- TCP Intercept -

TCP Intercept

TCP Intercept was developed to protect servers and other resources from denial of service (DoS) attacks, specifically TCP SYN attacks.

Just as the name says, TCP Intercept **captures** incoming TCP requests. Instead of allowing direct access to the server, TCP Intercept acts as an intermediary, establishing a connection to the server on behalf of the requesting client.

TCP Intercept will *block* a client if too many incoming connections are attempted.

To configure TCP Intercept, the desired traffic to be monitored must be identified. Traffic can be monitored *from* a certain address or network, *to* a certain address or network, or both:

Router(config)# *access-list 101 permit ip any 10.1.1.1 0.0.0.0*

This matches traffic from *any* source to the host *10.1.1.1*. TCP Intercept can then be configured to use this access list:

Router(config)# *ip tcp intercept list 101*

TCP Intercept can operate in one of two modes:

Router(config)# *ip tcp intercept mode intercept*

Router(config)# *ip tcp intercept mode watch*

In *intercept* (the **default**) mode, the router will actively capture TCP connections, and act as the buffer between the client and the server. To adjust how long TCP Intercept will manage a connection after no activity:

Router(config)# *ip tcp intercept connection-timeout 1800*

In *watch* mode, TCP connections pass through the router to the server, but are "observed" by the router. If a connection is not established within 30 seconds (by default), the router send a reset to the server to close down the session. This *watch* timer is configurable:

Router(config)# *ip tcp intercept watch-timeout 15*

* * *

TCP Intercept (continued)

Two thresholds can be configured with TCP Intercept, relating to the number of inbound TCP connections.

If the number of connections exceeds the *high* threshold (**1100** by default), TCP Intercept will begin aggressively dropping connections. By default, TCP Intercept will drop the oldest connections first, but can be configured to drop connections randomly instead:

Router(config)# *ip tcp intercept drop-mode random*

Router(config)# *ip tcp intercept drop-mode oldest*

TCP Intercept will *stop* dropping connections once the number falls below the *low* threshold (**900** by default). To configure the thresholds:

Router(config)# *ip tcp intercept max-incomplete low 600*

Router(config)# *ip tcp intercept max-incomplete high 800*

To troubleshoot TCP Intercept:

Router# *show tcp intercept connections*

Router# *show tcp intercept statistics*

* * *