IBM Aspera fasp.io Gateway 1.1

Admin Guide

Contents

Introduction

IBM Aspera fasp.io is a lightweight software component for high-speed bidirectional data transport. Using the patented Aspera FASP protocol, fasp.io achieves speeds of up to 2.5 Gb/sec per single process instance over unmanaged networks.

Aspera fasp.io fully utilizes available bandwidth to transfer data in byte-order sequence at the maximum possible speed with near-zero latency. It removes the barriers of size, distance, and complexity to move data between on-premises and cloud infrastructures.

Aspera fasp.io provides significant improvements in performance and service quality when transferring data between highly remote or dispersed locations in unfavorable network conditions, such as high latency and packet loss.

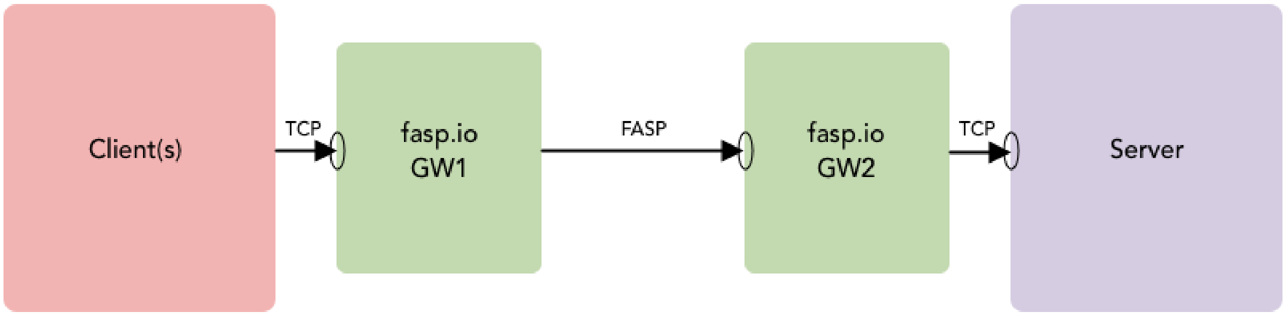
IBM Aspera extends its existing data transfer software portfolio with a new option, the Aspera fasp.io Gateway. Gateway is a software component that can be integrated quickly and easily with existing applications that use a TCP connection for their data flow. It improves nearly all server-to-server TCP-based data flows regardless of the distance and network conditions.

The IBM Aspera fasp.io Gateway acts as a transport layer proxy between TCP and Aspera FASP.

Usage

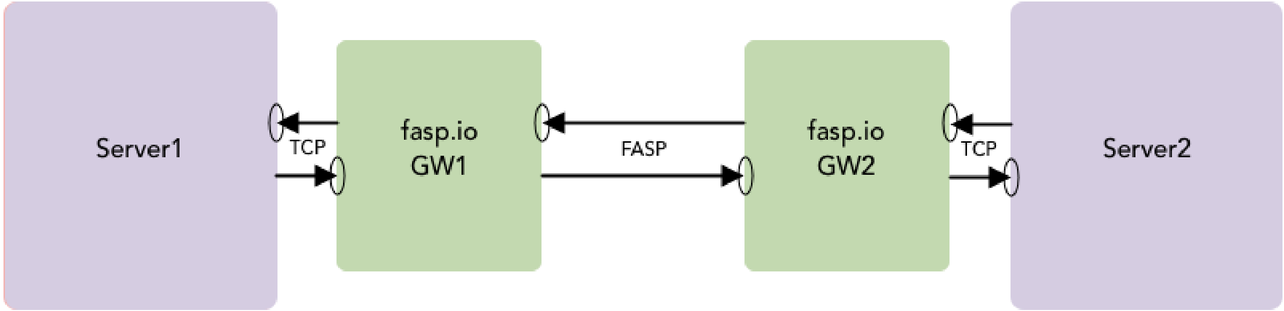
Gateway Client/Server Usage

In this configuration, two fasp.io Gateways are used to bridge one (or several) TCP connections from TCP clients to a TCP server over FASP:



Gateway Server/Server Usage

For some use cases, such as DB replication or messaging services (like MQ or Event Streams), communication must be established by both sides. In this mode, each server initiates a connection to the other:



For information on how to configure these two cases, see Configuration on page 5.

Licensing

At this time, IBM Aspera fasp.io Gateway includes no licensing enforcement system. However, usage of fasp.io Gateway is still subject to the terms of the IBM Aspera license agreement.

Installing on Linux

DEB or RPM Package

Install from the `.deb` or `.rpm` package as appropriate to your system.

- Linux `.deb`:

  ```
  $ sudo apt install ./ibm-fasp.io-gateway_version_arch.deb
  ```

- Linux `.rpm`:

  ```
  $ sudo yum install ./ibm-fasp.io-gateway_version_arch.rpm
  ```

Location of installed files:

Executable:	`/usr/bin`
Default configuration files:	`/etc/fasp.io`
Documentation:	`/usr/share/fasp.io-gateway`
SystemD startup file:	`/etc/systemd/system/fasp.io-gateway.service`

Uninstalling

- Linux `.deb`:

  ```
  $ sudo apt uninstall ./ibm-fasp.io-gateway
  ```

- Linux `.rpm`:

  ```
  $ sudo yum uninstall ./ibm-fasp.io-gateway
  ```

Installing on Windows

Running the Installer

To install IBM Aspera fasp.io Gateway onto your Windows machine:

1. Ensure that you're on a supported Windows machine, and are logged in with an account that has administrator privileges.
2. Download the installer. The default location is the `Downloads` folder:

   ```
   ibm-fasp.io-gateway_version_win64.msi
   ```

3. Open the `.msi` file. The the installer wizard launches. Follow the prompts to complete the installation.

Uninstalling

1. From the **Start** menu, go to **Settings** > **System** > **Apps & Features**.

2. Select the **IBM fasp.io Gateway** app. The **Uninstall** button appears.
3. Click **Uninstall** and confirm.

Configuration

The Gateway Configuration File

The IBM Aspera fasp.io Gateway configuration file, `gateway.toml`, is located here:

Linux:

```
/etc/fasp.io/gateway.toml
```

Windows:

```
C:\Program Files\IBM\fasp.io Gateway\config\gateway.toml
```

The `gateway.toml` file included in the installation is provided as a template. Modify this file to specify your ports, hostnames, and so on.

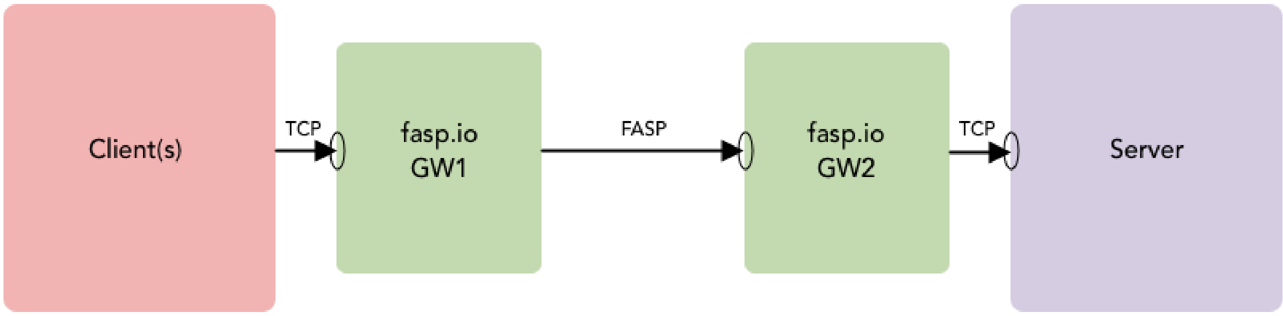
Note: Every time you modify `gateway.toml` to make changes for your configuration, you must restart the IBM fasp.io Gateway service. For information on starting and stopping the Gateway service, see Launching fasp.io Gateway on page 8.

Note: The version of the FASP protocol included in fasp.io Gateway uses a single UDP port. Whatever port you configure for your FASP connection over the WAN must have the same UDP port open on any firewalls along the connection path.

Configuration Examples

Example: Client to Server

In this configuration, two Gateways are used to bridge a TCP connection from TCP clients to a TCP server over FASP:



Given a server listening on port 12345, configure your client to point to Gateway 1 (`GW1` IP) and port 12345:

GW1 Configuration

```
[[bridge]]
   [bridge.local]
       protocol = "tcp"
           host = "0.0.0.0"
           port = 12345

   [bridge.forward]
       protocol = "fasp"
```

GW2 Configuration

```
[[bridge]]
   [bridge.local]
       protocol = "fasp"
           host = "0.0.0.0"
           port = 12345

   [bridge.forward]
       protocol = "tcp"
```

GW1 Configuration

```
            host = "GW2"
            port = 12345
```

GW2 Configuration

```
            host = "Server"
            port = 12345
```

Example: Forwarding to the first Available Host

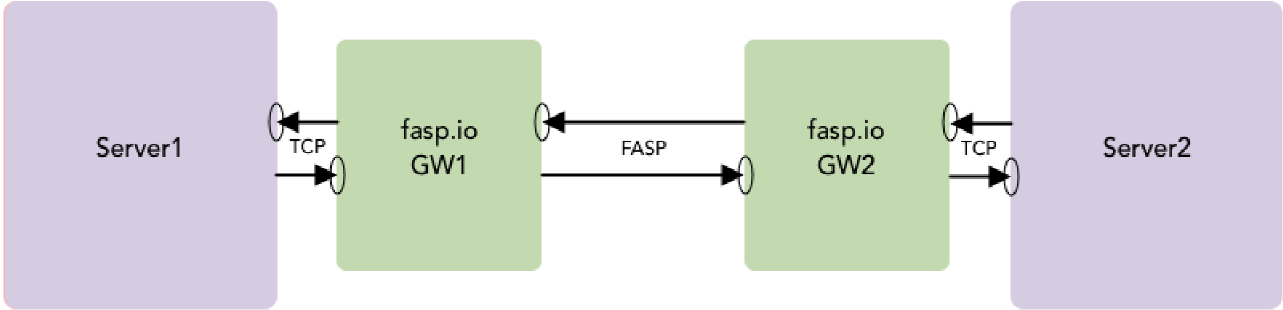
For `bridge.forward`, Gateway can loop through an array of specified hostnames or IP addresses and forward to the first available host it finds. For example:

```
[bridge.forward]
       protocol = "fasp"
           host = ["GW2", "10.0.0.2"]
           port = 12345
```

A hostname can resolve to multiple IP addresses. If a hostname is specified—either as a single entry or as an entry within the array—each of its IP addresses are tried until a connection is established.

Example: Server to Server

For some use cases, such as DB replication or messaging services (like MQ or Event Streams), communication must be established by both sides. In this mode, each server initiates a connection to the other:



GW1 Configuration

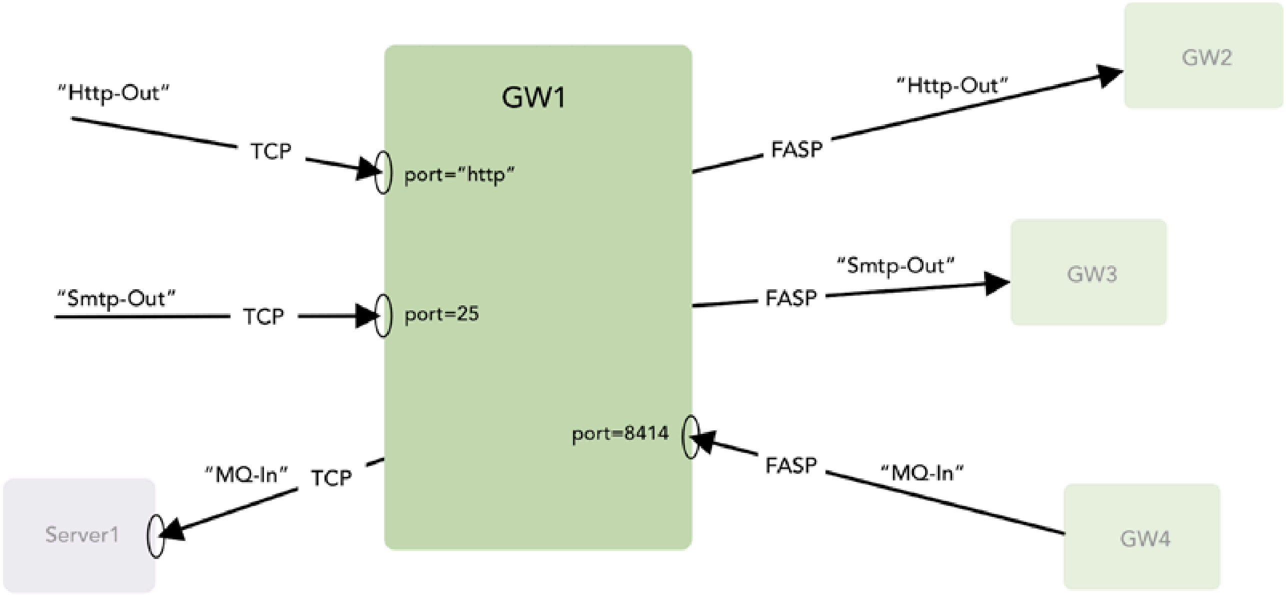
```
[[bridge]]
   name = "Outbound"
   [bridge.local]
       protocol = "tcp"
           host = "0.0.0.0"
           port = 12345

   [bridge.forward]
       protocol = "fasp"
           host = "GW2"
           port = 12345

[[bridge]]
   name = "Inbound"
   [bridge.local]
       protocol = "fasp"
           host = "0.0.0.0"
           port = 54321

   [bridge.forward]
       protocol = "tcp"
           host = "Server1"
           port = 54321
```

GW2 Configuration

```
[[bridge]]
   name = "Outbound"
   [bridge.local]
       protocol = "tcp"
           host = "0.0.0.0"
           port = 54321

   [bridge.forward]
       protocol = "fasp"
           host = "GW1"
           port = 54321

[[bridge]]
   name = "Inbound"
   [bridge.local]
       protocol = "fasp"
           host = "0.0.0.0"
           port = 12345

   [bridge.forward]
       protocol = "tcp"
           host = "Server2"
           port = 12345
```

Gateway Configuration Options

Gateway can also be configured with multiple bridges, multiple ports, multiple destinations, and multiple services. The image below shows a single gateway configured with examples of these combinations.



Bridge: HTTP-Out

```
[[bridge]]
   name = "Http-Out"
   [bridge.local]
      protocol = "tcp"
        host = "0.0.0.0"
        port = "http"

   [bridge.forward]
      protocol = "fasp"
        host = "GW2"
        port = "http"
```

Bridge: Smtp-Out

```
[[bridge]]
   name = "Smtp-Out"
   [bridge.local]
      protocol = "tcp"
        host = "0.0.0.0"
        port = 25

   [bridge.forward]
      protocol = "fasp"
        host = "GW3"
        port = 25
```

Bridge: MQ-In

```
[[bridge]]
   name = "MQ-In"
   [bridge.local]
      protocol = "fasp"
        host = "0.0.0.0"
        port = 8414

   [bridge.forward]
      protocol = "tcp"
        host = "Server1"
        port = 8414
```

Logging Configuration File

Logging configuration is defined in `logging.toml`, which is located here:

Linux:

```
/etc/fasp.io/logging.toml
```

Windows:

```
C:\Program Files\IBM\fasp.io Gateway\config\logging.toml
```

Error, warning, and info logs are sent to the console by default. In the default `logging.toml` file, four loggers are made available:

`gateway` – High-level logger for the gateway.
`s2s` – Stream-to-stream session class logger.
`fasp.io-cpp` – Logger for the Asio/C++ FASP SDK.
`fasp.io-c` – Logger for the FASP protocol.

Note: Every time you modify `logging.toml`, you must restart the fasp.io Gateway service.

For more information on how to configure logging, see the full reference at:

`https://github.com/guangie88/spdlog_setup`

Notes:

The `level` setting is optional for both sinks and loggers.
The `level` for error logging is `err`, not `error`.
The `_st` suffix means single-threaded.
The `_mt` suffix means multi-threaded.
`syslog_sink` is thread-safe by default. No `_mt` suffix is required.

The `spdlog` default logging format is:

```
[2014-10-31 23:46:59.678] [loggername] [info] message
```

For information on how to customize `spdlog` formatting, see:

`https://github.com/gabime/spdlog/wiki/3.-Custom-formatting`

Launching fasp.io Gateway

To start or stop IBM Aspera fasp.io Gateway, you start or stop the service using one of these methods as appropriate:

- Start `fasp.io-gateway` on the command line (Linux, Windows).
- Start **IBM fasp.io Gateway** from the Services panel (Windows).

Note: Each time you change your Gateway configuration—by modifying `gateway.toml`—you must restart the service. For information about configuring your system, see Configuration on page 5.

Linux

Run the commands below to start, stop, or enable the `fasp.io-gateway` service, as well as to check service status and access service logs:

- Start service:

  ```
  # sudo systemctl start fasp.io-gateway
  ```
- Stop service:

  ```
  # sudo systemctl stop fasp.io-gateway
  ```
- Enable service (to restart after reboot):

  ```
  # sudo systemctl enable fasp.io-gateway
  ```
- Service status:

  ```
  # sudo systemctl status fasp.io-gateway
  ```
- See service logs:

  ```
  # sudo journalctl --unit=fasp.io-gateway
  ```

Windows

Running Gateway requires that you start the Windows service `fasp.io-gateway` (on the command line) or **IBM fasp.io Gateway** (on the Services panel). Both methods require admin privileges.

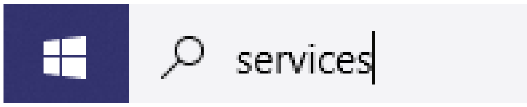
To start the service from the command line, run:

```
> net start fasp.io-gateway
```

To stop the service:

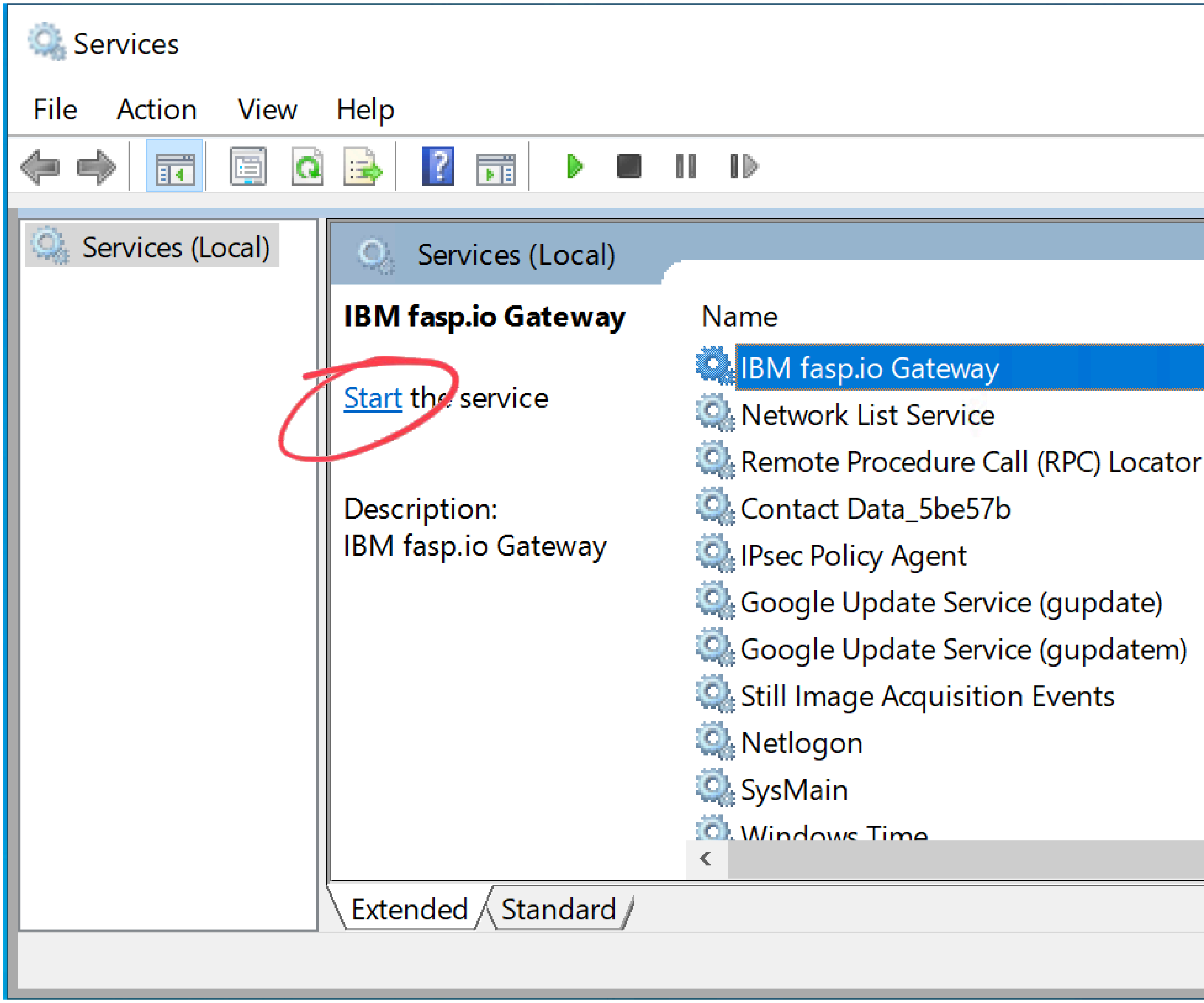
```
> net stop fasp.io-gateway
```

To start the service from Windows, open the Services panel. To access it quickly, enter "services" in the search box next to the **Start** button:



services

In the display that appears, click **Services**. The Windows **Services** panel opens. In the list of services, find and select **IBM fasp.io Gateway**. To launch the service, click **Start the service**.



IBM fasp.io Gateway

Start the service

Description:
IBM fasp.io Gateway

Name
IBM fasp.io Gateway
Network List Service
Remote Procedure Call (RPC) Locator
Contact Data_5be57b
IPsec Policy Agent
Google Update Service (gupdate)
Google Update Service (gupdatem)
Still Image Acquisition Events
Netlogon
SysMain
Windows Time

Testing the Gateway

Simple Echo Test Through netcat

On the Server machine, listen:

```
# nc -v -l 12345
```

On the client machine, connect to Gateway 1:

```
# nc GW1_ip_addr 12345
```

Stream Data Through netcat

On the Server machine, listen:

```
# nc -v -l 12345 > /dev/null
```

On the client machine, connect and stream 1 GB of data:

```
# dd if=/dev/zero count=1024 bs=1m  | nc -v -n GW1_ip_addr 12345 >/dev/null
```