# Determining the Traffic Not Recognized by NBAR

**Document ID: 21628**

# Contents

# Introduction

This document shows how to use the Custom Packet Description Language Module (PDLM) feature of Network–Based Application Recognition (NBAR) to match on unclassified traffic or traffic that is not specifically supported as a match protocol statement.

# Prerequisites

## Requirements

Readers of this document should have knowledge of these topics:

- Basic QoS methodologies
- Basic understanding of NBAR

## Components Used

The information in this document is based on these software and hardware versions:

- Cisco IOS® Software Release 12.2(2)T
- Cisco 7206 router

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

## Conventions

For more information on document conventions, refer to the Cisco Technical Tips Conventions.

# Understanding the Custom PDLM

NBAR supports a variety of static and stateful protocols. PDLMs allow new protocol support for NBAR without the requirement of an IOS release upgrade and router reload. Subsequent IOS releases incorporate support for these new protocols.

The Custom PDLM allows you to map protocols to static User Datagram Protocol (UDP) and TCP ports for protocols that are not currently supported in NBAR with a match protocol statement. In other words, it extends or enhances the list of protocols recognized by NBAR.

Here are the steps to adding the Custom PDLM to your router.

1. Locate and download the NBAR PDLM from the Software Download page (registered customers only) by downloading the **custom.pdlm file**.
2. Load the PDLM onto a flash memory device, such as PCMCIA card in slots 0 or 1, using the command below.

```
7206-15(config)# ip nbar pdlm slot0:custom.pdlm
```

3. Verify support for custom protocols using the **show ip nbar port−map | include custom** command (shown below) or the **show ip nbar pdlm** command.

```
7206-16# show ip nbar port-map | include custom
port-map custom-01              udp 0
port-map custom-01              tcp 0
port-map custom-02              udp 0
port-map custom-02              tcp 0
port-map custom-03              udp 0
port-map custom-03              tcp 0
port-map custom-04              udp 0
port-map custom-04              tcp 0
port-map custom-05              udp 0
port-map custom-05              tcp 0
port-map custom-06              udp 0
port-map custom-06              tcp 0
port-map custom-07              udp 0
port-map custom-07              tcp 0
port-map custom-08              udp 0
port-map custom-08              tcp 0
port-map custom-09              udp 0
port-map custom-09              tcp 0
port-map custom-10              udp 0
port-map custom-10              tcp 0
```

4. Assign ports to the custom protocols using the **ip nbar port−map custom−XY {tcp|udp} {port1 port2 ...}** command. For example, to match on traffic at TCP port 8877, use the **ip nbar port−map custom−01 tcp 8877** command.

# Classifying "Unclassified" Ports

Depending on your network traffic, you may need to use special classification mechanisms in NBAR. Once you classify this traffic, you then can use the custom PDLM and match the UDP and TCP port numbers to a custom port−map.

By default, the NBAR unclassified mechanisms are not enabled. The **show ip nbar unclassified−port−stats** command returns the following error message:

```
d11-5-7206-16# show ip nbar unclassified-port-stats
Port Statistics for unclassified packets is not turned on.
```

Under carefully controlled circumstances, use the **debug ip nbar unclassified−port−stats** command to configure the router to begin tracking on which ports that packets arrive. Then use the **show ip nbar unclassified−port−stats** command to verify the collected information. The output now displays a histogram of the most commonly used ports.

**Note:** Before issuing **debug** commands, refer to Important Information on Debug Commands. The **debug ip nbar** commands should be enabled only under carefully controlled circumstances.

If this information is not sufficient, you can enable the capture capability, which provides an easy way to capture packet traces of new protocols. Use the following **debug** commands, as shown below.

```
debug ip nbar filter destination_port tcp XXXX
debug ip nbar capture 200 10 10 10
```

The first command defines the packets in which you are interested for capture. The second command puts NBAR into capture mode. The arguments of the **capture** command are as follows:

- Number of bytes to capture per packet.
- Number of starting packets to capture, in other words, how many packets to capture after the TCP/IP SYN packet.
- Number of final packets to capture, in other words, how many packets at the end of the flow for which space should be reserved.
- Number of total packets to capture.

**Note:** Specifying the starting and final packet parameters captures only the relevant packets in a long flow.

Use the **show ip nbar capture** command to view the collected information. By default, capture mode waits for a SYN packet to arrive and then starts capturing the packets on that bidirectional flow.

# Blocking Gnutella with the Custom PDLM

Let's look at an example of how to use the Custom PDLM. We use Gnutella as the traffic we want to classify and then apply a QoS policy that blocks this traffic.

Gnutella uses six well–known TCP ports – 6346, 6347, 6348, 6349, 6355, and 5634. Other ports may be detected as Pongs are received. If users specify other ports for use in Gnutella file sharing, you can add these ports to your custom match protocol statement.

Here are the steps to creating a QoS service policy that matches on and drops Gnutella traffic.

1. As noted above, use the **show ip nbar unclassified–port–stats** command to view the NBAR "unclassified" traffic. If your network is transporting Gnutella traffic, you will see output similar to the following.

```
Port     Proto     # of Packets
-------  --------  -------
6346     tcp       347679
27005    udp       55043
```

2. Use the **ip nbar port–map custom** command to define a custom port–map that matches on the Gnutella ports.

```
ip nbar port-map custom-02 tcp 5634 6346 6347 6348 6349 6355
```

**Note:** Currently, you must use a name such as custom–xx. User–defined names for custom PDLMs will be supported in an upcoming release of Cisco IOS Software.

3. Use the **show ip nbar protocol stats** command to confirm matches to the custom statement.

```
2620# show ip nbar protocol stats byte-count
FastEthernet0/0
               Input              Output
Protocol       Byte Count         Byte Count
```

```
          ----------------------------------------------
custom-02       43880517                 52101266
```

4. Create a QoS service policy using the commands of the modular QoS CLI (MQC).

```
d11-5-7206-16(config)# class-map gnutella
d11-5-7206-16(config-cmap)# match protocol custom-02
d11-5-7206-16(config-cmap)# exit
d11-5-7206-16(config)# policy-map sample
d11-5-7206-16(config-pmap)# class gnutella
d11-5-7206-16(config-pmap-c)# police 1000000 31250 31250 conform-action
          drop exceed-action drop violate-action drop
```

Refer to Using Network–Based Application Recognition and Access Control Lists for Blocking the "Code Red" Worm for other configuration commands to block Gnutella and other unwanted traffic.

# Related Information

- **QoS Support Resources**
- **Technical Support – Cisco Systems**

Updated: Feb 15, 2008                                      Document ID: 21628