CheckPoint FireWall-1 White Paper

Version 2.0
September 1995

Table of Contents

Executive Summary

When you connect your local network to the Internet, the single most important measure you can take to prevent break-ins is to define a network security policy and establish a firewall to implement that policy. This document describes how to accurately and simply express such a security policy in FireWall-1 by presenting a number of typical example configurations. These examples and their design rationale will serve as a guide for your own implementation.

In addition, this document describes the architecture and unique characteristics of CheckPoint's FireWall-1 Internet Gateway, and outlines the major characteristics that enable CheckPoint FireWall-1 to establish full, transparent, and true Internet connectivity using the entire range of Internet protocols, while ensuring network security. Encryption, User and Client Authentication and other FireWall-1 features and techniques are described. Finally, performance data are presented.

The powerful combination of CheckPoint FireWall-1's sophisticated Stateful Multi-Layer Inspection (SMLI) technology and its intuitive GUI deliver unmatched security, connectivity and performance.

Internet Firewall Technologies

Overview

When you connect your network to the Internet or to another network, securing your network against intrusion is of critical importance. The most effective way to secure the Internet link is to put a firewall system between the local network and the Internet. The firewall's role is to ensure that all communication between an organization's network and the Internet conforms to the organization's security policies. Additional security measures, such as authentication and privacy enhancements, may follow and complement firewalls, but stopping the fire from spreading into the private network is the first step.

Two methods, usually implemented together, are commonly employed to establish Internet firewalls, the major difference between them being in the flow of communication:

Application gateways

With application and circuit gateways, all packets are addressed to a user-layer application on the gateway that relays the packets between the two communication points. In most application gateway implementations, additional packet filter machines are required to control and screen the traffic between the gateway and the networks. A typical configuration includes two routers, with a bastion host in the middle, to serve as the application gateway.

Application gateways are secure but inefficient. They are nontransparent to users and applications and more important, to the gateway host on which they run, and they are difficult to configure and manage. Only a limited number of applications is supported and special tailoring is required for each one.

Packet Filtering

A packet-filter gateway acts as a router between two networks. As packets flow from source to destination, the gateway either forwards or blocks them. A packet-filter gateway is less secure than an application gateway but more efficient. It is comprehensive and transparent to many protocols and applications. However, traditional packet filters are stateless, have only a low-level protocol understanding, and are difficult to configure and verify. Lack of auditing mechanisms is another major drawback.

CheckPoint FireWall-1 combines the advantages of both methods - with none of their disadvantages - to create an efficient, protocol independent and secure firewall engine. FireWall-1 is capable of application-layer security, user authentication, unified support and handling of all protocols, and auditing and alerting. CheckPoint FireWall-1's operation is also transparent to users and to system setup.

In addition to the inspection technology, CheckPoint FireWall-1 includes an object-oriented graphical user interface that enables simple and flexible system management and configuration.

Application-Layer and Circuit Gateways

For each application relayed, application-layer gateways use specific, special-purpose code. Application gateways can provide a high level of security, though they suffer from a number of deficiencies, and only a limited number (usually only a small basic subset) of applications and services are supported. In order to use application gateways, users must first connect to the gateway machine or install a specific client application for each application they expect to use. Each gatewayed application is a separate, proprietary piece of software and requires its own set of management tools and permissions.

Circuit gateways provide a more general way to implement application gateways. They support some TCP applications, but not all. Circuit gateways do not support other protocols. Users are often forced to install and use different client applications or to change their work habits. Installing client applications on each internal

computer is likely to be a cumbersome task since the internal network is typically heterogeneous with respect to platforms, operating systems, versions, etc.

Network performance is also affected by both application and circuit gateways; each packet must be copied and processed at least twice by all the communication layers, and user-layer processing and context switching is required. Moreover, a new process must be started for each connection.

The application gateway computer itself (bastion-host or dual-homed gateway) remains exposed to the network, and additional means, such as packet-filtering, must be implemented to protect it. Applications and daemons must be carefully managed, since even with router packet filters, they are still vulnerable at the non-privileged ports. These protective measures typically entail acquiring additional hardware, limiting available services, as well as tedious and error-prone administrative overhead.

Packet–Filtering Gateways

Packet-filtering technologies provide an efficient and general way to control any type of network traffic and applications. They require no changes in client applications, no specific application management or installation, and no additional hardware. Using a single, unified packet-filtering engine, all network traffic is processed and then either forwarded or blocked from a single point of control.

However, packet filtering technologies do not address all security requirements. The information available for filtering (for example, source and destination addresses and port numbers) is rarely sufficient. The number of rules is limited, and there is a high performance penalty when many rule instances are used. Lack of context or state information makes it impossible to use packet filters for datagram-based protocols like UDP (User Datagram Protocol), RPC (Remote Procedure Call), or even FTP (File Transfer Protocol - a commonly used and surprisingly complex TCP based service). In most cases, packet-filtering technologies provide no auditing or alerting mechanisms.

Previous packet-filtering technologies also suffer from poor management interfaces. Implementing them requires a high level of understanding of communication internals and writing low-level bit and byte code, so that these technologies are difficult to change and adapt. Some packet filters are implemented inside routers, thus limiting computing power and filtering capabilities and providing no auditing or reporting capabilities. Others are implemented as software packages that filter the packets in application-layer processes, an inefficient approach that requires multiple data copies, expensive delays and context switches, and delivers lower throughput.

CheckPoint FireWall-1

Most existing Internet firewalls use a combination of a packet-filter screening computer or a hardware-router for controlling the lower layers of communication, and application gateways for the enabled applications. This configuration provides only limited, non-transparent and nonflexible connectivity, and entails high costs in setup, management, and expertise.

In contrast, FireWall-1 combines the advantages of both concepts:

Packet Filtering

An efficient inspection module - applicable to all protocols - with Stateful Multi-Layer Inspection (SMLI) technology, understands data in the packet intended for all other layers, from the network layer (IP headers) up to the application layer, and provides stateful context.

In this way, FireWall-1 secures complex applications more effectively than technologies that have only data in some of the layers available to them. For example, while application gateways have access only to the application layer and routers have access only to to the lower layers, FireWall-1 integrates the information gathered from all layers into a single comprehensive inspection point.

At the same time, FireWall-1's SMLI technology provides transparent and efficient security to all protocols and applications.

Application Gateways

FireWall-1 provides secure application gateways (proxies) that add real value, for example, encryption and user authentication. There is no need for hardware routers or cumbersome system administration on the gateway.

FireWall-1 provides logging and alerting mechanisms, as well as simple installation and setup procedures.

FireWall-1's single integrated security solution provides enterprise-wide security - though the security policy can be enforced by any number of firewalls and any number of authenticated users can be controlled, there is still only one security policy, one Rule Base, and one centralized log. In addition to a single integrated security policy, the system administrator can, if required, maintain different Rule Bases to be implemented, for example, at different times of day.

An intuitive, objectoriented user interface enables easy, flexible, and uniform implementation of an organization's global security policy.

The following sections of this document demonstrate how to implement a security policy in FireWall-1, and explain the principles of FireWall-1's operation.

Configuring FireWall-1

This section demonstrates how to build a FireWall-1 Rule Base to implement a security policy for a simple network configuration using a "diode" policy, and then for a larger network using a more detailed configuration.

A Simple Configuration

The following example illustrates how to deploy FireWall-1 in the network configuration shown in the diagram below. Note that this is not a recommended configuration, but simply an example for the purposes of this document.

In this example, FireWall-1 will be installed on the gateway computer (named "monk" in the rules that follow).

A Typical Security Policy

For the configuration shown above, a typical security policy might be this:

external networks may only send mail to local computers

local computers may access the entire network: localnet and Internet

This policy protects the private network from non-local networks, but puts no restrictions on local computers.

You will begin by considering how this security policy can be implemented in FireWall-1. Next, you will see how this security policy can be "tightened up" so that the potential loopholes are blocked.

Implementing a Security Policy

In order to implement a security policy, you must perform the following actions:

Define the network objects used in the Rule Base.

You do not have to define the entire network to FireWall-1 - only those objects that are used in the Rule Base. For the configuration described here, you must define the gateway (monk), the mail server (mailsrvr) and the local network (localnet).

Define services used in your security policy (optional).

You do not have to define the commonly used services. These are already defined for you in FireWall-1. You should define only those special services that are part of your security policy.

Defining network objects and services is very straightforward. In most cases, you need only specify a name, because FireWall-1 can obtain the object's properties from the appropriate /etc, NIS (yp), and DNS databases.

Define the Rule Base - the rules for accepting, rejecting and logging packets.

Install the Rule Base - install the Inspection Code on the gateways.

The next step in implementing a security policy is to define the Rule Base, using the Rule Base editor. In the Rule Base editor, rules are expressed in terms of the following elements:

The first three elements describe the communication attempt.

Source - where the packet is coming from

Destination - where the packet is going

Service - the type of application

Note: If you specify "Any" under services, all TCP, UDP and RPC based applications, even those not defined in the Service Manager, are included. Even if you use an undefined database application, FireWall-1 will secure the outbound connection and ensure that replies are passed through but nothing else.

The next two elements indicate what is to be done.

Action - what is to be done with the given communication attempt

Track - whether to log the packet or to generate an alert

The last element indicates which firewall module will enforce the rule defined by the first five elements.

Install On - the firewall module that will enforce this rule

In this simple example, there are only two rules, corresponding to the policy given above.

The first rule (non-local networks may only send mail to the mail server) can be expressed in the Rule Base editor as follows:

Source	Destination	Services	Action	Track	Install On
Any	mailsrvr	smtp	Accept	Short Log	Gateways

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.