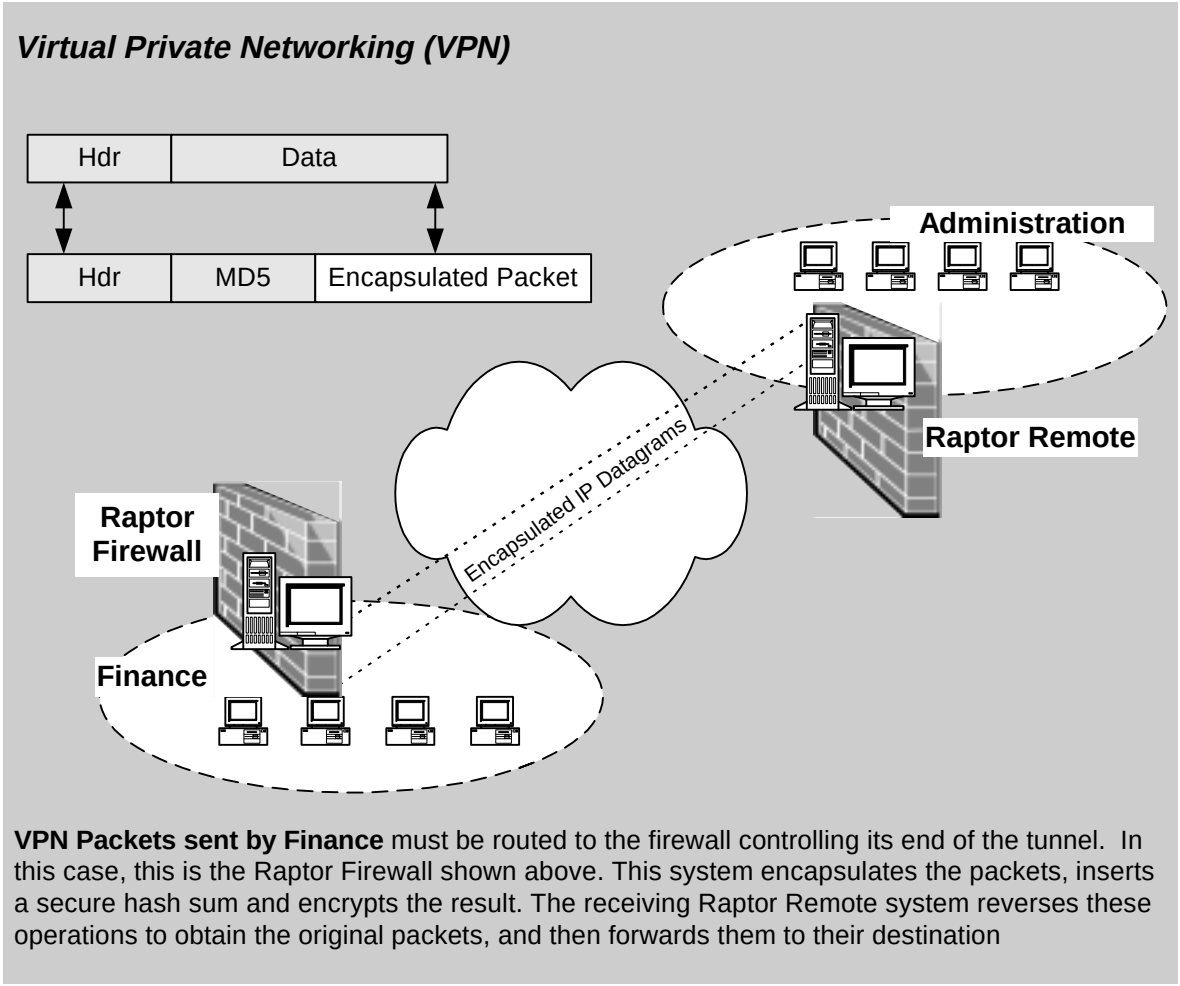


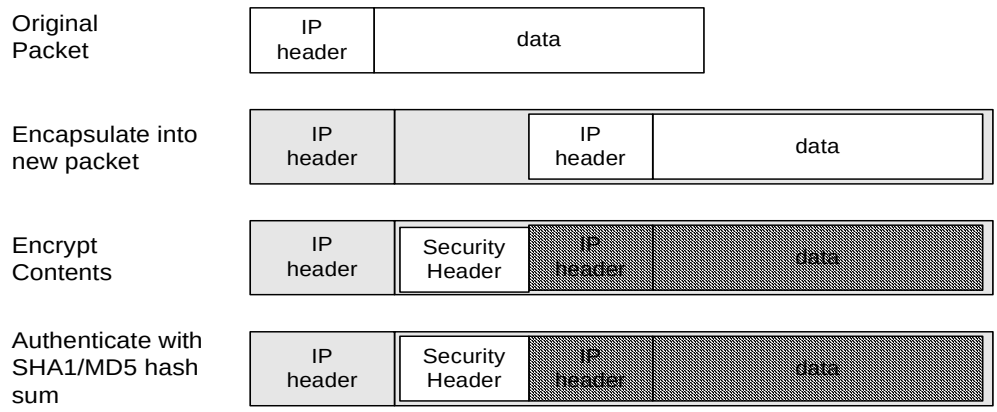
Figure 26. Raptor Firewall Virtual Private Network

VPN in a Nutshell: How it Works

Use of VPN to send data across the internet is transparent to both sending and receiving systems. All interactions necessary to ensure the integrity and privacy of the communication are handled by the cooperating Raptor Firewalls on each side of the VPN tunnel. Here is what happens when a machine on the *Finance* network (see Figure 26) sends data to a machine on the *Administration* LAN at a remote site:

1. Once the packets have been sent, a network router must forward them to the Raptor Firewall protecting the Finance network.
2. Upon receiving these packets, the Raptor's VPN component determines (based on their destination IP address) that they must be forwarded over a VPN tunnel. The Raptor then encrypts the entire packet including all headers, and attaches an authentication hash to the packet. These operations ensure the privacy and the security of the transmitted data.
3. The firewall then *encapsulates the encrypted packet* in an outer IP packet, and sends it to the Raptor Remote firewall protecting Human Resources (see Figure 27). The encapsulation and transmission of one IP packet inside another IP packet is known as *tunneling*.
4. On receiving the packet, the Raptor Remote system at Human Resources *validates* the authentication hash sum to ensure that it was sent by the Raptor Firewall protecting Finance, and *decrypts the packet*.
5. If authentication is successful, the Raptor Remote system strips the outer headers to retrieve the packet that was sent by the originating host in Finance. It then forwards the packet directly to the intended end system in the Human Resources network.
6. The firewall then encapsulates the encrypted packet in an outer IP packet, and sends it to the Raptor Remote firewall protecting Human Resources (see Figure 27). The encapsulation and transmission of one IP packet inside another IP packet is known as tunneling.

Repackaging Packets for Secure Transfer



Securing data for transfer across public networks involves several operations. The Raptor Firewall begins by encapsulating the packet into a new packet. It then inserts an authentication checksum into the packet, and encrypts the packet's contents. The receiving Raptor Firewall performs these operations in reverse to extract the original packet.

Figure 27. Handling of VPN Packets

Support for IPsec and swIPe Tunnels

In addition to the swIPe based technology available in earlier releases, Raptor Firewall 6.0 supports industry standard IPsec based virtual private networking. The following features of IPsec are supported:

- Authentication: None, EMAC (Keyed) MD5, HMAC MD5, HMAC SHA1.
- Encryption: None, DES, Triple DES.

Although administrators can use either swIPe or IPsec based tunnels, IPsec is becoming the preferred choice for the following reasons:

- Because they are standards based, Raptor's IPsec tunnels can interoperate with IPsec-based implementations from other vendors.
- IKE dynamic keying is available for IPsec based tunnels only.
- Customers who need more protection than afforded by 56-bit DES, can use IPsec tunnels with triple DES encryption.

Use of DES, Triple DES, or RC2 for Encryption

VPN supports DES, triple DES (3DES), or RC2 encryption. DES is a block cipher symmetric algorithm that is exportable to most international markets. It uses a 56-bit key to encrypt and decrypt data. Triple DES uses three 56-bit keys and three DES encryption operations. Use of 3DES is limited to the USA and Canada. RC2 supports variable key lengths up to a 40-bit key for encryption, and, as with 56-bit DES, can be exported internationally.

SHA1 and MD5 Secure Checksum for Privacy

VPN uses the SHA1 and MD5 algorithms to generate authentication signatures. The MD5 algorithm (RFC 1321) uses a one-way hash function to produce a non-reversible, unique 128-bit value or a digital signature for authentication. SHA1 produces a 160-bit value. Both mechanisms can generate a truncated 96-bit hash.

IKE Security and Authentication Protocol (ISAKMP)

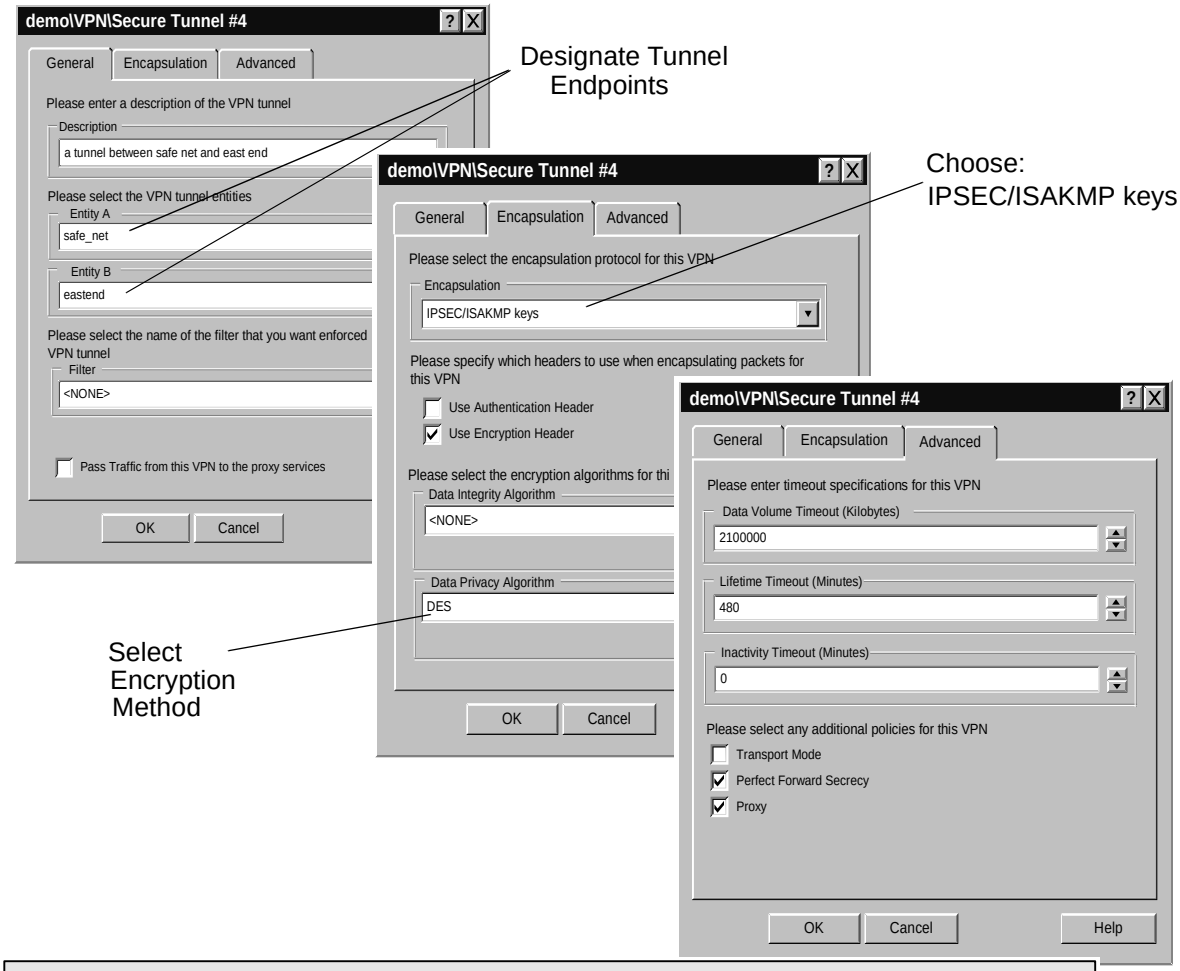
As part of Raptor Firewall 6.0, AXENT supports the IETF IKE protocol for key management and data security. Implementing IKE and IPsec protocols allows for the creation of VPN tunnels, key and transform negotiation, and establishing Security Parameter Indexes (SPI) dynamically. The ability to specify key expiration times is also supported.

By providing support for certificates and digital signatures, IKE also permits strong authentication of the peer machine during key negotiations. Incorporating the IKE standard (formerly called ISAKMP) into the Raptor Firewall 6.0 provides for increased tunnel security and enhanced flexibility in tunnel creation.

With dynamic allocation of resources, the overhead of manual tunnel management is removed from the administrator. In turn, the increased flexibility provided by IKE gives the administrator much greater control over tunnel parameters. In addition, by adopting the IKE standard, AXENT meets the needs of users requiring standards-based solutions.

Figure 28 shows the RMC Property Page for initiating IKE (ISAKMP) support.

Specifying the IKE (ISAKMP) Protocol



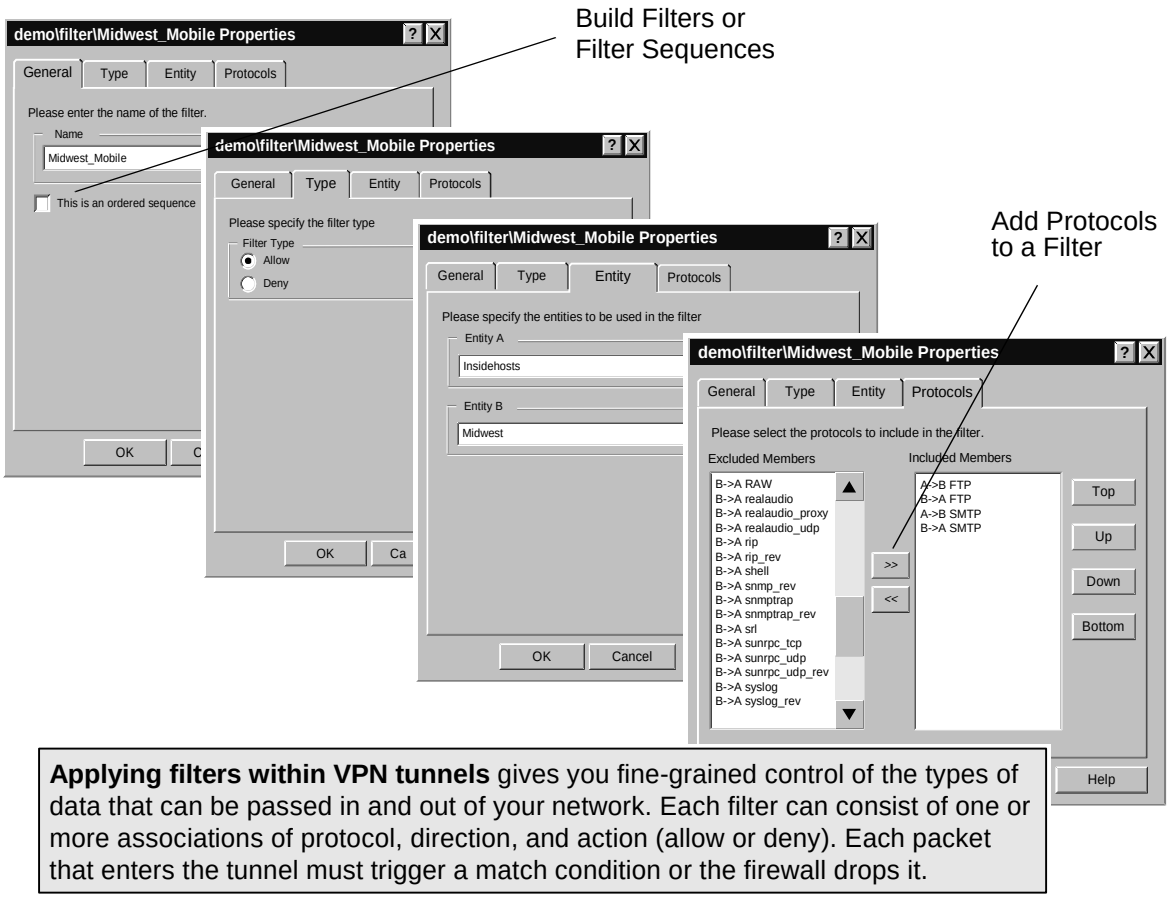
To define ISAKMP as the security and authentication method for tunnels, select ISAKMP at the Encapsulation field. Then, select the encryption method desired at the Data Privacy Algorithm field. Selecting the Advanced tab allows you to fine-tune tunnel settings.

Figure 28. Selecting the ISAKMP Protocol from RMC

Packet Filtering in VPN Tunnels

VPN tunnels provide a secure, private access method for hosts and trusted users. Even so, there may be instances where you wish to limit access to even trusted users. By applying packet filters within secure tunnels, you can assure that they can only be used to pass certain types of data, and in a specific direction only. Figure 29 shows the RMC Property Page for configuring packet filters.

Creating Packet Filters



Applying filters within VPN tunnels gives you fine-grained control of the types of data that can be passed in and out of your network. Each filter can consist of one or more associations of protocol, direction, and action (allow or deny). Each packet that enters the tunnel must trigger a match condition or the firewall drops it.

Figure 29. Use of Filters in VPN Tunnels

How Filters Work

Packet filters are based on the protocols defined with the management interface Protocols display. Each filter comprises an ordered set of match conditions, and an associated action. As packets pass through the tunnel, they are compared with each filter in turn. For instance, you could set up a filter called "mail_ftp_only" that specifies the following protocols and match conditions:

SMTP	A->B	Allow
FTP	A->B	Allow
FTP	B->A	Allow

This filter would allow each of these services in the direction indicated (A to B, or B to A). Packets that match any of the criteria listed are allowed; packets that do not are dropped. The result: a VPN tunnel that carefully limits access, rather than a wide-open IP pipe between designated systems.

Filters and Filter Sets

The Raptor Firewall allows you to group packet filters into ordered lists called filter sets for subsequent attachment to tunnels. As in the case of filters, packets associated with the tunnel are checked against this ordered list until a match is found.

You can attach packet filters or filter sets independently to each VPN tunnel. This gives you a high level of flexibility in setting up and managing tunnel traffic. For example, you can set up a VPN tunnel to a remote site that allows all kinds of traffic to pass through, and other tunnels to RaptorMobile platforms that only allow access to a customer database on a specific machine.

Using Filters to Qualify Tunnel Types

On the tunnel input side, you can use packet filters as part of the tunnel selection criteria to choose a tunnel that is most appropriate to a particular application. For example, you can set up two different tunnels between the same two sites. One of these could be encrypted, to pass sensitive traffic. Another, designed to pass http traffic, could be an authenticate-only tunnel with no encryption.

Use of Local Tunnels

The Raptor Firewall's VPN capability makes it possible to define a *local tunnel* between two network interfaces on the same firewall. The purpose in doing this is to allow IP network traffic types that are not supported by secure proxies to pass through the firewall. Traffic types that you would normally pass through a local tunnel include ICMP, RPCs, NFS, and streaming video and audio. In addition, a user may wish to create a local tunnel to pass trusted data without having to go through the firewall. Doing this can result in a slight increase in throughput.

Because the traffic passed through a local tunnel bypasses the firewall's security proxying mechanisms, AXENT recommends extreme caution when using this feature. Local tunnels should *always and only be used* along with packet filtering. Using appropriate packet filters provides an important measure of security in this case, since it assures that only defined types of traffic, in a specified direction, will be allowed through the local tunnel.

VPN Features

In addition to enhanced support for the IKE key management standard, AXENT continues to track the IPsec standards suite, modifying the firewall as necessary to maintain standards consistency. Further enhanced VPN features are described in the following section.

Proxy Support

The Raptor Firewall supports proxying of VPN traffic. With this enhancement, firewall administrators can specify that VPN traffic be forced up through the application proxies. This allows you to maintain tighter access control and content checking. This feature, which can be applied on a per tunnel basis, is available from the management interface as part of the VPN configuration process. As part of this enhancement, proxy rules have been extended to allow administrators to specify that only encrypted data be permitted at a particular rule.

Proxying allows you to maintain the privacy of a secure tunnel between clients while at the same time limiting access to sensitive areas within your network. Also, proxying provides a means for routing VPN traffic through networks that reside behind the firewall. Forcing VPN traffic up the protocol stack allows you to be precise in setting packet structure and specifying the final destination of tunnel traffic. This is particularly important when the protocol being used requires payload modification. Proxy support within VPN tunnels provides you with all the security features of the Raptor Firewall. These include:

- Ability to monitor tunnel data transfer sessions
- Ability to kill sessions
- Expanded message logging
- User authentication
- Finer control of tunnels through more extensive rules

Support for Network Address Translation

Support for Network Address Translation (NAT) for RaptorMobile clients simplifies the task of setting up routing tables that point to the firewall for RaptorMobile systems. With NAT, the firewall administrator can create an address pool to which RaptorMobile clients can be mapped. This pool of addresses can then be routed back to the Raptor Firewall 6.0.

For protocols that require modification to the data payload, such as command line ftp, administrators must use VPN proxying described above.

Importing User and Tunnel Information

To simplify the administration of multiple RaptorMobiles, the ability to import VPN user and tunnel configuration data is provided with Raptor Firewall 6.0. Data can be stored and used to initialize a VPN tunnel between the firewall and a RaptorMobile client. In addition, administrators can schedule periodic routine updates to this stored data to add, delete, or modify user and tunnel configuration parameters.

Cascaded and Nested Tunnels

The Raptor Firewall 6.0 supports the ability to cascade and nest VPN tunnels. At the management interface, users can cascade separate tunnels that have their endpoint at the Raptor Firewall. For example, a tunnel between Client A and the firewall can be cascaded with (joined to) a tunnel created between Client B and the firewall. This allows traffic from A to traverse both tunnels to reach Client B.

With nesting, a user can create a tunnel within a tunnel to transmit data between clients on different networks.

Configurable Tunnel Usage Limits

The Raptor Firewall 6.0 allows the administrator to set usage limits on VPN tunnels. The parameters *time in use, inactivity,* and *byte count* can be set to limit the life of a tunnel. By setting time and volume limits, an administrator can maintain a high degree of control over the use of tunnels by limiting their lifetime. Also, tunnels using dynamic keying can be forced to rekey on expiration of these configurable tunnel limits.

Summary

With the shift to electronic commerce and extra-nets, businesses and institutions rely heavily on network communications. As the nature of business changes, sensitive data is being moved out of the closed environment of isolated LANs. This data now resides, and must be accessed, on host systems, distributed LANs, and even remote PCs.

Protecting data from unauthorized users requires a *strategy* that encompasses the way business uses computers, and the *tools* to implement that strategy. With Raptor Firewall 6.0 and its enhanced and expanded suite of functions and integrated third-party applications, AXENT Technologies stands at the pinnacle in providing an *integrated, enterprise wide framework* for network security. AXENT's Raptor set of solutions provides secure transactions in each enterprise domain with products that leverage AXENT's application level firewall and virtual private networking technologies.

For More Information

For more details on AXENT'S Raptor unit security products, business relationships, and planned product enhancements visit our web site at http://www.axent.com. You can also reach AXENT Technologies by email at info@axent.com. You can reach the Raptor division by email at info@raptor.com. To reach us by phone, if you are calling from within the USA, dial 1.888.44.AXENT, or dial 1.300.258.5043 if you are calling from outside the USA.

