High-Speed Add-On Getting Started Guide

Version 1.0

Contents

Introduction

The Connect:Direct High Speed Add-On (HSAO) feature gives Connect:Direct the ability to use IBM Aspera's FASP protocol to transfer files. Under certain network conditions common in MFT, the FASP protocol transfers data much faster than other reliable network protocols, such as TCP/IP, and it can significantly accelerate file transfers.

When should HSAO be used?

In two very common (and often related) file transfer cases, TCP is unable to take advantage of all of the available bandwidth due to limitations in its flow control diagrams. HSAO accelerates file transfers in these same cases[1]:

1. When there is packet loss on the network link between the source and destination locations.
2. When there is latency on the network link between the source and destination locations. Latency can occur when the locations are physically far apart, for example, on the opposite sides of the U.S. or on different continents.

Packet loss can occur on any link, but because the packet loss rate typically increases with the length of the link, it often occurs on long network links. Hence, long network links are prime candidates for HSAO. See graph 1 for the combined effects of packet loss and latency on HSAO and TCP.

Additional factors to consider when deciding whether to use HSAO include:

1. The bandwidth of the link - The HSAO advantage over TCP increases with increasing bandwidth. See Graphs 2 and 3.
2. The size of the file being transferred - HSAO session establishment has greater overhead then TCP session establishment, and it typically takes longer to establish an HSAO session than a TCP session. Hence, larger files benefit more from HSAO. As a general rule, to see significant benefits, HSAO should be used for files of size greater than 1 GB. However, unusually high latency or packet loss, or a combination of the two, may lead to significant benefits for smaller files.
3. CPU usage - HSAO requires more CPU usage than TCP to overcome the effects of packet loss and latency. Typically, HSAO uses 2 to 3 times more CPU than TCP.
4. Secure+ - The higher CPU usage caused by using Secure+ may reduce the transfer rate of both HSAO and TCP. However, as long as a system's CPU is not saturated, Secure+ does not remove the advantage of HSAO over TCP. See graphs 4 and 6.
5. Deploying Sterling Secure Proxy (SSP) between the two Connect:Direct nodes - See the "Support through High-Speed Bridging" section, which describes how SSP can act as a protocol inspecting proxy for HSAO and can provide HSAO support itself. In either case, when SSP is deployed, HSAO maintains its advantage over TCP. See graphs 5 and 6.

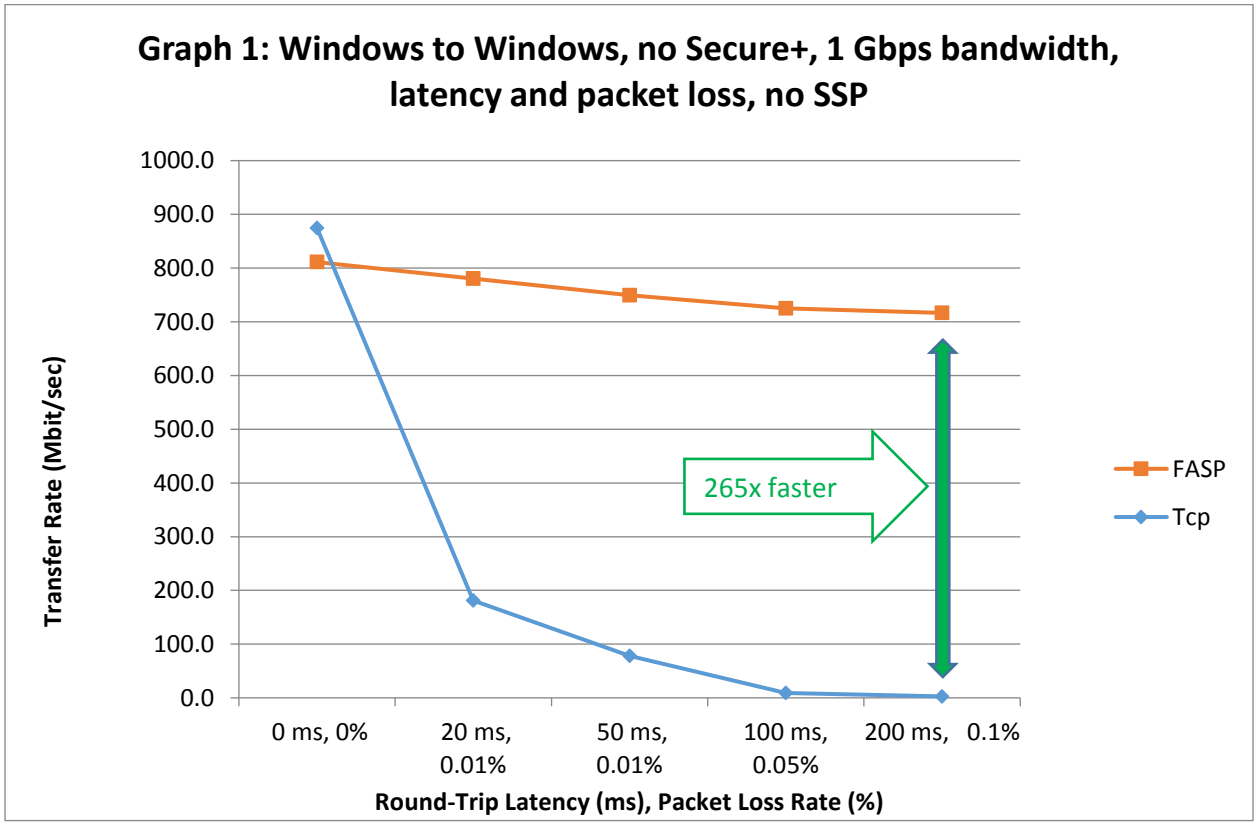
[1] Note that when the HSAO policy is set to Fair, its default and recommended value, HSAO, like TCP, shares network bandwidth fairly with other network traffic - its advantage over TCP is not obtained by taking more than its fair share of network bandwidth and hence starving non-HSAO applications of bandwidth.

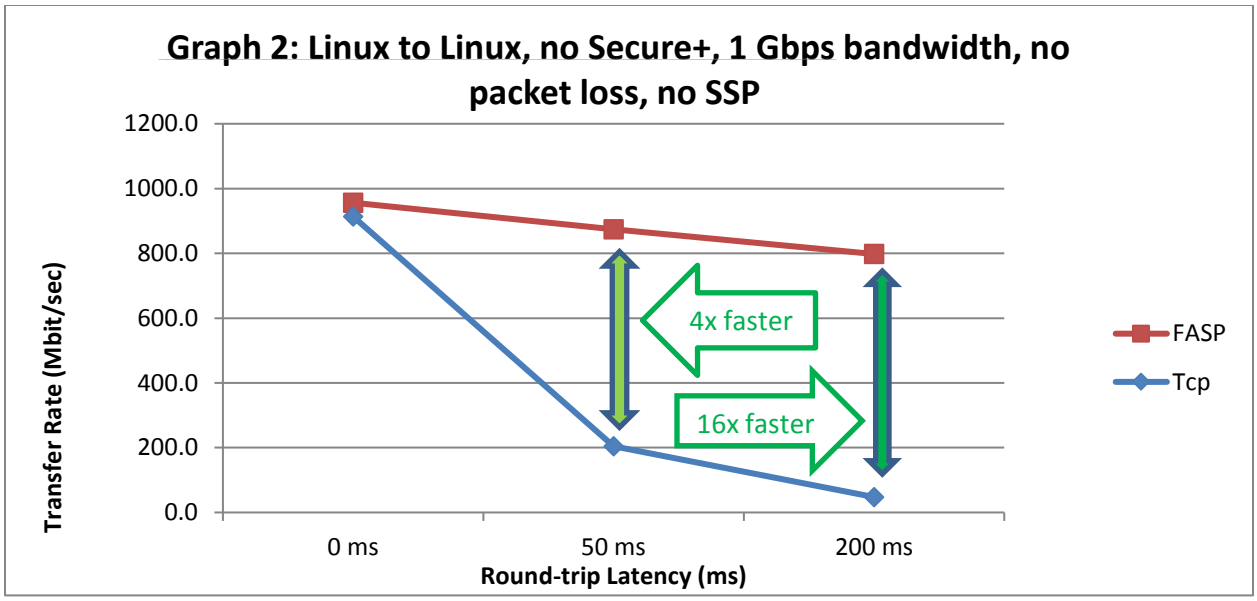
6. Concurrency - HSAO outperforms TCP under conditions in which TCP is unable to use all the available bandwidth. Increased concurrency may enable TCP to use more of the available bandwidth and therefore reduce HSAO's advantage over TCP.

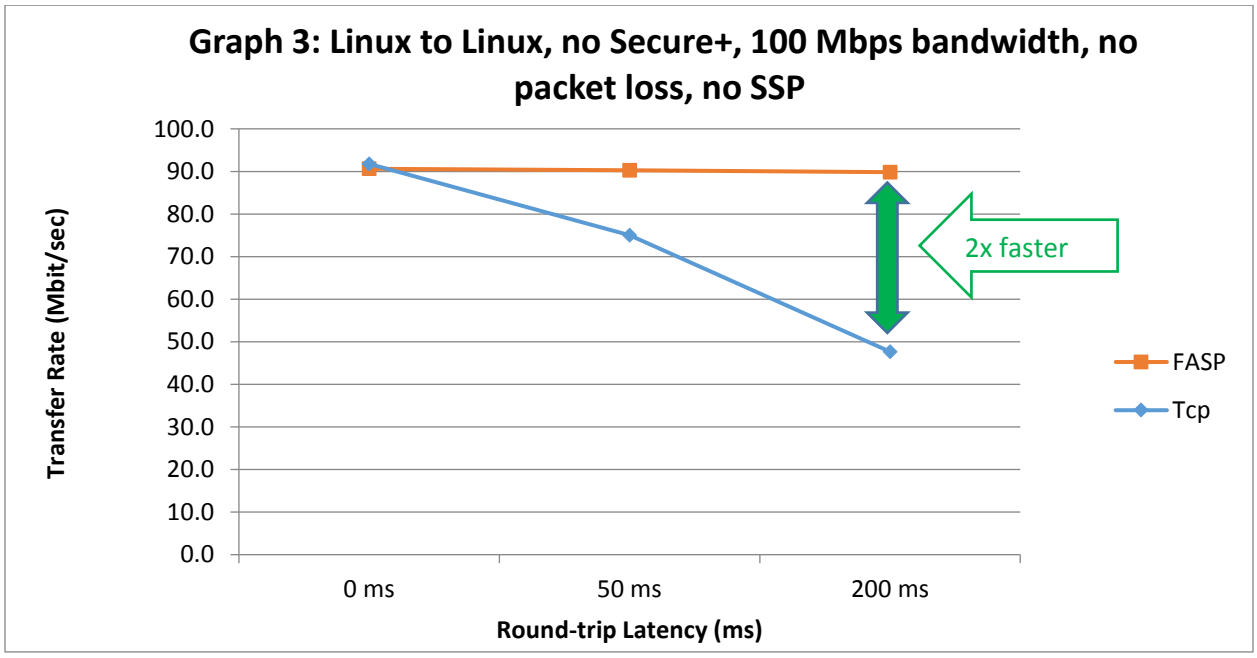
Because of the number of factors involved, there is no simple rule or formula for deciding when HSAO should be used. Use the following results that were measured in a laboratory using a network emulator to help you determine whether using HSAO will be beneficial in your environment. Your network link conditions will not likely match those covered by the following graphs, but one or more approximate matches can provide an indication of whether HSAO can benefit you.

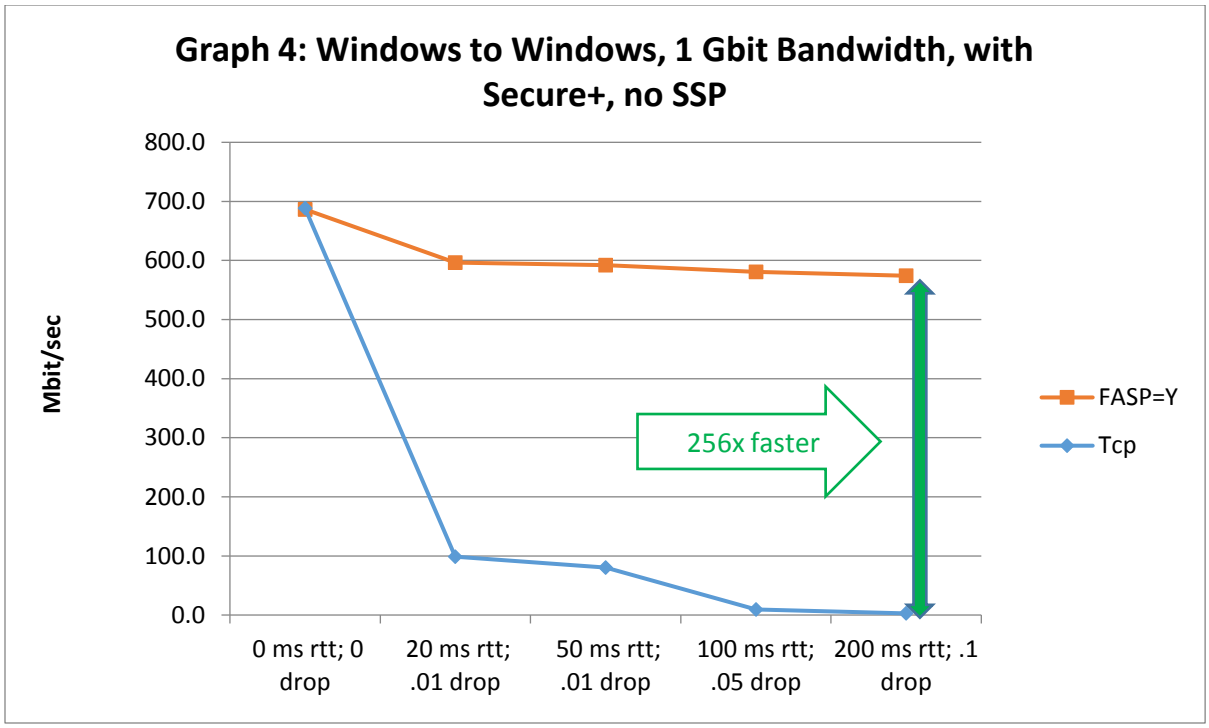
Comparison between HSAO and TCP

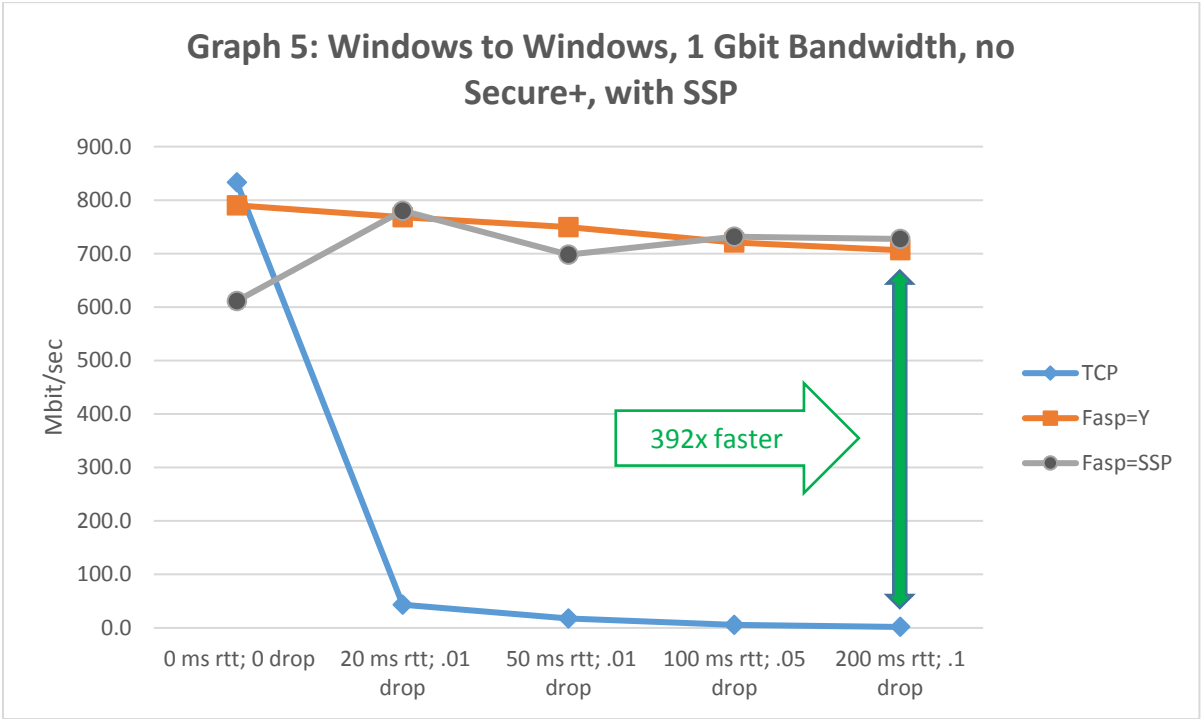
The following file transfer rate graphs show examples of the advantage of HSAO over TCP/IP in the presence of latency or packet loss or both. The use of HSAO is indicated by the protocol type "FASP". Graphs 1–4 are for Connect:Direct directly to Connect:Direct, without Sterling Secure Proxy (SSP) being deployed. Graphs 5 and 6 include an instance of SSP between the two Connect:Direct nodes, either in a standard proxying role (`FASP=Y`) or an HSAO-bridging role (`FASP=SSP`). See the "HSAO deployment options" section for more information about SSP.

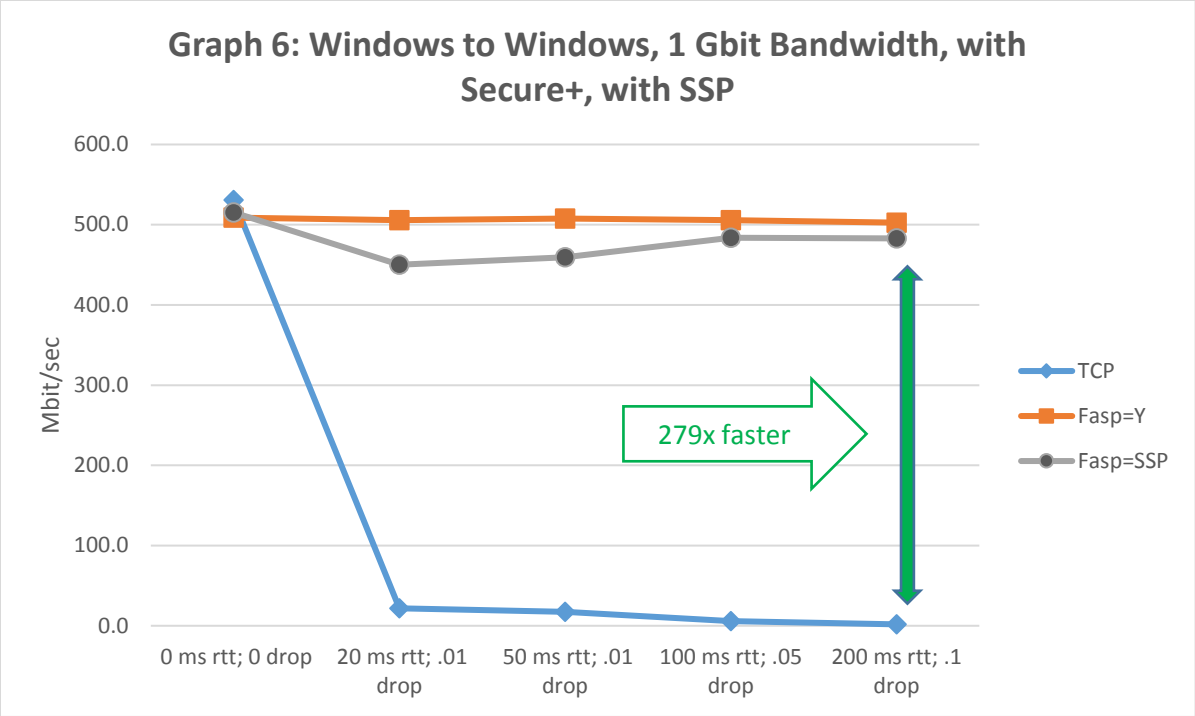



Graph 2: Linux to Linux, no Secure+, 1 Gbps bandwidth, no packet loss, no SSP
Transfer Rate (Mbit/sec)
1200.0
1000.0
800.0
600.0
400.0
200.0
0.0
0 ms
50 ms
200 ms
Round-trip Latency (ms)
4x faster
16x faster
FASP
Tcp


Graph 3: Linux to Linux, no Secure+, 100 Mbps bandwidth, no packet loss, no SSP
Transfer Rate (Mbit/sec)
100.0
90.0
80.0
70.0
60.0
50.0
40.0
30.0
20.0
10.0
0.0
0 ms
50 ms
200 ms
Round-trip Latency (ms)
2x faster
FASP
Tcp


Graph 4: Windows to Windows, 1 Gbit Bandwidth, with Secure+, no SSP
Mbit/sec
800.0
700.0
600.0
500.0
400.0
300.0
200.0
100.0
0.0
256x faster
0 ms rtt; 0 drop
20 ms rtt; .01 drop
50 ms rtt; .01 drop
100 ms rtt; .05 drop
200 ms rtt; .1 drop
FASP=Y
Tcp


Graph 5: Windows to Windows, 1 Gbit Bandwidth, no Secure+, with SSP
Mbit/sec
900.0
800.0
700.0
600.0
500.0
400.0
300.0
200.0
100.0
0.0
392x faster
0 ms rtt; 0 drop
20 ms rtt; .01 drop
50 ms rtt; .01 drop
100 ms rtt; .05 drop
200 ms rtt; .1 drop
TCP
Fasp=Y
Fasp=SSP



Determining if a link is a candidate for HSAO

A proof of concept (POC) is the most reliable way to determine if HSAO can provide significant benefits over a particular link. Before performing a POC, a Connect:Direct administrator can determine whether the link is a promising candidate for HSAO by measuring the latency, bandwidth, and packet loss of the link. After a link has been established as a candidate for HSAO, performing a POC is straightforward because HSAO is easily enabled in the Connect:Direct nodes at either end of the link.

If you would like a demo of Connect:Direct HSAO before performing a POC, contact your technical seller and request a demo using the CSP's Connect:Direct HSAO testing environment. If you do not know who to contact in IBM Sales and you are an existing customer, call 877-426-3774.

Measuring latency and detecting packet loss

The ping command can be used to determine a link's round trip latency. For example,

```
>ping www.ibm.com

Pinging e2874.x.akamaiedge.net [172.225.170.74] with 32 bytes of data:

Reply from 172.225.170.74: bytes=32 time=62ms TTL=51

Reply from 172.225.170.74: bytes=32 time=65ms TTL=51

Reply from 172.225.170.74: bytes=32 time=62ms TTL=51

Reply from 172.225.170.74: bytes=32 time=62ms TTL=51

Ping statistics for 172.225.170.74:

    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

```
Approximate round trip times in milli-seconds:

    Minimum = 62ms, Maximum = 65ms, Average = 62ms
```

This configuration performs 4 pings to determine that the latency of this link is approximately 62 ms.

Ping also reports the packet loss rate. On most links, a much larger number of pings will be required to detect packet loss.

Packet loss can also be detected with the `netstat –s` command. The output of the command varies from platform to platform. If the command reports that a significant fraction of the packets sent or received were associated with an error (for example, if the number of retransmitted packets is a significant fraction of the packets sent), this is an indication that packet loss may be occurring.

Measuring bandwidth

iPerf (http://software.es.net/iperf/) can be used to measure the bandwidth available to UDP (and hence to HSAO) and to TCP/IP.

UDP

For example, on Linux systems, the UDP server command is:

```
./iperf -s -u -l 32k -i 2
```

And the UDP client command is:

```
./iperf -c <IP address or DNS name of the server> -u -b 10m -l 32k -t 60 -w 1999k
```

The `–b 10m` parameter in the client command indicates the bandwidth rate to send the data. In this example, it is set to 10 Mbits/sec.

When UDP Iperf runs, you see output on the server similar to this:

```
[ ID] Interval       Transfer     Bandwidth       Jitter   Lost/Total Datagrams
[  4]  0.0- 1.0 sec   1.3 MBytes  10.0 Mbits/sec  0.209 ms    1/  894 (0.11%)
[  4]  1.0- 2.0 sec   1.3 MBytes  10.0 Mbits/sec  0.221 ms    0/  892 (0%)
[  4]  2.0- 3.0 sec   1.3 MBytes  10.0 Mbits/sec  0.277 ms    0/  892 (0%)
```

To determine the actual bandwidth of the connection, run the iperf UDP client multiple times while adjusting the value of the `–b` parameter until the Lost/Total Datagrams value reported by the iperf server is 1% or less. Increase the value of the `–b` parameter when the Lost/Total Datagrams less than 1%. Decrease the value of the `–b` parameter if the Lost/Datagrams is greater than 1%. The final value of the `–b` parameter is the bandwidth of the connection.

TCP

An example on Linux of TCP iperf on the server is:

```
./iperf -s -w 1999k -l 32k -i 2
```

For the Linux TCP iperf client, an example is:

```
./iperf -c <IP address or DNS name of the server> -l 56k -w 4096k -t 60 -i 2
```

When TCP iperf runs, you see output similar to this on the server:

```
[ ID] Interval       Transfer     Bandwidth
[  4] 0.0- 2.0 sec   1.8 MBytes   6.9 Mbits/sec
```

The “Bandwidth” column displays the bandwidth obtained during the iperf test.

Which members of the IBM Sterling MFT product family support HSAO?

HSAO is a licensed Connect:Direct option and requires an HSAO license key. See “Common installation and configuration steps for all scenarios” for information about license keys.

Native support

HSAO is supported natively by the following IBM Sterling MFT products:

- Connect:Direct UNIX (Linux and AIX) (4.2.0.4 and later)
- Connect:Direct Windows (4.7.0.4 and later)
- Sterling Secure Proxy (SSP) (3.4.3 and later)
- IBM Control Center (6.0.0.1 and later)

Support through High-Speed Bridging (SSP converting between TCP and FASP)

HSAO is also supported through SSP HSAO bridging by the following IBM Sterling MFT products:

- Connect:Direct z/OS (5.2 PTF UI36302 and later)
- Connect:Direct UNIX (zLinux) (4.2.0.4 iFix 13 and later)
- Connect:Direct UNIX (Linux and AIX) (4.2.0.4 iFix 13 and later)
- Connect:Direct Windows (4.7.0.4 iFix 7 and later)
- Sterling Secure Proxy (SSP) (3.4.3 and later)
- IBM Control Center (6.1.0.1 iFix 02 and later)

High-Speed Bridging provides HSAO support for Connect:Direct z/OS and Connect:Direct UNIX (zLinux), which do not have native HSAO support, by converting a file transfer between TCP/IP and FASP. Typically, SSP is used to bridge between TCP/IP and FASP. It is placed in or at the edge of a LAN, so that FASP is used over the vast majority of the length of the link to a remote Connect:Direct node.

Note: When SSP performs bridging, it continues to provide its usual session break and protocol inspection functionality.

When an organization prefers not to use a UDP-based network protocol within their LANs, High-Speed Bridging can be used with Connect:Direct UNIX (Linux and AIX) and Connect:Direct Windows as an alternative to native support..

Deployment scenarios

The following information provides several HSAO deployment scenarios along with the key configuration steps for each scenario.

HSAO deployment options

Because HSAO is available both natively and through SSP bridging, there is often more than one way for a given installation of Connect:Direct to take advantage of HSAO. The following table suggests HSAO

deployment options at a local site, given the platform that Connect:Direct is running on, the type of link to the Connect:Direct at the remote site, and whether a UDP-based protocol is allowed in the LAN or not.

Note: Apart from the link type, the HSAO deployments at the local and remote sites are independent.

There are three main deployment options, characterized by whether and how SSP is deployed:

- SSP optional

  SSP may not be required at the local site. For example, if the local Connect:Direct node has native HSAO support, and the link between the local and remote sites is a leased line.

- SSP proxy

  SSP is deployed as a standard proxy (protocol inspection and session break) at the local site. For example, if the local Connect:Direct node has native HSAO support, and the link between the local and remote sites goes over the internet.

- SSP bridging

  SSP is deployed as a bridge at the local site. For example, if the local Connect:Direct node does not have native HSAO support, or if the local Connect:Direct node has native HSAO support, and a UDP-based protocol is not allowed in the LAN.

Suggested HSAO deployment, based on Connect:Direct platform and network security factors

C:D Platform	HSAO Implementation Options	Suggested HSAO Deployment when particular network security factors apply		
		Private link	**Public link**	**UDP undesirable in LAN**
z/OS	SSP Bridging	*SSP Bridging*	*SSP Bridging*	*SSP Bridging*
zLinux	SSP Bridging	*SSP Bridging*	*SSP Bridging*	*SSP Bridging*
AIX	Native or SSP Bridge	*SSP Optional*	*SSP Proxy*	*SSP Bridging*
Windows	Native or SSP Bridge	*SSP Optional*	*SSP Proxy*	*SSP Bridging*
Linux	Native or SSP Bridge	*SSP Optional*	*SSP Proxy*	*SSP Bridging*

Product configuration

HSAO configuration is highly flexible. It can be configured globally, in a netmap entry for a remote node, in a Connect:Direct Process, and in a Copy Step. In the following scenarios, HSAO is configured in the netmap entry for the remote node because this is a straightforward way to enable HSAO for a test or a POC. Note: Configuration parameter syntax may vary from platform to platform. See the product documentation referenced at the end of this document to determine the exact syntax.

In the diagrams for the scenarios, node A (on the left) is the pnode and the sender and node B (on the right) is the snode and the receiver.

Common installation and configuration steps for all scenarios

1. Both Connect:Direct nodes and any SSP instances that are implemented must be at version levels that support HSAO.
2. Both Connect:Direct nodes must have an HSAO license key. License keys are available on Passport Advantage or, if a POC is being performed, from a Sales Representative. See the product documentation referenced at the end of this document for information on installing a license key.)
3. Any firewalls that the FASP protocol traverses must have the appropriate UDP ports open and meet the following criteria:
    a. The local firewall administrator must open all UDP ports to outbound UDP traffic from a system that initiates FASP connections.
    b. The remote firewall administrator must open a range of UDP ports to inbound traffic to a system that is receiving FASP traffic. The ports open in the firewall need to be consistent with the Connect:Direct FASP listen ports configuration or SSP FASP Port Range configuration. Ensure the number of ports opened is sufficient to handle the maximum number of concurrent, inbound sessions.

Confirming that HSAO was used

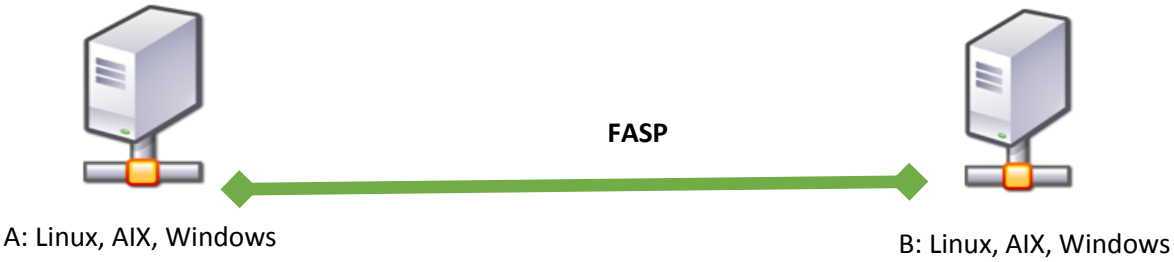
If HSAO was used, the Connect:Direct Copy Termination Records (CTRCs) include the flag `FASP=Y` or `FASP=SSP`.

Calculating the transfer rate

A CTRC contains the start and stop times of the transfer and the number of bytes transferred, which can be used to compute the transfer rate.

After the HSAO configuration is validated, conduct performance testing with large files (least 1 GB) to establish the benefits of HSAO in a particular environment. The `FASP file size threshold` parameter can be used to prevent HSAO from being used for files that are too small to benefit from HSAO.

Scenario 1: Local and remote Connect:Direct nodes have native HSAO support and the link is a leased line



Node A

- In the netmap entry for node B, set the following parameter values:

  ```
  fasp=(yes,yes)
  fasp.filesize.threshold=1
  ```

  to enable HSAO whether node A is the pnode or the snode and to transfer files of any size using HSAO.

Node B

- In the netmap entry for node A, set the following parameter values:

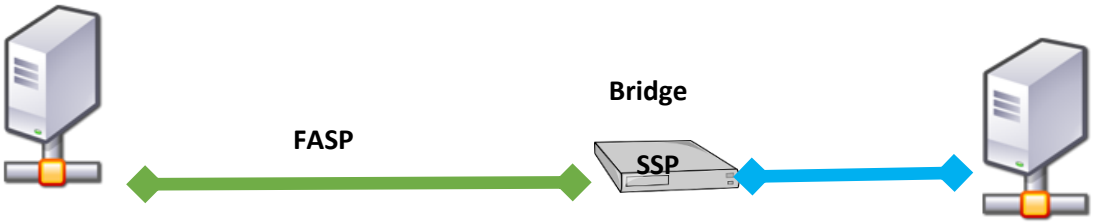
  ```
  fasp=(yes,yes)
  fasp.filesize.threshold=1
  ```

  This configuration enables HSAO whether node B is the pnode or the snode and to transfer files of any size using HSAO.

- Confirm that HSAO listen ports are configured in the initparms, for example:

  ```
  listen.ports=(44001, 33002-33005)
  ```

Scenario 2: Local Connect:Direct node has native HSAO support, remote Connect:Direct node does not, and the link is a leased line



A: Linux, AIX, Windows

B: z/OS, zLinux

Node A

- In the netmap entry for node B, set the following parameter values:

```
fasp=(yes,yes)

fasp.filesize.threshold=1
```

  This configuration enables HSAO whether node A is the pnode or the snode and to transfer files of any size using HSAO.

Node B

- In the netmap entry for node A, set the following parameter values:

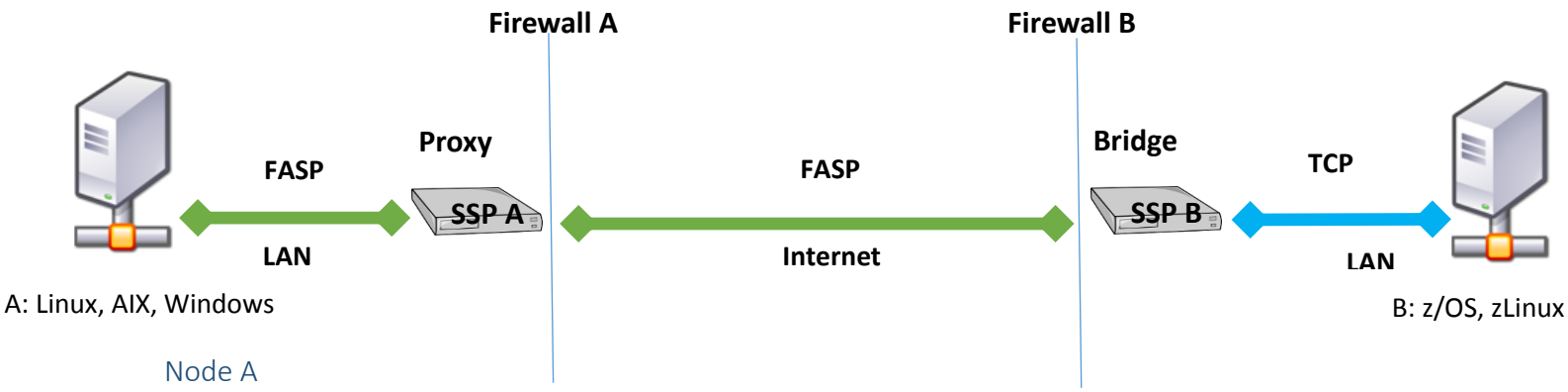
```
fasp=(SSP,SSP)

fasp.filesize.threshold=1
```

  This configuration enables SSP Bridging for transfers whether node B is the pnode or the snode and to transfer files of any size using HSAO.

SSP

- Optionally, configure the HSAO listen ports by setting the FASP port range field. The default is to listen on all ports.

Scenario 3: Local Connect:Direct node has native HSAO support, remote Connect:Direct node does not, and the link goes over the Internet



Node A

- In the netmap entry for node B, set the following parameter values:

```
fasp=(yes,yes)

fasp.filesize.threshold=1
```

  This configuration enables HSAO for transfers whether node A is the pnode or the snode and to transfer files of any size using HSAO.

SSP A

- Optionally, configure the HSAO listen ports.

Firewall A

- Open the firewall to all outbound UDP traffic from system A.

Node B

- In the netmap entry for node A, set the following parameter values:

```
fasp=(SSP,SSP)

fasp.filesize.threshold=1
```

  This configuration uses High-Speed Bridging for transfers whether node B is the pnode or the snode and to transfer files of any size using HSAO.

SSP B

- Optionally, configure the HSAO listen ports. The default is to listen on all ports.

Firewall B

- Open the firewall to inbound UDP traffic to Node B on the ports specified in the SSP B HSAO listen ports parameter. If no ports, were specified, all ports must be open to Node B.

Common problems with HSAO configuration

- An HSAO license key problem on either node. (Messages FASP030 - FASP034E)
- The `FASP file size threshold` parameter is set too low. (Message FASP008E)
- Insufficient FASP listen ports are configured on the snode. (Message FASP007E)
  To avoid this error, configure the number of ports configured to handle the maximum number of concurrent, inbound sessions.
- HSAO is not configured on the remote node. (Message FASP010E)
- A firewall is blocking the UDP ports. This is often the problem if TCP connections succeed but HSAO connections fail.

Special considerations

Connect:Direct firewall navigation feature

The Connect:Direct firewall navigation feature (the ability to specify the source ports to use for a connection with a particular trading partner) does not apply to HSAO. A local firewall has to be open outbound to all UDP ports from the system on which Connect:Direct is deployed.

HSAO connections through a load balancer

For special considerations when HSAO connections are made through a load balancer, see Using a Load Balancer with the High Speed Add-On (HSAO) Option for Connect:Direct.

Product documentation

- Connect:Direct z/OS - documentation for APAR PI55433 / PTF UI36302
- Connect:Direct UNIX - CD UNIX HSAO
- Connect:Direct Windows - CD Windows HSAO
- Sterling Secure Proxy (SSP) - SSP HSAO
- IBM Control Center – IBM Control Center V6.0.0 Knowledge Center (search for “FASP”)