Configuring WAAS Express

First Published: October 15, 2010
Last Updated: October 29, 2010

This module describes Cisco's WAAS Express software, which interoperates with WAN optimization headend applications from Cisco. WAAS Express improves WAN access and use by optimizing applications that require high bandwidth or are bound to a LAN, such as backup.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for WAAS Express" section on page 24.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Prerequisites for WAAS Express

- WAAS Express requires version 4.2.1 or higher installed on Cisco Network Module (NME) and Wide-area Application Engines (WAE).

WAAS Express Licensing

You must have a valid license for WAAS Express. The WAAS Express software includes a trial license for 60 days. WAAS Express switches into the following modes based on the available memory on the router

- WAAS_Standard—WAAS Express operates on this mode if maximum memory is available on the router.
- WAAS_Trial_Limited—WAAS Express operates on this mode if default memory is available on the router.
- WAAS_Disabled—WAAS Express operates on this mode if less than default memory is available on the router.

Note The maximum and default memory depend on the router.

Restrictions for WAAS Express

In Cisco 1905 and 1921 routers, DRE is disabled because the maximum memory in these routers is 512 Mb.

WAAS Express does not support the following:

- Data Redundancy Elimination (DRE) encoding
- Dual WAN interfaces.
- The maximum number of concurrent connections optimized by WAAS Express depends on the platform.
- You cannot define a user-defined map of type waas.
- WAAS Express accepts Selective Acknowledgement (SACK) notifications but does not generate SACK.

WAAS Express interoperates with the following features and may work with features not mentioned in the list:

- Dynamic Multipoint VPN (DMVPN)
- Zone-based Firewall
- Virtual tunnel interfaces (VTI)
- Network address translation (NAT)
- Quality of service (QoS)
- Flexible NetFlow

Note If WAAS Express and cryptomaps are configured on the same WAN interface and Flexible NetFlow is also configured to collect the WAAS Express flow information, Flexible NetFlow only collects small packets of the WAAS Express flows and does not report flows that are marked optimized and consumed by WAAS Express.

The following WAAS Express license restrictions apply:

- Low memory is not available in permanent license.
- WAAS_Trial_Limited:
  - Operates on limited flows and DRE memory and does not support IO resizing.

Information About WAAS Express

This section contains the following topics:

WAAS Express Overview

Cisco WAN optimization system consists of WAAS Express (WE) routers and Wide-area Application Engines (WAEs) that work together to optimize TCP traffic in your network. When client and server applications attempt to communicate with each other, the network intercepts the traffic and acts on behalf of the client application and the destination server. The WE and WAEs examine the traffic and use built-in application policies to determine whether the traffic in the network can be optimized.

WAAS Express helps enterprises meet the following objectives:

- Complements the Cisco WAN optimization system by adding the capability to the branch routers.
- Provide branch office employees with LAN-like access to information and applications across a geographically distributed network.
- Minimize unnecessary WAN bandwidth consumption through the use of advanced compression algorithms.
- Virtualize print and other local services to branch office users.
- Improve application performance over the WAN by addressing the following common issues:
  - Low data rates (constrained bandwidth)
  - Slow delivery of frames (high network latency)
  - Higher rates of packet loss (low reliability)

Table 1 describes how WAAS Express uses TCP optimization techniques to overcome the most common challenges associated with transporting traffic over a WAN.

Table 1 WAAS Express Solution

WAN Issue	WAAS Solution
Constrained bandwidth	Data caching provided with the file services feature and data compression reduces the amount of data sent over the WAN, which increases data transfer rates. These solutions improve total transaction time on congested links by reducing the amount of data sent across the WAN.
Packet loss	The optimized TCP stack in WAAS overcomes the issues associated with high packet loss and protects communicating endpoints from the state of the WAN.<<protects what>>

Traffic Optimization Process

The following steps describe how a WAAS Express-enabled network optimizes connections between the branch office client and destination server:

1. A branch office client attempts to connect to the destination server over the native application port.
2. The branch client intercepts the traffic.
3. The branch client performs the following actions:
   - Examines the parameters in the traffic's TCP headers and then refers to the application policies to determine if the intercepted traffic should be optimized. Information in the TCP header, such as the source and destination IP address, allows the branch client to match the traffic to an application policy. For a list of the default policies, see the "WAAS Application Policies" section on page 7.
   - Negotiates with the data-center WAE whether the traffic must be optimized.
   - Based on the negotiation, if the branch client determines that the traffic should be optimized, it adds information to the TCP header that informs the next client in the network path to optimize the traffic.
4. The branch client passes along the client request through the network to its original destination server.
5. The data center WAE performs the following actions:
   - Intercepts the traffic going to the destination server.
   - Establishes an optimized connection with the branch. If the data center WAE has optimization disabled, then an optimized connection is not established and the traffic passes over the network unoptimized.
6. WAAS optimizes subsequent traffic between the branch and data center WAE depending on the connection type.

WAAS Express does not optimize traffic in the following situations:

- WAAS Express is overloaded and does not have resources to optimize traffic.
- The WAE intercepts non-TCP traffic, such as Internet Control Message Protocol (ICMP).
- The WAE is overloaded and does not have the resources to optimize the traffic.
- The intercepted traffic matches an application policy that specifies to pass the traffic through unoptimized.

Note If unoptimized traffic reaches a WAE, the WAE forwards the traffic in pass-through mode without affecting the performance of the application using the passed-through connection.

Key Services of WAAS Express

The following sections describe WAAS Express services that help optimize traffic over your WAN:

Transport Flow Optimization

WAAS Express uses the transport flow optimization (TFO) features described in the following sections to optimize traffic intercepted by the WAAS devices. TFO protects communicating clients and servers from negative WAN conditions, such as bandwidth constraints, packet loss, congestion, and retransmission.

Windows Scaling

RFC 1323 describes TCP extensions for high performance. Window scaling allows the receiver of a TCP packet to advertise that its TCP receive window can exceed 64 KB. The receive window size determines the amount of space that the receiver has available for unacknowledged data. Windows scaling allows TCP endpoints to take advantage of available bandwidth in your network and not be limited to the default window size specified in the TCP header.

Note WAAS Express limits the maximum receive window size to 64 KB.

Selective Acknowledgment

RFC 2018 describes TCP Selective Acknowledgment (SACK) options. SACK is an efficient packet loss recovery and retransmission feature that allows clients to recover from packet losses more quickly than the default recovery mechanism used by TCP.

By default, TCP uses a cumulative acknowledgment scheme that forces the sender to either wait for a round trip to learn if any packets were not received by the recipient or to unnecessarily retransmit segments that may have been correctly received.

SACK allows the receiver to inform the sender about all segments that arrive successfully, so the sender needs to retransmit only the segments that are lost.

Binary Increase Congestion TCP

Binary Increase Congestion (BIC) TCP is a congestion management protocol that allows your network to recover more quickly from packet loss events.

When your network loses a packet, BIC TCP reduces the receiver's window size and sets that reduced size as the new value for the minimum window. BIC TCP then sets the maximum window size value to the size of the window just before the packet loss occurred. Because packet loss occurred at the maximum window size, the network can transfer traffic without dropping packets whose size falls within the minimum and maximum window size values.

If BIC TCP does not register packet loss at the updated maximum window size, that window size becomes the new minimum. If packet loss does occur, that window size becomes the new maximum. This process continues until BIC TCP determines the new optimum minimum and maximum window size values.

Compression

WAAS Express uses the following compression technologies to help reduce the size of data transmitted over a WAN:

- Data Redundancy Elimination (DRE)
- Lempel-Ziv (LZ) compression

These compression technologies reduce the size of transmitted data by removing redundant information before sending the shortened data stream over the WAN. By reducing the amount of transferred data, WAAS compression can reduce network utilization and application response times.

When WAAS Express uses compression to optimize TCP traffic, it replaces repeated data in the stream with a much shorter reference, then sends the shortened data stream out across the WAN. The receiving WAAS Express device uses its local redundancy library to reconstruct the data stream before passing it along to the destination client or server.

The DRE compression scheme is based on a shared cache architecture where each device involved in compression and decompression shares the same redundancy library. When the cache that stores the redundancy library on a WAE becomes full, DRE uses a first in, first out algorithm (FIFO) to discard old data and make room for new data. For WAAS Express, the cache is only used for decoding.

LZ compression operates on smaller data streams and keeps limited compression history. DRE operates on significantly larger streams (typically tens to hundreds of bytes or more) and maintains a much larger compression history. Large chunks of redundant data is common in file system operations when files are incrementally changed from one version to another or when certain elements are common to many files, such as file headers and logos.

Note DRE optimization is not supported for connections between WAAS Express devices.

WAAS Express does not compress uploading traffic using the DRE algorithm. WAAS Express decompresses the download traffic that is compressed by DRE.

Auto-discovery of WAAS Devices

WAAS Express includes an autodiscovery feature that enables WAEs and WAAS Express devices to automatically locate peer WAEs on your network by adding TCP options on the control packets. After discovering a peer device automatically, the WAEs terminate and separate the LAN-to-WAN TCP connections and add a buffering layer to resolve the differing speeds or WAAS Express proxies the connection on the router in different segments to achieve optimization benefits. Once a WAE establishes a connection to a peer WAE, the two devices can establish an optimized link for TCP traffic, or pass the non-TCP traffic through as unoptimized.

The autodiscovery of peer WAAS devices is achieved using TCP options. These TCP options are recognized and understood only by WAAS devices and are ignored by non-WAAS devices.

WAAS Application Policies

The WAAS software includes over 150 default application policies that help the WAAS system to classify and optimize some of the most common types of traffic in the network.

Table 2 lists the default applications and classifiers that WAAS will either optimize or pass through based on the policies that are provided with the system.

Cisco recommends that you review the default policies and modify them as appropriate before you create a new application policy. It is often easier to modify an existing policy than to create a new one.

When reviewing Table 2, note the following:

- The subheadings represent the application names. The associated classifiers are listed under these subheadings. For example, Authentication is a type of application and Kerberos is a classifier for that application.
- The word *Monitored* indicates that the applications are monitored by the WAAS Central Manager, which can only display statistics for 20 applications at a time.

The WAAS software supports the following optimization actions based on the type of traffic it encounters:

- TFO—A collection of optimization technologies such as automatic windows scaling, increased buffering, and selective acknowledgment that optimize all TCP traffic over your network.
- Data Redundancy Elimination (DRE)—A compression technology that reduces the size of transmitted data by removing redundant information before sending the shortened data stream over the WAN. DRE operates on significantly larger streams and maintains a much larger compression history than LZ compression.
- LZ (compression)—A compression technology that operates on smaller data streams and keeps limited compression history compared to DRE.

Table 2 Default Traffic Policies

Classifier	WAAS Action	Destination Ports
Authentication		
Kerberos	Pass-through	88, 464, 543, 544, 749, 754, 888, 2053
SASL	Pass-through	3659
TACACS	Pass-through	49
Backup *(monitored)*		
Amanda	TFO	10080
BackupExpress	TFO	6123
CommVault	TFO	8400–8403
Connected-DataProtector	TFO	16384
IBM-TSM	TFO+LZ+DRE	1500–1502
Legato-NetWorker	TFO	7937, 7938, 7939
Legato-RepliStor	TFO	7144, 7145

Table 2 Default Traffic Policies (continued)

Classifier	WAAS Action	Destination Ports
Veritas-BackupExec	TFO	1125, 3527, 6101, 6102, 6106
Veritas-NetBackup	TFO	13720, 13721, 13782, 13785
Computer-aided Design (CAD)		
PDMWorks	LZ+TFO+DRE	30000, 40000
Call Management		
Cisco-CallManager	Pass-through	2443, 2748
SIP-secure	Pass-through	5061
VoIP-Control	Pass-through	1300, 1718–1720, 2000–2002, 2428, 5060, 11000–11999
Conferencing		
CU-SeeMe	Pass-through	7640, 7642, 7648, 7649
ezMeeting	Pass-through	10101–10103, 26260–26261
GnomeMeeting	Pass-through	30000–30010
Intel-Proshare	Pass-through	5713–5717
MS-NetMeeting	Pass-through	522, 1503, 1731
VocalTec	Pass-through	1490, 6670, 22555, 25793
Console		
SSL-Shell	Pass-through	614
Telnet	Pass-through	23, 107, 513
Telnets	Pass-through	992
Unix-Remote-Execution	Pass-through	512, 514
Content-Management *(monitored)*		
Documentum	LZ+TFO+DRE	1489
Filenet	LZ+TFO+DRE	32768–32774
ProjectWise-FileTransfer	LZ+TFO+DRE	5800
Directory Services *(monitored)*		
LDAP	LZ+TFO+DRE	389, 8404
LDAP-Global-Catalog	LZ+TFO+DRE	3268
LDAP-Global-Catalog-Secure	Pass-through	3269
LDAP-secure	Pass-through	636
E-mail and Messaging *(monitored)*		
HP-OpenMail	LZ+TFO+DRE	5729, 5755, 5757, 5766, 5767, 5768
Internet-Mail	LZ+TFO+DRE	25, 110, 143, 220
Internet-Mail-secure	TFO	465, 993, 995
Lotus-Notes	LZ+TFO+DRE	1352
MAPI[1]	LZ+TFO+DRE	UUID:a4f1db00-ca47-1067-b31f-00dd010662da

Table 2 Default Traffic Policies (continued)

Classifier	WAAS Action	Destination Ports
MDaemon	LZ+TFO+DRE	3000, 3001
NNTP	LZ+TFO+DRE	119
NNTP-secure	TFO	563
Novell-Groupwise	LZ+TFO+DRE	1099, 1677, 2800, 3800, 7100, 7101, 7180, 7181, 7205, 9850
PCMail-Server	LZ+TFO+DRE	158
QMTP	LZ+TFO+DRE	209
X400	LZ+TFO+DRE	102
Enterprise Applications *(monitored)*		
SAP	LZ+TFO+DRE	3200–3219, 3221–3224, 3226–3267, 3270–3282, 3284–3305, 3307–3388, 3390–3399, 3600–3659, 3662–3699
Siebel	LZ+TFO+DRE	2320, 2321, 8448
File System *(monitored)*		
AFS	LZ+TFO+DRE	7000–7009
Apple-AFP	LZ+TFO+DRE	548
CIFS-non-wafs	LZ+TFO+DRE	139, 445
NFS	LZ+TFO+DRE	2049
Novell-NetWare	LZ+TFO+DRE	524
Sun-RPC	Pass-through	111
File Transfer *(monitored)*		
BFTP	LZ+TFO+DRE	152
FTP-Control[2]	Pass-through	21
FTP-Data[2]	LZ+TFO+DRE	src20
FTPS[2]	TFO	990
FTP-Control[2]	Pass-through	src989
Simple-FTP	LZ+TFO+DRE	115
TFTP	LZ+TFO+DRE	69
TFTPS	TFO	3713
Instant Messaging		
AOL	Pass-through	5190–5193
Apple-iChat	Pass-through	5297, 5298
IRC	Pass-through	531, 6660–6669
Jabber	Pass-through	5222, 5269
Lotus-Sametime-Connect	Pass-through	1533
MS-Chat	Pass-through	6665, 6667
MSN-Messenger	Pass-through	1863, 6891–6900

Table 2 Default Traffic Policies (continued)

Classifier	WAAS Action	Destination Ports
Yahoo-Messenger	Pass-through	5000, 5001, 5050, 5100
Name Services		
DNS	Pass-through	53
iSNS	Pass-through	3205
Service-Location	Pass-through	427
WINS	Pass-through	42, 137, 1512
Network Analysis		
Cisco-NetFlow	Pass-through	7544, 7545
Other *(monitored)*		
Basic-TCP-services	Pass-through	1–19
BGP	LZ+TFO+DRE	179
MS-Message-Queuing	LZ+TFO+DRE	1801, 2101, 2103, 2105
NTP	Pass-through	123
Other-Secure	Pass-through	261, 448, 684, 695, 994, 2252, 2478, 2479, 2482, 2484, 2679, 2762, 2998, 3077, 3078, 3183, 3191, 3220, 3410, 3424, 3471, 3496, 3509, 3529, 3539, 3660, 3661, 3747, 3864, 3885, 3896, 3897, 3995, 4031, 5007, 5989, 5990, 7674, 9802, 12109
SOAP	LZ+TFO+DRE	7627
Symantec-AntiVirus	LZ+TFO+DRE	2847, 2848, 2967, 2968, 38037, 38292
Peer-to-peer (P2P) *(monitored)*		
BitTorrent	Pass-through	6881–6889, 6969
eDonkey	Pass-through	4661, 4662
Gnutella	Pass-through	6346–6349, 6355, 5634
Grouper	Pass-through	8038
HotLine	Pass-through	5500–5503
Kazaa	Pass-through	1214
Laplink-ShareDirect	Pass-through	2705
Napster	Pass-through	6666, 6677, 6700, 6688, 7777, 8875
Qnext	Pass-through	44, 5555
SoulSeek	Pass-through	2234, 5534
WASTE	Pass-through	1337
WinMX	Pass-through	6699
Printing *(monitored)*		
AppSocket	LZ+TFO+DRE	9100
IPP	LZ+TFO+DRE	631

Table 2 Default Traffic Policies (continued)

Classifier	WAAS Action	Destination Ports
SUN-Xprint	LZ+TFO+DRE	8100
Unix-Printing	LZ+TFO+DRE	170, 515
Remote Desktop *(monitored)*		
Altiris-CarbonCopy	Pass-through	1680
Apple-NetAssistant	Pass-through	3283
Citrix-ICA	LZ+TFO+DRE	1494, 2598
ControlIT	TFO	799
Danware-NetOp	TFO	6502
Laplink-Host	TFO	1547
Laplink-PCSync	TFO	8444
Laplink-PCSync-secure	TFO	8443
MS-Terminal-Services	TFO	3389
Netopia-Timbuktu	TFO	407, 1417–1420
PCAnywhere	TFO	73, 5631, 5632, 65301
RAdmin	TFO	4899
Remote-Anything	TFO	3999, 4000
Vmware-VMConsole	TFO	902
VNC	TFO	5801–5809, 6900–6909
XWindows	TFO	6000–6063
Replication *(monitored)*		
Double-Take	LZ+TFO+DRE	1100, 1105
EMC-Celerra-Replicator	LZ+TFO+DRE	8888
MS-AD-Replication[1]	LZ+TFO+DRE	UUID:e3514235-4b06-11d1-ab04-00c04fc2dcd2
MS-Content-Replication-Service	TFO	560, 507
MS-FRS[1]	LZ+TFO+DRE	UUID:f5cc59b4-4264-101a-8c59-08002b2f8426
Netapp-SnapMirror	LZ+TFO+DRE	10565–10569
Remote-Replication-Agent	TFO	5678
Rsync	TFO	873
Structured Query Language (SQL) *(monitored)*		
Borland-Interbase	LZ+TFO+DRE	3050
IBM-DB2	LZ+TFO+DRE	523
InterSystems-Cache	LZ+TFO+DRE	1972
MS-SQL	LZ+TFO+DRE	1433
MS-SQL-RPC[1]	LZ+TFO+DRE	UUID:3f99b900-4d87-101b-99b7-aa0004007f07

Table 2 Default Traffic Policies (continued)

Classifier	WAAS Action	Destination Ports
MySQL	LZ+TFO+DRE	3306
Oracle	LZ+TFO+DRE	66, 1521, 1525
Pervasive-SQL	LZ+TFO+DRE	1583
PostgreSQL	LZ+TFO+DRE	5432
Scalable-SQL	LZ+TFO+DRE	3352
SQL-Service	LZ+TFO+DRE	156
Sybase-SQL	LZ+TFO+DRE	1498, 2439, 2638, 3968
UniSQL	LZ+TFO+DRE	1978, 1979
Secure Sockets Layer (SSL) *(monitored)*		
HTTPS	TFO	443
Secure Shell (SSH)		
SSH	TFO	22
Storage *(monitored)*		
EMC-SRDFA-IP	LZ+TFO+DRE	1748
FCIP	LZ+TFO+DRE	3225
iFCP	LZ+TFO+DRE	3420
iSCSI	LZ+TFO+DRE	3260
Streaming *(monitored)*		
Liquid-Audio	LZ+TFO+DRE	18888
MS-NetShow	LZ+TFO+DRE	1755
RTSP	LZ+TFO+DRE	554, 8554
VDOLive	LZ+TFO+DRE	7000
Systems Management *(monitored)*		
BMC-Patrol	Pass-through	6161, 6162, 6767, 6768, 8160, 8161, 10128
HP-OpenView	Pass-through	7426–7431, 7501, 7510
HP-Radia	LZ+TFO+DRE	3460, 3461, 3464, 3466
IBM-NetView	Pass-through	729–731
IBM-Tivoli	LZ+TFO+DRE	94, 627, 1580, 1581, 1965
LANDesk	LZ+TFO+DRE	9535, 9593–9595
NetIQ	Pass-through	2220, 2735, 10113–10116
Netopia-netOctopus	Pass-through	1917, 1921
Novell-ZenWorks	LZ+TFO+DRE	517, 1761–1763, 2037, 2544, 8039
WAAS-FlowMonitor	TFO	7878
WBEM	Pass-through	5987, 5988
Version Management *(monitored)*		
Clearcase	LZ+TFO+DRE	371

Table 2 Default Traffic Policies (continued)

Classifier	WAAS Action	Destination Ports
CVS	LZ+TFO+DRE	2401
Virtual Private Network (VPN)		
CIFS	LZ+TFO+DRE	139, 445
HTTP	LZ+TFO+DRE	80, 3128, 8000, 8001, 8080
HTTPS	TFO	443
L2TP	TFO	1701
OpenVPN	TFO	1194
PPTP	TFO	1723

1. These classifiers use the EndPoint Mapper (EPM) service in WAAS to accelerate traffic. EPM-based applications do not have predefined ports so the application's Unique Server Identity (UUID) must be used to identify the traffic.
2. These classifiers identify the source port instead of the destination port.

How to Configure WAAS Express

- Configuring WAN Optimization Parameters, page 13 (optional)
- Defining WAAS Express Policies, page 14 (optional)
- Enabling WAAS Express, page 20 (required)

Configuring WAN Optimization Parameters

Perform the following task to configure WAN optimization parameters globally.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type waas** *parameter-map-name*
4. **tfo auto-discovery blacklist** {**enable** | **hold-time** *minutes*}
5. **tfo optimize** {**full** | **dre** {**no** | **yes** {**compression**} {**lz** | **none**}}}
6. **cpu-threshold** *maximum-threshold*
7. **lz entropy-check**
8. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**parameter-map type waas** *parameter-map-name* **Example:** `Router(config)# parameter-map type waas waas_global`	Configures a parameter map of type waas and enters parameter map configuration mode. **Note** The only supported parameter map of type waas is **waas_global**.
Step 4	**tfo auto-discovery blacklist** {**enable** | **hold-time** *minutes*} **Example:** `Router(config-profile)# tfo auto-discovery blacklist hold-time 1000`	Enables, configures and integrates blacklist with auto-discovery for WAN globally. • **enable**—Enables a blacklist. • **hold-time** *minutes*—Specifies the period for holding the blacklist in the system.
Step 5	**tfo optimize** {**full** | **dre** {**yes** | **no** {**compression**} {**lz** | **none**}}} **Example:** `Router(config-profile)# tfo optimize dre no compression lz`	Configures the compression for WAN. • **full**—Turns on DRE and LZ compression. • **dre** {**yes** | **no**}—Toggles on or off DRE. • **compression** {**lz** | **none**}—Toggles on or off LZ compression.
Step 6	**cpu-threshold** *maximum-threshold* **Example:** `Router(config-profile)# cpu-threshold 90`	Sets the CPU threshold limit. • *maximum-threshold*—Specifies the maximum limit. The range is 1 to 100.
Step 7	**lz entropy-check** **Example:** `Router(config-profile)# lz entropy-check`	Enables adaptive LZ through entropy checking.
Step 8	**exit** **Example:** `Router(config-profile)# exit`	Exits the current mode.

Defining WAAS Express Policies

Note WAAS Express can be configured either with the default class maps and policy maps, or the class maps and policy maps can be defined and then WAAS Express configured.

Perform the following tasks to define class and policy maps if you do not want to use the default class and policy maps that are created when WAAS Express is enabled:

- Defining Class maps, page 15
- Associating Class maps to Policy maps, page 16

Defining Class maps

Perform the following task to define a class map.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map type waas** *class-name*
4. **match tcp** {**any** | **destination** | **source**}{**ip** *ip-address* [*inverse mask*] | **port** *start-port-number*1 [*end-port-number2*]}
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**class-map type waas** *class-name* **Example:** `Router(config)# class-map type waas waas_global`	Defines a class map of type waas and enters class map configuration mode.

	Command or Action	Purpose
Step 4	**match tcp** {**any** | **destination** | **source**}{**ip** *ip-address* [*inverse mask*] | **port** *start-port-number1* [*end-port-number2*]} **Example:** `Router(config-cmap)# match tcp destination port 7000 7009`	Matches the traffic based on the following criteria. • **any**—Matches all TCP traffic. • **destination**—Matches the TCP traffic with the destination IP address or port number. • **source**—Matches the TCP traffic with the source IP address or port number. • **ip** *ip-address*—Refers to the IP address of the source and destination. If NAT is used, the IP address refers to inside local address and outside global address. • **port** *port-number*—Refers to the port number of the source and destination. **Note** The class-map of type WAAS combines filters using the match-any logical operator. The match-all logical operator is not supported by the class map of type WAAS. This means that if one match criterion (filter) is matched, the entire class map is matched also.
Step 5	**exit** **Example:** `Router(config-cmap)# exit`	Exits the current mode.

Examples

The following example matches traffic having the destination TCP port number between 7000 and 7009:

```
Router(config)# class-map type waas waas_global
Router(config-cmap)# match tcp destination port 7000 7009

Router(config)# class-map type waas waas_global
Router(config-cmap)# match tcp destination ip 209.165.200.225 0.0.0.31 port 80 80
Router(config-cmap)# match tcp destination ip 209.165.200.225 0.0.0.31 port 8080 8080
```

In the example, traffic in either of the following two conditions is matched:

- Destination IP address is in the range 209.165.200.225, and the destination TCP port is 80.
- Destination IP address is in the range 209.165.200.225, and the destination TCP port is 8080.

What to Do Next

After defining the class maps, proceed to Associating Class maps to Policy maps.

Associating Class maps to Policy maps

Perform the following task to associate a class map to a policy map, before enabling WAAS Express on the device.

Note Any changes to the policy configuration (global policy map and the class maps of type WAAS) remain forever. For instance, if you modify the policy configuration, disable WAAS Express on the interfaces and reenable WAAS Express, the changes would still be visible.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map type waas** *policy-name*
4. **sequence-interval** *number*
5. **class** *class-map-name*
6. **optimize tfo** {**dre** | **lz**} **application** *application-name*
7. **passthrough application** *application-name*
8. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**policy-map type waas** *policy-name* **Example:** `Router(config)# policy-map type waas waas_global`	Defines a policy map of type waas and enters policy map configuration mode.
Step 4	**sequence-interval** *number* **Example:** `Router(config-pmap)# sequence-interval 100`	Assigns sequential numbers to the class-maps at the specified interval. • *number*—Specifies the sequential interval. The range is 1 to 65535.
Step 5	**class** *class-map-name* **Example:** `Router(config-pmap)# class-map class1`	Specifies the class on which optimization must be performed and enters class map configuration mode. • *class-map-name*—the name of the class-map.

	Command or Action	Purpose
Step 6	**optimize tfo** {**dre** | **lz**} **application** *application-name* **Example:** `Router(config-pmap-c)# optimize dre application web`	Applies WAN optimization for the matching traffic as follows: • **tfo**—Applies TFO optimization only. • **application** *application-name*—Class-map application. • **dre**—Applies TFO and DRE optimization. • **lz**—Applies TFO and LZ optimization.
Step 7	**passthrough application** *application-name* **Example:** `Router(config-pmap-c)# passthrough application web`	Passes through match traffic and does not apply WAN optimization to the matching traffic. • **application** *application-name*—Class-map application. **Note** **passthrough** is the default WAN optimization for the matching traffic.
Step 8	**exit** **Example:** `Router(config-pmap-c)# exit`	Exits the current mode.

Examples

This example shows how to create a new policy with actions and application tagging:

```
Router(config)# policy-map type waas waas_global
Router(config-pmap)# class AFS
Router(config-pmap-c)# optimize dre lz application Web
Router(config-pmap-c)# exit
Router(config-pmap)# class Http
Router(config-pmap-c)# optimize lz application Filesystem
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# exit
Router(config-pmap)# exit
```

The following is sample output from the **show policy-map type waas** command:

```
Policy Map type waas waas_global
 Class AFS
  optimize dre lz application Web
 Class Http
  optimize lz application Filesystem
 Class class-default
```

The following example shows how to create a policy map with Insert-Before:

```
Router(config)# policy-map type waas_global
Router(config-pmap)# class AFS
Router(config-pmap-c)# optimize lz application Filesystem
Router(config-pmap-c)# exit
Router(config-pmap)# class Http insert-before AFS
Router(config-pmap-c)# optimize dre lz application Web
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# exit
Router(config-pmap)# exit
```

The following is sample output from the **show policy-map type waas** command:

```
Router# show policy-map type waas
Policy Map type waas waas_global
 Class Http
  optimize dre lz application Web
 Class AFS
  optimize lz application Filesystem
 Class class-default
```

The following example shows how to create a policy map with sequence numbers:

```
Router(config)# policy-map type waas_global
Router(config-pmap)# sequence-interval 10
Router(config-pmap)# class AFS
Router(config-pmap-c)# optimize dre lz application Web
Router(config-pmap-c)# exit
Router(config-pmap)# class Http
Router(config-pmap-c)# optimize lz application Filesystem
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# exit
Router(config-pmap)# exit
```

The following is sample output from the **show policy-map type waas** command:

```
Router# show policy-map type waas
Policy Map type waas_global
    sequence-interval 10
10 Class AFS
     optimize dre lz application Web
20 Class Http
     optimize lz application Filesystem
30 Class class-default
```

The following example shows how to remove a class from a policy map:

```
Router(config)# policy-map type waas_global
Router(config-pmap)# no class AFS
Router(config-pmap)# exit
```

The following is sample output from the **show policy-map type waas** command:

```
Router# show policy-map type waas
Policy Map type waas_global
    sequence-interval 10
20 Class Http
      optimize dre lz application Web
30 Class class-default
```

Troubleshooting Tips

To clear the DRE cache, enable WAAS Express and execute the **no waas enable** command with the *forced* argument on the interface.

What to Do Next

After defining the policy maps, you must enable WAAS Express.

Enabling WAAS Express

Perform this task to enable WAAS Express on an WAN interface. The **waas enable** command must be explicitly applied on each WAN interface. You can enable WAAS Express by using the default class and policy maps created automatically or define your class and policy maps.

The global policy map governs the behavior of optimization on the interface. All traffic exiting the WAN interface or entering from the WAN interface is screened for optimization according to the global policy map. However, the traffic on other interfaces will not be touched by WAAS Express. WAAS Express supports flows that travel over multiple WAN interfaces, entering one interface and exiting another.

Note WAAS Express does not support the option of selecting a user-defined policy-map to associate with the **waas enable** command. The default policy waas_global is used on the interface where WAAS Express is enabled. You can modify the default waas_global policy. The default WAAS Express policy is extracted from the default WAAS policy.

Perform the following task to enable WAAS Express on an interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-type/number*
4. **waas enable**
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**interface** *interface-type/number* **Example:** `Router(config)# interface GigabitEthernet0/0`	Specifies an interface for configuration and enters interface configuration mode

	Command or Action	Purpose
Step 4	**waas enable** **Example:** `Router(config-if)# waas enable`	Enables WAAS Express on the WAN interface.
Step 5	**exit** **Example:** `Router(config-if)# exit`	Exits the current mode.

Troubleshooting Tips

To troubleshoot, execute the following commands:

- **debug waas**—To detect errors.
- **monitor**—To monitor and collect packet capture.
- **show waas**—To verify the configuration.

The **no waas enable** command neither removes the WAAS Express configuration nor affects the existing flows execute flows that are already optimized by WAAS Express on an interface nor removes the default maps from the device. This command only removes the configuration. To terminate the flows and disable WAAS Express, use the command with the *forced* argument.

Use the **waas config remove-all** command, to remove the default maps from the device.

To replace the policy configuration that you defined with the default policy configuration, use the **waas config restore-default** command. This command replaces the existing policy configuration with the predefined default policy configuration.

Note You can execute the **waas config restore-default** command only if WAAS Express is disabled.

Configuration Examples for WAAS Express

- Configuring WAAS Express: Example

Configuring WAAS Express: Example

```
Router(config)# class-map type waas match-any http
Router(config-cmap)# match tcp destination port 80 80
Router(config-cmap)# match tcp destination port 8080 8082
Router(config-cmap)# exit

Router(config)# class-map type waas waas_global
Router(config-cmap)# match tcp destination port 5190 5193
Router(config-cmap)# exit

Router(config)# class-map type waas match-any bittorrent
Router(config-cmap)# match tcp destination port 6969
Router(config-cmap)# match tcp destination port 6881 6889
Router(config-cmap)# exit
```

```
Router(config)# policy-map type waas global
Router(config-pmap)# class http
Router(config-pmap-c)# optimize DRE LZ application web-traffic
Router(config-pmap-c)# exit

Router(config-pmap)# class aol
Router(config-pmap-c)# optimize tfo-only application IM
Router(config-pmap-c)# exit

Router(config-pmap)# class bittorrent
Router(config-pmap-c)# optimize LZ application p2p
Router(config-pmap-c)# exit
Router(config-pmap)# exit

Router(config)# interface E0
Router(config-if)# description WAN Connection
Router(config-if)# waas enable
Router(config-if)# exit
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	*Cisco IOS Master Commands List, All Releases*
WAN commands	*Cisco IOS Wide-Area Networking Command Reference*
Cisco Wide Area Application Services	• *Cisco Wide Area Application Services Quick Configuration Guide, Software Version 4.2.1* • *Cisco Wide Area Application Services Configuration Guide, Software Version 4.2.1* • *Cisco WAAS Installation and Configuration Guide for ACNS on a Virtual Blade (ACNS-VB)*

Standards

Standard	Title
None	—

MIBs

MIB	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 1323	*TCP Extensions for High Performance*
RFC 2018	*TCP Selective Acknowledgment Options*
RFC 3390	*Increasing TCP's Initial Window*

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for WAAS Express

Table 3 lists the features in this module and provides links to specific configuration information.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 3 lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Table 3 Feature Information for WAAS Express

Feature Name	Releases	Feature Information
WAAS Express	15.1(2)T	Cisco's WAAS Express software, which interoperates with WAN optimization headend applications from Cisco. WAAS Express improves WAN access and use by optimizing applications that require high bandwidth or are bound to a LAN, such as backup. The following sections provide information about this feature: • Information About WAAS Express, page 3 • How to Configure WAAS Express, page 13 The following commands were introduced or modified: **class-map type waas**, **clear waas**, **cpu-threshold**, **debug waas**, **lz entropy-check**, **match tcp**, **optimize**, **parameter-map type waas**, **passthrough**, **policy-map type waas**, **sequence-interval**, **show waas alarms**, **show waas auto-discovery**, **show waas connection**, **show waas statistics aoim**, **show waas statistics application**, **show waas statistics auto-discovery**, **show waas statistics class**, **show waas statistics dre**, **show waas statistics global**, **show waas statistics lz**, **show waas statistics pass-through**, **show waas statistics peer**, **show waas status**, **show waas token**, **tfo auto-discovery**, **tfo optimize**, **waas cm-register url**, **waas config**, **waas enable**, **waas export**.