

Configuring TCP Intercept (Prevent Denial-of-Service Attacks)

This chapter describes how to configure your router to protect TCP servers from TCP SYN-flooding attacks, a type of denial-of-service attack. This is accomplished by configuring the Cisco IOS feature known as “TCP Intercept.”

For a complete description of TCP Intercept commands, refer to the “TCP Intercept Commands” chapter of the *Security Command Reference*. To locate documentation of other commands that appear in this chapter, use the command reference master index or search online.

In This Chapter

This chapter has the following sections:

- About TCP Intercept
- TCP Intercept Configuration Task List
- Enable TCP Intercept
- Set the TCP Intercept Mode
- Set the TCP Intercept Drop Mode
- Change the TCP Intercept Timers
- Change the TCP Intercept Aggressive Thresholds
- Monitor and Maintain TCP Intercept
- TCP Intercept Configuration Example

About TCP Intercept

The TCP intercept feature implements software to protect TCP servers from TCP SYN-flooding attacks, which are a type of denial-of-service attack.

A SYN-flooding attack occurs when a hacker floods a server with a barrage of requests for connection. Since these messages have unreachable return addresses, the connections cannot be established. The resulting volume of unresolved open connections eventually overwhelms the server and can cause it to deny service to valid requests, thereby preventing legitimate users from connecting to a Web site, accessing e-mail, using FTP service, and so on.

The TCP intercept feature helps prevent SYN-flooding attacks by intercepting and validating TCP connection requests. In intercept mode, the TCP intercept software intercepts TCP synchronization (SYN) packets from clients to servers that match an extended access list. The software establishes a connection with the client on behalf of the destination server, and if successful, establishes the

connection with the server on behalf of the client and knits the two half-connections together transparently. Thus, connection attempts from unreachable hosts will never reach the server. The software continues to intercept and forward packets throughout the duration of the connection.

In the case of illegitimate requests, the software's aggressive timeouts on half-open connections and its thresholds on TCP connection requests protect destination servers while still allowing valid requests.

When establishing your security policy using TCP intercept, you can choose to intercept all requests or only those coming from specific networks or destined for specific servers. You can also configure the connection rate and threshold of outstanding connections.

You can choose to operate TCP intercept in watch mode, as opposed to intercept mode. In watch mode, the software passively watches the connection requests flowing through the router. If a connection fails to get established in a configurable interval, the software intervenes and terminates the connection attempt.

TCP options that are negotiated on handshake (such as RFC 1323 on window scaling, for example) will not be negotiated because the TCP intercept software does not know what the server can do or will negotiate.

TCP Intercept Configuration Task List

Perform the following tasks to configure TCP intercept. The first task is required; the rest are optional.

- Enable TCP Intercept
- Set the TCP Intercept Mode
- Set the TCP Intercept Drop Mode
- Change the TCP Intercept Timers
- Change the TCP Intercept Aggressive Thresholds
- Monitor and Maintain TCP Intercept

Enable TCP Intercept

To enable TCP intercept, perform the following tasks in global configuration mode:

Task	Command
Step 1 Define an IP extended access list.	access-list <i>access-list-number</i> { deny permit } tcp any <i>destination destination-wildcard</i>
Step 2 Enable TCP intercept.	ip tcp intercept list <i>access-list-number</i>

You can define an access list to intercept all requests or only those coming from specific networks or destined for specific servers. Typically the access list will define the source as **any** and define specific destination networks or servers. That is, you do not attempt to filter on the source addresses because you don't necessarily know who to intercept packets from. You identify the destination in order to protect destination servers.

If no access list match is found, the router allows the request to pass with no further action.

Set the TCP Intercept Mode

The TCP intercept can operate in either active intercept mode or passive watch mode. The default is intercept mode.

In intercept mode, the software actively intercepts each incoming connection request (SYN) and responds on behalf of the server with an ACK and SYN, then waits for an ACK of the SYN from the client. When that ACK is received, the original SYN is set to the server and the software performs a three-way handshake with the server. When this is complete, the two half-connections are joined.

In watch mode, connection requests are allowed to pass through the router to the server but are watched until they become established. If they fail to become established within 30 seconds (configurable with the **ip tcp intercept watch-timeout** command), the software sends a Reset to the server to clear up its state.

To set the TCP intercept mode, perform the following task in global configuration mode:

Task	Command
Set the TCP intercept mode.	ip tcp intercept mode {intercept watch}

Set the TCP Intercept Drop Mode

When under attack, the TCP intercept feature becomes more aggressive in its protective behavior. If the number of incomplete connections exceeds 1100 or the number of connections arriving in the last one minute exceeds 1100, each new arriving connection causes the oldest partial connection to be deleted. Also, the initial retransmission timeout is reduced by half to 0.5 seconds (so the total time trying to establish a connection is cut in half).

By default, the software drops the oldest partial connection. Alternatively, you can configure the software to drop a random connection. To set the drop mode, perform the following task in global configuration mode:

Task	Command
Set the drop mode.	ip tcp intercept drop-mode {oldest random}

Change the TCP Intercept Timers

By default, the software waits for 30 seconds for a watched connection to reach established state before sending a Reset to the server. To change this value, perform the following task in global configuration mode:

Task	Command
Change the time allowed to reach established state.	ip tcp intercept watch-timeout <i>seconds</i>

By default, the software waits for 5 seconds from receipt of a reset or FIN-exchange before it ceases to manage the connection. To change this value, perform the following task in global configuration mode:

Task	Command
Change the time between receipt of a reset or FIN-exchange and dropping the connection.	ip tcp intercept finrst-timeout <i>seconds</i>

Change the TCP Intercept Aggressive Thresholds

By default, the software still manages a connection for 24 hours after no activity. To change this value, perform the following task in global configuration mode:

Task	Command
Change the time the software will manage a connection after no activity.	ip tcp intercept connection-timeout <i>seconds</i>

Change the TCP Intercept Aggressive Thresholds

Two factors determine when aggressive behavior begins and ends: total incomplete connections and connection requests during the last one-minute sample period. Both thresholds have default values that can be redefined.

When a threshold is exceeded, the TCP intercept assumes the server is under attack and goes into aggressive mode. When in aggressive mode, the following occurs:

- Each new arriving connection causes the oldest partial connection to be deleted. (You can change to a random drop mode.)
- The initial retransmission timeout is reduced by half to 0.5 seconds, and so the total time trying to establish the connection is cut in half. (When not in aggressive mode, the code does exponential back-off on its retransmissions of SYN segments. The initial retransmission timeout is 1 second. The subsequent timeouts are 2 seconds, 4 seconds, 8 seconds, and 16 seconds. The code retransmits 4 times before giving up, so it gives up after 31 seconds of no acknowledgment.)
- If in watch mode, the watch timeout is reduced by half. (If the default is in place, the watch timeout becomes 15 seconds).

The drop strategy can be changed from the oldest connection to a random connection with the **ip tcp intercept drop-mode** command.

Note The two factors that determine aggressive behavior are related and work together. When *either* of the **high** values is exceeded, aggressive behavior begins. When *both* quantities fall below the **low** value, aggressive behavior ends.

You can change the threshold for triggering aggressive mode based on the total number of incomplete connections. The default values for **low** and **high** are 900 and 1100 incomplete connections, respectively. To change these values, perform the following tasks in global configuration mode:

Task	Command
Set the threshold for stopping aggressive mode.	ip tcp intercept max-incomplete low <i>number</i>
Set the threshold for triggering aggressive mode.	ip tcp intercept max-incomplete high <i>number</i>

You can also change the threshold for triggering aggressive mode based on the number of connection requests received in the last 1-minute sample period. The default values for **low** and **high** are 900 and 1100 connection requests, respectively. To change these values, perform the following tasks in global configuration mode:

Task	Command
Set the threshold for stopping aggressive mode.	ip tcp intercept one-minute low <i>number</i>
Set the threshold for triggering aggressive mode.	ip tcp intercept one-minute high <i>number</i>

Monitor and Maintain TCP Intercept

To display TCP intercept information, perform either of the following tasks in EXEC mode:

Task	Command
Display incomplete connections and established connections.	show tcp intercept connections
Display TCP intercept statistics.	show tcp intercept statistics

TCP Intercept Configuration Example

The following configuration defines extended IP access list 101, causing the software to intercept packets for all TCP servers on the 192.168.1.0/24 subnet:

```
ip tcp intercept list 101
!
access-list 101 permit tcp any 192.168.1.0 0.0.0.255
```

