McAfee Network Security Platform

(DoS Prevention Techniques)
COPYRIGHT
Copyright © 2018 McAfee, LLC

TRADEMARK ATTRIBUTIONS
McAfee and the McAfee logo, McAfee Active Protection, ePolicy Orchestrator, McAfee ePO, McAfee EMM, Foundstone, McAfee LiveSafe, McAfee QuickClean, Safe Eyes, McAfee SECURE, SecureOS, McAfee Shredder, SiteAdvisor, McAfee Stinger, True Key, TrustedSource, VirusScan are trademarks or registered trademarks of McAfee, LLC or its subsidiaries in the US and other countries. Other marks and brands may be claimed as the property of others.

LICENSE INFORMATION
License Agreement
NOTICE TO ALL USERS: CAREFULLY READ THE APPROPRIATE LEGAL AGREEMENT CORRESPONDING TO THE LICENSE YOU PURCHASED, WHICH SETS FORTH THE GENERAL TERMS AND CONDITIONS FOR THE USE OF THE LICENSED SOFTWARE. IF YOU DO NOT KNOW WHICH TYPE OF LICENSE YOU HAVE ACQUIRED, PLEASE CONSULT THE SALES AND OTHER RELATED LICENSE GRANT OR PURCHASE ORDER DOCUMENTS THAT ACCOMPANY YOUR SOFTWARE PACKAGING OR THAT YOU HAVE RECEIVED SEPARATELY AS PART OF THE PURCHASE (AS A BOOKLET, A FILE ON THE PRODUCT CD, OR A FILE AVAILABLE ON THE WEBSITE FROM WHICH YOU DOWNLOADED THE SOFTWARE PACKAGE). IF YOU DO NOT AGREE TO ALL OF THE TERMS SET FORTH IN THE AGREEMENT, DO NOT INSTALL THE SOFTWARE. IF APPLICABLE, YOU MAY RETURN THE PRODUCT TO MCAFEE OR THE PLACE OF PURCHASE FOR A FULL REFUND.
Contents

Preface ........................................................................................................................................... 5
About this guide ................................................................. 5
Audience ................................................................. 5
Conventions ................................................................. 5
What's in this guide ................................................................. 6
Find product documentation ................................................................. 6

1 Overview .................................................................................................................................... 7
What is a Denial of Service Attack? ................................................................. 7
What Is a Distributed Denial of Service Attack? ................................................................. 7

2 Types of DoS/DDoS Attacks ........................................................................................................ 9

3 Detection techniques using Network Security Platform .................................................................. 11
Volume-based DoS Attack Detection ............................................................................................. 11
Threshold-based Attack Detection ................................................................................................. 11
Exploit-based DoS Attack Detection ............................................................................................. 13

Mitigation techniques for DoS and DDoS ........................................................................................ 15
Statistical Anomaly ......................................................................................................................... 15
SYN Cookie ................................................................................................................................... 21
Stateful TCP Engine ....................................................................................................................... 22
DNS Protect .................................................................................................................................... 23
Rate Limiting .................................................................................................................................... 24
ACLs ............................................................................................................................................... 24
Connection Limiting with GTI Integration enabled ......................................................................... 25
Custom Reconnaissance Attack Definition .................................................................................... 26
Web Server - Denial-of-Service Protection ...................................................................................... 26
Selecting specific DoS attacks for blocking in the Manager ................................................................. 27

Index ............................................................................................................................................... 29
Preface

Contents
- About this guide
- Find product documentation

About this guide
This information describes the guide's target audience, the typographical conventions and icons used in this guide, and how the guide is organized.

Audience
McAfee documentation is carefully researched and written for the target audience. The information in this guide is intended primarily for:

- **Administrators** — People who implement and enforce the company's security program.
- **Users** — People who use the computer where the software is running and can access some or all of its features.

Conventions
This guide uses the following typographical conventions and icons.

- **Book title or Emphasis** — Title of a book, chapter, or topic; introduction of a new term; emphasis.
- **Bold** — Text that is strongly emphasized.
- **User input or Path** — Commands and other text that the user types; the path of a folder or program.
- **Code** — A code sample.
- **User interface** — Words in the user interface including options, menus, buttons, and dialog boxes.
- **Hypertext blue** — A live link to a topic or to a website.
- **Note** — Additional information, like an alternate method of accessing an option.
- **Tip** — Suggestions and recommendations.
- **Important/Caution** — Valuable advice to protect your computer system, software installation, network, business, or data.
- **Warning** — Critical advice to prevent bodily harm when using a hardware product.
What's in this guide

Find product documentation

McAfee provides the information you need during each phase of product implementation, from installation to daily use and troubleshooting. After a product is released, information about the product is entered into the McAfee online KnowledgeBase.

Task
2. Under Self Service, access the type of information you need:

<table>
<thead>
<tr>
<th>To access...</th>
<th>Do this...</th>
</tr>
</thead>
</table>
| User documentation    | 1. Click Product Documentation.  
                          | 2. Select a product, then select a version.  
| KnowledgeBase         | • Click Search the KnowledgeBase for answers to your product questions.  
                          | • Click Browse the KnowledgeBase for articles listed by product and version. |
Overview

This document caters to McAfee Network Security Platform users in the following categories:

- You want to have an overview of the types of Denial-of-Service (DoS)/Distributed Denial-of-Service (DDoS) attacks that Network Security Platform can detect
- You would like to know the response action(s) that can be taken against each type of DoS/DDoS attack

Contents

- What is a Denial of Service Attack?
- What Is a Distributed Denial of Service Attack?

What is a Denial of Service Attack?

A Denial-of-Service (DoS) attack is a malicious attempt to render a service, system, or network unusable by its legitimate users. Unlike most other hacks, a Denial of Service (DoS) does not require the attacker to gain access or entry into the targeted server. The primary goal of a DoS attack is instead to deny legitimate users access to the service provided by that server.

Attackers achieve their DoS objective by flooding the target until it crashes, becomes unreachable from the outside network, or can no longer handle legitimate traffic. The actual volume of the attack traffic involved depends on the type of attack traffic payload used. With crafted payload such as malformed IP address fragments, several such packets might be sufficient to crash a vulnerable TCP/IP stack; on the other hand, it might take a large volume of perfectly conforming IP address fragments to overwhelm the defragmentation processing in the same TCP/IP stack. Sophisticated attackers might choose to use a mixture of normal and malformed payloads for a DoS attack. DoS attacks can vary in impact from consuming the bandwidth of an entire network, to preventing service use of a single targeted host, or crashing of a single service on the target host.

Most DoS attacks are flood attacks; that is, attacks aimed at flooding a network with TCP connection packets that are normally legitimate, but consume network bandwidth when sent in heavy volume. The headers of malicious packets are typically forged, or spoofed, to fool the victim into accepting the packets as if they are originating from a trusted source.

What Is a Distributed Denial of Service Attack?

A Distributed Denial-of-Service (DDoS) is a type of DoS attack that is launched from compromised hosts distributed within or across networks. A DDoS attack is coordinated across many systems all controlled by a single attacker, known as a master. Prior to the attack, the master compromises a large number of hosts, typically without their owners’ knowledge, and installed software that will later enable the coordinated attack. These compromised hosts, called zombies (also called daemons, agents, slaves, or bots), are then used to perform the actual attack. When the master is ready to launch the attack, every available zombie is contacted and instructed to attack a single victim. The master is not a part of the attack, thus tracing the true origin of a
DDoS attack is very difficult. As with a DoS attack, packets sent from each zombie may be spoofed to fool the victim into accepting data from the trusted source. DDoS allows the attackers to utilize the network to multiplex low-volume sources into a high-volume stream in order to overwhelm the targets. Through the master-zombie communications, the real attackers can potentially hide their identities behind the zombies.
Types of DoS/DDoS Attacks

DoS/DDoS attacks can be classified differently depending on how you view them. On the basis of how they operate, you can broadly classify them as volume-based or exploit-based attacks. On the basis of what they target, you can classify them as infrastructure or application attacks. You also have a third category of attacks which are DoS attacks that developed using DoS toolkits, which give you the ability to target both infrastructure as well as applications.

We will need to understand volume-based and exploit-based attacks to understand how Network Security Platform protects against DoS/DDoS attacks in general. However, further in the document you will find categorization based on infrastructure-based and application-based attacks and attacks that are developed using DoS/DDoS toolkits.
Types of DoS/DDoS Attacks
McAfee Network Security Platform provides an integrated hardware and software solution, which delivers comprehensive protection from known, first strike (unknown), DoS, and DDoS attacks from several hundred Mbps to multi-gigabit speeds.


With threshold-based detection, you, as a network administrator, can configure data traffic limits to ensure your servers will not become unavailable due to overload. These thresholds are selected based on coverage of different DDoS attacks and on the availability of statistics that will help the users to configure them. Meanwhile, self-learning methodologies enable Network Security Platform to study the patterns of network usage and traffic over time; thus understanding the wide variety of lawful, though unusual, usage patterns that may occur during legitimate network operations. The learning algorithm takes into account sudden bursts that is common in all network traffic, and differentiates it from the real onset of DDoS traffic. In addition to learning the intensity behavior, it also learns the correlational behavior of different types of packets, which reliably captures TCP/IP protocol behavior, route configuration, and so on. Highly accurate DoS detection techniques are essential because popular Web sites and networks do experience legitimate and sometimes unexpected traffic surges during external events, or for a particularly compelling new program, service, or application.

The combination of these two techniques yields the highest accuracy of detection for the full spectrum of DoS and DDoS attacks, when hundreds or even thousands of hosts are co-opted by a malicious programmer to strike against a single victim.

Contents

- Volume-based DoS Attack Detection
- Exploit-based DoS Attack Detection

**Volume-based DoS Attack Detection**

Network Security Platform detects volume-based DoS attacks through threshold-based and statistical anomaly-based methods. Often a combination of these two methods is used.

**Threshold-based Attack Detection**

The threshold method involves specifying the count and interval thresholds while configuring a DoS policy in the Network Security Manager. When the threshold is crossed, the DoS attack is detected.
Threshold value and interval can be customized in the Network Security Manager for threshold attacks, such as:

- Inbound Link Utilization (Bytes/Sec) Too High
- Too Many Inbound Rejected TCP Packets
- Too Many Inbound ICMP Packets
- Too Many Inbound TCP Connections
- Too Many Inbound IP Fragments
- Too Many Inbound TCP SYN
- Too Many Inbound Large ICMP packets
- Too Many Inbound UDP Packets
- Too Many Inbound Large UDP packets

### Learning-based [Statistical Anomaly-based] Attack Detection

A new Sensor runs for its first 48 hours in learning mode. After 48 hours are complete, the Sensor automatically changes to detection mode, having established a baseline of the "normal" traffic pattern for the network, or a long-term profile. The assumption is that no DoS attack takes place during those first 48 hours.

After moving to detection mode, the Sensor continues to gather statistical data and update its long-term profile. In this way, the long-term profile evolves with the network.

The Sensor also builds short time profiles with a time window of a few minutes.

Learning mode profiles can be customized at the Sensor level. Sub-interfaces and individual CIDR hosts within a VLAN tag or CIDR block can be created and protected against DoS attacks with specific learning mode settings. This is useful in preventing a server in your DMZ or other location from being shut down by a DoS attack. A separate profile is created for each resource.

The Statistical method uses statistical data gathered over a time window to create normal short-term and long-term profiles. DoS attacks are detected when there are anomalies between the traffic pattern in normal profiles and network traffic. Statistical anomalies in traffic are monitored by the Sensor with reference to learned data on normal traffic.

After moving to detection mode, the Sensor continues to gather statistical data and update its long-term profile. In this way, the long-term profile evolves with the network.

The Sensor uses the following checks and counter checks to ensure accuracy of detection:

- Counter profile contamination
- Source IP classification. See section
- Statistical Anomaly.

### Countering Profile Contamination

The goal behind the long-term profile is to define normal traffic levels. The Sensor can identify anomalous spikes in traffic with reference to the defined normal levels.

The Sensor also uses the gathered statistical data to calculate short-term profiles (statistical data averaged over a time window of a few minutes).

If a short-term profile that includes DoS attack data is used to update the long-term profile, it contaminates the long-term profile.
Network Security Platform uses the following countermeasures to help prevent contamination:

- When in detection mode, the Sensor temporarily ceases updating the long-term profile if too many statistical anomalies are seen over a short period.

- The Sensor uses percentile measure. A few large spikes in the short-term data will probably upset a simple average, but are less likely to affect a percentile measure. For example, imagine a group of four students taking an exam with percentile measure ranges of 0-29, 30-49, 50-69 and 70-100 for judging the effectiveness of the exam. Let us say three of the students receive grades of 95%, 93%, and 92% and the fourth receives a grade of 0%. The average score is only 70% but three of the four students are still in the 70-100 range. The teacher can therefore use the percentile ranges as a valid measure for judging the effectiveness of the exam.

Exploit-based DoS Attack Detection

Exploit attack are manifested in attack signatures, which the Network Security Platform uses to detect specific exploit attacks.

A signature is a profile of an attack. Detection of specific attacks is possible through signatures. Network Security Platform also uses exploit signatures for DoS attacks that are not caused by traditional means such as volume overload. For example, the HTTP: Microsoft IIS...SLASH... DenialofService exploit identifies a single request that prevents older IIS servers from responding to clients until they are restarted.

The Sensor uses signatures to perform different levels of traffic processing and analysis. Network Security Platform signatures operate on a framework of Flows, Protocol parsing and Packet searches to detect vulnerability based DoS attacks and attacks using DDoS attack tools.

For example, Network Security Platforms detection mechanisms enable a signature to identify every HTTP traffic flow, every HTTP traffic flow using the GET mechanism, every HTTP traffic flow using GET with “/cgi-bin/calendar.pl” as the path and even every GET with that path and a parameter named “month” with a value of “February”.

Network Security Platform supports the aggregation of multiple signatures into every attack. Each signature within an attack can be more or less specific to identify everything from generic network activity that affects a given platform in a particular way to a specific piece of code that has very specific and identifiable effects. Based on their specificity and severity, signatures are assigned different confidence and severity values.

When a network event occurs that matches an existing Network Security Platform attack, several signatures.
Detection techniques using Network Security Platform
Exploit-based DoS Attack Detection

Once DoS/DDoS attacks have been detected, Network Security Platform offers several methods to block various types of DoS Attacks.

Mitigation techniques for DoS and DDoS

Network Security Platform uses specific methods to prevent DoS attacks. These methods work independently but can also be applied in combination. To effectively mitigate DoS and DDoS use one of the following methods depending on the attack type:

<table>
<thead>
<tr>
<th>Attack type</th>
<th>Description</th>
<th>Mitigation techniques</th>
<th>Attack signature to be enabled (Only for IPS Policy)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Infrastructure attacks: Stateless protocol attacks</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>UDP floods</strong></td>
<td>Sending a flood of UDP attacks to a targeted system constitutes a UDP Flood attack. When communication is established between two UDP services, a UDP Flood attack is initiated by sending a large number of UDP packets to random ports of the targeted system. The targeted system is forced into sending many &quot;Destination Unreachable&quot; ICMP packets, thus consuming its resources and leading to DoS. As UDP does not require any connection setup procedure to transfer data, anyone with network connectivity can launch an attack; no account access is needed. Another example of UDP flood is connecting a host's &quot;chargen&quot; service to the &quot;echo&quot; service on the same or another machine. All affected machines may be effectively taken out of service because of the excessively high number of packets produced. In addition, if two or more hosts are so engaged, the intervening network may also become congested and deny service to all hosts whose traffic traverse that network.</td>
<td><strong>IPS Policy</strong></td>
<td>• DoS: UDP Land Attack</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DoS: NewTear Attack</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Inbound UDP Packet Volume Too High / Outbound UDP Packet Volume Too High</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td><strong>Connection limiting policy with McAfee GTI integration enabled</strong></td>
</tr>
<tr>
<td>Attack type</td>
<td>Description</td>
<td>Mitigation techniques</td>
<td>Attack signature to be enabled (Only for IPS Policy)</td>
</tr>
<tr>
<td>-------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------</td>
<td>------------------------------------------------------</td>
</tr>
<tr>
<td>DNS floods</td>
<td>A flood of DNS requests is sent to a server. This constitutes a DNS Flood attack. Since DNS uses UDP, no hand-shake process is involved. The difference between a DNS Flood and a UDP Flood is that a DNS Flood is directed at port 53. A flood of DNS requests can tie down the resources of DNS infrastructure and creates a DoS condition.</td>
<td>IPS Policy</td>
<td>• DNS: Ethereal Name Expansion DoS Overflow</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DNS: MailEnable SMTP Service SPF Lookup</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Remote DoS Vulnerability</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DNS: Ethereal Endless Decompression DoS</td>
</tr>
<tr>
<td>NTP amplification attacks</td>
<td>In the most basic instance, an attacker sends a get monlist query to an NTP server. The requesting IP address is spoofed to contain an IP address belonging to a target server. The NTP server responds to the request by directing its response to the target IP address. Since the response is considerably larger than the request, the amount of traffic directed at the target server is amplified and ultimately leads to a DoS condition.</td>
<td>IPS Policy</td>
<td>NTP: NTPd Reserved Mode DoS</td>
</tr>
<tr>
<td>ICMP floods</td>
<td>ICMP Echo Request or Reply packets flood a network, creating a DoS condition. A flood of Echo requests to a target system engages the system to respond to the requests. If there is a flood of Reply packets, then it is likely that the remote attacker has forged an IP address from within your network and is sending ICMP Echo Request packets to another network. That network replies to the address in the requests, thus starting a request/reply flood between the two networks.</td>
<td>IPS Policy</td>
<td>• Inbound ICMP Packet Volume Too High /</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Outbound ICMP Packet Volume Too High</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• ICMPv6: Destination Unreachable DoS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DoS: ICMP-Based Jolt2 Attack</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DoS: Jolt Attack</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DoS: Ping-of-Death Attack</td>
</tr>
<tr>
<td>IGMP floods</td>
<td>IGMP floods although uncommon in modern DDoS attacks use protocol 2 with limited message variations. This type of flood can potentially consume large amounts of network bandwidth.</td>
<td>IPS Policy</td>
<td>IGMP: Microsoft IGMP DoS</td>
</tr>
<tr>
<td>Infrastructure attacks: Out-of-state TCP attacks</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYN floods</td>
<td>A large number of spoofed TCP SYN packets are sent to a victim to consume its TCP memory.</td>
<td>SYN cookies</td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IPS Policy</td>
<td>Inbound TCP SYN or FIN Volume Too High /</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Outbound TCP SYN or FIN Volume Too High</td>
</tr>
<tr>
<td>Attack type</td>
<td>Description</td>
<td>Mitigation techniques</td>
<td>Attack signature to be enabled (Only for IPS Policy)</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>TCP full-connect</td>
<td>An attacker compromises several endpoints across the network. Each of these compromised endpoints is known as a slave or a zombie. A remote computer instructing each of these zombies is called the Master. At the time of an attack, the Master instructs every zombie to establish TCP connections that appear to be legitimate since they originate from a real computer with an IP address, with a targeted server until that server’s resources are exhausted. The resultant action is a denial-of-service condition to legitimate TCP requests.</td>
<td>IPS Policy</td>
<td>• TCP: 3-Way Handshake PAWS Fail DoS • DoS: Sensor Management Port • DCERPC: Microsoft Windows RPCSS Memory Leak DoS • DCERPC: Malformed Request DoS • DoS: Axent Raptor Crash</td>
</tr>
<tr>
<td>ACK / FIN floods</td>
<td>A TCP ACK/FIN attack takes place when the attacker sends a large volume of TCP ACK/FIN packets intentionally to the target host. This consumes bandwidth and creates a DoS condition.</td>
<td>Stateful TCP engine</td>
<td>NA</td>
</tr>
<tr>
<td>TCP RST floods</td>
<td>In a stream of TCP packets in a connection, every packet contains a TCP header. Each header consists of a reset (RST) flag, which can be set to 0 or 1. When set to 0, the recipient takes no action. However, when set to 1, the recipient interprets this as a signal to immediately stop using the TCP connection. In a TCP RST Flood attack, an attacker, monitoring traffic between two computers, sends crafted TCP RST headers to one or both computers participating in the connection. The computers assume that the connection must be closed and end the connection. When this action is constantly simulated, it results in a DoS condition. For the attacker to succeed, every parameter of the TCP RST headers must be set to convincing values in order to appear coming from an endpoint.</td>
<td>Stateful TCP engine</td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IPS Policy</td>
<td>• TCP: Inbound TCP RST Volume Too High / TCP: Outbound TCP RST Volume Too High • TCP: RST Socket Exhaustion DoS • DoS: TCP RST BGP Denial of Service • ICMP: Destination Unreachable DOS • ICMP: Destination Unreachable DOS II • DoS: TCP RST BGP Denial of Service</td>
</tr>
</tbody>
</table>

**Application attacks: Full-Connect HTTP flood**
<table>
<thead>
<tr>
<th>Attack type</th>
<th>Description</th>
<th>Mitigation techniques</th>
<th>Attack signature to be enabled (Only for IPS Policy)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Low-Orbit Ion Cannon (LOIC)</strong></td>
<td>Low Orbit Ion Cannon (LOIC) is an open source network stress testing and denial-of-service attack application available in the public domain, written in C#. LOIC performs a DoS or, when used by multiple individuals, a DDoS attack by flooding a network with TCP or UDP packets. LOIC inspired the creation of a JavaScript-based variant of the tool called JS LOIC.</td>
<td>Web Server Protection against Denial-of-Service attacks</td>
<td>NA</td>
</tr>
<tr>
<td><strong>R U Dead Yet (RUDY)</strong></td>
<td>R U Dead Yet (R.U.D.Y or RUDY) is a slow-rate HTTP POST DoS tool. It creates a DoS condition by using long form field submissions. By injecting one byte of information into an application POST field at a time and waiting, RUDY causes an application to wait for never-ending posts to process. Web servers need to support POSTs to accommodate users with slower connections. RUDY causes the target web server to hang while it waits for the rest of the POST request. Simultaneously, several similar connections are initiated to the target web server, thereby exhausting the connection table of the server.</td>
<td>Web Server Protection against Denial-of-Service attacks</td>
<td>NA</td>
</tr>
<tr>
<td><strong>High-Orbit Ion Cannon (HOIC)</strong></td>
<td>The High Orbit Ion Cannon (HOIC) DoS tool is considered the next generation replacement for LOIC. This attack is able to target up to 256 URLs simultaneously and randomize signatures to avoid detection. Used in isolation, HOIC is limited because it require a coordinated group attack to bring down a site. However, when used in conjunction with booster files, customizable scripts that randomize attack signatures, the attack traffic is hard to distinguish from legitimate traffic.</td>
<td>Web Server Protection against Denial-of-Service attacks</td>
<td>NA</td>
</tr>
<tr>
<td><strong>Application attacks: HTTP Slow Attacks</strong></td>
<td></td>
<td>Web Server Protection against Denial-of-Service attacks</td>
<td>NA</td>
</tr>
<tr>
<td><strong>Slowheader</strong></td>
<td>Slowloris keeps as many connections, to a target web server, open as possible. This attack works by sending only a partial request to a web server. Periodically, it sends HTTPS headers which add to but never complete the request. In the meanwhile, it establishes several similar connections forcing target web servers to keep these connections open, exhausting the concurrent connection.</td>
<td>Web Server Protection against Denial-of-Service attacks</td>
<td>NA</td>
</tr>
<tr>
<td>Attack type</td>
<td>Description</td>
<td>Mitigation techniques</td>
<td></td>
</tr>
<tr>
<td>-----------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------------------------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Network Security Platform technique</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Attack signature to be enabled (Only for IPS Policy)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Network Security</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Platform technique</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>pools of such servers. Subsequent</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>legitimate connection requests are denied</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>creating a DoS condition.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>IPS Policy</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• HTTP: Possible Slowloris Denial of Service Attack Detected</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• HTTP: ASP.NET Remote DoS Vulnerability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Slow Post attacks</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>IPS Policy</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• HTTP: ASP .NET VIEWSTATE Replay Attacks and DoS vulnerability</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• HTTP: W32/ Mydoom@MM DoS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Attacks that send the network into an</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>infinite loop</td>
<td><strong>IPS Policy</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DoS: Bonk Attack</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DoS: SynDr op Attack</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DoS: Land Attack</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• BOT: Pushdo Bot SSL DoS</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• BOT: Ddose r Bot Traffic Detected</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• BOT: Spike Ddos traffic Detected</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DDoS toolkit-based attacks</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Stacheldraht</td>
<td>A piece of software written for Linux and Solaris systems which act as</td>
<td><strong>IPS Policy</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td>DDoS agents.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>IPS Policy</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DDoS: Stacheldraht Handler-check-gag</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DDoS: Stacheldraht Agent-response-gag</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DDoS: Stacheldraht Master-to-Agent Communication</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DDoS: Stacheldraht Agent-to-Master</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DDoS: Stacheldraht Master-Spoofworks</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DoS: UDP-Based Jolt2 Attack</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DDoS: Stacheldraht Master-Response</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DoS: UDP Bomb</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Attack type</td>
<td>Description</td>
<td>Mitigation techniques</td>
<td>Attack signature to be enabled (Only for IPS Policy)</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
<td>--------------------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Trin00</td>
<td>A set of computer programs to conduct a DDoS attack. Trinoo networks are believed to include thousands of computers across the Internet that have been compromised by remote buffer overrun exploits.</td>
<td>IPS Policy</td>
<td>• DDoS: Trin00 Daemon-to-Master (PONG)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DDoS: Trin00 Daemon-to-Master</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DDoS: Trin00 Attacker-to-Master Remote Password</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DDoS: Trin00 Master-to-Agent Communication</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DDoS: Trin00 Attacker-to-Master Default mdie Password</td>
</tr>
<tr>
<td>Mstream</td>
<td>A distributed denial of service tool that is based on the “stream.c” attack, this tool consists of a master controller and a zombie. The master controller control all zombies. An attacker uses Telnet to connect with the master controller to control zombies.</td>
<td>IPS Policy</td>
<td>• DDoS: mstream Handler Ping to Agent</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DDoS: mstream Master-to-Handler Communication</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DDoS: mstream Handler-to-Agent Communication</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DDoS: mstream Agent-to-Handler Communication</td>
</tr>
<tr>
<td>TFN / TFN2k</td>
<td>Tribal Flood Network (TFN) is a set of computer programs that orchestrate DDoS attacks such as ICMP flood, SYN flood, UDP flood, and Smurf attacks.</td>
<td>IPS Policy</td>
<td>• DDoS: TFN2k ICMP Possible Communication</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DDoS: TFN Client Command</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• DDoS: TFN Agent Response</td>
</tr>
<tr>
<td>Attack type</td>
<td>Description</td>
<td>Mitigation techniques</td>
<td>Attack signature to be enabled (Only for IPS Policy)</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
<td>-----------------------</td>
<td>---------------------------------------------------</td>
</tr>
<tr>
<td>Trinity</td>
<td>Trinity is a DDoS attack tool, which uses Internet relay chat (IRC) as the communications channel for the agent (zombie) and the controller (master). Trinity can launch eight different flood attacks: ACK, establish, fragment, null, random flags, RST, SYN, and UDP floods.</td>
<td>Network Security Platform technique IPS Policy</td>
<td>IRC: Trinity DDoS</td>
</tr>
<tr>
<td>Shaft</td>
<td>Shaft belongs in the family of DDoS tools such as Trin00, TFN, Stacheldraht, and TFN2K. As in those tools, there are handler (or master) and agent programs. The Shaft &quot;network&quot; is made up of one or more handler programs (&quot;shaftmaster&quot;) and a large set of agents (&quot;shaftnode&quot;). Shaft uses &quot;tickets&quot; to keep track of its individual agents. Both passwords and ticket numbers have to match for the agent to execute the request. A simple, letter-shifting Caesar encryption cipher is in use. An attacker may use a telnet program (&quot;client&quot;) to connect to and communicate with the handlers.</td>
<td>Network Security Platform technique IPS Policy</td>
<td>• DDoS: Shaft Agent-to-Handler Communication • DDoS: Shaft Handler-to-Agent Communication</td>
</tr>
</tbody>
</table>

### Statistical Anomaly


The Sensor builds a profile for each tracked packet type in each direction. Within each source IP profile, the entire IP address space is divided into a maximum of 128 mutually exclusive IP address blocks, or bins, much in the same way CIDR addressing divides the address space. Each bin is uniquely identified by a prefix and prefix length (from 2 to 32 bits). An IP address falls into a bin when the first 'n' number of bits of the address matches the bin's prefix. The Sensor then associates each source IP with a particular bin in the appropriate profile.

Each bin has the following two properties:

- The percentage of long-term (good) traffic originating from the source IPs that belongs to this bin.
- The percentage of the overall IP address space that the IP range in this bin occupies.

With the source IPs properly classified, the Sensor can now protect a network from DoS attacks. When a statistical anomaly occurs, the Sensor takes the following actions on the source IP profile in question:

- The Sensor blocks all packets with source IPs in the bins that occupy a large percentage of the IP space, but represent a small percentage of the long-term traffic. This combats attacks that are generated with random, wide-ranging, spoofed source IP addresses.
- The Sensor blocks all packets with source IPs in the bins that occupy a large percentage of the short-term traffic together with a significantly higher percentage of short-term traffic than historically seen. This combats attacks that are initiated from a handful of networks with authentic source IP addresses.
- The Sensor does not block packets with source IPs in the bins that occupy a small percentage of the IP space and represent a high percentage of the long-term traffic. This protects against blocking hosts that are known to be good.
The exception to the third criterion is when the traffic also meets the second criterion. In other words, source IPs from the “good” bins are blocked if their short-term traffic level is significantly higher than their peak long-term level. This combats attacks that are initiated from good hosts that have recently been compromised.

Source IP classification is more effective than using devices such as firewalls that limits the rate of SYN packets on the network to block DoS attacks. The key difference in such an approach and Network Security Platform is that a rate-limiting device blocks traffic randomly. “Good” traffic has the same probability of being blocked as attack traffic. On the other hand, source IP classification used by Network Security Platform attempts to differentiate good traffic from attack traffic, so attack traffic is more likely to be blocked.

SYN Cookie

In a SYN flood attack, Server resources are targeted to consume TCP memory by sending SYN, and after the Server responds with a SYN+ACK, force the Server to hold state information while waiting for the Client’s ACK message. As the Server maintains state for half-open connections, Server resources are constrained. The Server may no longer be able to accept TCP connections, resulting in a DoS condition.

Network Security Platform uses a specific choice of initial TCP number as a defense against SYN flood attacks.

With SYN cookies enabled in the Network Security Manager, whenever a new connection request arrives at a server, the server sends back a SYN+ACK with an Initial Sequence Number (ISN) uniquely generated using the information present in the incoming SYN packet and a secret key. If the connection request is from a legitimate host, the server gets back an ACK from the host. If the connection request is not from a legitimate host, a TCP session is not created, and a potential DoS condition is prevented.

To enable SYN cookies in a Network Security Platform 8.2 Manager, follow these steps.

1. Click the **Devices** tab.
2. Select the domain from the **Domain** drop-down list.
3. On the left pane, click the **Devices** tab.
4. Select the device from the **Device** drop-down list.
5. Select **Setup | Advanced | Protocol Settings**.
6. Look for **SYN Cookie** in the **TCP Settings** page and enable it for inbound, outbound, or both using the drop-down list.
7. Enter a **Threshold Value** for the directions that you have enabled.
8. To edit the parameter, click **Update** for that parameter.
To restore the default values, scroll down to the bottom of the page and click **Restore**.

![Statistics - Device Packet Drops](image)

**Figure 4-1 SYN cookie enabled**

**Stateful TCP Engine**

Sensor has a built-in TCP stack that follows the state required for TCP protocol. Sensor uses the TCP state to prevent out of context packets and packets without a valid state. The TCP stateful engine can be used effectively to drop spoofed TCP RST, TCP FIN, TCP ACK floods.

To enable this engine, follow these steps:

1. Click the **Devices** tab.
2. Select the domain from the **Domain** drop-down list.
3. On the left pane, click the **Devices** tab.
4. Select the device from the **Device** drop-down list.

6. Look for TCP Flow Violation in the TCP Settings page and select Deny no TCB from the drop-down list.

Figure 4-2 Stateful TCP engine

**DNS Protect**

DNS Protect feature in sensor can be used to protect DNS Servers from DoS spoof attack by forcing the DNS clients to use TCP instead of UDP as their transport protocol. Since TCP uses three-way-handshake, it is comparatively tough to launch spoofed attacks when TCP is used.

You can set the DNS protection mode, add to, or delete existing DNS Spoof protection IP addresses from the protected server list using CLI commands.

**Rate Limiting**

Rate limiting is used to control the rate of egress (traffic going out) traffic sent through the ports of a Sensor. When deployed in inline mode, Sensor permits rate limiting of traffic by limiting the bandwidth of the traffic that goes through the Sensor ports. Traffic that is less than or equal to the specified bandwidth value is allowed, whereas traffic that exceeds the bandwidth value is dropped. The Sensor uses the token bucket approach for rate limiting traffic. An Administrator can set appropriate values in the Manager to prevent DoS by setting Sensor port specific bandwidth limits that are relevant for preventing DoS in a particular network.

You can rate-limit by protocol such as P2P, HTTP and ICMP as by TCP ports, UDP ports, and IP Protocol Number.

Network Security Platform provides rate limiting configuration at individual sensor ports. For example, if 1A-1B is a port-pair, traffic management is configured separately for 1A and 1B. Traffic Management configuration for a port applies to egress traffic only.
In the Manager, every rate limiting queue of a Sensor is uniquely identified by a name. The traffic management queues are configured based on Protocol, TCP ports, UDP ports, and IP Protocol Number. You can create multiple queues for each port of the sensor. The traffic management configuration in the Manager must be followed by a configuration update to the sensor.

Rate limiting option is available as one of the traffic management options in the Manager.

Rate limiting is very effective when applied in a specific context with full knowledge of the nature of traffic in a particular network. Rate limiting needs to be applied carefully as it might drop legitimate traffic as well.

Rate limiting is applicable only to M-series Sensors.

**ACLs**

You can also use ACLs to prevent DoS attacks.

ACLs can be created for a combination of any source IP addresses, destination IP addresses, specified CIDR blocks, destination protocol/port, by TCP/UDP port, by ICMP type, and by IP protocol for the Sensor as a whole and per individual port pair.

ACL can be used to mitigate DoS attacks by creating ACLs specific to the nature of traffic in a network. For instance, if you are aware of what protocols are normally seen in your network, you can configure ACL to drop the type of traffic that is not normally expected in your network.

Permit, Drop or Deny response action can be set while enabling intrusion prevention matching a configured rule. When used within an enterprise, ACL Permit can be configured from all know CIDRs.

**Connection Limiting with GTI Integration enabled**

You can define a threshold value to limit the number of connections/ per second or the number of active connections to prevent connection based DoS attacks.

The Sensor provides the ability to define threshold values to limit number of connections (three-way TCP handshakes) a host can establish. The number of connections or connection rate that is less than or equal to the defined threshold value is allowed. When this number is exceeded, the subsequent connections are dropped. This helps in minimizing the connection-based DoS attacks on server.

The threshold value is defined as the number of connections/second or active connections. For example, if you define 1 connection/sec as the threshold value then, there are 10 connections in the 1st second, all the other connections from the 2nd to the 10th second will be dropped. On the other hand, if you have 1 connection for each second, all the 10 connections till the 10th second will be allowed. This is also known as Traffic sampling.

You can define the Connection Limiting rules of the following types:

- **Protocol** - use this to limit TCP/UDP/ICMP active connections or connection rate from a host.
- **GTI** - In this case, the Sensor integrates with McAfee GTI IP Reputation to obtain the reputation score and geo-location of the external host. Therefore, use this to define connection limiting rules for traffic to/from external hosts based on reputation and geo-location of the external hosts.
Custom Reconnaissance Attack Definition

You can create custom signature-based attack definitions to prevent exploit based DoS attacks. Using the Custom Attack Editor, you can define correlated attacks using these individual attack definitions. For example, custom attacks that check for URI can be further correlated to test for multiple occurrences in a defined time interval to raise a correlated attack. The correlation methods supported are:

- Brute Force
- Host Sweep
- Port Scan
- Service Sweep
- Finger Printing

When traffic passing through the Sensor exceeds the threshold count set for the custom reconnaissance attack within a configured interval, the Sensor raises an alert to the Manager. You can then opt to take the response actions such as blocking or quarantining the host.

Web Server - Denial-of-Service Protection

Attacks on web servers can either target the infrastructure or target a particular URL/path which results in overloading of web servers or exceeding the server capacity. Since it is hard to distinguish bot traffic, you can use Network Security Platform to configure different response actions based on a threshold, and revert with a challenge to determine whether the HTTP requests originated from valid browsers or bots.

Once the challenge/response has been validated the NSP Sensor forwards the request to the web server. Invalid responses to the challenge are dropped and not forwarded to the web server.

To combat DoS attacks on web servers, you can configure your Sensor to send different response actions:

- Configure the Maximum Simultaneous Connections Allowed to All Web Servers which limits the active number of connections and prevents your server from being overloaded.
- Recover from a DoS attack by enabling Prevent Slow Connection Web Server Attacks. This feature clears 10% of the oldest connections that have been alive for more than 5 minutes.
You can also configure your Sensor to protect web servers by restricting web clients:

- Configure **Maximum URL Request Rate Allowed to All Website Paths** to restrict the number of requests.
- Set the **Enable Web Client Browser Detection** option to send a Javascript/HTML challenge to the client. The Sensor will then wait for a response before allowing the request to proceed.
- Specify **Website Paths to Protect** and the **Maximum URL Requests/second** that must be allowed to protect critical assets within your network.

![Figure 4-3 L7 DDoS web server protection](image)

**Selecting specific DoS attacks for blocking in the Manager**

Network Security Platform's IPS signature set provides you with a wide range of DoS attack signatures for attacks such as SYN floods and UDP floods, attacks using DDoS toolkits such as Stacheldraht, Trin00, and TFN, and HTTP-based attacks.

You can block any or all of these attacks in your IPS deployment if you enable the appropriate attack signatures. For instance, your network might be witnessing a high incidence of DoS attacks of a particular type, say Trin00 or another type of DoS attack.

In these circumstances, you can choose to mitigate risks by modifying your existing IPS policy and swiftly block such attacks. You can achieve this by filtering just Trin00 attack signatures from the signature set in your IPS policy editor.

To filter just Trin00 attack signatures in a Network Security Platform 8.2 Manager:

1. Within the **Policy** tab, go to **Intrusion Prevention | IPS Policies**.
   - The list of IPS policies appears.
2. Select the policy that you intend to use.
3 In the top-right section of this window, enter the attack signature that you want to view in the Quick Search field. For example, DDoS: Trin00 Daemon-to-Master.

4 Double-click the attack and specify the parameters in the right side pane.
   There are no recommended settings for these parameters. You must select the options that best suit your requirements.
Index

A
about this guide 5

C
conventions and icons used in this guide 5

D
documentation
   audience for this guide 5
   product-specific, finding 6
   typographical conventions and icons 5

M
McAfee ServicePortal, accessing 6

S
ServicePortal, finding product documentation 6

T
Technical Support, finding product information 6