FTP66 ALG Support for IPv6 Firewalls

The FTP66 ALG Support for IPv6 Firewalls feature allows FTP to work with IPv6 firewalls. This module describes how to configure a firewall, Network Address Translation (NAT), and Stateful NAT64 to work with the FTP66 application-level gateway (ALG).

- Restrictions for FTP66 ALG Support for IPv6 Firewalls, on page 1
- Information About FTP66 ALG Support for IPv6 Firewalls, on page 1
- How to Configure FTP66 ALG Support for IPv6 Firewalls, on page 4
- Configuration Examples for FTP66 ALG Support for IPv6 Firewalls, on page 14
- Additional References for FTP66 ALG Support for IPv6 Firewalls, on page 15
- Feature Information for FTP66 ALG Support for IPv6 Firewalls, on page 16

Restrictions for FTP66 ALG Support for IPv6 Firewalls

The FTP66 ALG does not support the following:

- Box-to-box high availability.
- Per-subscriber firewalls.
- Stateless Network Address Translation 64 (NAT64).
- Virtual routing and forwarding (VRF) when stateful NAT64 is configured.
- Virtual TCP (vTCP) or the breaking up of packets into smaller packets after translation.

Information About FTP66 ALG Support for IPv6 Firewalls

Application-Level Gateways

An application-level gateway (ALG), also known as an application-layer gateway, is an application that translates the IP address information inside the payload of an application packet. An ALG is used to interpret the application-layer protocol and perform firewall and Network Address Translation (NAT) actions. These actions can be one or more of the following depending on your configuration of the firewall and NAT:

- Allow client applications to use dynamic TCP or UDP ports to communicate with the server application.

- Recognize application-specific commands and offer granular security control over them.
- Synchronize multiple streams or sessions of data between two hosts that are exchanging data.
- Translate the network-layer address information that is available in the application payload.

The firewall opens a pinhole, and NAT performs translation service on any TCP or UDP traffic that does not carry the source and destination IP addresses in the application-layer data stream. Specific protocols or applications that embed IP address information require the support of an ALG.

FTP66 ALG Support Overview

Firewalls support the inspection of IPv6 packets and stateful Network Address Translation 64 (NAT64). For FTP to work over IPv6 packet inspection, the application-layer gateway (ALG) (also called the application-level gateway [ALG]), FTP66, is required. The FTP66 ALG is also called all-in-one FTP ALG and one FTP ALG.

The FTP66 ALG supports the following:

- Firewall IPv4 packet inspection
- Firewall IPv6 packet inspection
- NAT configuration
- NAT64 configuration (along with FTP64 support)
- NAT and firewall configuration
- NAT64 and firewall configuration

The FTP66 ALG has the following security vulnerabilities:

- Packet segmentation attack—The FTP ALG state machine can detect segmented packets, and the state machine processing is stopped until a complete packet is received.
- Bounce attack—The FTP ALG does not create doors (for NAT) or pinholes (for firewalls) with a data port number less than 1024. The prevention of a bounce attack is activated only when the firewall is enabled.

FTP Commands Supported by FTP66 ALG

The FTP66 application-level gateway (ALG) is based on RFC 959. This section describes the main RFC 959 and RFC 2428 FTP commands and responses that the FTP66 ALG processes.

PORT Command

The PORT command is used in active FTP mode. The PORT command specifies the address and the port number to which a server should connect. When you use this command, the argument is a concatenation of a 32-bit Internet host address and a 16-bit TCP port address. The address information is broken into 8-bit fields, and the value of each field is transmitted as a decimal number (in character string representation). The fields are separated by commas.

The following is a sample PORT command, where *h1* is the highest order 8-bit of the Internet host address:

```
PORT h1,h2,h3,h4,p1,p2
```

PASV Command

The PASV command requests a server to listen on a data port that is not the default data port of the server and to wait for a connection, rather than initiate another connection, when a TRANSFER command is received. The response to the PASV command includes the host and port address the server is listening on.

Extended FTP Commands

Extended FTP commands provide a method by which FTP can communicate the data connection endpoint information for network protocols other than IPv4. Extended FTP commands are specified in RFC 2428. In RFC 2428, the extended FTP commands EPRT and EPSV, replace the FTP commands PORT and PASV, respectively.

EPRT Command

The EPRT command allows you to specify an extended address for data connection. The extended address must consist of a network protocol, network address, and transport address. The format of an EPRT command is as follows:

```
EPRT<space><d><net-prt><d><net-addr><d><tcp-port><d>
```

- The <net-prt> argument must be an address family number and must be defined as described in the table below.

Table 1: The <net-prt> Argument Definitions

Address Family Number	Protocol
1	IPv4 (Pos81a)
2	IPv6 (DH96)

- The <net-addr> argument is a protocol-specific string representation of the network address. For the two address family numbers specified in the table above (address family numbers 1 and 2), the addresses must be in the format listed in the table below.

Address Family Number	Address Format	Example
1	Dotted decimal	10.135.1.2
2	IPv6 string representations defined in DH96	2001:DB8:1::1

- The <tcp-port> argument must be a string representation of the number of the TCP port on which the host is listening for data connection.
- The following command shows how to specify the server to use an IPv4 address to open a data connection to host 10.235.1.2 on TCP port 6275:

```
EPRT |1|10.235.1.2|6275|
```

- The following command shows how to specify the server to use an IPv6 network protocol and a network address to open a TCP data connection on port 5282:

```
EPRT |2|2001:DB8:2::2:417A|5282|
```

- The <d> argument is the delimiter character and it must be in ASCII format, in the range from 33 to 126.

EPSV Command

The EPSV command requests that a server listen on a data port and wait for a connection. The response to this command includes only the TCP port number of the listening connection. The response code for entering passive mode by using an extended address must be 229.

The text returned in response to an EPSV command must be in the following format:

```
(<d><d><d><tcp-port><d>)
```

- The portion of the string enclosed in parentheses must be the exact string needed by the EPRT command to open the data connection.

  The first two fields in parentheses must be blank. The third field must be a string representation of the TCP port number on which the server is listening for a data connection. The network protocol used by the data connection is the same network protocol used by the control connection. The network address used to establish the data connection is the same network address used for the control connection.

- The following is a sample response string:

  ```
  Entering Extended Passive Mode (|||6446|)
  ```

The following FTP responses and commands are also processed by the FTP66 ALG. The results of processing these commands are used to drive the transition in the state machine.

- 230 response
- AUTH
- USER
- PASS

How to Configure FTP66 ALG Support for IPv6 Firewalls

Configuring a Firewall for FTP66 ALG Support

You need to explicitly enable the FTP66 ALG by using the **match protocol ftp** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **class-map type inspect match-any** *class-map-name*
4. **match protocol** *protocol-name*
5. **exit**
6. **policy-map type inspect** *policy-map-name*
7. **class type inspect** *class-map-name*
8. **inspect**
9. **exit**
10. **class class-default**
11. **exit**

12. **exit**
13. **zone security** *zone-name*
14. **exit**
15. **zone-pair security** *zone-pair* **source** *source-zone* **destination** *destination-zone*
16. **service-policy type inspect** *policy-map-name*
17. **exit**
18. **interface** *type number*
19. **no ip address**
20. **ip virtual-reassembly**
21. **zone-member security** *zone-name*
22. **negotiation auto**
23. **ipv6 address** *ipv6-address/prefix-length*
24. **cdp enable**
25. **exit**
26. **ipv6 route** *ipv6-prefix/prefix-length interface-type interface-number*
27. **ipv6 neighbor** *ipv6-address interface-type interface-number hardware-address*
28. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**class-map type inspect match-any** *class-map-name* **Example:** `Device(config)# class-map type inspect match-any in2out-class`	Creates an inspect type class map and enters QoS class-map configuration mode.
Step 4	**match protocol** *protocol-name* **Example:** `Device(config-cmap)# match protocol ftp`	Configures a match criteria for a class map on the basis of the named protocol.
Step 5	**exit** **Example:** `Device(config-cmap)# exit`	Exits QoS class-map configuration mode and enters global configuration mode.

	Command or Action	Purpose
Step 6	**policy-map type inspect** *policy-map-name* **Example:** `Device(config)# policy-map type inspect in-to-out`	Creates an inspect type policy map and enters QoS policy-map configuration mode.
Step 7	**class type inspect** *class-map-name* **Example:** `Device(config-pmap)# class type inspect in2out-class`	Specifies the class on which an action is performed and enters QoS policy-map class configuration mode.
Step 8	**inspect** **Example:** `Device(config-pmap-c)# inspect`	Enables stateful packet inspection.
Step 9	**exit** **Example:** `Device(config-pmap-c)# exit`	Exits QoS policy-map class configuration mode and enters QoS policy-map configuration mode.
Step 10	**class class-default** **Example:** `Device(config-pmap)# class class-default`	Applies the policy map settings to the predefined default class and enters QoS policy-map class configuration mode. • If the traffic does not match any of the match criteria in the configured class maps, it is directed to the predefined default class.
Step 11	**exit** **Example:** `Device(config-pmap-c)# exit`	Exits QoS policy-map class configuration mode and enters QoS policy-map configuration mode.
Step 12	**exit** **Example:** `Device(config-pmap)# exit`	Exits QoS policy-map configuration mode and enters global configuration mode.
Step 13	**zone security** *zone-name* **Example:** `Device(config)# zone security inside`	Creates a security zone to which interfaces can be assigned and enters security zone configuration mode. • Your configuration must have two security zones to create a zone pair: a source and a destination zone. • In a zone pair, you can use the default zone as either the source or the destination zone.
Step 14	**exit** **Example:** `Device(config-sec-zone)# exit`	Exits security zone configuration mode and enters global configuration mode.

	Command or Action	Purpose
Step 15	**zone-pair security** *zone-pair* **source** *source-zone* **destination** *destination-zone* **Example:** `Device(config)# zone-pair security in2out source inside destination outside`	Creates a pair of security zones and enters security zone-pair configuration mode. • To apply a policy, you must configure a zone pair.
Step 16	**service-policy type inspect** *policy-map-name* **Example:** `Device(config-sec-zone-pair)# service-policy type inspect in-to-out`	Attaches a firewall policy map to the destination zone pair. • If a policy is not configured between a pair of zones, traffic is dropped by default.
Step 17	**exit** **Example:** `Device(config-sec-zone-pair)# exit`	Exits security zone-pair configuration mode and enters global configuration mode.
Step 18	**interface** *type number* **Example:** `Device(config)# interface gigabitethernet 0/0/1`	Configures an interface and enters interface configuration mode.
Step 19	**no ip address** **Example:** `Device(config-if)# no ip address`	Removes an IP address or disables IP processing.
Step 20	**ip virtual-reassembly** **Example:** `Device(config-if)# ip virtual-reassembly`	Enables virtual fragmentation reassembly (VFR) on an interface.
Step 21	**zone-member security** *zone-name* **Example:** `Device(config-if)# zone-member security inside`	Assigns an interface to a specified security zone. • When you make an interface a member of a security zone, all traffic into and out of that interface (except traffic bound for the device or initiated by the device) is dropped by default. To let traffic through the interface, you must make the zone part of a zone pair to which you apply a policy. If the policy permits traffic, traffic can flow through that interface.
Step 22	**negotiation auto** **Example:** `Device(config-if)# negotiation auto`	Enables the autonegotiation protocol to configure the speed, duplex, and automatic flow control of the Gigabit Ethernet interface.
Step 23	**ipv6 address** *ipv6-address/prefix-length* **Example:** `Device(config-if)# ipv6 address 2001:DB8:1::1/96`	Configures an IPv6 address based on an IPv6 general prefix and enables IPv6 processing on an interface.

	Command or Action	Purpose
Step 24	**cdp enable** **Example:** `Device(config-if)# cdp enable`	Enables Cisco Discovery Protocol on an interface.
Step 25	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 26	**ipv6 route** *ipv6-prefix/prefix-length interface-type interface-number* **Example:** `Device(config)# ipv6 route 2001::/96 gigabitethernet 0/0/1`	Establishes static IPv6 routes.
Step 27	**ipv6 neighbor** *ipv6-address interface-type interface-number hardware-address* **Example:** `Device(config)# ipv6 neighbor 2001:DB8:1::1 gigabitethernet 0/0/1 0000.29f1.4841`	Configures a static entry in the IPv6 neighbor discovery cache.
Step 28	**end** **Example:** `Device(config)# end`	Exits global configuration mode and enters privileged EXEC mode.

Configuring NAT for FTP66 ALG Support

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip address** *ip-address mask*
5. **ip nat inside**
6. **zone-member security** *zone-name*
7. **exit**
8. **interface** *type number*
9. **ip address** *ip-address mask*
10. **ip nat outside**
11. **zone-member security** *zone-name*
12. **exit**
13. **ip nat inside source static** *local-ip global-ip*
14. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type number* **Example:** `Device(config)# interface gigabitethernet 0/1/2`	Configures an interface and enters interface configuration mode.
Step 4	**ip address** *ip-address mask* **Example:** `Device(config-if)# ip address 10.1.1.1 255.255.255.0`	Sets a primary or secondary IP address for an interface.
Step 5	**ip nat inside** **Example:** `Device(config-if)# ip nat inside`	Indicates that an interface is connected to the inside network (the network that is subject to NAT translation).
Step 6	**zone-member security** *zone-name* **Example:** `Device(config-if)# zone-member security inside`	Assigns an interface to a specified security zone. • When you make an interface a member of a security zone, all traffic into and out of that interface (except traffic bound for the device or initiated by the device) is dropped by default. To let traffic through the interface, you must make the zone part of a zone pair to which you apply a policy. If the policy permits traffic, traffic can flow through that interface.
Step 7	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 8	**interface** *type number* **Example:** `Device(config)# interface gigabitethernet 0/1/1`	Configures an interface and enters interface configuration mode.
Step 9	**ip address** *ip-address mask* **Example:**	Indicates that an interface is connected to the inside network (the network that is subject to NAT translation).

	Command or Action	Purpose
	`Device(config-if)# ip address 10.2.1.1 255.255.255.0`	
Step 10	**ip nat outside** **Example:** `Device(config-if)# ip nat outside`	Indicates that the interface is connected to the outside network.
Step 11	**zone-member security** *zone-name* **Example:** `Device(config-if)# zone-member security outside`	Assigns an interface to a specified security zone. • When you make an interface a member of a security zone, all traffic into and out of that interface (except traffic bound for the device or initiated by the device) is dropped by default. To let traffic through the interface, you must make the zone part of a zone pair to which you apply a policy. If the policy permits traffic, traffic can flow through that interface.
Step 12	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 13	**ip nat inside source static** *local-ip global-ip* **Example:** `Device(config)# ip nat inside source static 10.1.1.10 10.1.1.80`	Enables NAT of the inside source address.
Step 14	**end** **Example:** `Device(config)# end`	Exits global configuration mode and enters privileged EXEC mode.

Configuring NAT64 for FTP66 ALG Support

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 unicast-routing**
4. **interface** *type number*
5. **no ip address**
6. **ipv6 virtual-reassembly**
7. **zone-member security** *zone-name*
8. **negotiation auto**
9. **ipv6 address** *ipv6-address*
10. **ipv6 enable**
11. **nat64 enable**
12. **cdp enable**

13. **exit**
14. **interface** *type number*
15. **ip address** *type number*
16. **ip virtual-reassembly**
17. **zone member security** *zone-name*
18. **negotiation auto**
19. **nat64 enable**
20. **exit**
21. **ipv6 route** *ipv6-address interface-type interface-number*
22. **ipv6 neighbor** *ipv6-address interface-type interface-number hardware-address*
23. **nat64 v6v4 static** *ipv6-address ipv4-address*
24. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 unicast-routing** **Example:** `Device(config)# ipv6 unicast-routing`	Enables the forwarding of IPv6 unicast datagrams.
Step 4	**interface** *type number* **Example:** `Device(config)# interface gigabitethernet 0/0/0`	Configures an interface and enters interface configuration mode.
Step 5	**no ip address** **Example:** `Device(config-if)# no ip address`	Removes an IP address or disables IP processing.
Step 6	**ipv6 virtual-reassembly** **Example:** `Device(config-if)# ipv6 virtual-reassembly`	Enables virtual fragmentation reassembly (VFR) on an interface.
Step 7	**zone-member security** *zone-name* **Example:** `Device(config-if)# zone-member security inside`	Assigns an interface to a specified security zone. • When you make an interface a member of a security zone, all traffic into and out of that interface (except

	Command or Action	Purpose
		traffic bound for the device or initiated by the device) is dropped by default. To let traffic through the interface, you must make the zone part of a zone pair to which you apply a policy. If the policy permits traffic, traffic can flow through that interface.
Step 8	**negotiation auto** **Example:** `Device(config-if)# negotiation auto`	Enables the autonegotiation protocol to configure the speed, duplex, and automatic flow control of the Gigabit Ethernet interface.
Step 9	**ipv6 address** *ipv6-address* **Example:** `Device(config-if)# ipv6 address 2001:DB8:1::2/96`	Configures an IPv6 address based on an IPv6 general prefix and enables IPv6 processing on an interface.
Step 10	**ipv6 enable** **Example:** `Device(config-if)# ipv6 enable`	Enables IPv6 processing on an interface that has not been configured with an explicit IPv6 address.
Step 11	**nat64 enable** **Example:** `Device(config-if)# nat64 enable`	Enables NAT64 on an interface.
Step 12	**cdp enable** **Example:** `Device(config-if)# cdp enable`	Enables Cisco Discovery Protocol on an interface.
Step 13	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 14	**interface** *type number* **Example:** `Device(config)# interface gigabitethernet 0/1/1`	Configures an interface and enters interface configuration mode.
Step 15	**ip address** *type number* **Example:** `Device(config-if)# ip address 209.165.201.25 255.255.255.0`	Sets a primary or secondary IP address for an interface.
Step 16	**ip virtual-reassembly** **Example:** `Device(config-if)# ip virtual-reassembly`	Enables VFR on an interface.

	Command or Action	Purpose
Step 17	**zone member security** *zone-name* **Example:** `Device(config-if)# zone member security outside`	Assigns an interface to a specified security zone. • When you make an interface a member of a security zone, all traffic into and out of that interface (except traffic bound for the router or initiated by the router) is dropped by default. To let traffic through the interface, you must make the zone part of a zone pair to which you apply a policy. If the policy permits traffic, traffic can flow through that interface.
Step 18	**negotiation auto** **Example:** `Device(config-if)# negotiation auto`	Enables the autonegotiation protocol to configure the speed, duplex, and automatic flow control of the Gigabit Ethernet interface.
Step 19	**nat64 enable** **Example:** `Device(config-if)# nat64 enable`	Enables NAT64 on an interface.
Step 20	**exit** **Example:** `Device(config-if)# exit`	Exits interface configuration mode and enters global configuration mode.
Step 21	**ipv6 route** *ipv6-address interface-type interface-number* **Example:** `Device(config)# ipv6 route 2001:DB8:1::2/96 gigabitethernet 0/0/0`	Establishes static IPv6 routes and specifies the IPv6 address of the next hop that can be used to reach a specified network.
Step 22	**ipv6 neighbor** *ipv6-address interface-type interface-number hardware-address* **Example:** `Device(config)# ipv6 neighbor 2001:DB8:1::103 gigabitethernet 0/0/0 0000.29f1.4841`	Configures a static entry in the IPv6 neighbor discovery cache.
Step 23	**nat64 v6v4 static** *ipv6-address ipv4-address* **Example:** `Device(config)# nat64 v6v4 static 2001:DB8:1::103 209.165.201.32`	Translates an IPv6 source address to an IPv4 source address and an IPv4 destination address to an IPv6 destination address for NAT64.
Step 24	**end** **Example:** `Device(config)# end`	Exits global configuration mode and enters privileged EXEC mode.

Configuration Examples for FTP66 ALG Support for IPv6 Firewalls

Example: Configuring an IPv6 Firewall for FTP66 ALG Support

```
Device# configure terminal
Device(config)# class-map type inspect match-any in2out-class
Device(config-cmap)# match protocol ftp
Device(config-cmap)# exit
Device(config)# policy-map type inspect in-to-out
Device(config-pmap)# class type inspect in2out-class
Device(config-pmap-c)# inspect
Device(config-pmap-c)# exit
Device(config-pmap)# class class-default
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# zone security inside
Device(config-sec-zone)# exit
Device(config)# zone security outside
Device(config-sec-zone)# exit
Device(config)# zone-pair security in2out source inside destination outside
Device(config-sec-zone-pair)# service-policy type inspect in-to-out
Device(config-sec-zone-pair)# exit
Device(config)# interface gigabitethernet 0/0/1
Device(config-if)# no ip address
Device(config-if)# ip virtual-reassembly
Device(config-if)# zone-member security inside
Device(config-if)# negotiation auto
Device(config-if)# ipv6 address 2001:DB8:1::1/96
Device(config-if)# cdp enable
Device(config-if)# exit
Device(config)# interface gigabitethernet 0/1/1
Device(config-if)# no ip address
Device(config-if)# ip virtual-reassembly
Device(config-if)# zone-member security outside
Device(config-if)# negotiation auto
Device(config-if)# ipv6 address 2001:DB8:2::2/96
Device(config-if)# exit
Device(config)# ipv6 route 2001::/96 gigabitethernet 0/0/1
Device(config)# ipv6 route 2001::/96 gigabitethernet 0/1/1
Device(config)# ipv6 neighbor 2001:DB8:1::1 gigabitethernet 0/0/1 0000.29f1.4841
Device(config)# ipv6 neighbor 2001:DB8:2::2 gigabitethernet 0/1/1 0000.29f1.4842
Device(config)# end
```

Example: Configuring NAT for FTP66 ALG Support

```
Device# configure terminal
Device(config)# interface gigabitethernet 0/1/2
Device(config-if)# ip address 10.1.1.1 255.255.255.0
Device(config-if)# ip nat inside
Device(config-if)# zone-member security inside
Device(config-if)# exit
Device(config)# interface gigabitethernet 0/1/1
Device(config-if)# ip address 10.2.1.1 255.255.255.0
Device(config-if)# ip nat outside
Device(config-if)# zone-member security outside
```

```
Device(config-if)# exit
Device(config-if)# ip nat inside source static 10.1.1.10 10.1.1.80
```

Example: Configuring NAT64 for FTP66 ALG Support

```
Device# configure terminal
Device(config)# ipv6 unicast-routing
Device(config)# interface gigabitethernet 0/0/0
Device(config-if)# no ip address
Device(config-if)# ipv6 virtual-reassembly
Device(config-if)# zone-member security inside
Device(config-if)# negotiation auto
Device(config-if)# ipv6 address 2001:DB8:1::2/96
Device(config-if)# ipv6 enable
Device(config-if)# nat64 enable
Device(config-if)# cdp enable
Device(config-if)# exit
Device(config)# interface gigabitethernet 0/1/1
Device(config-if)# ip address 209.165.201.25 255.255.255.0
Device(config-if)# ip virtual-reassembly
Device(config-if)# zone member security outside
Device(config-if)# negotiation auto
Device(config-if)# nat64 enable
Device(config-if)# exit
Device(config)# ipv6 route 2001:DB8:1::2/96 gigabitethernet 0/0/0
Device(config)# 2001:DB8:1::103 gigabitethernet 0/0/0 0000.29f1.4841
Device(config)# nat64 v6v4 static 2001:DB8:1::103 209.165.201.32
```

Additional References for FTP66 ALG Support for IPv6 Firewalls

Related Documents

Related Topic	Document Title
Cisco IOS commands	Master Command List, All Releases
Security commands	• Security Command Reference: Commands A to C • Security Command Reference: Commands D to L • Security Command Reference: Commands M to R • Security Command Reference: Commands S to Z
NAT commands	IP Addressing Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 959	*File Transfer Protocol*
RFC 2428	*FTP Extensions for IPv6 and NATs*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for FTP66 ALG Support for IPv6 Firewalls

The following table provides release information about the feature or features described in this module. This table 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.

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

Table 2: Feature Information for FTP66 ALG Support for IPv6 Firewalls

Feature Name	Releases	Feature Information
FTP66 ALG Support for IPv6 Firewalls	Cisco IOS XE Release 3.7S	The FTP66 ALG Support for IPv6 Firewalls feature allows FTP to work with IPv6 firewalls. This module describes how to configure a firewall, Network Address Translation (NAT), and NAT64 to work with the FTP66 application-level gateway (ALG).