**How to enable TCP Fast Open in NetScaler?**

TCP Fast Open (TFO) is a mechanism in TCP connection establishment process, which helps to speed up the opening of the connections and data flow. It allows data to be carried during the initial TCP connection handshake, i.e. in SYN and SYN-ACK packets and enables the data to be consumed by the receiving node during the connection establishment thus speeding up data transfer while the connection is being established.

**Why TCP Fast Open?**

Let us understand how TFO works and how it is useful under different use cases. The idea of TFO originated because of the increasing performance requirements of today’s applications.

TFO process saves up to one full round trip time commonly referred as RTT in TCP handshake when compared with the conventional TCP three way handshake. For applications that have short web transfers, this additional RTT makes a significant impact in overall latency in the network.

For example in Chrome browser, it is found out that on an average, one-third of the connections are new TCP connections and thus with TFO involved, it can provide substantial improvements in performance of the network.

Now what about the security of data that is transferred in the initial handshake? TFO makes the data exchange secure by using a TFO cookie, a cryptographic cookie which is sent from receiving node and gets stored on the client who initiated the connection. When the client tries connect to the same node again, it sends the TFO cookie along with the SYN packet during the handshake thus authenticating itself with the receiving node.

Up on successful authentication, the receiving node will send data to the client without receiving the final acknowledgement thereby saving one RTT to start data transmission. This reduces the overall network latency to a significant level when short lived connections are very high.
TCP TFO flow

1. **Server**
   - Receives SYN + TFO cookie request
   - Generates cookie by encrypting client IP
   - Sends SYN-ACK + TFO cookie
2. **Client**
   - Caches cookie for this server IP
3. **Server**
   - Receives SYN + TFO cookie + data
   - Validates client TFO cookie +
     accepts connection + data is made available to application
   - Sends SYN + ACK for SYN+ Data in SYN packet
   - Sends more data packets to client while handshake is in progress

To enable TCP TFO by using the NetScaler GUI

Navigate to **System > Profiles > TCP Profiles** and click **Edit** after selecting the TCP Profile to be edited
Navigate to the bottom of TCP Profile and check TCP Fast Open and click OK to enable it.
To enable TCP TFO by using the NetScaler Command prompt

>set tcpprofile nstcp_default_profile -tcpFastOpen ENABLED

show tcpprofile

1) Name: nstcp_default_profile
   Window Scaling status: DISABLED
   Window Scaling factor: 4
   Selective Acknowledgement(SACK) status: DISABLED
   Maximum Segment Size(MSS): 1460
   Maximum TCP segments allowed in a burst: 6 MSS
   Initial congestion window(cwnd) setting : 4 MSS
   TCP Delayed-ACK Timer: 100 millisec
   Nagle's Algorithm: DISABLED
   Maximum out-of-order packets to queue: 64
   Immediate ACK on PUSH packet: ENABLED
   Maximum packets per MSS: 0
   Maximum packets per retransmission: 1
   TCP minimum Retransmission Timeout(RTO) in millisec: 1000
   TCP Slow start increment: 2
   TCP Buffer Size: 8190 bytes
   TCP Send Buffer Size: 8190 bytes
   TCP Syncookie: ENABLED
   Update Last activity on KA Probes: ENABLED
   TCP flavor: Default
   TCP Dynamic Receive Buffering: DISABLED
   Keep-alive probes: DISABLED
   Connection idle time before starting keep-alive probes: 900 seconds
   Keep-alive probe interval: 75 seconds
   Maximum keep-alive probes to be missed before dropping connection: 3
   Establishing Client Connection: AUTOMATIC
   TCP Segmentation Offload: AUTOMATIC
   TCP Timestamp Option: DISABLED
   RST window attenuation (spoof protection): DISABLED
   Accept RST with last acknowledged sequence number: ENABLED
   SYN spoof protection: DISABLED
   TCP Explicit Congestion Notification(ECN): DISABLED
   Multipath TCP: DISABLED
   Multipath TCP drop data on pre-established subflow: DISABLED
   Multipath TCP fastopen: DISABLED
   Multipath TCP session timeout: 0 seconds
   Duplicate Selective Acknowledgement(DSACK): ENABLED
   ACK Aggregation: DISABLED
   Forward RTO recovery(FRTO): DISABLED
TCP Max congestion window(CWND): 524288 bytes
TCP Forward Acknowledgment(FACK): DISABLED
TCP Optimization mode: TRANSPARENT
TCP Fastopen: ENABLED  >>> TCP Fast Open is Enabled
TCP Hybrid Start(HYSTART): DISABLED
TCP dupack threshold: 3
Burst Rate Control: DISABLED
TCP Rate: 0
TCP Rate Maximum Queue: 0
Reference count: 21

Note: Default value for TCP Fast Open is DISABLED. NetScaler supports TCP Fast open feature from release 11.1 onwards.

Thus using TCP Fast Open feature in NetScaler, overall network latency can be improved significantly in networks with short lived web connections and high rate of new TCP connections.