None.
Internet-Draft
Expires: October 12, 2006

C. Morrow
UUNET Technologies
G. Jones
The MITRE Corporation
V. Manral
IPInfusion
May 12, 2006

# Filtering Capabilities for IP Network Infrastructure
# draft-ietf-opsec-filter-caps-01

## Status of this Memo



## Copyright Notice



## Abstract

[I-D.practices] lists operator practices related to securing networks. This document lists filtering capabilities needed to support those practices.

Capabilities are defined without reference to specific technologies.

This is done to leave room for deployment of new technologies that implement the capability. Each capability cites the practices it supports. Current implementations that support the capability are cited. Special considerations are discussed as appropriate listing operational and resource constraints, limitations of current implementations, tradeoffs, etc.

Table of Contents

1. Introduction . . . . . 6
1.1. Threat Model . . . . . 6
1.2. Capabilities or Requirements ? . . . . . 7
1.3. Format . . . . . 7
1.4. Definitions . . . . . 7
2. Functional Capabilities - Filtering non-transit traffic (management plane) . . . . . 9
2.1. Filtering TO the Device . . . . . 9
2.1.1. Ability to Filter Traffic on All Interfaces TO the Device . . . . . 9
2.1.2. Ability to Filter Traffic To the Device . . . . . 10
2.1.3. Ability to Filter Traffic To the Device - Minimal Performance Degradation . . . . . 10
2.1.4. Ability to Filter To the Device - Specify Filter Actions . . . . . 12
2.1.5. Ability to Filter To the Device - Log Filter Actions . . . . . 13
2.1.6. Ability to Filter To the Device - Specify Log Granularity . . . . . 14
2.1.7. Ability to Filter To the Device - Ability to Filter Protocols . . . . . 15
2.1.8. Ability to Filter To the Device - Ability to Filter Addresses . . . . . 15
2.1.9. Ability to Filter To the Device - Ability to Filter Protocol Header Fields . . . . . 16
2.1.10. Ability to Filter To the Device - Ability to Filter Inbound and Outbound . . . . . 17
2.1.11. Ability to Filter To the Device - Ability to Accurately Count Filter Hits . . . . . 18
2.1.12. Ability to Filter To the Device - Ability to Display Filter Counters . . . . . 19
2.1.13. Ability to Filter To the Device - Ability to Display Filter Counters per Filter Application . . . . 20
2.1.14. Ability to Filter To the Device - Ability to Reset Filter Counters . . . . . 21
2.1.15. Ability to Filter To the Device - Filter Counters are Accurate . . . . . 22
2.2. Rate Limit TO the Device . . . . . 22

2.2.1. Ability to Rate limit Traffic on All Interfaces TO the Device . . . . . 22
2.2.2. Ability to Rate Limit Traffic To the Device . . . . . 23
2.2.3. Ability to Rate Limit Traffic To the Device - Minimal Performance Degradation . . . . . 24
2.2.4. Ability to Rate Limit To the Device - Specify Rate Limit Actions . . . . . 26
2.2.5. Ability to Rate Limit To the Device - Log Rate Limit Actions . . . . . 27
2.2.6. Ability to Rate Limit To the Device - Specify Log Granularity . . . . . 28
2.2.7. Ability to Rate Limit To the Device - Ability to Rate Limit Protocols . . . . . 28
2.2.8. Ability to Rate Limit To the Device - Ability to Rate Limit Addresses . . . . . 29
2.2.9. Ability to Rate Limit To the Device - Ability to Rate Limit Protocol Header Fields . . . . . 30
2.2.10. Ability to Rate Limit To the Device - Ability to Rate Limit Inbound and Outbound . . . . . 31
2.2.11. Ability to Rate Limit To the Device - Ability to Accurately Count Rate Limit Hits . . . . . 32
2.2.12. Ability to Rate Limit To the Device - Ability to Display Rate Limit Counters . . . . . 33
2.2.13. Ability to Rate Limit To the Device - Ability to Display Rate Limit Counters per Rate Limit Application . . . . . 33
2.2.14. Ability to Rate Limit To the Device - Ability to Reset Rate Limit Counters . . . . . 34
2.2.15. Ability to Rate Limit To the Device - Rate Limit Counters are Accurate . . . . . 35
3. Functional Capabilities - Filtering transit traffic (data plane) . . . . . 37
3.1. Filtering THROUGH the Device . . . . . 37
3.1.1. Ability to Filter Traffic on All Interfaces THROUGH the Device . . . . . 37
3.1.2. Ability to Filter Traffic Through the Device . . . . . 38
3.1.3. Ability to Filter Traffic Through the Device - Minimal Performance Degradation . . . . . 38
3.1.4. Ability to Filter Through the Device - Specify Filter Actions . . . . . 40
3.1.5. Ability to Filter Through the Device - Log Filter Actions . . . . . 41
3.1.6. Ability to Filter Through the Device - Specify Log Granularity . . . . . 42
3.1.7. Ability to Filter Through the Device - Ability to Filter Protocols . . . . . 43
3.1.8. Ability to Filter Through the Device - Ability to Filter Addresses . . . . . 43

3.1.9. Ability to Filter Through the Device - Ability to Filter Protocol Header Fields . . . . . 44
3.1.10. Ability to Filter Through the Device - Ability to Filter Inbound and Outbound . . . . . 45
3.1.11. Ability to Filter Through the Device - Ability to Accurately Count Filter Hits . . . . . 46
3.1.12. Ability to Filter Through the Device - Ability to Display Filter Counters . . . . . 47
3.1.13. Ability to Filter Through the Device - Ability to Display Filter Counters per Filter Application . . . . 48
3.1.14. Ability to Filter Through the Device - Ability to Reset Filter Counters . . . . . 49
3.1.15. Ability to Filter Through the Device - Filter Counters are Accurate . . . . . 50
3.2. Rate Limit THROUGH the Device . . . . . 50
3.2.1. Ability to Rate limit Traffic on All Interfaces THROUGH the Device . . . . . 51
3.2.2. Ability to Rate Limit Traffic Through the Device . . . 51
3.2.3. Ability to Rate Limit Traffic Through the Device - Minimal Performance Degradation . . . . . 52
3.2.4. Ability to Rate Limit Through the Device - Specify Rate Limit Actions . . . . . 54
3.2.5. Ability to Rate Limit Through the Device - Log Rate Limit Actions . . . . . 55
3.2.6. Ability to Rate Limit Through the Device - Specify Log Granularity . . . . . 56
3.2.7. Ability to Rate Limit Through the Device - Ability to Rate Limit Protocols . . . . . 57
3.2.8. Ability to Rate Limit Through the Device - Ability to Rate Limit Addresses . . . . . 57
3.2.9. Ability to Rate Limit Through the Device - Ability to Rate Limit Protocol Header Fields . . . . . 58
3.2.10. Ability to Rate Limit Through the Device - Ability to Rate Limit Inbound and Outbound . . . . . 59
3.2.11. Ability to Rate Limit Through the Device - Ability to Accurately Count Rate Limit Hits . . . . . 60
3.2.12. Ability to Rate Limit Through the Device - Ability to Display Rate Limit Counters . . . . . 61
3.2.13. Ability to Rate Limit Through the Device - Ability to Display Rate Limit Counters per Rate Limit Application . . . . . 62
3.2.14. Ability to Rate Limit Through the Device - Ability to Reset Rate Limit Counters . . . . . 63
3.2.15. Ability to Rate Limit Through the Device - Rate Limit Counters are Accurate . . . . . 64
4. Functional Capabilities - Filtering Layer 2 Attributes . . . . 65
4.1. Filtering Layer 2 . . . . . 65
4.1.1. Ability to partition layer-2 network to provide

different levels of security . . . . . 65
4.1.2. Ability to restrict access to specified hardware (MAC) addresses . . . . . 66
4.1.3. Ability to restrict based on layer-2 packet type [etherType] field . . . . . 67
5. Additional Operational Practices . . . . . 68
5.1. Profile Current Traffic . . . . . 68
5.2. Block Malicious Packets . . . . . 68
5.3. Limit Sources of Management . . . . . 68
6. Security Considerations . . . . . 69
7. References . . . . . 70
7.1. Normative References . . . . . 70
7.2. Non-normative References . . . . . 70
Appendix A. Acknowledgments . . . . . 71
Authors' Addresses . . . . . 72
Intellectual Property and Copyright Statements . . . . . 73

## 1. Introduction

This document is defined in the context of [I-D.practices]. [I-D.practices] defines the goals, motivation, scope, definitions, intended audience,threat model, potential attacks and give justifications for each of the practices. Many of the capabilities listed here refine or add to capabilities listed in [RFC3871]

### 1.1. Threat Model

Threats in today's networked environment range from simple packet floods with overwhelming bandwidth toward a leaf network to subtle attacks aimed at subverting known vulnerabilities in existing applications. The attacked network or host might not be an end user, it may be the networking device or links inside the provider core.

Networks must have the ability to place mitigation in order to limit these threats. These mitigation steps could include routing updates, traffic filters, and routing filters. It is possible that the mitigation steps might have to affect transit traffic as well as traffic destined to the device on which the mitigation steps are activated.

The scope of the threat includes simply denying services to an individual customer on one side of the scale to exploiting a newly discovered protocol vulnerability which affects the entire provider core. The obvious risk to the business requires mitigation capabilities which can span this range of threats.

Threat: An indication of impending danger or harm to the network or its parts. This could be formed from the projected loss of revenue to the business. Additionally, it could be formed from the increased cost to the business caused by the event. (more interfaces, more bandwidth, more personnel to support the increased size or complexity)

Risk: The possibility of suffering harm or loss of network services due to a threat.

Attack: To set upon with violent force the network or its parts. Typically this is a form of flood of packets to or through a network. This could also be a much smaller stream of packets created with the intent of exploiting a vulnerability in the infrastructure of the network.

Asset: Either a customer, network device or network link. Any of these could be assets from a business perspective.

These terms are more completely defined in RFC2828 we have added some scope specific information only.

## 1.2. Capabilities or Requirements ?

Capabilities may or may not be requirements. That is a local determination that must be made by each operator with reference to the policies that they must support. It is hoped that this document, together with [I-D.practices] will assist operators in identifying their security capability requirements and communicating them clearly to vendors.

## 1.3. Format

Each capability has the following subsections:

- Capability (what)
- Supported Practices (why)
- Current Implementations (how)
- Considerations (caveats, resource issues, protocol issues, etc.)

The Capability section describes a feature to be supported by the device. The Supported Practice section cites practices described in [I-D.practices] that are supported by this capability. The Current Implementation section is intended to give examples of implementations of the capability, citing technology and standards current at the time of writing. See [RFC3631]. It is expected that the choice of features to implement the capabilities will change over time. The Considerations section lists operational and resource constraints, limitations of current implementations, tradeoffs, etc.

[EDITORS NOTE: this is a first draft. At least two editing passes will be made over the capabilities listed below in future drafts: one will break out compound capabilities into individual capabilities, the other will try to align the supported practices with the practices listed in [I-D.practices]]

## 1.4. Definitions

RFC 2119 Keywords

> The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

The use of the RFC 2119 keywords is an attempt, by the editor, to assign the correct requirement levels ("MUST", "SHOULD", "MAY"...). It must be noted that different organizations, operational environments, policies and legal environments will generate different requirement levels.

## 2. Functional Capabilities - Filtering non-transit traffic (management plane)

The capabilities in this section are intended to list testable, functional capabilities that are needed to operate devices securely. Focusing on filtering non-transit packets on devices, controlling access to the management plane.

### 2.1. Filtering TO the Device

#### 2.1.1. Ability to Filter Traffic on All Interfaces TO the Device

Capability.

The device provides a means to filter IP packets on any interface implementing IP that are non-transit packets.

Supported Practices.

* Profile Current Traffic ([I-D.practices] Section x.x.x)
* Block Malicious Packets (Section 5.2)
* Limit Sources of Management (Section 5.3)

Current Implementations.

Many devices currently implement access control lists or filters that allow filtering based on protocol and/or source/destination address and or source/destination port and allow these filters to be applied to interfaces.

Considerations.

None.

### 2.1.2. Ability to Filter Traffic To the Device

Capability.

It is possible to apply the filtering mechanism to traffic that is addressed directly to the device via any of its interfaces - including loopback interfaces.

Supported Practices.

* This allows the operator to apply filters that protect the device itself from attacks and unauthorized access.

Current Implementations.

Many devices currently implement access control lists or filters that allow filtering based on protocol and/or source/destination address and or source/destination port and allow these filters to be applied to services offered by the device.

Examples of this might include filters that permit only BGP from peers and SNMP and SSH from an authorized management segment and directed to the device itself, while dropping all other traffic addressed to the device.

Considerations.

None.

### 2.1.3. Ability to Filter Traffic To the Device - Minimal Performance Degradation

Capability.

The device provides a means to filter packets without significant performance degradation. This specifically applies to stateless packet filtering operating on layer 3 (IP) and layer 4 (TCP or UDP) headers, as well as normal packet forwarding information such as incoming and outgoing interfaces.

The device is able to apply stateless packet filters on ALL interfaces (up to the maximum number possible) simultaneously and with multiple filters per interface (e.g., inbound and outbound).

The filtering of traffic destined to interfaces on the device, including the loopback interface, should not degrade performance significantly.

Supported Practices.

* This enables the implementation of filters on whichever services necessary. To the extent that filtering causes degradation, it may not be possible to apply filters that implement the appropriate policies.

Current Implementations.

Another way of stating the capability is that filter performance should not be the limiting factor in device throughput. If a device is capable of forwarding 30Mb/sec without filtering, then it should be able to forward the same amount with filtering in place.

Considerations.

The definition of "significant" is subjective. At one end of the spectrum it might mean "the application of filters may cause the box to crash". At the other end would be a throughput loss of less than one percent with tens of thousands of filters applied.

The level of performance degradation that is acceptable will have to be determined by the operator.

Repeatable test data showing filter performance impact would be very useful in evaluating this capability. Tests should include such information as packet size, packet rate, number of interfaces tested (source/destination), types of interfaces, routing table size, routing protocols in use, frequency of routing updates, etc.

This capability does not address stateful filtering, filtering above layer 4 headers or other more advanced types of filtering that may be important in certain operational environments.

### 2.1.4. Ability to Filter To the Device - Specify Filter Actions

Capability.

The device provides a mechanism to allow the specification of the action to be taken when a filter rule matches. Actions include "permit" (allow the traffic), "reject" (drop with appropriate notification to sender), and "drop" (drop with no notification to sender).

Supported Practices.

* This capability is essential to the use of filters to enforce policy.

Current Implementations.

Assume that your management devices for deployed networking devices live on several subnets, use several protocols, and are controlled by several different parts of your organization. There might exist a reason to have disparate policies for access to the devices from these parts of the organization.

Actions such as "permit", "deny", "drop" are essential in defining the security policy for the services offered by the network devices.

Considerations.

While silently dropping traffic without sending notification may be the correct action in security terms, consideration should be given to operational implications. See [RFC3360] for consideration of potential problems caused by sending inappropriate TCP Resets.

### 2.1.5. Ability to Filter To the Device - Log Filter Actions

Capability.

It is possible to log all filter actions. The logging capability is able to capture at least the following data:

* permit/deny/drop status
* source and destination IP address
* source and destination ports (if applicable to the protocol)
* which network element received the packet (interface, MAC address or other layer 2 information that identifies the previous hop source of the packet).

Supported Practices.

* Logging is essential for auditing, incident response, and operations

Current Implementations.

Actions such as "permit", "deny", "drop" are essential in defining the security policy for the services offered by the network devices. Auditing the frequency, sources and destinations of these attempts is essential for tracking ongoing issues today.

Considerations.

Logging can be burdensome to the network device, at no time should logging cause performance degradation to the device or services offered on the device.

### 2.1.6. Ability to Filter To the Device - Specify Log Granularity

Capability.

It is possible to enable/disable logging on a per rule basis.

Supported Practices.

* The ability to tune the granularity of logging allows the operator to log the information that is desired and only the information that is desired. Without this capability, it is possible that extra data (or none at all) would be logged, making it more difficult to find relevant information.

Current Implementations.

If a filter is defined that has several rules, and one of the rules denies telnet (tcp/23) connections, then it should be possible to specify that only matches on the rule that denies telnet should generate a log message.

Considerations.

None.

### 2.1.7. Ability to Filter To the Device - Ability to Filter Protocols

Capability.

The device provides a means to filter traffic based on the value of the protocol field in the IP header.

Supported Practices.

* Being able to filter on protocol is necessary to allow implementation of policy, secure operations and for support of incident response.

Current Implementations.

Some denial of service attacks are based on the ability to flood the victim with ICMP traffic. One quick way (admittedly with some negative side effects) to mitigate the effects of such attacks is to drop all ICMP traffic headed toward the victim.

Considerations.

None.

### 2.1.8. Ability to Filter To the Device - Ability to Filter Addresses

Capability.

The function is able to control the flow of traffic based on source and/or destination IP address or blocks of addresses such as Classless Inter-Domain Routing (CIDR) blocks.

Supported Practices.

* The capability to filter on addresses and address blocks is a fundamental tool for establishing boundaries between different networks.

Current Implementations.

One example of the use of address based filtering is to implement ingress filtering per [RFC2827].

Considerations.

None.

### 2.1.9. Ability to Filter To the Device - Ability to Filter Protocol Header Fields

Capability.

The filtering mechanism supports filtering based on the value(s) of any portion of the protocol headers for IP, ICMP, UDP and TCP. It supports filtering of all other protocols supported at layer 3 and 4. It supports filtering based on the headers of higher level protocols. It is possible to specify fields by name (e.g., "protocol = ICMP") rather than bit- offset/length/numeric value (e.g., 72:8 = 1).

Supported Practices.

* Being able to filter on portions of the header is necessary to allow implementation of policy, secure operations, and support incident response.

Current Implementations.

This capability implies that it is possible to filter based on TCP or UDP port numbers, TCP flags such as SYN, ACK and RST bits, and ICMP type and code fields. One common example is to reject "inbound" TCP connection attempts (TCP, SYN bit set+ACK bit clear or SYN bit set+ACK,FIN and RST bits clear). Another common example is the ability to control what services are allowed in/out of a network. It may be desirable to only allow inbound connections on port 80 (HTTP) and 443 (HTTPS) to a network hosting web servers.

Considerations.

None.

### 2.1.10. Ability to Filter To the Device - Ability to Filter Inbound and Outbound

Capability.

It is possible to filter both incoming and outgoing traffic on any interface.

Supported Practices.

* This capability allows flexibility in applying filters at the place that makes the most sense. It allows invalid or

malicious traffic to be dropped as close to the source as possible.

Current Implementations.

It might be desirable on a border router, for example, to apply an egress filter outbound on the interface that connects a site to its external ISP to drop outbound traffic that does not have a valid internal source address. Inbound, it might be desirable to apply a filter that blocks all traffic from a site that is known to forward or originate lots of junk mail.

Considerations.

None.

### 2.1.11. Ability to Filter To the Device - Ability to Accurately Count Filter Hits

Capability.

The device supplies a facility for accurately counting all filter matches.

Supported Practices.

* Accurate counting of filter rule matches is important because it shows the frequency of attempts to violate policy. This enables resources to be focused on areas of greatest need.

Current Implementations.

Assume, for example, that a ISP network implements anti-spoofing egress filters (see [RFC2827]) on interfaces of its edge routers that support single-homed stub networks. Counters could enable the ISP to detect cases where large numbers of spoofed packets are being sent. This may indicate that the customer is performing potentially malicious actions (possibly in violation of the ISPs Acceptable Use Policy), or that system(s) on the customers network have been "owned" by hackers and are being (mis)used to launch attacks.

Considerations.

None.

### 2.1.12. Ability to Filter To the Device - Ability to Display Filter Counters

Capability.

The device provides a mechanism to display filter counters.

Supported Practices.

* Information that is collected is not useful unless it can be displayed in a useful manner.

Current Implementations.

Assume there is a router with four interfaces. One is an up-link to an ISP providing routes to the Internet. The other three connect to separate internal networks. Assume that a host on one of the internal networks has been compromised by a hacker and is sending traffic with bogus source addresses. In such a situation, it might be desirable to apply ingress filters to each of the

internal interfaces. Once the filters are in place, the counters can be examined to determine the source (inbound interface) of the bogus packets.

Considerations.

None.

### 2.1.13. Ability to Filter To the Device - Ability to Display Filter Counters per Filter Application

Capability.

If it is possible for a filter to be applied more than once at the same time, then the device provides a mechanism to display filter counters per filter application.

Supported Practices.

* It may make sense to apply the same filter definition simultaneously more than one time (to different interfaces, etc.). If so, it would be much more useful to know which instance of a filter is matching than to know that some instance was matching somewhere.

Current Implementations.

One way to implement this capability would be to have the counter display mechanism show the interface (or other entity) to which the filter has been applied, along with the name (or other designator) for the filter. For example if a filter named "desktop_outbound" applied two different interfaces, say, "ethernet0" and "ethernet1", the display should indicate something like "matches of filter 'desktop_outbound' on ethernet0 ..." and "matches of filter 'desktop_outbound' on ethernet1 ..."

Considerations.

None.

### 2.1.14. Ability to Filter To the Device - Ability to Reset Filter Counters

Capability.

It is possible to reset counters to zero on a per filter basis.

For the purposes of this capability it would be acceptable for the system to maintain two counters: an "absolute counter", C[now], and a "reset" counter, C[reset]. The absolute counter would maintain counts that increase monotonically until they wrap or overflow the counter. The reset counter would receive a copy of the current value of the absolute counter when the reset function was issued for that counter. Functions that display or retrieve the counter could then display the delta (C[now] - C[reset]).

Supported Practices.

* This allows operators to get a current picture of the traffic matching particular rules/filters.

Current Implementations.

Assume that filter counters are being used to detect internal hosts that are infected with a new worm. Once it is believed that all infected hosts have been cleaned up and the worm removed, the next step would be to verify that. One way of doing so would be to reset the filter counters to zero and see if traffic indicative of the worm has ceased.

Considerations.

None.

### 2.1.15. Ability to Filter To the Device - Filter Counters are Accurate

Capability.

Filter counters are accurate. They reflect the actual number of matching packets since the last counter reset. Filter counters are be capable of holding up to 2^32 - 1 values without overflowing and should be capable of holding up to 2^64 - 1 values.

Supported Practices.

* Inaccurate data can not be relied on as the basis for action. Underreported data can conceal the magnitude of a problem.

Current Implementations.

If N packets matching a filter are sent to/through a device, then the counter should show N matches.

Considerations.

None.

## 2.2. Rate Limit TO the Device

### 2.2.1. Ability to Rate limit Traffic on All Interfaces TO the Device

Capability.

The device provides a means to rate limit IP packets on any interface implementing IP that are non-transit packets.

Supported Practices.

* Profile Current Traffic ([I-D.practices] Section x.x.x)
* Block Malicious Packets (Section 5.2)
* Limit Sources of Management (Section 5.3)

Current Implementations.

Many devices currently implement rate limits that allow rate limiting based on protocol and/or source/destination address and or source/destination port or raw bandwidth and allow these limits to be applied to interfaces.

Considerations.

None.

### 2.2.2. Ability to Rate Limit Traffic To the Device

Capability.

It is possible to apply the rate-limiting mechanism to traffic that is addressed directly to the device via any of its interfaces - including loopback interfaces.

Supported Practices.

* This allows the operator to apply rate-limits that protect the device itself from attacks and unauthorized access.

Current Implementations.

Many devices currently implement rate-limits that allow limiting based on protocol and/or source/destination address and or source/destination port and allow these limits to be applied to services offered by the device.

Examples of this might include rate-limits that permit BGP traffic rates up to 100 megabits per second from an authorized peer, while dropping all other traffic addressed to the device which exceeds this limit.

Considerations.

None.

### 2.2.3. Ability to Rate Limit Traffic To the Device - Minimal Performance Degradation

Capability.

The device provides a means to rate-limit packets without significant performance degradation.

The device is able to apply rate-limits on ALL interfaces (up to the maximum number possible) simultaneously and with multiple rate-limits per interface (e.g., inbound, outbound, differing traffic classifications in either direction).

The rate-limiting of traffic destined to interfaces on the device, including the loopback interface, should not degrade performance significantly.

Supported Practices.

* This enables the implementation of rate-limits on whichever services are necessary. To the extent that rate-limiting causes degradation, it may not be possible to apply rate-limits that implement the appropriate policies.

Current Implementations.

Another way of stating the capability is that rate-limit performance should not be the limiting factor in device throughput. If a device is capable of forwarding 30Mb/sec without rate-limits, then it should be able to forward the same amount with rate-limits in place.

Considerations.

The definition of "significant" is subjective. At one end of the spectrum it might mean "the application of rate-limits may cause the box to crash". At the other end would be a throughput loss of less than one percent with tens of thousands of rate-limits applied. The level of performance degradation that is acceptable will have to be determined by the operator.

Repeatable test data showing rate-limiting performance impact would be very useful in evaluating this capability. Tests should include such information as packet size, packet rate, number of interfaces tested (source/destination), types of interfaces, routing table size, routing protocols in use, frequency of routing updates, etc.

### 2.2.4. Ability to Rate Limit To the Device - Specify Rate Limit Actions

Capability.

The device provides a mechanism to allow the specification of the action to be taken when a rate-limit rule matches. Actions include "permit" (allow the traffic), "reject" (drop with appropriate notification to sender), and "drop" (drop with no notification to sender).

Supported Practices.

* This capability is essential to the use of rate limits to enforce policy.

Current Implementations.

Assume that your management devices for deployed networking devices live on several subnets, use several protocols, and are controlled by several different parts of your organization. There might exist a reason to have disparate policies for access to the devices from these parts of the organization. Further you may want to limit traffic levels for these types of traffic from these known sources.

Actions such as "permit", "deny", "drop" are essential in defining the security policy for the services offered by the network devices.

Considerations.

While silently dropping traffic without sending notification may be the correct action in security terms, consideration should be given to operational implications. See [RFC3360] for consideration of potential problems caused by sending inappropriate TCP Resets.

### 2.2.5. Ability to Rate Limit To the Device - Log Rate Limit Actions

Capability.

It is possible to log rate limit actions. The logging capability is able to capture at least the following data:

* permit/deny/drop status
* source and destination IP address
* source and destination ports (if applicable to the protocol)
* which network element received the packet (interface, MAC address or other layer 2 information that identifies the previous hop source of the packet).

Supported Practices.

* Logging is essential for auditing, incident response, and operations

Current Implementations.

Actions such as "permit", "deny", "drop" are essential in defining the security policy for the services offered by the network devices. Auditing the frequency, sources and destinations of these attempts is essential for tracking ongoing issues today.

Considerations.

Logging can be burdensome to the network device, at no time should logging cause performance degradation to the device or services offered on the device.

### 2.2.6. Ability to Rate Limit To the Device - Specify Log Granularity

Capability.

It is possible to enable/disable logging on a per rule basis.

Supported Practices.

* The ability to tune the granularity of logging allows the operator to log the information that is desired and only the information that is desired. Without this capability, it is possible that extra data (or none at all) would be logged, making it more difficult to find relevant information.

Current Implementations.

If a rate limit is defined that has several rules, and one of the rules denies telnet (tcp/23) connections, then it should be possible to specify that only matches on the rule that denies telnet should generate a log message.

Considerations.

None.

### 2.2.7. Ability to Rate Limit To the Device - Ability to Rate Limit Protocols

Capability.

The device provides a means to rate limit traffic based on the value of the protocol field in the IP header.

Supported Practices.

* Being able to rate limit on protocol is necessary to allow implementation of policy, secure operations and for support of incident response.

Current Implementations.

Some denial of service attacks are based on the ability to flood the victim with ICMP traffic. One quick way (admittedly with some negative side effects) to mitigate the effects of such attacks is to rate limit all ICMP traffic headed toward the victim.

Considerations.

None.

### 2.2.8. Ability to Rate Limit To the Device - Ability to Rate Limit Addresses

Capability.

The function is able to control the flow of traffic based on source and/or destination IP address or blocks of addresses such as Classless Inter-Domain Routing (CIDR) blocks.

Supported Practices.

* The capability to rate limit on addresses and address blocks is a fundamental tool for establishing boundaries between different networks.

Current Implementations.

One example of the use of address based rate limits is to implement ingress filtering per [RFC2827].

Considerations.

None.

### 2.2.9. Ability to Rate Limit To the Device - Ability to Rate Limit Protocol Header Fields

Capability.

The rate limit mechanism supports rate limiting based on the value(s) of any portion of the protocol headers for IP, ICMP, UDP and TCP. It supports rate limiting of all other protocols supported at layer 3 and 4. It supports rate limiting based on the headers of higher level protocols. It is possible to specify fields by name (e.g., "protocol = ICMP") rather than bit- offset/ length/numeric value (e.g., 72:8 = 1).

Supported Practices.

* Being able to rate limit on portions of the header is necessary to allow implementation of policy, secure operations, and support incident response.

Current Implementations.

This capability implies that it is possible to rate limit based on TCP or UDP port numbers, TCP flags such as SYN, ACK and RST bits, and ICMP type and code fields. One common example is to reject "inbound" TCP connection attempts (TCP, SYN bit set+ACK bit clear

or SYN bit set+ACK,FIN and RST bits clear). Another common example is the ability to control what services are allowed in/out of a network. It may be desirable to only allow inbound connections on port 80 (HTTP) and 443 (HTTPS) to a network hosting web servers.

Considerations.

None.

### 2.2.10. Ability to Rate Limit To the Device - Ability to Rate Limit Inbound and Outbound

Capability.

It is possible to rate limit both incoming and outgoing traffic on any interface.

Supported Practices.

* This capability allows flexibility in applying rate limits at the place that makes the most sense. It allows invalid or malicious traffic to be dropped as close to the source as possible.

Current Implementations.

It might be desirable on a router to apply an egress rate limit to its external connections to limit outbound traffic that does not have a high priority. Inbound, it might be desirable to apply a rate limit to all traffic of a certain classification in order to preserve limited resources on the router's management components.

Considerations.

None.

### 2.2.11. Ability to Rate Limit To the Device - Ability to Accurately Count Rate Limit Hits

Capability.

The device supplies a facility for accurately counting all rate limit matches.

Supported Practices.

* Accurate counting of rate limit rule matches is important because it shows the frequency of attempts to violate policy. This enables resources to be focused on areas of greatest need.

Current Implementations.

Assume, for example, that a ISP network implements anti-spoofing egress rate limits (see [RFC2827]) on interfaces of its edge routers that support single-homed stub networks. Counters could enable the ISP to detect cases where large numbers of spoofed packets are being sent. This may indicate that the customer is performing potentially malicious actions (possibly in violation of the ISPs Acceptable Use Policy), or that system(s) on the customers network have been compromised by hackers and are being (mis)used to launch attacks.

Considerations.

None.

### 2.2.12. Ability to Rate Limit To the Device - Ability to Display Rate Limit Counters

Capability.

The device provides a mechanism to display rate limit counters.

Supported Practices.

* Information that is collected is not useful unless it can be displayed in a useful manner.

Current Implementations.

Assume there is a router with four interfaces. One is an up-link to an ISP providing routes to the Internet. The other three connect to separate internal networks. Assume that a host on one of the internal networks has been compromised by a hacker and is sending traffic with bogus source addresses. In such a situation, it might be desirable to apply ingress rate limits to each of the internal interfaces. Once the rate limits are in place, the counters can be examined to determine the source (inbound interface) of the bogus packets.

Considerations.

None.

### 2.2.13. Ability to Rate Limit To the Device - Ability to Display Rate Limit Counters per Rate Limit Application

Capability.

If it is possible for a rate limit to be applied more than once at the same time, then the device provides a mechanism to display rate limit counters per rate limit application.

Supported Practices.

* It may make sense to apply the same rate limit definition simultaneously more than one time (to different interfaces, etc.). If so, it would be much more useful to know which instance of a rate limit is matching than to know that some instance was matching somewhere.

Current Implementations.

One way to implement this capability would be to have the counter display mechanism show the interface (or other entity) to which the rate limit has been applied, along with the name (or other designator) for the rate limit. For example if a rate limit named "desktop_outbound" applied two different interfaces, say, "ethernet0" and "ethernet1", the display should indicate something like "matches of rate limit 'desktop_outbound' on ethernet0 ..." and "matches of rate limit 'desktop_outbound' on ethernet1 ..."

Considerations.

None.

### 2.2.14. Ability to Rate Limit To the Device - Ability to Reset Rate Limit Counters

Capability.

It is possible to reset counters to zero on a per rate limit basis.

For the purposes of this capability it would be acceptable for the system to maintain two counters: an "absolute counter", C[now], and a "reset" counter, C[reset]. The absolute counter would maintain counts that increase monotonically until they wrap or overflow the counter. The reset counter would receive a copy of the current value of the absolute counter when the reset function was issued for that counter. Functions that display or retrieve the counter could then display the delta (C[now] - C[reset]).

Supported Practices.

* This allows operators to get a current picture of the traffic matching particular rules/rate limit.

Current Implementations.

Assume that rate limit counters are being used to detect internal hosts that are infected with a new worm. Once it is believed that all infected hosts have been cleaned up and the worm removed, the next step would be to verify that. One way of doing so would be to reset the rate limit counters to zero and see if traffic indicative of the worm has ceased.

Considerations.

None.

### 2.2.15. Ability to Rate Limit To the Device - Rate Limit Counters are Accurate

Capability.

Rate limit counters are accurate. They reflect the actual number of matching packets since the last counter reset. Rate limit counters are be capable of holding up to 2^32 - 1 values without overflowing and should be capable of holding up to 2^64 - 1 values.

Supported Practices.

* Inaccurate data can not be relied on as the basis for action. Underreported data can conceal the magnitude of a problem.

Current Implementations.

If N packets matching a Rate limit are sent to/through a device, then the counter should show N matches.

Considerations.

None.

## 3. Functional Capabilities - Filtering transit traffic (data plane)

The capabilities in this section are intended to list testable, functional capabilities that are needed to operate devices securely. Focusing on filtering transit packets on devices, controlling the data plane.

### 3.1. Filtering THROUGH the Device

#### 3.1.1. Ability to Filter Traffic on All Interfaces THROUGH the Device

Capability.

The device provides a means to filter IP packets on any interface implementing IP that are transit packets.

Supported Practices.

* Profile Current Traffic ([I-D.practices] Section x.x.x)
* Block Malicious Packets (Section 5.2)
* Limit Sources of Management (Section 5.3)

Current Implementations.

Many devices currently implement access control lists or filters that allow filtering based on protocol and/or source/destination address and or source/destination port and allow these filters to be applied to interfaces.

Considerations.

None.

### 3.1.2. Ability to Filter Traffic Through the Device

Capability.

It is possible to apply the filtering mechanism to traffic that is flowing through the device via any of its interfaces - transit traffic.

Supported Practices.

* This allows the operator to apply filters that protect the networks supported by the device from attacks and unauthorized access.

Current Implementations.

Many devices currently implement access control lists or filters that allow filtering based on protocol and/or source/destination address and or source/destination port and allow these filters to be applied to interfaces of the device for the purposes of protecting the networks that connect to the device.

Examples of this might include filters that permit only HTTP from known good sources and SMTP and SSH from a known subset of the entire network, while dropping all other traffic.

Considerations.

None.

### 3.1.3. Ability to Filter Traffic Through the Device - Minimal Performance Degradation

Capability.

The device provides a means to filter packets without significant performance degradation. This specifically applies to stateless packet filtering operating on layer 3 (IP) and layer 4 (TCP or UDP) headers, as well as normal packet forwarding information such as incoming and outgoing interfaces.

The device is able to apply stateless packet filters on ALL interfaces (up to the maximum number possible) simultaneously and with multiple filters per interface (e.g., inbound and outbound).

The filtering of traffic through the device should not degrade performance significantly.

Supported Practices.

* This enables the implementation of filters on necessary services for the networks supported by the device. To the extent that filtering causes degradation, it may not be possible to apply filters that implement the appropriate policies.

Current Implementations.

Another way of stating the capability is that filter performance should not be the limiting factor in device throughput. If a device is capable of forwarding 30Mb/sec without filtering, then it should be able to forward the same amount with filtering in place.

Considerations.

The definition of "significant" is subjective. At one end of the spectrum it might mean "the application of filters may cause the box to crash". At the other end would be a throughput loss of less than one percent with tens of thousands of filters applied.

The level of performance degradation that is acceptable will have to be determined by the operator.

Repeatable test data showing filter performance impact would be very useful in evaluating this capability. Tests should include such information as packet size, packet rate, number of interfaces tested (source/destination), types of interfaces, routing table size, routing protocols in use, frequency of routing updates, etc.

This capability does not address stateful filtering, filtering above layer 4 headers or other more advanced types of filtering that may be important in certain operational environments.

### 3.1.4. Ability to Filter Through the Device - Specify Filter Actions

Capability.

The device provides a mechanism to allow the specification of the action to be taken when a filter rule matches. Actions include "permit" (allow the traffic), "reject" (drop with appropriate notification to sender), and "drop" (drop with no notification to sender).

Supported Practices.

* This capability is essential to the use of filters to enforce policy.

Current Implementations.

Assume that your network's services live on several subnets, use several protocols, and are controlled by several different parts of your organization. There might exist a reason to have disparate policies for access to the devices from these parts of the organization.

Actions such as "permit", "deny", "drop" are essential in defining the security policy for the services offered by the network devices.

Considerations.

While silently dropping traffic without sending notification may be the correct action in security terms, consideration should be given to operational implications. See [RFC3360] for consideration of potential problems caused by sending inappropriate TCP Resets.

### 3.1.5. Ability to Filter Through the Device - Log Filter Actions

Capability.

It is possible to log all filter actions. The logging capability is able to capture at least the following data:

* permit/deny/drop status
* source and destination IP address
* source and destination ports (if applicable to the protocol)
* which network element received the packet (interface, MAC address or other layer 2 information that identifies the previous hop source of the packet).

Supported Practices.

* Logging is essential for auditing, incident response, and operations

Current Implementations.

Actions such as "permit", "deny", "drop" are essential in defining the security policy for the services offered by the network devices. Auditing the frequency, sources and destinations of these attempts is essential for tracking ongoing issues today.

Considerations.

Logging can be burdensome to the network device, at no time should logging cause performance degradation to the device or services offered on the device.

### 3.1.6. Ability to Filter Through the Device - Specify Log Granularity

Capability.

It is possible to enable/disable logging on a per rule basis.

Supported Practices.

* The ability to tune the granularity of logging allows the operator to log the information that is desired and only the information that is desired. Without this capability, it is possible that extra data (or none at all) would be logged, making it more difficult to find relevant information.

Current Implementations.

If a filter is defined that has several rules, and one of the rules denies telnet (tcp/23) traffic, then it should be possible to specify that only matches on the rule that denies telnet should generate a log message.

Considerations.

None.

### 3.1.7. Ability to Filter Through the Device - Ability to Filter Protocols

Capability.

The device provides a means to filter traffic based on the value of the protocol field in the IP header.

Supported Practices.

* Being able to filter on protocol is necessary to allow implementation of policy, secure operations and for support of incident response.

Current Implementations.

Some denial of service attacks are based on the ability to flood the victim with ICMP traffic. One quick way (admittedly with some negative side effects) to mitigate the effects of such attacks is to drop all ICMP traffic headed toward the victim.

Considerations.

None.

### 3.1.8. Ability to Filter Through the Device - Ability to Filter Addresses

Capability.

The function is able to control the flow of traffic based on source and/or destination IP address or blocks of addresses such as Classless Inter-Domain Routing (CIDR) blocks.

Supported Practices.

* The capability to filter on addresses and address blocks is a fundamental tool for establishing boundaries between different networks.

Current Implementations.

One example of the use of address based filtering is to implement ingress filtering per [RFC2827].

Considerations.

None.

### 3.1.9. Ability to Filter Through the Device - Ability to Filter Protocol Header Fields

Capability.

The filtering mechanism supports filtering based on the value(s) of any portion of the protocol headers for IP, ICMP, UDP and TCP. It supports filtering of all other protocols supported at layer 3 and 4. It supports filtering based on the headers of higher level protocols. It is possible to specify fields by name (e.g., "protocol = ICMP") rather than bit- offset/length/numeric value (e.g., 72:8 = 1).

Supported Practices.

* Being able to filter on portions of the header is necessary to allow implementation of policy, secure operations, and support incident response.

Current Implementations.

This capability implies that it is possible to filter based on TCP or UDP port numbers, TCP flags such as SYN, ACK and RST bits, and ICMP type and code fields. One common example is to reject TCP connection attempts (TCP, SYN bit set+ACK bit clear or SYN bit set+ACK,FIN and RST bits clear). Another common example is the ability to control what services are allowed in/out of a network. It may be desirable to only allow inbound connections on port 80 (HTTP) and 443 (HTTPS) to a network hosting web servers.

Considerations.

None.

### 3.1.10. Ability to Filter Through the Device - Ability to Filter Inbound and Outbound

Capability.

It is possible to filter both incoming and outgoing traffic on any interface.

Supported Practices.

* This capability allows flexibility in applying filters at the place that makes the most sense. It allows invalid or malicious traffic to be dropped as close to the source as

possible.

Current Implementations.

It might be desirable on a border router, for example, to apply an egress filter outbound on the interface that connects a site to its external ISP to drop outbound traffic that does not have a valid internal source address. Inbound, it might be desirable to apply a filter that blocks all traffic from a site that is known to forward or originate lots of junk mail.

Considerations.

None.

### 3.1.11. Ability to Filter Through the Device - Ability to Accurately Count Filter Hits

Capability.

The device supplies a facility for accurately counting all filter matches.

Supported Practices.

* Accurate counting of filter rule matches is important because it shows the frequency of attempts to violate policy. This enables resources to be focused on areas of greatest need.

Current Implementations.

Assume, for example, that a ISP network implements anti-spoofing egress filters (see [RFC2827]) on interfaces of its edge routers that support single-homed stub networks. Counters could enable the ISP to detect cases where large numbers of spoofed packets are being sent. This may indicate that the customer is performing potentially malicious actions (possibly in violation of the ISPs Acceptable Use Policy), or that system(s) on the customers network have been "owned" by hackers and are being (mis)used to launch attacks.

Considerations.

None.

### 3.1.12. Ability to Filter Through the Device - Ability to Display Filter Counters

Capability.

The device provides a mechanism to display filter counters.

Supported Practices.

* Information that is collected is not useful unless it can be displayed in a useful manner.

Current Implementations.

Assume there is a router with four interfaces. One is an up-link to an ISP providing routes to the Internet. The other three connect to separate internal networks. Assume that a host on one of the internal networks has been compromised by a hacker and is sending traffic with bogus source addresses. In such a situation, it might be desirable to apply ingress filters to each of the internal interfaces. Once the filters are in place, the counters can be examined to determine the source (inbound interface) of the

bogus packets.

Considerations.

None.

### 3.1.13. Ability to Filter Through the Device - Ability to Display Filter Counters per Filter Application

Capability.

If it is possible for a filter to be applied more than once at the same time, then the device provides a mechanism to display filter counters per filter application.

Supported Practices.

* It may make sense to apply the same filter definition simultaneously more than one time (to different interfaces, etc.). If so, it would be much more useful to know which instance of a filter is matching than to know that some instance was matching somewhere.

Current Implementations.

One way to implement this capability would be to have the counter display mechanism show the interface (or other entity) to which the filter has been applied, along with the name (or other designator) for the filter. For example if a filter named "desktop_outbound" applied two different interfaces, say, "ethernet0" and "ethernet1", the display should indicate something like "matches of filter 'desktop_outbound' on ethernet0 ..." and "matches of filter 'desktop_outbound' on ethernet1 ..."

Considerations.

None.

### 3.1.14. Ability to Filter Through the Device - Ability to Reset Filter Counters

Capability.

It is possible to reset counters to zero on a per filter basis.

For the purposes of this capability it would be acceptable for the system to maintain two counters: an "absolute counter", C[now], and a "reset" counter, C[reset]. The absolute counter would maintain counts that increase monotonically until they wrap or overflow the counter. The reset counter would receive a copy of the current value of the absolute counter when the reset function was issued for that counter. Functions that display or retrieve the counter could then display the delta (C[now] - C[reset]).

Supported Practices.

* This allows operators to get a current picture of the traffic matching particular rules/filters.

Current Implementations.

Assume that filter counters are being used to detect internal hosts that are infected with a new worm. Once it is believed that all infected hosts have been cleaned up and the worm removed, the next step would be to verify that. One way of doing so would be to reset the filter counters to zero and see if traffic indicative of the worm has ceased.

Considerations.

None.

### 3.1.15. Ability to Filter Through the Device - Filter Counters are Accurate

Capability.

Filter counters are accurate. They reflect the actual number of matching packets since the last counter reset. Filter counters are be capable of holding up to 2^32 - 1 values without overflowing and should be capable of holding up to 2^64 - 1 values.

Supported Practices.

* Inaccurate data can not be relied on as the basis for action. Underreported data can conceal the magnitude of a problem.

Current Implementations.

If N packets matching a filter are sent to/through a device, then the counter should show N matches.

Considerations.

None.

## 3.2. Rate Limit THROUGH the Device

### 3.2.1. Ability to Rate limit Traffic on All Interfaces THROUGH the Device

Capability.

The device provides a means to rate limit IP packets on any interface implementing IP that are transit packets.

Supported Practices.

* Profile Current Traffic ([I-D.practices] Section x.x.x)
* Block Malicious Packets (Section 5.2)
* Limit Sources of Management (Section 5.3)

Current Implementations.

Many devices currently implement rate limits that allow rate limiting based on protocol and/or source/destination address and or source/destination port or raw bandwidth and allow these limits to be applied to interfaces.

Considerations.

None.

### 3.2.2. Ability to Rate Limit Traffic Through the Device

Capability.

It is possible to apply the rate-limiting mechanism to traffic that is transiting the device via any of its interfaces.

Supported Practices.

* This allows the operator to apply rate-limits that protect the networks transiting the device from attacks and unauthorized access.

Current Implementations.

Many devices currently implement rate-limits that allow limiting based on protocol and/or source/destination address and or source/destination port and allow these limits to be applied to services offered by the networks which transit the device.

Examples of this might include rate-limits that permit SSH traffic rates up to 100 megabits per second from an authorized peer, while dropping all other traffic addressed to the network which exceeds this limit.

Considerations.

None.

### 3.2.3. Ability to Rate Limit Traffic Through the Device - Minimal Performance Degradation

Capability.

The device provides a means to rate-limit packets without significant performance degradation.

The device is able to apply rate-limits on ALL interfaces (up to the maximum number possible) simultaneously and with multiple rate-limits per interface (e.g., inbound, outbound, differing traffic classifications in either direction).

The rate-limiting of traffic destined to networks transiting the device should not degrade performance significantly.

Supported Practices.

* This enables the implementation of rate-limits on whichever services are necessary. To the extent that rate-limiting causes degradation, it may not be possible to apply rate-limits that implement the appropriate policies.

Current Implementations.

Another way of stating the capability is that rate-limit performance should not be the limiting factor in device throughput. If a device is capable of forwarding 30Mb/sec without rate-limits, then it should be able to forward the same amount with rate-limits in place.

Considerations.

The definition of "significant" is subjective. At one end of the spectrum it might mean "the application of rate-limits may cause the box to crash". At the other end would be a throughput loss of less than one percent with tens of thousands of rate-limits applied. The level of performance degradation that is acceptable will have to be determined by the operator.

Repeatable test data showing rate-limiting performance impact would be very useful in evaluating this capability. Tests should include such information as packet size, packet rate, number of interfaces tested (source/destination), types of interfaces, routing table size, routing protocols in use, frequency of routing updates, etc.

### 3.2.4. Ability to Rate Limit Through the Device - Specify Rate Limit Actions

Capability.

The device provides a mechanism to allow the specification of the action to be taken when a rate-limit rule matches. Actions include "permit" (allow the traffic), "reject" (drop with appropriate notification to sender), and "drop" (drop with no notification to sender).

Supported Practices.

* This capability is essential to the use of rate limits to enforce policy.

Current Implementations.

Assume that your management devices for deployed networking devices live on several subnets, use several protocols, and are controlled by several different parts of your organization. There might exist a reason to have disparate policies for access to the devices from these parts of the organization. Further you may want to limit traffic levels for these types of traffic from these known sources as close to the sources as possible via interface rate limits implemented on the supporting network devices for those source networks.

Actions such as "permit", "deny", "drop" are essential in defining the security policy for the services offered by the network devices.

Considerations.

While silently dropping traffic without sending notification may be the correct action in security terms, consideration should be given to operational implications. See [RFC3360] for consideration of potential problems caused by sending inappropriate TCP Resets.

### 3.2.5. Ability to Rate Limit Through the Device - Log Rate Limit Actions

Capability.

It is possible to log rate limit actions. The logging capability is able to capture at least the following data:

* permit/deny/drop status
* source and destination IP address
* source and destination ports (if applicable to the protocol)
* which network element received the packet (interface, MAC address or other layer 2 information that identifies the previous hop source of the packet).

Supported Practices.

* Logging is essential for auditing, incident response, and operations

Current Implementations.

Actions such as "permit", "deny", "drop" are essential in defining the security policy for the services offered by the network devices. Auditing the frequency, sources and destinations of these attempts is essential for tracking ongoing issues today.

Considerations.

Logging can be burdensome to the network device, at no time should logging cause performance degradation to the device or services offered on the device.

### 3.2.6. Ability to Rate Limit Through the Device - Specify Log Granularity

Capability.

It is possible to enable/disable logging on a per rule basis.

Supported Practices.

* The ability to tune the granularity of logging allows the operator to log the information that is desired and only the information that is desired. Without this capability, it is possible that extra data (or none at all) would be logged, making it more difficult to find relevant information.

Current Implementations.

If a rate limit is defined that has several rules, and one of the rules denies telnet (tcp/23) connections, then it should be possible to specify that only matches on the rule that denies telnet should generate a log message.

Considerations.

None.

### 3.2.7. Ability to Rate Limit Through the Device - Ability to Rate Limit Protocols

Capability.

The device provides a means to rate limit traffic based on the value of the protocol field in the IP header.

Supported Practices.

* Being able to rate limit on protocol is necessary to allow implementation of policy, secure operations and for support of incident response.

Current Implementations.

Some denial of service attacks are based on the ability to flood the victim with ICMP traffic. One quick way (admittedly with some negative side effects) to mitigate the effects of such attacks is to rate limit all ICMP traffic headed toward the victim.

Considerations.

None.

### 3.2.8. Ability to Rate Limit Through the Device - Ability to Rate Limit Addresses

Capability.

The function is able to control the flow of traffic based on source and/or destination IP address or blocks of addresses such as Classless Inter-Domain Routing (CIDR) blocks.

Supported Practices.

* The capability to rate limit on addresses and address blocks is a fundamental tool for establishing boundaries between different networks.

Current Implementations.

One example of the use of address based rate limits is to implement ingress filtering per [RFC2827].

Considerations.

None.

### 3.2.9. Ability to Rate Limit Through the Device - Ability to Rate Limit Protocol Header Fields

Capability.

The rate limit mechanism supports rate limiting based on the value(s) of any portion of the protocol headers for IP, ICMP, UDP and TCP. It supports rate limiting of all other protocols supported at layer 3 and 4. It supports rate limiting based on the headers of higher level protocols. It is possible to specify fields by name (e.g., "protocol = ICMP") rather than bit- offset/ length/numeric value (e.g., 72:8 = 1).

Supported Practices.

* Being able to rate limit on portions of the header is necessary to allow implementation of policy, secure operations, and support incident response.

Current Implementations.

This capability implies that it is possible to rate limit based on TCP or UDP port numbers, TCP flags such as SYN, ACK and RST bits, and ICMP type and code fields. One common example is to reject "inbound" TCP connection attempts (TCP, SYN bit set+ACK bit clear or SYN bit set+ACK,FIN and RST bits clear). Another common example is the ability to control what services are allowed in/out of a network. It may be desirable to only allow inbound connections on port 80 (HTTP) and 443 (HTTPS) to a network hosting web servers.

Considerations.

None.

### 3.2.10. Ability to Rate Limit Through the Device - Ability to Rate Limit Inbound and Outbound

Capability.

It is possible to rate limit both incoming and outgoing traffic on any interface to or from any transiting network.

Supported Practices.

* This capability allows flexibility in applying rate limits at the place that makes the most sense. It allows invalid or malicious traffic to be dropped as close to the source as possible.

Current Implementations.

It might be desirable on a router to apply an egress rate limit to its external connections to limit outbound traffic that does not have a high priority. Inbound, it might be desirable to apply a rate limit to all traffic of a certain classification in order to preserve limited resources on the sported networks behind the device.

Considerations.

None.

### 3.2.11. Ability to Rate Limit Through the Device - Ability to Accurately Count Rate Limit Hits

Capability.

The device supplies a facility for accurately counting all rate limit matches.

Supported Practices.

* Accurate counting of rate limit rule matches is important because it shows the frequency of attempts to violate policy. This enables resources to be focused on areas of greatest need.

Current Implementations.

Assume, for example, that a ISP network implements anti-spoofing egress rate limits (see [RFC2827]) on interfaces of its edge routers that support single-homed stub networks. Counters could enable the ISP to detect cases where large numbers of spoofed packets are being sent. This may indicate that the customer is performing potentially malicious actions (possibly in violation of the ISPs Acceptable Use Policy), or that system(s) on the

customers network have been compromised by hackers and are being (mis)used to launch attacks.

Considerations.

None.

### 3.2.12. Ability to Rate Limit Through the Device - Ability to Display Rate Limit Counters

Capability.

The device provides a mechanism to display rate limit counters.

Supported Practices.

* Information that is collected is not useful unless it can be displayed in a useful manner.

Current Implementations.

Assume there is a router with four interfaces. One is an up-link to an ISP providing routes to the Internet. The other three connect to separate internal networks. Assume that a host on one of the internal networks has been compromised by a hacker and is sending traffic with bogus source addresses. In such a situation, it might be desirable to apply ingress rate limits to each of the internal interfaces. Once the rate limits are in place, the counters can be examined to determine the source (inbound interface) of the bogus packets.

Considerations.

None.

### 3.2.13. Ability to Rate Limit Through the Device - Ability to Display Rate Limit Counters per Rate Limit Application

Capability.

If it is possible for a rate limit to be applied more than once at the same time, then the device provides a mechanism to display rate limit counters per rate limit application.

Supported Practices.

* It may make sense to apply the same rate limit definition simultaneously more than one time (to different interfaces, etc.). If so, it would be much more useful to know which instance of a rate limit is matching than to know that some instance was matching somewhere.

Current Implementations.

One way to implement this capability would be to have the counter display mechanism show the interface (or other entity) to which the rate limit has been applied, along with the name (or other designator) for the rate limit. For example if a rate limit named "desktop_outbound" applied two different interfaces, say, "ethernet0" and "ethernet1", the display should indicate something like "matches of rate limit 'desktop_outbound' on ethernet0 ..." and "matches of rate limit 'desktop_outbound' on ethernet1 ..."

Considerations.

None.

### 3.2.14. Ability to Rate Limit Through the Device - Ability to Reset Rate Limit Counters

Capability.

It is possible to reset counters to zero on a per rate limit basis.

For the purposes of this capability it would be acceptable for the system to maintain two counters: an "absolute counter", C[now], and a "reset" counter, C[reset]. The absolute counter would maintain counts that increase monotonically until they wrap or overflow the counter. The reset counter would receive a copy of the current value of the absolute counter when the reset function was issued for that counter. Functions that display or retrieve the counter could then display the delta (C[now] - C[reset]).

Supported Practices.

* This allows operators to get a current picture of the traffic matching particular rules/rate limit.

Current Implementations.

Assume that rate limit counters are being used to detect internal hosts that are infected with a new worm. Once it is believed that all infected hosts have been cleaned up and the worm removed, the next step would be to verify that. One way of doing so would be to reset the rate limit counters to zero and see if traffic indicative of the worm has ceased.

Considerations.

None.

### 3.2.15. Ability to Rate Limit Through the Device - Rate Limit Counters are Accurate

Capability.

Rate limit counters are accurate. They reflect the actual number of matching packets since the last counter reset. Rate limit counters are be capable of holding up to 2^32 - 1 values without overflowing and should be capable of holding up to 2^64 - 1 values.

Supported Practices.

* Inaccurate data can not be relied on as the basis for action. Underreported data can conceal the magnitude of a problem.

Current Implementations.

If N packets matching a Rate limit are sent to/through a device, then the counter should show N matches.

Considerations.

None.

## 4. Functional Capabilities - Filtering Layer 2 Attributes

The capabilities in this section are intended to list testable, functional capabilities that are needed to operate devices securely.

A layer-2 domain permits all devices in the domain to establish communication at layer-2. Devices thus connected have an implicit trust relationship among themselves. If there are devices in a layer-2 domain which are at different trust levels, we may want to filter traffic between such devices based on the trust levels or any other fields in the layer-2 header. The following filtering capabilities are required at layer-2.

### 4.1. Filtering Layer 2

#### 4.1.1. Ability to partition layer-2 network to provide different levels of security

Capability.

The device provides a means to partition the physical layer-2 domain into multiple virtual domains, thus allowing the filtering of unwarranted traffic.

Supported Practices.

* Being able to partition a layer-2 domain provides the same level of security within a layer-2 domain as can be guaranteed if they were different layer-2 domains.

Current Implementations.

Most Ethernet networks use the concept of VLAN [8021Q] to partition a layer-2 broadcast domain. Private VLAN's [PVLAN] allow further partitioning of VLANs's into smaller domains.

Considerations.

Not all layer-2 network technologies may lend themselves to virtual partitioning.

### 4.1.2. Ability to restrict access to specified hardware (MAC) addresses

Capability.

The device provides a means to filter traffic based on the source and/ or destination hardware address.

Supported Practices.

* Being able to filter on hardware Address provides an ability to block frames between devices in the same layer-2 domain to communicate.

Current Implementations.

The ability to filter and monitor traffic in layer-2 allows for security at layer-2 itself, between devices on the same network. Allowing filtering based on hardware address allows a simple filtering interface to the administrator to apply simple policy rules.

Considerations.

Different Link layer technologies use different addressing mechanisms.

### 4.1.3. Ability to restrict based on layer-2 packet type [etherType] field

Capability.

The device provides a means to filter packets based on the packet type field in the layer-2 header.

Supported Practices.

* Being able to filter on packets based on the packet type field helps in preventing packets not understandable by a particular device to be processed by it.

Current Implementations.

The ability to filter and monitor traffic in layer-2 allows for security at layer-2 itself. This capability can also prevent IPX packets to inadvertently be sent to an IP device and vice versa.

Considerations.

None.

# 5. Additional Operational Practices

This section describes practices not covered in [I-D.practices]. They are included here to provide justification for capabilities that reference them.

## 5.1. Profile Current Traffic

Discuss practice. Use same format as [I-D.practices].

## 5.2. Block Malicious Packets

Discuss practice. Use same format as [I-D.practices].

## 5.3. Limit Sources of Management

Discuss practice. Use same format as [I-D.practices].

## 6. Security Considerations

General

Security is the subject matter of this entire memo. The capabilities listed cite practices in [I-D.practices] that they are intended to support. [I-D.practices] defines the threat model, practices and lists justifications for each practice.

# 7. References

## 7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

## 7.2. Non-normative References

[8021Q] "802.1Q - Virtual LANs", IEEE Standard 802.1Q - Virtual LANs, August 2001.

[I-D.practices] Kaeo, M., "Operational Security Current Practices", Internet-Draft (to be published) draft-ietf-opsec-current-practices-00, February 2005.

[PVLAN] HomChaudhuri, S. and M. Foschiano, "Private VLANs: Addressing VLAN scalability and security issues in a multi-client environment", Internet Draft (to be published) draft-sanjib-private-vlan-02, June 2004.

[RFC2828] Shirey, R., "Internet Security Glossary", RFC rfc2828.txt, May 2000.

[RFC3631] Bellovin, S. and J. Schiller, "Security Mechanisms for the Internet", RFC 3631, December 2003.

[RFC3871] Jones, G., "Operational Security Requirements for Large Internet Service Provider (ISP) IP Network Infrastructure", RFC 3871, September 2004.

## Appendix A. Acknowledgments

The editors gratefully acknowledges the contributions of:

- xxx
- yyy
- The MITRE Corporation for supporting development of this document. NOTE: The editor's affiliation with The MITRE Corporation is provided for identification purposes only, and is not intended to convey or imply MITRE's concurrence with, or support for, the positions, opinions or viewpoints expressed by the editor.
- Others who have provided significant feedback are: zzz
- This listing is intended to acknowledge contributions, not to imply that the individual or organizations approve the content of this document.
- Apologies to those who commented on/contributed to the document and were not listed.

Authors' Addresses

Christopher L. Morrow
UUNET Technologies
21830 UUNet Way
Ashburn, Virginia 21047
U.S.A.

Phone: +1 703 886 3823
Email: chris@uu.net

George M. Jones
The MITRE Corporation
7515 Colshire Drive, M/S WEST
McLean, Virginia 22102-7508
U.S.A.

Phone: +1 703 488 9740
Email: gmjones@mitre.org

Vishwas Manral
IPInfusion,
Bangalore,
India

Phone: +91-98456-61911
Email: vishwas@ipinfusion.com

## Intellectual Property Statement



## Disclaimer of Validity



## Copyright Statement



## Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.