



Defining Static Filters

This chapter covers these topics:

- [Introduction to Ascend filters](#)
- [Defining packet filters](#)
- [Applying packet filters](#)
- [Predefined filters](#)

Introduction to Ascend filters

A packet filter contains rules describing packets and what to do when those packets are encountered. When you apply a packet filter to an interface, the MAX monitors the data stream on that interface and takes a specified action when packet contents match the filter rules. Depending on the filter definition, it may apply to inbound or outbound packets, or both. In addition, filter rules are flexible enough to take an action (such as forward or drop) on those packets that match the rules, or all packets *except* those that match the rules.

Note: The MAX ships with three predefined filters. Many sites use these filters as is or add rules pertinent to their networks. See [Predefined filters](#).

Packet filters and firewalls

The MAX supports these types of *static* packet filters:

- Generic filters, which examine the byte- or bit-level contents of any packet.

Generic filters focus on certain bytes or bits in a packet and compare the contents of that location with a value defined in the filter. To use generic filters effectively, you need to know the contents of certain bytes in the packets you wish to filter. Protocol specifications are usually the best source of such information.

- IP filters, which examine higher-level fields specific to IP packets.

IP filters focus on known fields in IP packets, such as source or destination address, protocol number, and so forth. They operate on logical information, which is relatively easy to obtain.

- IPX filters, which examine higher-level fields specific to IPX packets.

IPX filters focus on known fields in IPX packets, such as source or destination address, node, socket, and so forth. They operate on logical information, which is relatively easy to obtain.

The MAX also supports Secure Access, which provides *dynamic* firewalls. Firewalls differ from filters in that they alter their behavior as traffic passes through them, where filters remain unchanged through their lifetimes. Unlike the static packet filters, which have a limited number of rules, router memory is the only limitation in Secure Access firewalls.

If your MAX unit has Secure Access support installed, see the *Ascend Secure Access User's Guide* (part number 7820-0429-001) for complete instructions on creating and applying firewalls. You can refer to a firewall set up in SAM in a RADIUS user profile, so that the firewall is applied for the connection defined in the user profile. For more information, see the *MAX RADIUS Configuration Guide*.

Ways to apply packet filters to an interface

After you define a packet filter, you apply it to an interface to monitor packets crossing that interface. You can apply the filter as one of the following:

- A data filter, to define which packets can or cannot cross the interface
- A call filter, to define which packets can or cannot bring up a connection or reset the idle-timer for an established connection (WAN interfaces only)

Packets can pass through both a data filter and call filter on a WAN interface. If you apply both a data and call filter, the data filter comes first.

Data filters for dropping or forwarding certain packets

Data filters are commonly used for security, but they can apply to any purpose that requires the MAX to drop or forward only specific packets. For example, you can use data filters to drop packets addressed to particular hosts or to prevent broadcasts from going across the WAN. You can also use data filters to allow users to access only specific devices across the WAN.

When you apply a data filter, its forwarding action (forward or drop) affects the actual data stream by preventing certain packets from reaching the Ethernet from the WAN, or vice versa. Data filters do not affect the idle timer, and a data filter applied to a Connection profile does not affect the answering process ([Figure 7-1](#)).

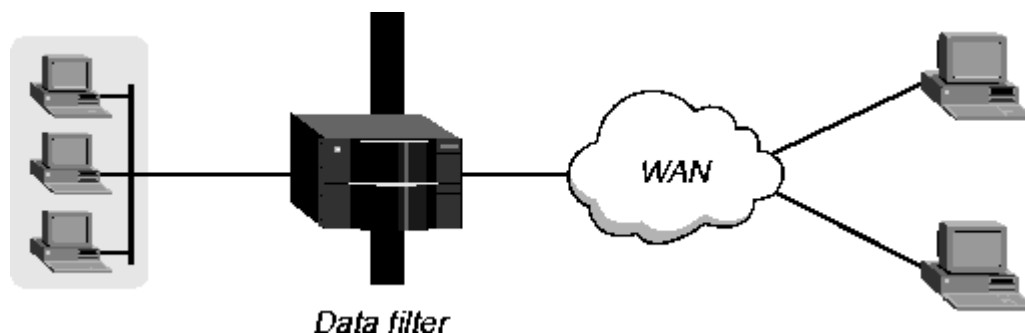


Figure 7-1. Data filters can drop or forward certain packets

Call filters for managing connections

A call filter defines which packets can or cannot bring up a connection or reset the idle timer for an established link ([Figure 7-2](#)).

Call filters prevent unnecessary connections and help the MAX distinguish active traffic from *noise*. By default, any traffic to a remote site triggers a call, and any traffic across an active connection resets the connection's idle timer.

When you apply a call filter, its forwarding action (forward or drop) does not affect which packets the MAX sends across an active connection. The forwarding action of a call filter determines which packets can either initiate a connection or reset a session's timer. When a session's idle-timer expires, the session terminates. The default for the idle timer is 120 seconds, so if a connection is inactive for two minutes, the MAX terminates the connection.

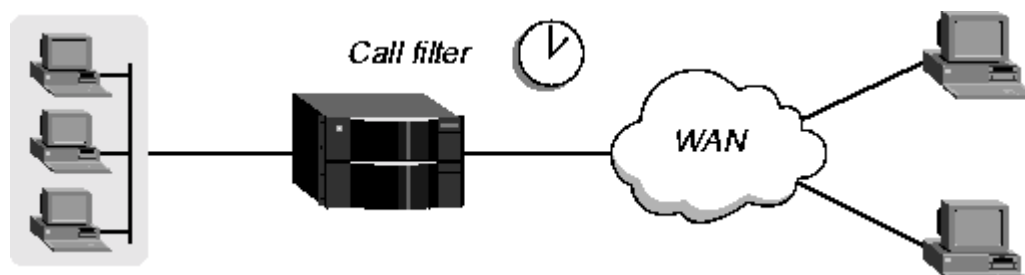


Figure 7-2. Call filters can prevent certain packets from resetting the timer

How packet filters work

This section provides an overview of packet filters and the processes they follow. For more details on filter matching a value in a packet, see [Understanding the packet filter parameters](#).

A Filter profile can contain up to 12 input and output filter specifications (rules). Each rule has its own forwarding action-forward or drop. A match occurs at the first successful comparison between a filter and the packet being examined. When a comparison succeeds, the filtering process stops and the forward action in that rule is applied to the packet.

If no comparisons succeed, the packet does not match this filter. However, this does not mean that the MAX forwards the packet. When no filter is in use, the MAX forwards all packets, but once you apply a filter to an interface, this default is *reversed*. For security purposes, the MAX does not automatically forward non-matching packets. It requires a rule that explicitly allows those packets to pass. For an example of an input filter that forwards all packets that did not match a previous rule, see [Defining a filter to prevent IP address spoofing](#).

Note: For a call filter to prevent an interface from remaining active unnecessarily, you must define rules for both input and output packets. Otherwise, if only input rules are defined, output packets will keep a connection active, or vice versa.

In a generic filter, all parameter settings in a rule work together to specify a location in a packet and a number to be compared to that location. The Compare parameter specifies whether a comparison succeeds when the contents of the packet equals or do not equal that number.

In an IP filter, a set of distinct comparisons are made in order. When a comparison fails, the MAX

allows a packet to go on to the next comparison. When a comparison succeeds, the filtering process stops and the MAX applies the forward action in that rule to the packet. The IP filter tests proceed in this order:

1. Compare source address parameters to the source address of the packet. If they are not equal, the comparison fails.
2. Compare destination address parameters to the destination address in the packet. If they are not equal, the comparison fails.
3. If the protocol parameter is zero (which matches any protocol), the comparison succeeds. If it is non-zero and not equal to the protocol field in the packet, the comparison fails.
4. If the Src Port Cmp parameter is not set to none, compare the source port parameter to the source port of the packet. If they do not match as specified in the Src-Port-Cmp parameter, the comparison fails.
5. If the Dst Port Cmp parameter is not set to none, compare the destination port parameter to the destination port of the packet. If they do not match as specified in the Dst-Port-Cmp parameter, the comparison fails.
6. If TCP Estab is Yes and the protocol number is 6, the comparison succeeds.

Defining packet filters

Filter profiles provide rules for defining which packets will be affected. The rules are the same for Input or Output filters. These are the filter parameters:

```

Ethernet
  Filters
    Name=filter-name
    Input filters...
      In filter 01-12
        Valid=Yes
        Type=GENERIC
        Generic...
          Forward=No
          Offset=14
          Length=8
          Mask=ffffffffffffffff
          Value=aaaa0300000080f3
          Compare=Equals
          More=No
      Ip...
        Forward=No
        Src Mask=255.255.255.192
        Src Adrs=192.100.50.128
        Dst Mask=0.0.0.0
        Dst Adrs=0.0.0.0
        Protocol=0
        Src Port Cmp=None
        Src Port #=N/A
        Dst Port Cmp=None
        Dst Port #=N/A
        TCP Estab=N/A
      Ipx...
        Forward=No
        Src Network Adrs=cfff0000
  
```

```

                                Dst Network Adrs=cf088888
                                Src Node Adrs=111222333
                                Dst Node Adrs=aaabbbccc
                                Src Socket Cmp=equal
                                Src Socket #=0451
                                Dst Socket Cmp=equal
                                Dst Socket #=0015

Output filters...
  Out filter 01-12
    Valid=Yes
    Type=GENERIC
    Generic...
      Forward=No
      Offset=14
      Length=8
      Mask=ffffffffffffffff
      Value=aaaa0300000080f3
      Compare=Equals
      More=No
    Ip...
      Forward=No
      Src Mask=255.255.255.192
      Src Adrs=192.100.50.128
      Dst Mask=0.0.0.0
      Dst Adrs=0.0.0.0
      Protocol=0
      Src Port Cmp=None
      Src Port #=N/A
      Dst Port Cmp=None
      Dst Port #=N/A
      TCP Estab=N/A
    Ipx...
      Forward=No
      Src Network Adrs=cfff0000
      Dst Network Adrs=cf088888
      Src Node Adrs=111222333
      Dst Node Adrs=aaabbbccc
      Src Socket Cmp=equal
      Src Socket #=0451
      Dst Socket Cmp=equal
      Dst Socket #=0015

```

Note that the parameters for defining the actual packet conditions are identical for Input and Output filters. For more information about each parameter, see the *MAX Reference Guide*.

Understanding the packet filter parameters

This section provides some background information on configuring packet filters.

Assigning a name to the Filter profile

Each filter must be assigned a name so it can be referenced from other profiles. The names of defined filters appear in the main Filters menu.

Input and Output filters

Each filter can contain up to 12 Input filters and Output filters, each defined individually and applied in order (1-12) to the packet stream. The MAX applies Input filters to inbound packets and Output filters to outbound packets.

Enabling a specific In or Out filter

Valid enables or disables the current In or Out filter. When you deactivate a filter, all of its parameters do not apply. (You cannot configure the filter until you enable it.)

Specifying a generic or IP filter type

Set Type to GENERIC or IP. Only the parameters in the corresponding subprofile (Generic or Ip) are applicable.

Generic filter rules

Generic filters can affect any packet, regardless of its protocol type or header fields. They use these parameters:

```
Generic...
  Forward=No
  Offset=14
  Length=8
  Mask=fffffffffffffffffff
  Value=aaaa0300000080f3
  Compare=Equals
  More=No
```

This section provides some background information on how these parameters work together.

Defining the action to take when a packet matches the filter

Forward specifies whether the MAX discards or forwards packets that match the filter specification. When no filters are in use, the MAX forwards all packets by default. When a filter is in use, the default, Forward = No, discards matching packets.

Specifying an offset to the bytes in a packet to be examined

Offset specifies a byte-offset from the start of a frame to the data in the packet to be tested against this filter. For example, with this filter specification:

```
Generic...
  Forward=No
  Offset=2
  Length=8
  Mask=0F FF FF FF 00 00 00 F0
  Value=07 FE 45 70 00 00 00 90
  Compare=Equals
  More=No
```

and the following packet contents:

```
2A 31 97 FE 45 70 12 22 33 99 B4 80 75
```

The first two bytes in the packet (2A and 31) are ignored due to the two-byte offset.

Note: If the MAX links the current filter to the previous one (if More=Yes in the previous filter), the offset starts at the endpoint of the previous segment.

Specifying the number of bytes to test

Length specifies the number of bytes to test in a frame, starting at the specified Offset. The MAX compares the contents of those bytes to the value specified in the filter's Value parameter. For example, with this specification:

```
Generic...
  Forward=No
  Offset=2
  Length=8
  Mask=0F FF FF FF 00 00 00 F0
  Value=07 FE 45 70 00 00 00 90
  Compare=Equals
  More=No
```

and the following packet contents:

```
2A 31 97 FE 45 70 12 22 33 99 B4 80 75
```

The filter applies the mask only to the eight bytes following the two-byte offset.

Masking the value before comparison

Mask is a 16-bit mask to apply to the Value before comparing it to the packet contents at the specified offset. You can use it to fine-tune exactly which bits you want to compare.

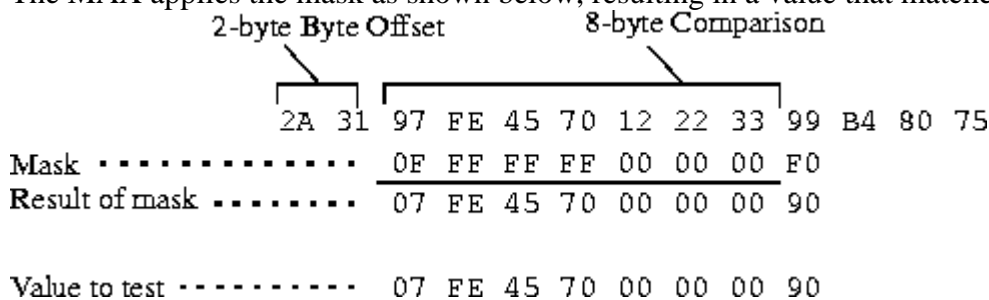
The MAX applies the mask to the specified value using a logical *AND* after the mask and value are both translated into binary format. The mask hides the bits that appear behind each binary 0 (zero) in the mask. A mask of all ones (FF FF FF FF FF FF FF FF) masks no bits, so the full Compare To value must match the packet contents. For example, with this filter specification:

```
Generic...
  Forward=No
  Offset=2
  Length=8
  Mask=0F FF FF FF 00 00 00 F0
  Value=07 FE 45 70 00 00 00 90
  Compare=Equals
  More=No
```

and the following packet contents:

```
2A 31 97 FE 45 70 12 22 33 99 B4 80 75
```

The MAX applies the mask as shown below, resulting in a value that matches the Value.



The packet matches this filter. The Filter Action is "Discard", so the MAX drops the packet. The byte comparison works as follows:

- 2A and 31 are ignored due to the two-byte offset.
- 9 in the lower half of the third byte is ignored, because the mask has a 0 in its place. The 7 in the third byte matches the value parameter's 7 in the upper half of that byte.
- F and E in the fourth byte match the value parameter for that byte.
- 4 and 5 in the fifth byte match the value parameter for that byte.
- 7 and 0 in the sixth byte match the value parameter for that byte.
- 12 and 22 and 33 in the seventh, eighth and ninth bytes are ignored because the mask has a 0 in those places.
- 9 in the tenth byte equals the matches the value parameter's 9 in the lower half of that byte. The second 9 in the upper-half of the packet's tenth byte is ignored because the mask has a 0 in its place.

The value to match up in the packet contents

Value specifies a hexadecimal number to be compared to specific bits contained in packets after the Offset, Length, and Mask calculations have been applied.

The type of comparison to be performed when matching the packet

Compare specifies the type of comparison to make between the specified value and the packet's contents: less than, equal, greater than, or not equal.

Linking the filter to the next In filter or Out filter in sequence

More specifies whether the MAX includes the next filter condition before determining whether the frame matches the filter. If checked, the MAX links the current filter condition to the one immediately following it, so the filter can examine multiple non-contiguous bytes within a packet. In effect, this parameter *marries* the current filter to the next one, so that the MAX applies the next filter before the MAX decides the forwarding decision. The match occurs only if *both* non-contiguous bytes contain the specified values. The next filter must be enabled; otherwise, the MAX ignores the filter.

IP filter rules

IP filter rules affect only IP and related packets. IP filters use these parameters:

```
Ip...
  Forward=No
  Src Mask=255.255.255.192
  Src Adrs=192.100.50.128
  Dst Mask=0.0.0.0
  Dst Adrs=0.0.0.0
  Protocol=0
  Src Port Cmp=None
  Src Port #=N/A
  Dst Port Cmp=None
  Dst Port #=N/A
  TCP Estab=N/A
```

This section provides some background information on how these parameters work.

Defining what action to take when a packet matches the filter

Forward specifies whether the MAX discards or forwards packets that match the filter specification. When no filters are in use, the MAX forwards all packets by default. When a filter is in use, the default discards matching packets.

Specifying which part of the source IP address to use for comparison

Src Mask specifies a mask to apply to the Src Adrs value before comparing it to the source address in a packet. You can use it to mask out the host portion of an address, for example, or the host and subnet portion.

The MAX applies the mask to the address using a logical *AND* after the mask and address are both translated into binary format. The mask hides the portion of the address that appears behind each binary 0 (zero) in the mask. A mask of all zeros (the default) masks all bits, so all source addresses match. A mask of all ones (255.255.255.255) masks no bits, so the full source address from a single host is matched.

Filtering the packet's source IP address

This parameter specifies a source IP address. After you modify this value by applying the specified Src Mask, the MAX compares it to a packet's source address.

Specifying which part of the destination IP address to use for comparison

Dst Mask specifies a mask to apply to the Dst Adrs before comparing it to the destination address in a packet. You can use it to mask out the host portion of an address, for example, or the host and subnet portion. The MAX applies the mask to the address using a logical *AND* after the mask and address are both translated into binary format. The mask hides the portion of the address that appears behind each binary 0 (zero) in the mask. A mask of all zeros (the default) masks all bits, so all destination addresses are matched. A mask of all ones (255.255.255.255) masks no bits, so the full destination address to a single host is matched.

Filtering on the packet's destination IP address

Dst Adrs specifies a destination IP address. After you modify this value by applying the specified Mask, the MAX compares it to a packet's destination address.

Filtering on the protocol number field in IP packets

If you specify a protocol number, the MAX compares it to the protocol number field in packets to match them to this filter. The default protocol number of zero matches all protocols. A list of common protocols appears below. For a complete list of protocol numbers, see the section on Well-Known Port Numbers in RFC 1700, *Assigned Numbers*, by Reynolds, J. and Postel, J., October 1994.

- 1: ICMP
- 5: STREAM
- 8: EGP

- 6: TCP
- 9: Any private interior gateway protocol (such as Cisco's IGRP)
- 11: Network Voice Protocol
- 17: UDP
- 20: Host Monitoring Protocol
- 22: XNS IDP
- 27: Reliable Data Protocol
- 28: Internet Reliable Transport Protocol
- 29: ISO Transport Protocol Class 4
- 30: Bulk Data Transfer Protocol
- 61: Any Host Internal Protocol
- 89: OSPF

Filtering on source port numbers

Src Port # specifies a value to compare with the source port number in a packet. The default setting (zero) indicates that the MAX disregards the source port in this filter. Port 25 is reserved for SMTP; that socket is dedicated to receiving mail messages. Port 20 is reserved for FTP data messages, port 21 for FTP control sessions, and port 23 for telnet.

The Src Port Cmp parameter specifies the type of comparison to be made.

Filtering on destination port numbers

Dst Port # specifies a value to compare with the destination port number in a packet. The default setting (zero) indicates that the MAX disregards the destination port in this filter. Port 25 is reserved for SMTP; that socket is dedicated to receiving mail messages. Port 20 is reserved for FTP data messages, port 21 for FTP control sessions, and port 23 for telnet.

The Dst Port Cmp parameter specifies the type of comparison to be made.

Filtering based only on established TCP sessions.

TCP Estab can be used to restrict the filter to packets in an established TCP session. You can only use it if the Protocol number has been set to 6 (TCP). Otherwise, it is not applicable.

Example filter specifications

This section shows some example generic and IP filter specifications.

Defining a filter to drop AppleTalk broadcasts

This example shows a generic filter whose purpose is to prevent local AppleTalk AEP and NBP traffic from going across the WAN. It is supposed to drop packets, so it will be applied as a data filter. The filter first defines packets that should be forwarded across the WAN: AARP (AppleTalk Address Resolution Protocol) packets, AppleTalk packets that are not addressed to the AppleTalk multicast address (such as regular traffic related to an actual AppleTalk File Server connection), and all non-AppleTalk traffic.

The filter then specifies that AEP (AppleTalk Echo Protocol) and NBP (Name Binding Protocol) packets should be dropped. To define this filter:

1. Open a Filter profile and assign it a name. For example:

```
Ethernet
  Filters
    Name=AppleTalk Broadcasts
```

2. Open Output Filters > Out filter 01.
3. Set Valid to Yes and Type to GENERIC.

```
Output filters...
  Out filter 01
    Valid=Yes
    Type=GENERIC
```

4. Open the Generic subprofile and specify the following rules:

```
Generic...
  Forward=Yes
  Offset=14
  Length=8
  Mask=fffffffffffffffffff
  Value=aaaa0300000080f3
  Compare=Equals
  More=No
```

These rules define the bytes in AARP packets that contain the protocol type number (0x80f3). The Value setting specifies the same value (0x80f3), so AARP packets match these rules.

5. Close this filter. Then open Out filter 02, and set Valid to Yes and Type to GENERIC.

```
Output filters...
  Out filter 02
    Valid=Yes
    Type=GENERIC
```

6. Open the Generic subprofile and specify the following rules:

```
Generic...
  Forward=Yes
  Offset=32
  Length=6
  Mask=ffffffffffff0000
  Value=090007ffff0000
  Compare=NotEquals
  More=No
```

These rules specify the multicast address used by AppleTalk broadcasts. The MAX forwards

any AppleTalk packet that does not match the specified rules.

7. Close this filter. Then open Out filter 03, and set Valid to Yes and Type to GENERIC.

```
Output filters...
  Out filter 03
    Valid=Yes
    Type=GENERIC
```

8. Open the Generic subprofile and specify the following rules:

```
Generic...
  Forward=Yes
  Offset=14
  Length=8
  Mask=fffffffffffffffffff
  Value=aaaa03080007809b
  Compare=NotEquals
  More=No
```

These rules define the bytes in AppleTalk packets that specifies the protocol type number (0x809b). These rules define non-AppleTalk traffic (packets that do not contain that value in the specified location). The MAX will forward non-AppleTalk outbound packets.

9. Close this filter. Then open Out filter 04, and set Valid to Yes and Type to GENERIC.

```
Output filters...
  Out filter 04
    Valid=Yes
    Type=GENERIC
```

10. Open the Generic subprofile and specify the following rules:

```
Generic...
  Forward=No
  Offset=32
  Length=3
  Mask=fffffffffffffffffff
  Value=0404040000000000
  Compare=Equals
  More=No
```

These rules specify AEP packets. For details, see *Inside AppleTalk* (Addison Wesley, Inc.)

11. Close this filter. Then open Out filter 05, and set Valid to Yes and Type to GENERIC.

```
Output filters...
  Out filter 05
    Valid=Yes
    Type=GENERIC
```

12. Open the Generic subprofile and specify the following rules:

```
Generic...
  Forward=No
  Offset=32
  Length=4
  Mask=ff00fff000000000
  Value=0200022000000000
```

```
Compare=Equals
More=Yes
```

Notice that More = Yes, linking Out filter 05 with the Out filter 06. Together, these two Out filters specify NBP lookup packets with a wildcard entity name.

13. Close this filter. Then open Out filter 06, and set Valid to Yes and Type to GENERIC.

```
Output filters...
  Out filter 06
    Valid=Yes
    Type=GENERIC
```

14. Open the Generic subprofile and specify the following rules:

```
Generic...
  Forward=No
  Offset=42
  Length=2
  Mask=ffff000000000000
  Value=013d000000000000
  Compare=Equals
  More=No
```

15. Close this filter.

16. Close the Filter profile.

Defining a filter to prevent IP address spoofing

IP address spoofing occurs when a remote device illegally acquires a local address to break through a firewall. This example filter first defines input filters that drop packets whose source address is on the local IP network or the loopback address (127.0.0.0). In effect, these filters say: "If you see an inbound packet with one of these source addresses, drop the packet." The third input filter defines every other source address (0.0.0.0) and specifies "Forward everything else to the local network."

Note: If you apply this filter to the Ethernet interface, the MAX drops IP packets it receives from local LAN and you will not be able to Telnet to the unit.

This example filter then defines an output filter that specifies: "If an outbound packet has a source address on the local network, forward it; otherwise, drop it." The MAX drops all outbound packets with a non-local source address. This filter uses a local IP network address of 192.100.50.128, with a subnet mask of 255.255.255.192. These addresses are just examples. To define this IP filter:

1. Open a Filter profile and assign it a name. For example:

```
Ethernet
  Filters
    Name=IP Spoofing
```

2. Open Input Filters > In filter 01.
3. Set Valid to Yes and Type to IP.

```

Input filters...
  In filter 01
    Valid=Yes
    Type=IP

```

4. Open the IP subprofile and specify the following rules:

```

Ip...
  Forward=No
  Src Mask=255.255.255.192
  Src Adrs=192.100.50.128
  Dst Mask=0.0.0.0
  Dst Adrs=0.0.0.0
  Protocol=0
  Src Port Cmp=None
  Src Port #=N/A
  Dst Port Cmp=None
  Dst Port #=N/A
  TCP Estab=N/A

```

The Src Mask parameter specifies the local netmask The Src Adrs parameter specifies the local IP address. If an incoming packet has the local address, the MAX does not forward it onto the Ethernet.

5. Close this filter. Then open In filter 02, and set Valid to Yes and Type to IP.

```

Input filters...
  In filter 02
    Valid=Yes
    Type=IP

```

6. Open the IP subprofile and specify the following rules:

```

Ip...
  Forward=No
  Src Mask=255.0.0.0
  Src Adrs=127.0.0.0
  Dst Mask=0.0.0.0
  Dst Adrs=0.0.0.0
  Protocol=0
  Src Port Cmp=None
  Src Port #=N/A
  Dst Port Cmp=None
  Dst Port #=N/A
  TCP Estab=N/A

```

These rules specify the loopback address in the Src Mask and Src Adrs fields. If an incoming packet has this address, the MAX does not forward it onto the Ethernet.

7. Close this filter. Then open In filter 03, and set Valid to Yes and Type to IP.

```

Input filters...
  In filter 03
    Valid=Yes
    Type=IP

```

8. Open the IP subprofile and specify the following rules:

```

Ip...

```

```

Forward=Yes
Src Mask=0.0.0.0
Src Adrs=0.0.0.0
Dst Mask=0.0.0.0
Dst Adrs=0.0.0.0
Protocol=0
Src Port Cmp=None
Src Port #=N/A
Dst Port Cmp=None
Dst Port #=N/A
TCP Estab=N/A

```

These rules specify every other source address (0.0.0.0) If an incoming packet has any non-local source address, the MAX forwards it onto the Ethernet.

9. Close this In filter and the Input filters subprofile. Then, open the Output filters subprofile and select the first Out filter in the list (01).
10. Set Valid to Yes and Type to IP.

```

Output filters...
  Out filter 01
    Valid=Yes
    Type=IP

```

11. Open the IP subprofile and specify the following rules:

```

Ip...
  Forward=Yes
  Src Mask=255.255.255.192
  Src Adrs=192.100.40.128
  Dst Mask=0.0.0.0
  Dst Adrs=0.0.0.0
  Protocol=0
  Src Port Cmp=None
  Src Port #=N/A
  Dst Port Cmp=None
  Dst Port #=N/A
  TCP Estab=N/A

```

The Src Mask parameter specifies the local netmask The Src Adrs parameter specifies the local IP address. If an outgoing packet has a local source address, the MAX forwards it.

12. Close the Filter profile.

Defining a filter for more complex IP security issues

This example illustrates some of the issues you may need to consider when writing your own IP filters. The sample filter presented here does not address the fine points of network security. You may want to use this sample filter as a starting point and augment it to address your security requirements. See the *MAX Security Supplement* for details.

In this example, the local network supports a Web server and the administrator needs to carry out these tasks:

- Provide dial-in access to the server's IP address.
- Restrict dial-in traffic to all other hosts on the local network.

However, many local IP hosts need to dial out to the Internet and use IP-based applications such as Telnet or FTP; therefore, their response packets need to be directed appropriately to the originating host. In this example, the Web server's IP address is 192.9.250.5. Apply this filter in Connection profiles as a data filter.

To define this filter:

1. Open a Filter profile and assign it a name.

```
Ethernet
  Filters
    Name=Web Safe
```

2. Open Input Filters > In filter 01.
3. Set Valid to Yes and Type to IP.

```
Input filters...
  In filter 01
    Valid=Yes
    Type=IP
```

4. Open the IP subprofile and specify the following rules:

```
Ip...
  Forward=Yes
  Src Mask=0.0.0.0
  Src Adrs==0.0.0.0
  Dst Mask=255.255.255.255
  Dst Adrs=192.9.250.5
  Protocol=6
  Src Port Cmp=None
  Src Port #=N/A: c(          +6(          Dst ETT11
```

```
Dst Port #=1023
TCP Estab=No
```

These rules specify TCP packets (Protocol = 6) *from* any address and *to* any address. The filter forwards them if the destination port is greater than the source port. For example, Telnet requests go out on port 23 and responses come back on some random port greater than port 1023. So, this filter defines packets coming back to respond to a user's request to Telnet to a remote host.

7. Close this filter. Then open In filter 03, and set Valid to Yes and Type to IP.

```
Input filters...
  In filter 03
    Valid=Yes
    Type=IP
```

8. Open the IP subprofile and specify the following rules:

```
Ip...
  Forward=Yes
  Src Mask=0.0.0.0
  Src Adrs=0.0.0.0
  Dst Mask=0.0.0.0
  Dst Adrs=0.0.0.0
  Protocol=17
  Src Port Cmp=None
  Src Port #=N/A
  Dst Port Cmp=Gtr
  Dst Port #=1023
  TCP Estab=No
```

These rules specify UDP packets (Protocol = 17) *from* any address and *to* any address. The filter forwards them if the destination port is greater than the source port. For example, suppose a RIP packet goes out as a UDP packet to destination port 520. The response to this request goes to a random destination port greater than 1023.

9. Close this filter. Then open In filter 04, and set Valid to Yes and Type to IP.

```
Input filters...
  In filter 04
    Valid=Yes
    Type=IP
```

10. Open the IP subprofile and specify the following rules:

```
Ip...
  Forward=Yes
  Src Mask=0.0.0.0
  Src Adrs=0.0.0.0
  Dst Mask=0.0.0.0
  Dst Adrs=0.0.0.0
  Protocol=1
  Src Port Cmp=None
  Src Port #=N/A
  Dst Port Cmp=None
  Dst Port #=N/A
  TCP Estab=No
```

These rules specify unrestricted pings and traceroutes. ICMP does not use ports like TCP and

UDP, so a port comparison is unnecessary.

11. Close the Filter profile.

Applying packet filters

Filters must be applied to an interface to examine packets passed across that interface in the MAX. They can be applied as a data filter, to forward or drop certain packets, or as a call filter, to affect which packets reset the Idle timer. See [Introduction to Ascend filters](#) for background information on these two applications. These are the relevant parameters:

```
Ethernet
  Answer
    Session options...
      Data Filter=0
      Call Filter=0
      Filter Persistence=No

Ethernet
  Connections
    Session options...
      Data Filter=5
      Call Filter=0
      Filter Persistence=No

Ethernet
  Mod Config
    Ether options...
      Filter=1
```

For more information about each parameter, see the *MAX Reference Guide*.

Understanding how filters are applied

This section provides some background information about the parameters for applying filters to a local or WAN interface.

- Applying filters in the Answer profile

The MAX does not apply filters in the Answer profile if the caller has a Connection profile. Use filters only if configured profiles are not required for callers, or if the caller is authenticated using a Name profile. If you use the Answer profile filters, they have the same effect as those ordinarily specified in a Connection profile, described next.

- Specifying a data filter

A data filter affects the actual data stream on the WAN interface, forwarding or dropping packets according to its rules. See [Data filters for dropping or forwarding certain packets](#). When you apply a filter to a WAN interface, the filter takes effect when the MAX brings up a connection up on that interface.

- Specifying a call filter

A call filter does not forward or drop packets. When the filter rules specify "forward", the call filter lets matching packets initiate the connection or reset the idle time if the connection is

active. See [Call filters for managing connections](#).

If you apply both a data filter and call filter, the data filter comes first. This means that only those packets that pass the data filter reach the call filter.

- Filter persistence

Before the MAX supported Secure Access, the MAX simply constructed a filter on a WAN interface when the connection was established and destroyed the filter when the connection was brought down, even if the connection just timed out momentarily. This works fine for static packet filters, but does not accommodate Secure Access firewalls. Filter Persistence is needed to allow firewalls to persist across connection state changes, but it is not needed for filters. If you do set it for a static packet filter, the filter persists across connection state changes. See the *MAX Security Supplement* for details.

- Applying a data filter on Ethernet

Call filters do not apply to the local network interface, so you need only one Filter parameter in the Ethernet profile. This is a data filter that affects which packets are allowed to reach the Ethernet or leave the Ethernet for another interface.

A filter applied to the Ethernet interface takes effect immediately. If you change the Filter profile definition, the changes apply as soon as you save the Filter profile.

Note: Use caution when applying a filter to the Ethernet interface. You could inadvertently render the MAX inaccessible from the local LAN.

Example configurations applying filters

After you create a filter, as described in [Defining packet filters](#), you can apply it as a data filter or call filter. This section shows some example configurations.

Applying a data filter in a Connection profile

To apply a data filter in a Connection profile:

1. Open the Session Options subprofile of the Connection profile.
2. Specify the filter's number in the Data Filter parameter. For example:

```
Ethernet
  Connections
    Session options...
      Data Filter=5
      Call Filter=0
      Filter Persistence=No
```

Specify the unique portion of the number preceding the filter's name in the Filters menu.

3. Close the Connection profile.

Applying a call filter and resetting the idle timer

When you apply a call filter in a Connection profile, it determines which packets reset the idle timer for a connection. In this example, the idle timer is reset to 20 seconds, so if no packets pass the call filter for 20 seconds, the MAX terminates the connection.

To apply a call filter and reset the idle timer in a Connection profile:

1. Open Connections > Session Options.
2. Specify the filter's number in the Call Filter parameter.

The filter's number is the unique portion of the number preceding the filter's name in the Filters menu.

3. Specify 20 seconds in the Idle parameter.

```
Ethernet
  Connections
    Session options...
      Data Filter=0
      Call Filter=2
      Filter Persistence=No
      Idle=20
```

Or, if the profile specifies a terminal server call, use the TS Idle Mode and TS Idle parameters instead; for example:

```
Ethernet
  Connections
    Session options...
      Data Filter=0
      Call Filter=2
      Filter Persistence=No
      Idle=0
      TS Idle Mode=Input/Output
      TS Idle=20
```

4. Close the Connection profile.

Applying a data filter to the Ethernet interface

To apply a data filter to the local network interface:

1. Open the Ethernet > Mod Config > Ether Options.
2. Specify the filter's number in the Filter parameter. For example:

```
Ethernet
  Mod Config
    Ether options...
      Filter=1
```

(Call filters are not applicable to the local network interface.)

3. Close the Ethernet profile.

Predefined filters

The MAX ships with three predefined Filter profiles, one for each commonly used protocol suite. Some sites modify the predefined call filters to make them more full-featured for the types of packets commonly seen at that site. As shipped, they provide a base that you can build on to fine-tune how the MAX handles routine traffic on your network. They are intended for use as call filters, to help keep connectivity costs down. These are the predefined filters:

- IP Call (for managing connectivity on IP connections)
- NetWare Call (for managing connectivity on IPX connections)
- AppleTalk Call (for managing connectivity on bridged AppleTalk connections)

IP Call filter

The predefined IP Call filter prevents inbound packets from resetting the Idle Timer. It does not prevent any type of outbound packets from resetting the timer or placing a call. The definitions for the IP Call filter parameters are:

```

Ethernet
  Filters
    IP Call...
      Name=IP Call
      Input filters...
        In filter 01
          Valid=Yes
          Type=GENERIC
          Generic...
            Forward=No
            Offset=0
            Length=0
            Mask=00000000000000000000
            Value=00000000000000000000
            Compare=None
            More=No
        Output filters...
          Out filter 01
            Valid=Yes
            Type=GENERIC
            Generic...
              Forward=Yes
              Offset=0
              Length=0
              Mask=00000000000000000000
              Value=00000000000000000000
              Compare=None
              More=No

```

The IP Call filter contains one input filter, which defines all inbound packets, and one output filter, which defines all outbound packets (all outbound packets destined for the remote network).

NetWare Call filter

The design of predefined NetWare Call filter prevents SAP (Service Advertising Protocol) packets originating on the local IPX network from resetting the Idle Timer or initiating a call. NetWare servers broadcast SAP packets every 60 seconds to make sure that all routers and bridges know about available services. To prevent these packets from keeping a connection up unnecessarily, apply the predefined NetWare Call filter in the Session Options subprofile of Connection profiles in which you configure IPX routing.

The predefined NetWare Call filter contains six output filters, which identify outbound SAP packets and prevent them from resetting the Idle Timer or initiating a call. The definitions for the NetWare Call filter parameters are:

```
Ethernet
  Filters
    NetWare Call...
      Name=NetWare Call
      Output filters...
        Out filter 01
          Valid=Yes
          Type=GENERIC
          Generic...
            Forward=No
            Offset=14
            Length=3
            Mask=ffffff00000000000000
            Value=e0e0030000000000
            Compare=Eqls
            More=Yes
        Out filter 02
          Valid=Yes
          Type=GENERIC
          Generic...
            Forward=No
            Offset=27
            Length=8
            Mask=fffffffffffffffffff
            Value=ffffffffffff0452
            More=Yes
        Out filter 03
          Valid=Yes
          Type=GENERIC
          Generic...
            Forward=No
            Offset=47
            Length=2
            Mask=ffff00000000000000
            Value=0002000000000000
            More=No
        Out filter 04
          Valid=Yes
          Type=GENERIC
          Generic...
            Forward=No
            Offset=12
            Length=4
            Mask=fc00ffff00000000
            Value=0000ffff00000000
            More=Yes
        Out filter 05
          Valid=Yes
          Type=GENERIC
          Generic...
            Forward=No
            Offset=24
            Length=8
            Mask=fffffffffffffffffff
            Value=ffffffffffff0452
            More=Yes
        Out filter 06
          Valid=Yes
          Type=GENERIC
```

```

Generic...
  Forward=No
  Offset=44
  Length=2
  Mask=ffff000000000000
  Value=0002000000000000
  More=No

```

AppleTalk Call filter

The AppleTalk Call filter instructs the MAX to place a call and reset the Idle Timer based on AppleTalk activity on the LAN, but to prevent inbound packets or AppleTalk Echo (AEP) packets from resetting the timer or initiating a call. It includes one input and five output filters.

The input filter prevents inbound packets from resetting the timer or initiating a call. The output filters identify the AppleTalk Phase II and Phase I AEP protocols. The last filter allows all other outbound packets to reset the timer or initiate a call.

```

Ethernet
  Filters
    AppleTalk Call...
      Name=AppleTalk Call
      Input filters...
        In filter 01
          Valid=Yes
          Type=GENERIC
          Generic...
            Forward=No
            Offset=0
            Length=0
            Mask=0000000000000000
            Value=0000000000000000
            More=No
        Output filters...
          Out filter 01
            Valid=Yes
            Type=GENERIC
            Generic...
              Forward=No
              Offset=14
              Length=8
              Mask=ffffff000000ffff
              Value=aaaa03000000809b
              More=Yes
          Out filter 02
            Valid=Yes
            Type=GENERIC
            Generic...
              Forward=No
              Offset=32
              Length=3
              Mask=ffffff0000000000
              Value=0404040000000000
              More=No
          Out filter 03
            Valid=Yes
            Type=GENERIC
            Generic...
              Forward=No
              Offset=12
              Length=2
              Mask=ffff000000000000

```

```
Value=809b000000000000
More=Yes
Out filter 04
Valid=Yes
Type=GENERIC
Generic...
Forward=No
Offset=24
Length=3
Mask=ffffff0000000000
Value=0404040000000000
More=No
Out filter 05
Valid=Yes
Type=GENERIC
Generic...
Forward=Yes
Offset=0
Length=0
Mask=0000000000000000
Value=0000000000000000
More=No
```

[HOME](#) [CONTENTS](#) [PREVIOUS](#) [NEXT](#) [INDEX](#)

techpubs@eng.ascend.com

Copyright © 1998, Ascend Communications, Inc. All rights reserved.