



**GLOBAL LEADER IN
BROADCAST QUALITY
LIVE VIDEO OVER IP**

Zixi Feeder User Guide

Software Version 14
Document Version DOC16-85-0018
Copyright © Zixi 2022
All Rights Reserved

Legal Notice

This document contains proprietary and confidential material of Zixi LLC. Any unauthorized reproduction, use, or disclosure of this material, or any part thereof, is strictly prohibited. This document is solely for the use of Zixi employees and authorized Zixi customers.

The material furnished in this document is believed to be accurate and reliable. However, no responsibility is assumed by Zixi LLC. for the use of this document or any material included herein. Zixi LLC. reserves the right to make changes to this document or any material included herein at any time and without notice.

For more information visit: www.zixi.com.

Copyright © Zixi 2022 Waltham, MA U.S.A.

All Rights Reserved.

** See back for additional licensing information*

Table of Contents

Legal Notice.....	2
Introduction to Zixi Feeder	6
Zixi Technology.....	6
Zixi Platform Architecture.....	7
Supported Protocols	8
Supported Devices	9
Hardware-based IP encoder devices with integrated Zixi Feeder	9
Server-based encoders	9
Desktop PC encoders	10
ZEP Lite Feeder or Zixi Server Appliance.....	10
Generic Server/Appliance	10
Latency Considerations.....	10
Multicast Options.....	11
Recording Options	11
Prerequisites	12
Firewall Settings.....	12
Bandwidth Availability	12
Source/Destination Devices Compatibility	12
Remote Connection	12
Logging in to Feeder.....	13
Activating Your Zixi Feeder License	14
Activating Your Zixi Feeder License – Online Activation.....	14
Loading Feeder License File - Offline Activation.....	14
Reloading the Feeder License	15
Viewing Current Zixi Feeder License Information.....	15
Viewing Status in Feeder	16
Adding Input Streams to Feeder	18
Adding a UDP Stream to Feeder	18
Input Stream Parameters Table.....	19
Adding an RTMP Stream	20

Input Stream Parameters Table	20
Adding an ASI Stream to Feeder	21
Adding a File Stream to Feeder.....	21
Input Stream Parameters Table.....	22
Adding an HTTP Pull Stream	22
Input Stream Parameters Table.....	23
Adding a Failover Stream	23
Input Stream Parameters Table.....	25
Searching for Input Streams in Feeder	25
Viewing Existing Input Streams in Feeder	25
Performing Actions on Input Streams in Feeder	27
Stopping/Starting Incoming Streams in Feeder.....	27
Editing Input Settings in Feeder.....	27
Deleting Input Streams from Feeder	27
Duplicating an Input Stream in Feeder	27
Adding Analysis to an Input Stream.....	28
Adding Filters to Input Streams (PID Filtering)	29
Filter Parameters Table.....	30
Adding Outputs in Feeder	31
Adding a File Output	31
Adding a Push to Broadcaster Output	32
Adding a UDP Output.....	38
Adding an HTTP Push Output.....	39
Creating Outputs in Feeder.....	42
Creating a File Output in Feeder.....	42
Output Parameters Table	43
Creating a Push to Broadcaster Output	44
Output Parameters Table	45
Creating a UDP Output in Feeder	48
Output Parameters Table	49
Creating an HTTP Push Output in Feeder	51
Output Parameters Table	52

Legal Notice

Zixi Technology

Output Groups	52
Creating a New Output Group	52
Using an Existing Output Group.....	53
Network Bonding	53
Using Network Bonding for Hitless Failover	54
Feeder Output with Network Bonding	55
Encrypting an Output Stream in Zixi Feeder.....	57
Searching for Outputs in Feeder	58
Viewing Existing Outputs in Feeder	58
Viewing Output Statistics in Feeder.....	60
Configuring Encoder Feedback Script.....	61
Performing Actions on Outputs in Feeder	62
Stopping/Starting Outputs in Feeder.....	62
Editing Output Settings in Feeder.....	62
Deleting Outputs from Feeder.....	62
Duplicating Outputs in Feeder	62
Feeder Settings	64
Feeder General Settings.....	64
Feeder ZEN Master Settings	65
Feeder SSH Tunnels.....	66
RTMP Server.....	68
Feeder Logging.....	68
Export/Import Settings	69
Feeder Network Settings	69
Viewing Feeder Events Log	71

Introduction to Zixi Feeder

Zixi Feeder resides at the point of acquisition and is either embedded inside Zixi EcoSystem partner encoder HW or SW, inside a camera (no backpack or field pack required), or can be installed on a pc that will be operated with any non-Zixi enabled encoder or camera. Zixi Feeder accepts and prepares the encoded video for transmission over standard internet connections using Zixi's UDP based transport stream protocol. By communicating with the Zixi Broadcaster, it creates a secure reliable network-aware delivery path.

Zixi Feeder offers the following features:

- **Robust content delivery over IP** - Communicates with Zixi Broadcaster¹ using UDP-based, video-optimized protocols on standard IP networks.
- **Broad range of encoders** - Accepts inputs from a broad range of hardware/software encoders and professional-grade cameras.
- **Recording** - Stores streams as MPEG-TS files.
- **Transport Stream Analyzer** – MPEG-TS ETSI TR 101-290 analyzer (priority 1 and 2)
- **Monitoring** - Captures network and content specific statistical information in real-time.
- **Supports many to many and any to any** - Gateway platform between UDP (unicast/multicast), Zixi protected stream (unicast/multicast), RTMP, or other supported formats.
- **Network bonding with hit-less fail-over** - Divides a stream into several network channels and then subsequently reunites them at the Zixi Broadcaster side, enabling simultaneous load balancing between multiple networks and ensuring high availability of the stream. Zixi Feeder can be configured to automatically use all available network interfaces, while automatically adapting to changes in the IP addresses without requiring additional configuration.
- **AES stream encryption** - Zixi Feeder can encrypt outputs using a fixed key (AES 128, AES 192, AES 256) which can be entered into the output settings or automatically generated. In addition, selection of Automatic mode will generate the encryption and exchange it securely between the applications automatically.

Zixi Technology

Zixi's transport stream protocol is a content and network-aware protocol that dynamically adjusts to varying network conditions and employs error correction techniques for error-free video streaming over IP. With minimum overhead to physical bandwidth, this dynamic mechanism provides low end-to-end latency, removes jitter, recovers and re-orders packets, smooths video delivery and regenerates video to its original form, *all in real-time*.

¹ It is possible to connect Zixi Feeder directly to an embedded Zixi Receiver based on point-to-point communication, however this capability requires specific integration and is not generally available.

Introduction to Zixi Feeder

Zixi Platform Architecture

Zixi delivers outstanding performance (at low predictable latency), superior reliability (no packet loss) and broadcast-grade video quality (SD, HD, and UHD) with no tradeoffs to delay, resolution or stutter.

Streaming from one Zixi-enabled device/server to another Zixi-enabled device/server protects the stream from quality degradations along the path. It enables the streaming of high-quality video over any distance, while overcoming the varying network conditions of the public Internet, where the amount of network errors, packet loss, jitter and out-of-order packets fluctuate "every second".

The Zixi transport stream protocol can also be deployed on part of the path. For example, from a Zixi Feeder to a Zixi Broadcaster (Zixi protected path) and on to the end user through UDP or RTMP. In this case, the path that has challenging conditions (e.g. long distance or unstable wireless conditions) will be protected by Zixi and then delivered on to the end-user on a standard UDP or RTMP protocol.

Zixi's transport stream technology includes the following features for ultimate quality and security:

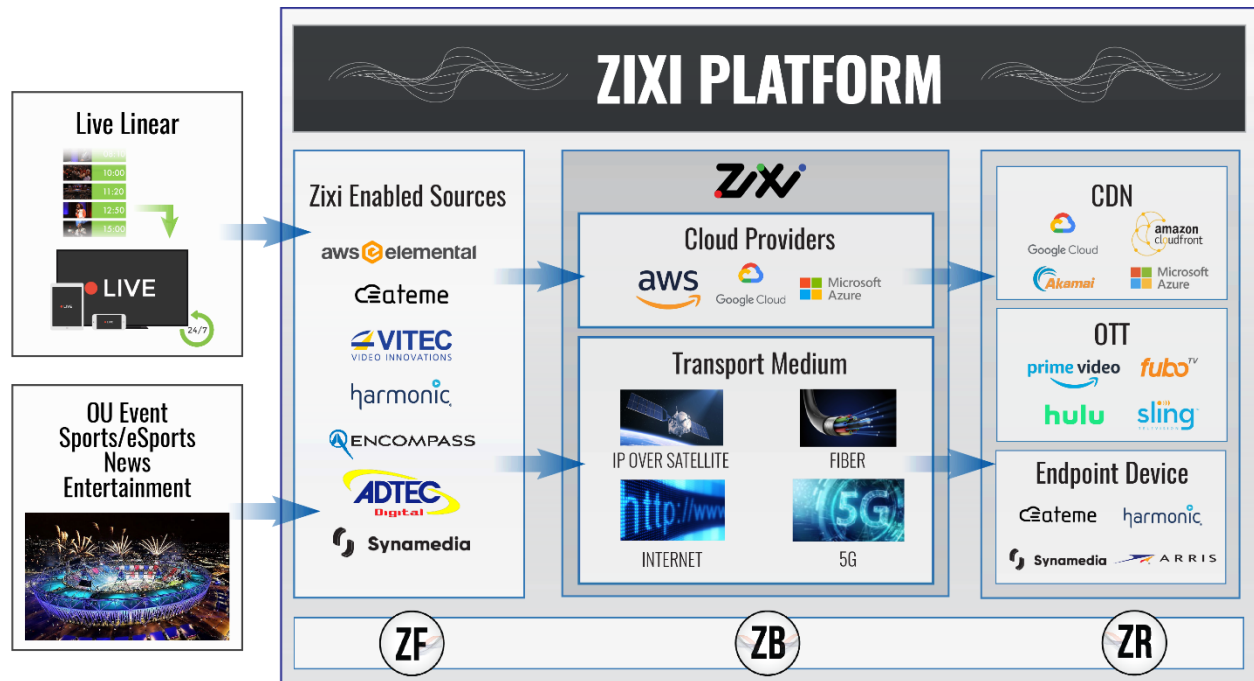
- Network sensing
- Dynamic de-jitter
- MPEG specific optimizations
- Z-ARQ error recovery
- Z-FEC - Dynamic content aware forward error correction
- Active multi-path error recovery
- Adaptive bitrate over UDP, unicast or multicast
- Rate control and congestion avoidance
- 256 bit AES transport encryption

Zixi Platform Architecture

The Zixi platform architecture shows how a live linear broadcast or event is ingested through a Zixi Feeder (ZF) or Zixi embedded device running Zixi Feeder at acquisition. Zixi Feeder encapsulates the stream in the Zixi transport protocol and delivers point to point or point to multi-point over standard IP connections to Zixi Broadcaster (ZB). Zixi Broadcaster can be deployed on premise or in the cloud with the ability to monitor streams anywhere along the path. For management, processing and larger scale distribution capabilities, Zixi can support complex production workflows for live events, including transcoding, recording, and more, all in a clustered environment that supports reliability and scalability. The stream is distributed through a CDN, OTT, or endpoint device, running Zixi Receiver (ZR) to any device.

Introduction to Zixi Feeder

Supported Protocols



Supported Protocols

Input:

- MPEG-TS over UDP and/or RTP with SMPTE-2022
- RTMP push
- Transport stream files from local file system
- ASI

Output:

- Zixi protected transport
- MPEG-TS over UDP and/or RTP with SMPTE-2022
- Transport stream files to local file system

Supported Devices

Zixi Feeder can be installed (or is pre-installed) in the following devices:

Hardware-based IP encoder devices with integrated Zixi Feeder

The following hardware-based IP encoders offer an integrated Zixi Feeder:

- Adtec Digital – EN31 and EN200 Encoder - <http://www.adtecdigital.com/products-and-solutions/end-to-end-contribution>
- ATEME – CM5000 Contribution Encoder - <https://www.ateme.com/kyrion-cm5000/>
- Blonder Tongue – LiveCast Encoder - www.blondertongue.com
- Drake - PEG-NE24-IP-C - <https://www.rldrake.com/product/peg-ne24-ip/>
- Harmonic – ViBE CP3000 and CP6000 - <https://www.harmonicinc.com/products/>
- Miravid – Live Stream - <http://www.miravid.com>
- Teradek – Cube and Vidui
- <http://teradek.com/collections/cube#encoder>; <http://teradek.com/pages/vidui-pro-sign-up>
- VITEC – MGW Nano, Premium and ACE
- <https://www.vitec.com/products/encoders/portable-encoders/>
- ZyCast – HDZ-800 - <http://www.zycasttech.com/zycast-technologys-new-hdz-800-live-video-streaming-over-the-internet-solution/>

Server-based encoders

The following server-based encoders offer an integrated Zixi Feeder:

- ATEME- Titan Live - <https://www.ateme.com/TITAN-Live/>
- ATX Networks – DigiVu - <https://atxnetworks.com/products/video-processing/encoders/mpeg2-h264-sd-hd/digivu-multichannel-encoder-rf/>
- Elemental - <http://www.elemental.com/products/elemental-delta>
- Open Broadcast Encoder - <http://obe.tv/products/encoding-and-decoding>

Desktop PC encoders

The following desktop PC encoders offer an integrated Zixi Feeder:

- Open Broadcast Systems - <http://obe.tv/products/encoding-and-decoding>
- Wirecast - <http://www.telestream.net/wirecast/overview.htm>

ZEP Lite Feeder or Zixi Server Appliance

Zixi Edge Point Appliance (ZEP Lite) and Zixi Server Appliance can be used with virtually any IP Encoder to input MPEG-TS over UDP, RTP or RTMP and output Zixi TS over the internet to a Zixi Broadcaster.

- ZEP Lite - can be used for a single OU contribution feed up to 20Mbps including encryption.
- Zixi Server Appliance - is shipped with Zixi software and CentOS Linux and can be used as a Feeder for processing multiple simultaneous video streams.

Generic Server/Appliance

The following are our minimum host server recommendations:

- Processor - Intel Dual Core i3 processor or better,
- Memory - 4 GB RAM,
- Networking - 2x 1GB network cards (one to connect the source device (Enc/Mux/IRD) and the other to be connected to the internet).
- OS - Linux CentOS 6/7, 64 bit (recommended), Ubuntu 12/14, Debian 7/8, 64 bit, Windows 7/8/ server 2008/2012, 64 bit.

Latency Considerations

Latency defines the delay in which the video stream will be delivered to the end-point. This delay is essentially a buffer that is used for additional processing, thereby improving the quality of video stream. The latency should be considered based on the importance of immediacy. For example, if the video requires interaction (e.g. between a reporter in the field and an anchor in the studio), the latency should be set to a minimum so that the interaction does not suffer from any delay.

Latency can vary from tens of milliseconds to several seconds.

Zixi protocol can provide a delay of a single frame (30ms), while increasing the latency to provide more robustness for the error-recovery.

In low latency (below 1500ms) the error recovery is based on a hybrid approach of FEC and ARQ. FEC will add overhead of up to 30% (adaptive—without congesting the network).

To allow sufficient error-recovery it is recommended to set the Latency at least 3 times larger than the RTT (i.e Latency $\geq 3 * RTT$)

Multicast Options

Latency can be set to any value, we recommend two common modes:

1. Interactive mode - 500ms
2. Non-interactive mode – 4000ms

Latency below 1500ms will add an FEC overhead of up to 30%.

Multicast Options

Multicast streams can be received or sent by the Zixi Feeder. Zixi Feeder serves as a gateway between unicast and multicast streams, while allowing reliable streaming over multicast.

Recording Options

Live stream inputs can be stored as MPEG-TS files to a local disk drive or to a mounted drive.

Prerequisites

Before you begin using Zixi Feeder, make sure you prepare your deployment by following the instructions below.

Firewall Settings

- Make sure UDP traffic coming to and from the Zixi Feeder is not blocked.
- External firewall should allow communication on UDP port 2088 (inbound & outbound).
- Remote web management should allow TCP port 4200.
- Make sure outgoing TCP 80 to license.zixi.com and UDP 53 to your DNS server.

Bandwidth Availability

Verify that both the uplink and downlink internet bandwidth are sufficient to carry the feeds.

Source/Destination Devices Compatibility

Verify that the MPEG-TS coming from your source device (Encoder/Mux/IRD/Playout, etc.) is compatible with the destination device (IRD/Playout/etc.).

Remote Connection

Allow SSH or TeamViewer Ver.10 remote connection details for support issues.

Logging in to Feeder

➔ To log in to Zixi Feeder:

1. From your web browser, navigate to: <http://localhost:4200>
A user authentication window opens.
2. In the **User Name** field, type **admin** and in the **Password** field, type '1234'.
3. Click **OK**.

Activating Your Zixi Feeder License

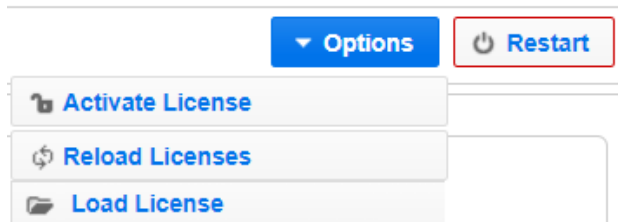
In order to start using Zixi Feeder, you must first activate your license. Zixi Feeder and its optional features are activated by license files. Before you first use Zixi Feeder and when you wish to add optional features, you will need to activate the appropriate license.

Activating Your Zixi Feeder License – Online Activation

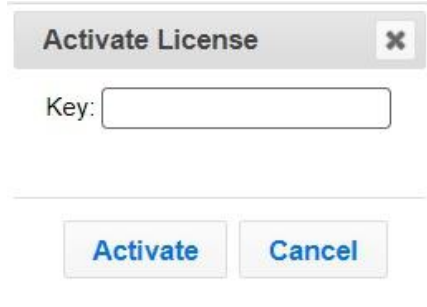
When logging in for the first time to a Zixi Feeder, you must activate the Zixi license.

➔ To Activate the License using a License Key:

1. In the **Status** page of the Zixi Feeder, click **Options** in the menu bar. The drop-down menu is displayed.



2. Click **Activate License**. The **Activate License** window appears.



3. Enter your **Activation Key** and click **Activate**. Once entered, The Zixi Feeder is enabled.

Loading Feeder License File - Offline Activation

If you want to activate your license without an internet connection, you can upload a license file. To obtain a license file, please contact Zixi support.

➔ To Load the License File:

1. In the **Status** page of the Zixi Feeder, click **Options** in the menu bar.
2. Click **Load License**. The directory of the local drive opens.
3. Select the license file and click **Open**.

Activating Your Zixi Feeder License

Reloading the Feeder License

4. A prompt window instructs you to Reload the license to activate the license. See [Reloading the Feeder License](#).

Reloading the Feeder License

In the event that your license expires, you must re-enter a new Zixi license key.

➔ To Reload the License:

1. In the **Status** page of the Zixi Feeder, click **Options** in the menu bar.
The drop-down options menu is displayed.
2. Click **Reload**.
The license is reloaded.

Viewing Current Zixi Feeder License Information

You can view the current license information on the **Status** page in the **License Information** section. See [Viewing Status in Feeder](#).

Viewing Status in Feeder

Viewing Current Zixi Feeder License Information

Viewing Status in Feeder

The **Status** page shows information about the current status of your Zixi Feeder. The **Program Information** section shows aggregated statistics for all input and output streams on the Feeder. The **License Information** section shows the current status of your license.

Zixi Feeder Version 14.0.38790 Linux 64-bit | ip-172-31-0-246.us-east-2.compute.internal | Copyright (c) 2007-2020, Zixi LLC. CPU: 41.57% Mem: 69% Clock: 11:34:03

STATUS **INPUTS** **OUTPUTS** **SETTINGS** **EVENT LOG** **LOGOUT**

Status Options Restart

Program Information

- ID: ip-172-31-0-246.us-east-2.compute.internal
- Host ID: 02164ffa8d56
- In Bitrate: 12085 kbps
- Out Bitrate: 5995 kbps
- Start Time: 2020-05-18 08:15:35
- Uptime: 51:18:30
- In Streams: 1
- Out Streams: 1
- Kbytes Received: 266861340
- Kbytes Sent: 116997361

License Information

- Product: Feeder
- Activation Key: [Activation Key](#)
- Expiration: 16-Mar-2021

The Information shown on the Status page is described in the following table:

Field	Description
Program Information	
ID	The Zixi Feeder ID number.
Host ID	Displays the ID of Zixi Feeder host.
In Bitrate	The aggregated bitrate of all active input streams.
Out Bitrate	The aggregated bitrate of all active output streams.
Start Time	Displays the time that the Zixi Feeder began to actively stream content.
Uptime	The duration of time that the Zixi Feeder is running.
In Streams	The number of current active inputs.
Out Streams	The number of current active outputs.
kBytes Received	The cumulative traffic received (in Kbps).
kBytes Sent	The cumulative traffic transmitted (in Kbps).
License Info	

Viewing Status in Feeder

Viewing Current Zixi Feeder License Information

Product	The product name.
Activation Key	Click on the Activation Key button to show the Activation Key as well as additional license details.
Expiration	The expiration date of the license.

Adding Input Streams to Feeder

You can add input streams to the Zixi Feeder from one of the following sources:

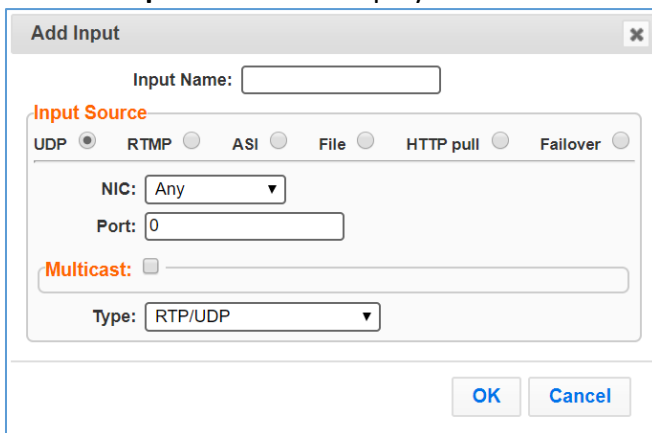
- **UDP streams** - the source is a UDP or RTP stream (with or without Forward Error Correction enabled), over unicast or multicast.
- **RTMP streams** - the source is an RTMP stream.
- **ASI streams** – the source is an ASI stream, which is converted into an IP-MPEG-TS stream before it is inputted into the Zixi Feeder.
- **File** - the source is an MPEG-TS file that is located on the machine.
- **HTTP pull** - the source is a Transport Stream over HTTP, which is pulled by the Zixi Feeder.
- **Failover** – creates a hitless failover group that uses two or more binary-identical sources to ensure an uninterrupted stream.

Adding a UDP Stream to Feeder

This enables the Feeder to receive UDP/RTP/RTP + SMPTE 2022 FEC streams.

➔ To Add a UDP Stream:

1. In the Zixi Feeder **Inputs** page, click the **+ New Input** button in the Menu bar. The **Add Input** window is displayed:



2. Enter the unique **Input Name** (an internal name) for the Feeder.
3. Select **UDP** for the Input Source.
4. Select the relevant **NIC** (network card) to listen to for the stream.
5. Enter the **Port** address of the stream.
6. If you want to receive the stream as **Multicast**, select this checkbox (see Input Stream Parameters Table below for additional parameters).
7. Select the desired protocol **Type** to be used to transmit the encoded stream (see Input Stream Parameters Table below for details).

Adding Input Streams to Feeder

Adding a UDP Stream to Feeder

- Click **OK**.
The “Input added” message appears on the top of the screen.
The UDP Stream is added to the Zixi Feeder and now appears in the list of Input streams available.

Input Stream Parameters Table

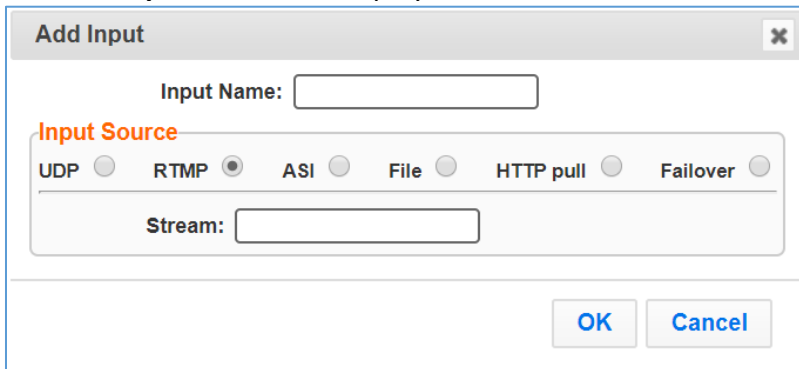
Parameter	Description
Input Name	Specify the internal name for the input stream.
<i>Input Source</i>	
NIC	Specify the local network card (at the source) that will be used for receiving/listening on for the stream. Default: <i>Any NIC</i> .
Port	Specify the port to listen on. This port must be open on all firewalls between the Zixi Feeder and other devices it is communicating with.
<i>Multicast</i>	
Address	Specify the multicast address when the input stream is transmitted over multicast.
SSM Source	Specify the source IP for a source-specific multicast. Note: This is the mechanism by which multicast sources can be specified, and then either rejected or accepted. In essence, this provides a very primitive firewall system for multicast networks.
Type	Select the appropriate protocol type from the drop-down list. <ul style="list-style-type: none">• RTP/UDP – the stream can be either RTP or UDP• RTP + SMPTE 2022 FEC – the stream is an RTP stream with Forward Error Correction enabled• UDP – the stream is a UDP stream only.

Adding an RTMP Stream

This enables the Feeder to receive an RTMP stream.

➔ To Add an RTMP Stream:

1. In the Zixi Feeder **Inputs** page, click the **+ New Input** button in the Menu bar. The **Add Input** window is displayed:



2. Enter the unique **Input Name** (an internal name) for the Feeder.
3. Select **RTMP** for the **Input Source**.
4. Specify the **Stream** name.
5. Click **OK**.
The “Input added” message appears on the top of the screen.
The RTMP Stream is added to the Zixi Feeder and now appears in the list of streams available in the Zixi Feeder inputs screen.
6. If RTMP servers haven't yet been enabled in your Feeder, then use the following procedure.
 - a. On the **Settings** screen, select the **RTMP Server** tab.
 - b. Select the **Enable** checkbox.
 - c. In the **RTMP server port** field, enter the port number.
 - d. Click **Apply**.
You will be required to restart Zixi Feeder.

Input Stream Parameters Table

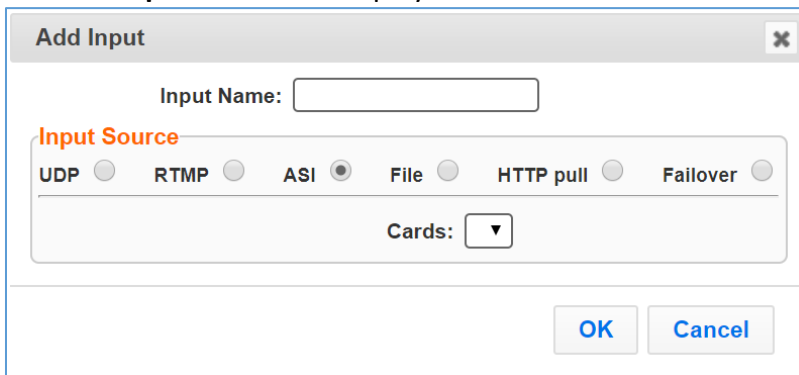
Parameter	Description
Input Name	Specify the unique identifier for this stream.
Stream	Enter the unique name for the stream that has been defined in the RTMP server.

Adding an ASI Stream to Feeder

If you have an ASI card that is compatible with Zixi¹, the ASI card will convert the ASI-MPTS stream to an IP/MPTS stream and transmit it to the Zixi Feeder. By adding an ASI input stream, you can configure Zixi Feeder to listen to the specified ASI card from which the stream will be transmitted.

➔ To Add an ASI Stream:

1. In the Zixi Feeder **Inputs** page, click the **+ New Input** button in the Menu bar. The **Add Input** window is displayed:



2. Enter the unique **Input Name** (an internal name) for the Feeder.
3. Select **ASI** for the **Input Source**.
4. In the **Cards** field, select the available ASI card from the drop-down list. Only the physical ASI cards which have been installed on the machine will appear in the drop-down list.
5. Click **OK**.
The “Input added” message appears on the top of the screen.
The ASI Stream is added to the Zixi Feeder and now appears in the list of streams available in the Zixi Feeder inputs screen.

Adding a File Stream to Feeder

This creates a TS file input for the Broadcaster from a local or mounted disk. Zixi Feeder will transmit the file located in the specified path.

➔ To Add a File Stream:

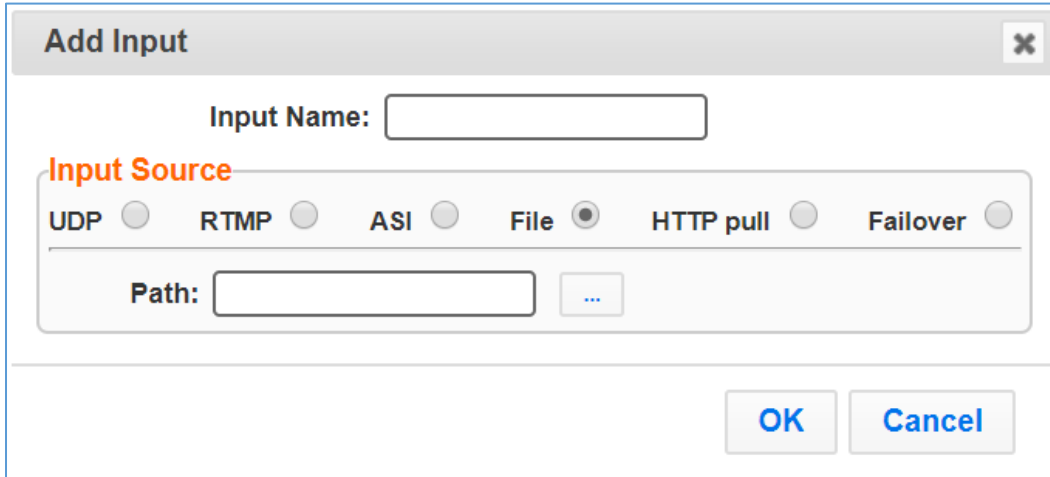
1. In the Zixi Feeder **Inputs** page, click the **+ New Input** button in the Menu bar.

¹ The following ASI cards have been tested and found to be compatible with Zixi on both Linux CentOS and Windows systems: DTA-2144B (4 ports) or DTA-2145 (2 ports). Additional types of ASI cards may also be compatible.

Adding Input Streams to Feeder

Adding an HTTP Pull Stream

The **Add Input** window is displayed:



2. Enter the unique **Input Name** (an internal name) for the Feeder.
3. Select **File** for the **Input Source**.
4. Specify the **Path** on the local machine where the file is located.
5. Click **OK**.

The “Input added” message appears on the top of the screen.

The File Input from the Zixi Feeder is added to the Zixi Broadcaster and now appears in the list of streams available in the Zixi Feeder inputs screen.

Input Stream Parameters Table

Parameter	Description
Input Name	Specify the unique identifier for this stream.
Path	Enter the path of the .ts file, which is located on the local or mounted (Linux) disk.

Adding an HTTP Pull Stream

This enables the Feeder to receive an HTTP Pull stream.

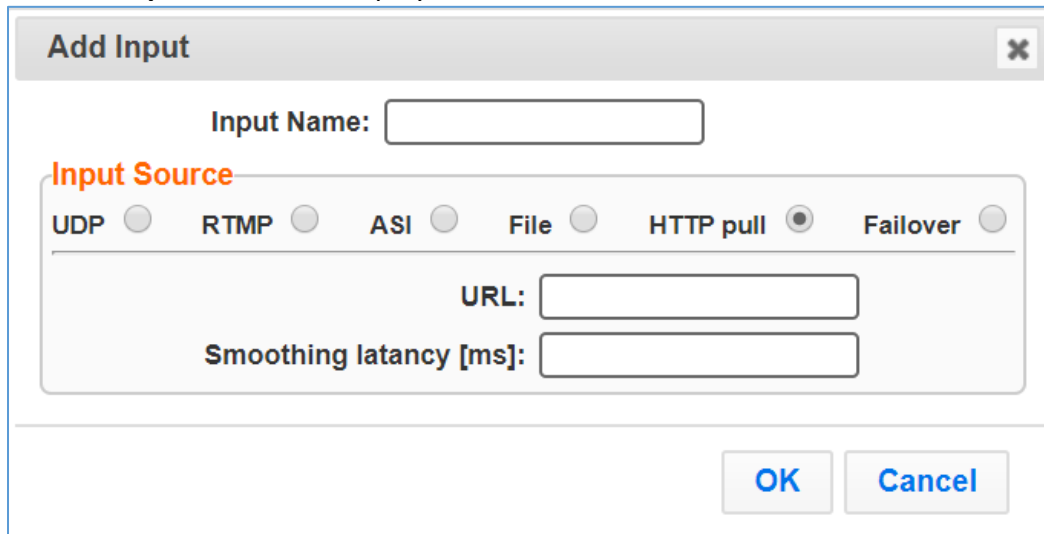
➔ To Add an HTTP Pull Stream:

1. In the Zixi Feeder **Inputs** page, click the **+ New Input** button in the Menu bar.

Adding Input Streams to Feeder

Adding a Failover Stream

The **Add Input** window is displayed:



2. Enter the unique **Input Name** (an internal name) for the Feeder.
3. Select **HTTP pull** for the **Input Source**.
4. Specify the **URL** of the Transport Stream.
5. Specify the **Smoothing latency** [ms] (see Input Stream Parameters Table below for details).
6. Click **OK**.

The “Input added” message appears on the top of the screen.

The HTTP Stream is added to the Zixi Feeder and now appears in the list of streams available in the Zixi Feeder inputs screen.

Input Stream Parameters Table

Parameter	Description
URL	Specify the URL of the Transport Stream.
Smoothing latency [ms]	The Smoothing feature uses a buffer to time packets according to the stream's clock. This is useful when there are bursts in the rate in which the stream arrives. The smoothing features creates a queue in the buffer, allowing smoother entry into the Feeder. The Smoothing parameter defines the size of the buffer in Milliseconds. A larger buffer will enable smoother entry, however it may influence the latency rate.

Adding a Failover Stream

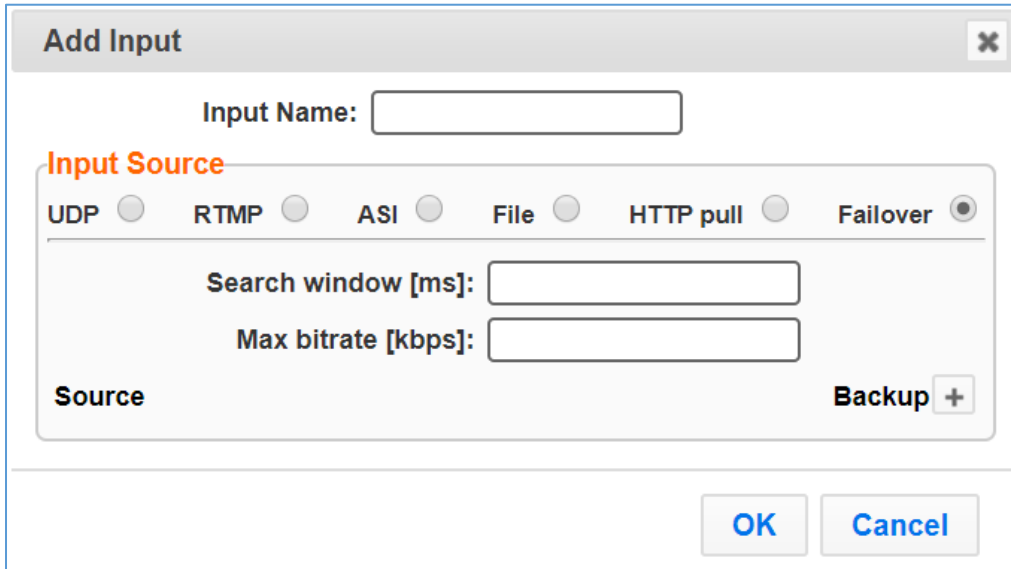
This enables the Feeder to use two or more binary-identical streams for hitless failover. For additional information about hitless failover and redundancy options, refer to the Zixi Redundancy Guide.

Adding Input Streams to Feeder

Adding a Failover Stream

➔ To Add a Failover Stream:

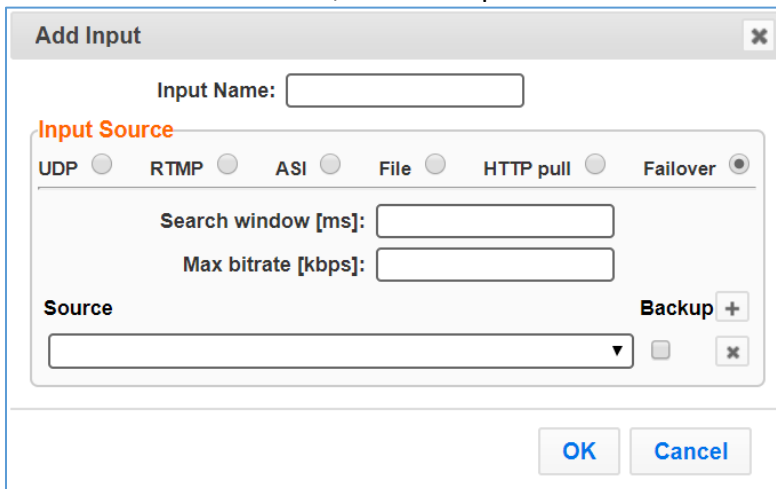
1. In the Zixi Feeder **Inputs** page, click the **+ New Input** button in the Menu bar. The **Add Input** window is displayed:



The screenshot shows the 'Add Input' dialog box. It has a title bar with 'Add Input' and a close button. Below the title bar is an 'Input Name' text box. Underneath is the 'Input Source' section with radio buttons for UDP, RTMP, ASI, File, HTTP pull, and Failover. The 'Failover' radio button is selected. Below the radio buttons are two text boxes: 'Search window [ms]:' and 'Max bitrate [kbps]:'. At the bottom left of the dialog is a 'Source' label and a 'Backup +' button. At the bottom right are 'OK' and 'Cancel' buttons.

2. Enter the unique **Input Name** (an internal name) for the Feeder.
3. Select **Failover** for the **Input Source**.
4. Specify the **Search window** [ms] (see Input Stream Parameters Table below for details).
5. Specify the **Max bitrate** [kbps] (see Input Stream Parameters Table below for details).
6. Click the **+** button.

The **Source** field is shown, with a dropdown list of available Sources.



This screenshot is similar to the previous one, but the 'Source' field now contains a dropdown menu with a downward arrow. The 'Backup +' button is now positioned to the right of the dropdown menu. The 'OK' and 'Cancel' buttons remain at the bottom.

7. Select a source from the list of Sources.
8. Repeat steps 6 and 7 for each Source that you would like to add to this Input.
9. Select the checkbox next to the Source/s that you would like to designate as a backup.
10. Click **OK**.

Adding Input Streams to Feeder

Searching for Input Streams in Feeder

The “Input added” message appears on the top of the screen.


The Failover Stream is added to the Zixi Feeder and now appears in the list of streams available in the Zixi Feeder inputs screen.

Input Stream Parameters Table

Parameter	Description
Search window [ms]	The time period for which packets are compared.
Max bitrate [kbps]	Specify the maximum expected bitrate for memory allocation. Recommended: 2X the actual bitrate, which will prevent buffer overruns.
Source	Specify each Source that you would like to add to this Failover stream.
Backup (checkbox)	Select the Source/s that you would like to designate as a backup.

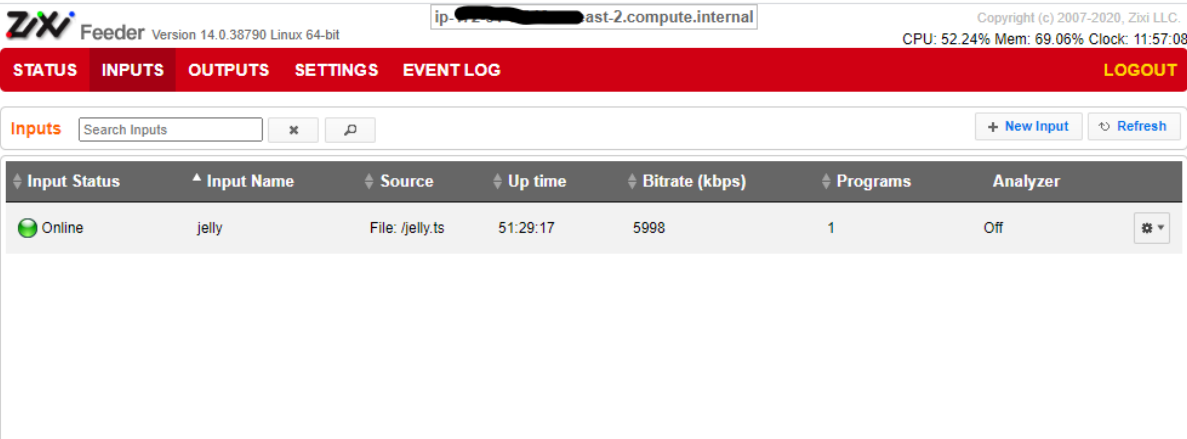
Searching for Input Streams in Feeder

➔ To Search for an Input Stream:

- In the **Inputs** page of Zixi Feeder, type in the specific input stream name in the search text box on the left- hand top of the screen and click . The **Input** screen displays the requested output stream(s).

Viewing Existing Input Streams in Feeder

The Inputs page displays all the input streams that have been created.



The screenshot shows the Zixi Feeder web interface. At the top, there is a navigation bar with tabs for STATUS, INPUTS, OUTPUTS, SETTINGS, and EVENT LOG, along with a LOGOUT button. Below the navigation bar, there is a search bar for inputs and a table of input streams. The table has the following columns: Input Status, Input Name, Source, Up time, Bitrate (kbps), Programs, and Analyzer. The first row of the table shows an input named 'jelly' with a status of 'Online', a source of 'File: /jelly.ts', an up time of '51:29:17', a bitrate of '5998', 1 program, and the analyzer is 'Off'.

Adding Input Streams to Feeder

Viewing Existing Input Streams in Feeder

Any of the columns in the following table can be sorted in ascending/descending order by clicking on its heading.

The Search text box in the upper left-hand side of the screen can be used to search for a specific input stream by entering the desired Stream ID.


Field	Description
Input Status	Displays the status of the input stream. The following colored indicators are displayed: <ul style="list-style-type: none">• Green = Connected• Yellow = Connecting• Red = Disconnected with an error condition• Blue = Offline with no connection attempts• Grey = Offline and stopped
Input Name	Displays the unique ID of the stream, as defined in the Zixi Feeder.
Source	Displays the source host and port number.
Up time	Displays the amount of time that the stream is live.
Bitrate [kbps]	Displays bitrate of the connected input stream.
Programs	Displays the number of channels or programs that compose the stream.
Analyzer	Displays if the Analyzer is active and the streams that are being analyzed according to the TR101 protocol. P1 and P2 are error priorities.

Performing Actions on Input Streams in Feeder

From the Inputs screen you have centralized control on the input streams and can do the following:

Stopping/Starting Incoming Streams in Feeder


➔ To Stop/Start an Input Streams:

1. In the Zixi Feeder **Inputs** page, click the menu button  for the incoming stream that you want to start or stop.
2. In the drop-down menu, click **Start/Stop** (depending on the activity of the stream). The Input stream in that row is either started or stopped.
To continue the stream, click again.

Editing Input Settings in Feeder


You can edit an input stream which has been previously configured.

➔ To Edit Input Settings:

1. In the Zixi Feeder **Inputs** page, click the menu button  for the incoming stream that you want to edit.
2. In the drop-down menu, click **Edit**. The stream details window of the input selected is displayed.
3. Make the necessary modifications and click **OK**.
The newly modified input takes effect.

Deleting Input Streams from Feeder

➔ To Delete Input Streams from Feeder:

1. In the Zixi Feeder **Inputs** page, click the menu button  for the incoming stream that you want to delete.
2. In the drop-down menu, click **Delete**.
3. The input stream in that row is deleted.


Duplicating an Input Stream in Feeder

In the event that there are similar channels coming from the same source, you can duplicate the configuration and make minor modifications.

Performing Actions on Input Streams in Feeder

Adding Analysis to an Input Stream


➔ To Duplicate the Configuration of an Input Stream:

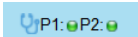
1. In the Zixi Feeder **Inputs** page, click the menu button  for the incoming stream that you want to duplicate.
2. In the drop-down menu, click **Duplicate**.
The Stream details windows opens with similar parameters to the original stream.
3. Type the new Input Name, and make any additional modifications that you want.
4. Click **OK**.
The duplicate input stream with its new settings is created.

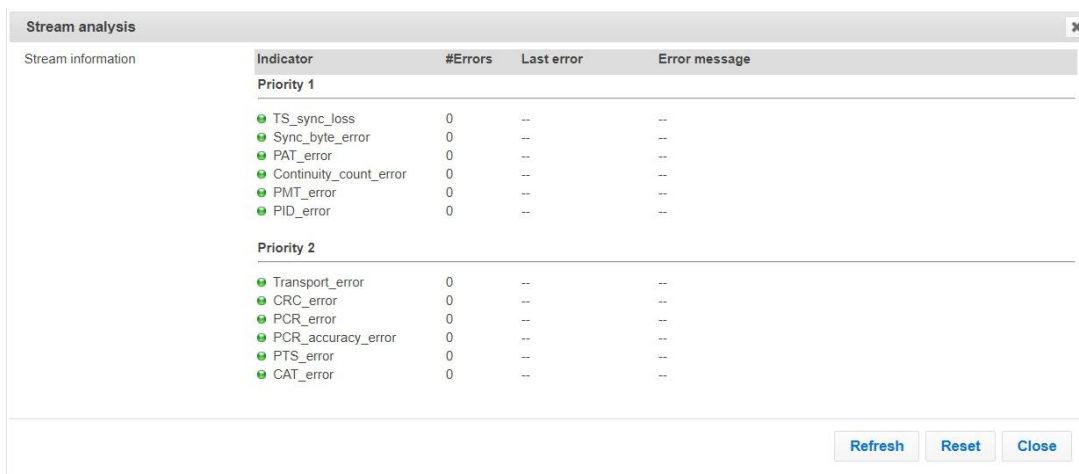
Adding Analysis to an Input Stream













A TR101 analysis can be added to the input stream for monitoring the TS while in operation. The analysis can be used to check the integrity of the TS and the most important elements of the stream. The analysis include priority 1 and 2.

➔ To Start Analysis of the Input Stream:

1. In the Zixi Feeder **Inputs** page, click the menu button  for the incoming stream that you want to analyze.
2. In the drop-down menu, click **Analyze**. The statuses of a **P1** (priority-1 errors) and **P2** (priority-2 errors) are indicated by an adjacent green or red button displayed in the Analyzer column. A green notification temporarily appears on the top of the screen.

You can view the detailed analysis by clicking on the adjacent  stethoscope icon. The Stream Analysis screen is displayed:



Stream information	Indicator	#Errors	Last error	Error message
Priority 1				
	 TS_sync_loss	0	--	--
	 Sync_byte_error	0	--	--
	 PAT_error	0	--	--
	 Continuity_count_error	0	--	--
	 PMT_error	0	--	--
	 PID_error	0	--	--
Priority 2				
	 Transport_error	0	--	--
	 CRC_error	0	--	--
	 PCR_error	0	--	--
	 PCR_accuracy_error	0	--	--
	 PTS_error	0	--	--
	 CAT_error	0	--	--

Refresh Reset Close

The following actions can be done on the Stream Analysis screen:


- **Refresh** – refreshes the data that is displayed.
- **Reset** – resets all the field's values to zero.

Clicking **Stop Analysis** in the drop-down menu discontinues the analysis.

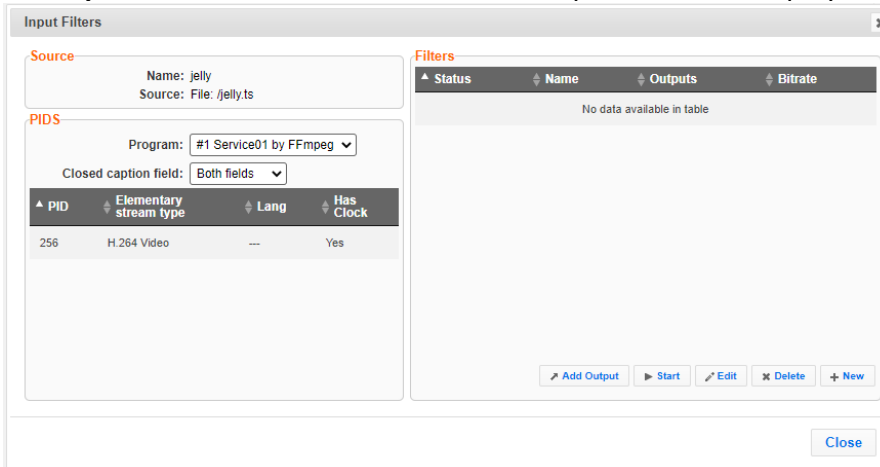
Adding Filters to Input Streams (PID Filtering)

Zixi Feeder can receive MPTS (Multi Program Transport Stream). If you do not wish to stream out all the programs included in an input stream, you can specify a PID filter for the input stream. The PID Filtering feature allows the user to filter the PIDs he would like to stream out as an output. The input MPEG-TS is composed of several PIDs, and the user can choose which PIDs to keep and which to remove.

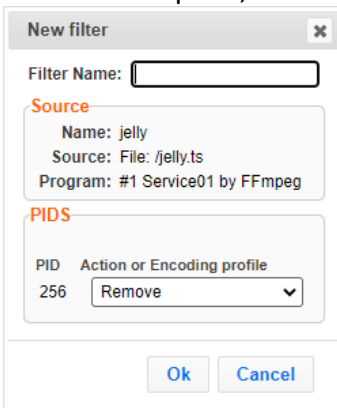
➔ To Add Filters to an Input Stream:

1. In the Zixi Feeder **Inputs** page, click the menu button  for the incoming stream that you want to filter.
2. Click, **Filters**.

The **Input Filters** window for the selected input stream is displayed:



3. In the **Program** drop-down list, select which programs you would like to include in the filter(s).
4. In the **Closed caption field** drop-down list select which field includes closed captions. The list of available PIDs is displayed.
5. In the **Filters** pane, click **+ New**. The **New filter** window is displayed.



6. Specify the **Filter Name**. This is an internal name for the filter.

Performing Actions on Input Streams in Feeder

Adding Filters to Input Streams (PID Filtering)

7. Under **PIDS**, for each channel select **Remove** (will be removed from the filtered stream) or **Keep** (will be included in the filtered stream).
8. Click **OK**. The new filter configuration is displayed under Filters. You can configure multiple filters for the same input stream.
9. Select the desired filter and click **Start** to begin streaming the filtered stream.
10. To configure the output for the filtered stream, See [Adding Outputs in Feeder](#).
11. Click **Close**.

Filter Parameters Table

Field	Description
Source	
Name	Displays the name of the input stream.
Source	Displays the input stream type source host and port number.
PIDS (Program IDs)	
Program	Specify the programs to display in the list below.
Closed caption field	Specify whether the program includes closed captions.
PID	Displays the Program IDs.
Elementary stream type	Displays the stream type (e.g. H264 video).
Lang	Displays the language of the audio track.
Has Clock	Displays whether the stream data includes a clock.
Filters	Displays a list of filters that were created and their statuses.
Status	Displays the status of the filter: <ul style="list-style-type: none">• Green – Active filter• Red – Offline filter
Name	Displays the filter name.
Outputs	Displays the number of outputs that use this filter.
Bitrate	Displays the bitrate of the filter stream.
Add Output	Defined an output stream for the selected filtered stream.
Start	Start transmitting the selected filtered stream.
Edit	Edits the selected filtered stream.

Performing Actions on Input Streams in Feeder

Adding Outputs in Feeder

Delete	Deletes the selected filtered stream.
New	Creates a new filter.

Adding Outputs in Feeder


This section describes how to add an output to a specific input stream as an alternative method (shortcut) as opposed to configuring it through the **Outputs** screen.

The available types of Zixi Feeder outputs are:

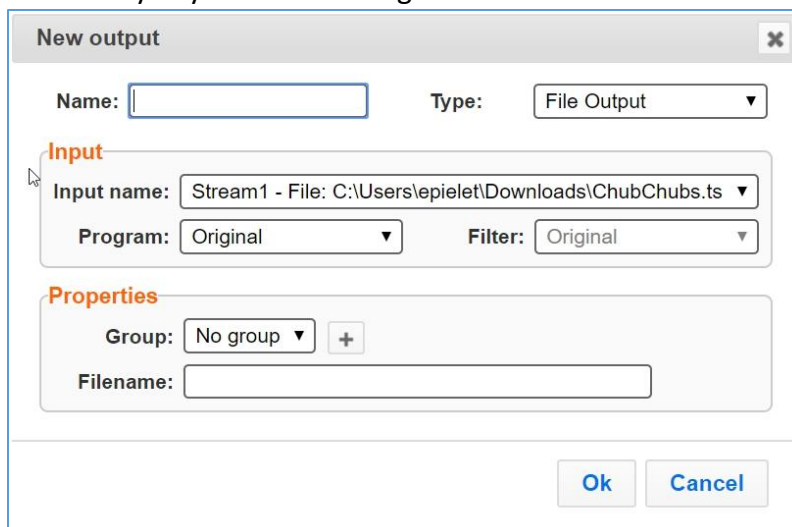
- **UDP Output** – Outputs the stream in UDP or RTP format over unicast or multicast.
- **File Output** – Zixi Feeder will output the stream as a file.
- **Push to Broadcaster** – Pushes streams to a Zixi Broadcaster (Zixi protocol).
- **HTTP Push** - Pushes a Transport Stream over HTTP to a target URL.

Adding a File Output

➔ To Add a File Output in Feeder:

1. In the Zixi Feeder **Inputs** page, click the menu button  for the incoming stream for which you want to configure an output.
2. In the drop-down menu, click **New Output**.

The **New output** window with the settings of the selected input stream is displayed. You can modify any of these settings.



3. Enter the unique **Name** that you want to configure on the Feeder.
4. Select **File Output** as the **Type** of output.
5. Enter the **Input** Parameters (see Output Parameters Table below for additional parameters).
6. Enter the **Filename** for saving the outputted file.
7. Click **OK**.

Performing Actions on Input Streams in Feeder

Adding Outputs in Feeder


The **New Output** with its new settings takes effect.

Output Parameters Table

Parameter	Description
Name	Specify the unique name for the stream.
Type	Specify the output type. Select File Output .
Input	
Input Name	Select the Input which has been previously configured as Input in the Feeder.
Program	If the stream is a Multi Program Transport Stream and you want to stream a specific program, select the relevant Program from the drop-down list. Select Original to stream all the programs in the stream.
Filter	Within the selected program you can stream specific PIDs by creating a Filter (see: Adding Filters to Input Streams (PID Filtering) section) and then selecting it from the Filter drop-down list. Select Original to stream all PIDs within the program.
Properties	
Group	Not applicable.
Filename	Specify the location in which the file will be saved.

Adding a Push to Broadcaster Output

➔ To Add a Push to Broadcaster Output:

1. In the Zixi Feeder **Inputs** page, click the menu button  for the incoming stream for which you want to configure an output.
2. In the drop-down menu, click **New Output**.

Performing Actions on Input Streams in Feeder

Adding Outputs in Feeder


3. The **New Output** window with the settings of the selected input stream is displayed. You can modify any of these settings:

The screenshot shows the 'New output' dialog box. The 'Name' field is empty. The 'Type' dropdown is set to 'Push to Broadcaster'. Under the 'Input' section, the 'Input name' dropdown is set to 'bbb - udp: 4554@Localhost', the 'Program' dropdown is set to 'Original', and the 'Filter' dropdown is set to 'Original'. Under the 'Properties' section, the 'Group' dropdown is set to 'No group', 'Stream ID' and 'Password' are empty, 'Max bitrate (kbps)' is set to '8000', 'Max latency (ms)' is set to '6000', and 'Ignore TLS certificate' is unchecked. Below this is a table with columns 'Host', 'Port', and 'IP'. The 'Port' is set to '2088' and 'IP' is set to 'Any'. There is a '+' button to the right of the table and an 'x' button to the right of the 'IP' dropdown. Below the table is 'Bond Links' which is unchecked. At the bottom right of the dialog is 'Show all settings' which is unchecked. At the very bottom are 'Ok' and 'Cancel' buttons.

4. Enter the unique **Name** that you want to configure on the Feeder.
5. Select **Push to Broadcaster** as the **Type** of output.
6. Select the **Input name** from the drop-down list.



The drop-down list consists of all previously configured inputs in the Feeder.

7. If the stream is a *Multi Program Transport Stream* and you want to stream a specific program, select the relevant **Program** from the drop-down list. Select **Original** to stream all the programs in the stream.
8. Within the selected program you can stream specific PIDs by creating a Filter (see: [Adding Filters to Input Streams \(PID Filtering\)](#) section) and then selecting it from the **Filter** drop-down list. Select **Original** to stream all PIDs within the program.
9. If you would like to add this stream to an Output Group, select an existing **Group** from the dropdown list OR create a new group by clicking  (see Output Parameters Table below for additional details).

Performing Actions on Input Streams in Feeder

Adding Outputs in Feeder

10. Enter the **Properties** (see table below for additional parameters).
11. If desired, enter the **Advanced Properties** (see Output Parameters Table below for additional parameters).
12. If desired, enter the **Remote RTMP Output** (see Output Parameters Table below for additional parameters).
13. Click **OK**.

The **New Output** with its new settings takes effect.

Output Parameters Table

Parameter	Description
Name	Specify the unique name of the stream.
Type	Specify the output type. Select Push to Broadcaster .
Input	
Input Name	Select the Input which has been previously configured as Input in the Feeder.
Program	If the stream is a Multi Program Transport Stream and you want to stream a specific program, select the relevant Program from the drop-down list. Select Original to stream all the programs in the stream.
Filter	Within the selected program you can stream specific PIDs by creating a Filter (see: Adding Filters to Input Streams (PID Filtering) section) and then selecting it from the Filter drop-down list. Select Original to stream all PIDs within the program.
Properties	
Group	If your deployment includes a DNS Load Balancer, it is possible to group multiple outputs into a single Output Group. All the outputs created in Zixi Feeder within the same group and with the same host name, will be resolved into the same IP address. You can create a new output group or use an existing group. See OUTPUT GROUPS .
Stream ID	Specify the unique Stream ID that the Broadcaster expects. Must be identical to the one configured on the Broadcaster.
Password	Optional. Specify a password string that can be used to authenticate that the Feeder is allowed to push this stream to the configured Zixi Broadcaster. Only a Feeder who has the Password and the Stream ID of the input stream that was configured on the Broadcaster can stream into it.
Max bitrate [kbps]	Specify the maximum bitrate of the stream. Default: 8000.

Performing Actions on Input Streams in Feeder

Adding Outputs in Feeder

Parameter	Description
	<p>Note: Will be used for internal memory allocation and will not affect the transmitted bitrate.</p> <p>Recommended: double the bitrate of the stream.</p>
Max Latency [ms]	Specify the maximum number of milliseconds for which the Zixi Broadcaster should buffer the stream. Default: 6000
Ignore TLS certificate	Select this option when streaming to a trusted server that does not have a valid certificate. When this option is selected, Zixi Feeder will ignore the TLS certificate errors and transmit the stream to its destination.
Host	Specify the destination host for the push stream. Note: additional failover destinations can be added by clicking the adjacent “+” button.
Port	Specify the destination port (to listen on for Push streams). Default: 2088.
IP	Specify the local IP address. The default enables Any IP address.
Backup	Select Backup to define the destination as a backup destination. In this case the destination will be active only if the non-backup destinations are not available. Additional destinations that are not defined as backups will receive the stream simultaneously, allowing hitless failover in case the other fails.
Bond Links	Select this checkbox to split the stream into multiple links, which will be forwarded to a single Zixi Broadcaster.
Max Bitrate (for Bond Links enabled)	Relevant for Bond Links option. Specify the max bitrate in Kbps for the specific link.
Backup (for Bond Links enabled)	Relevant for Bond Links option. Select this checkbox to define the link as a backup. The backup will be used only if the accumulated bandwidth of the non-backup links is not sufficient.
Advanced Properties	
PCR Smoothing delay (ms)	Enables transmission of the output at the correct rate (pacing). Required when the receiving device is sensitive and can't lock on to the stream (in such a case, the recommended latency is 100 - 1000 ms).
Encryption Type	Specify the type of Encryption. Options are: None (default), AES 128, AES 192, AES 256 or Automatic. Selection of Automatic will generate the encryption and exchange it securely between the applications automatically.

Performing Actions on Input Streams in Feeder

Adding Outputs in Feeder

Parameter	Description
Key (If encryption was enabled)	Click Generate and the encryption key for the input stream will appear in the adjacent text box. Send it to the end-user to decipher the received encoded stream.
DSCP	Specify the DSCP decimal value. This setting is relevant for networks that support Differentiated Services, a mechanism for classifying and managing network traffic and providing quality of service (QoS) on modern IP networks.
ECN	<p>Zixi Feeder supports Explicit Congestion Notification. To enable ECN, select one of the following options that represent codepoint encoding:</p> <p>ECT0 - ECN Capable Transport (10) ECT1 - ECN Capable Transport (01)</p> <p>ECN allows end-to-end notification of network congestion without dropping packets. If the underlying network infrastructure supports ECN, this optional feature may be used between two ECN-enabled endpoints to allow notifications of network congestions without dropping packets. Conventionally, TCP/IP networks signal congestion by dropping packets. When ECN is successfully negotiated, an ECN-aware router may set a mark in the IP header instead of dropping a packet in order to signal impending congestion. The receiver of the packet echoes the congestion indication to the sender, which reduces its transmission rate as if it detected a dropped packet.</p>
RTMP input feedback	For RTMP streams only. Select this checkbox to adjust the Zixi Feeder reading speed of the RTMP input according to the network connection speed (between the Zixi Feeder and Zixi Broadcaster). This can be useful if the RTMP source (encoder) can adjust its bitrate. When enabled, the Zixi Broadcaster will send a message through the Zixi Feeder the encoder that specifies whether to decrease or increase the bitrate.
Encoder feedback script	<p>Select this checkbox to receive reports back to the encoder about the current available bitrate according to Zixi's network condition analysis. Supporting encoders will adjust their output bitrate accordingly.</p> <p>Note: for supporting encoders only.</p> <p>This option requires installation of a special script. For more information, see Configuring Encoder Feedback Script.</p>
Minimum bitrate [kbps]	If Encoder feedback script option is enabled, specify the minimal bitrate for the stream. This will be the lowest bitrate, which can be recommended by the Zixi Feeder to the encoder.


Performing Actions on Input Streams in Feeder

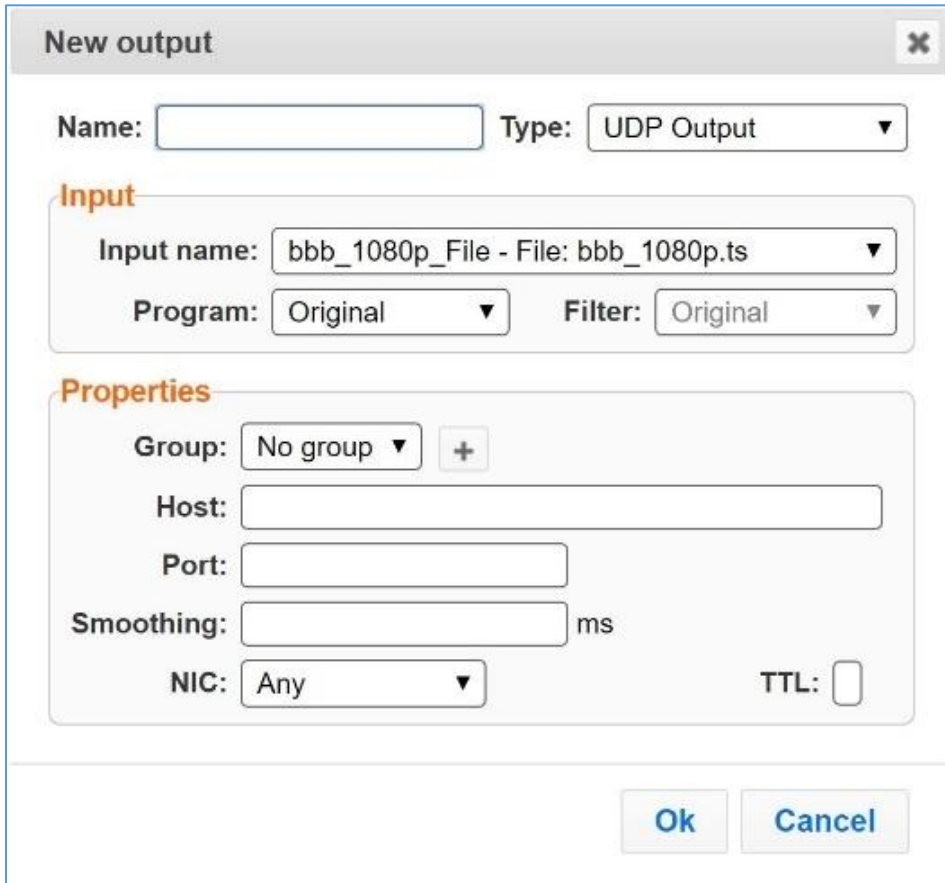
Adding Outputs in Feeder

Parameter	Description
Encoder address	If Encoder feedback script option is enabled, specify the IP address of the source encoder.
Remote RTMP Output	This setting allows sending a Zixi-protected stream to Zixi Broadcaster and then Zixi Broadcaster automatically creates an RTMP output to the remote RTMP Server in the specified destination. Note: Allow automatic RTMP Output option must be enabled on the Zixi Broadcaster.
URL	Specify the URL of the remote RTMP server.
Backup URL	Specify a backup URL (similar to the primary URL) in case the primary RTMP server is faulty, then the Broadcaster will switch and stream to the backup one.
Stream	Specify the unique name for the stream that the RTMP server expects at the other end of the connection.
Username	Specify the Username to authenticate on the remote RTMP server.
Password	Specify the string that is used for authorization on the remote RTMP server.
Hot Backup	Click this checkbox to set the Broadcaster to output to the two RTMP servers (Primary & Backup) in parallel.

Adding a UDP Output

➔ To Add UDP Output:

1. In the Zixi Feeder **Inputs** page, click the menu button  for the incoming stream for which you want to configure an output.
2. In the drop-down menu, click **New Output**.
The **New output** window with the settings of the selected input stream is displayed. You can modify any of these settings.



New output

Name: Type: UDP Output ▼

Input

Input name: bbb_1080p_File - File: bbb_1080p.ts ▼

Program: Original ▼ Filter: Original ▼

Properties

Group: No group ▼ +


Host:

Port:

Smoothing: ms

NIC: Any ▼ TTL:

Ok Cancel

3. Enter the unique **Name** that you want to configure on the Feeder.
4. Select **UDP Output** as the **Type** of output.
5. Enter the **Input** parameters (see **Output Parameters Table** below for additional parameters).
6. If you would like to add this stream to an Output Group, select an existing **Group** from the dropdown list OR create a new group by clicking . (see Output Parameters Table below for additional details).
7. In the **Host** field, specify the IP address of the stream destination.
8. In the **Port** field, specify the port of the destination.
9. In the **NIC** field, select the network card from which the stream will be transmitted.

Performing Actions on Input Streams in Feeder

Adding Outputs in Feeder

10. Click **OK**.


The new UDP Output with its settings is created and now appears on the Outputs screen.

Output Parameters Table

Parameter	Description
Name	Specify the unique name of the stream.
Type	Specify the output type. Select UDP Output .
Input	
Input Name	Select the Input which has been previously configured as Input in the Feeder.
Program	If you have created a filter, you can select the relevant program.
Filter	If you have created a filter, you can select the relevant filter.
Properties	
Group	If your deployment includes a DNS Load Balancer, it is possible to group multiple outputs into a single Output Group. All the outputs created in Zixi Feeder within the same group and with the same host name, will be resolved into the same IP address. You can create a new output group or use an existing group. See OUTPUT GROUPS .
Host	Specify the IP address of the stream destination.
Port	Specify the ports to listen on for the output streams.
Smoothing	Enables transmission of the output at the correct rate. Required when the receiving device is sensitive and can't lock on to the stream (in such a case, the recommended latency is 100 - 1000 ms).
NIC	Specify the Network card used.
TTL	Specify the network Time-to-Live parameter.

Adding an HTTP Push Output

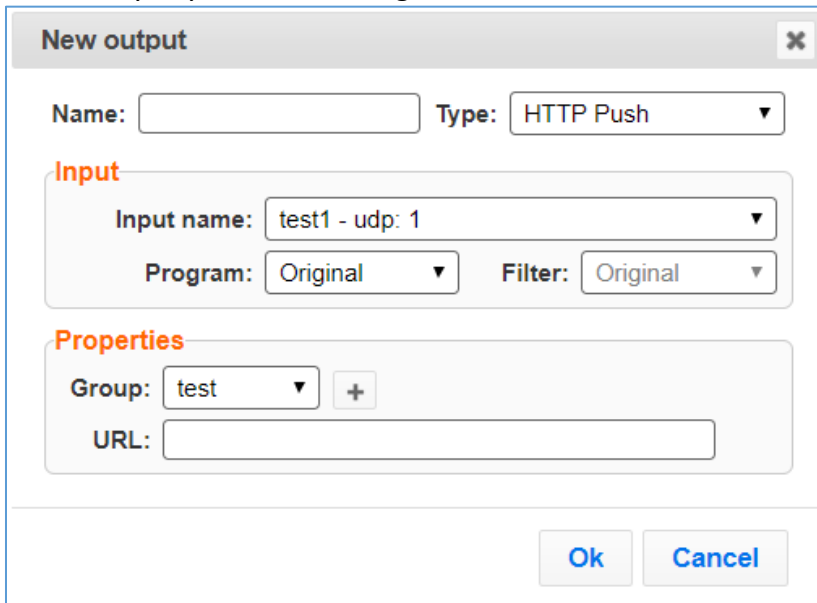
➔ To Add an HTTP Push Output in Feeder:

1. In the Zixi Feeder **Inputs** page, click the menu button  for the incoming stream for which you want to configure an output.
2. In the drop-down menu, click **New Output**.

Performing Actions on Input Streams in Feeder

Adding Outputs in Feeder

The **New output** window with the settings of the selected input stream is displayed. You can modify any of these settings.



3. Enter the unique **Name** that you want to configure on the Feeder.
4. Select **File Output** as the **Type** of output.
5. Enter the **Input** Parameters (see Output Parameters Table below for additional parameters).
6. If you would like to add this stream to an Output Group, select an existing **Group** from the dropdown list OR create a new group by clicking **+**. (see Output Parameters Table below for additional details).
7. Specify the **URL** of the target.
8. Click **OK**.

The new HTTP Push Output with its new settings takes effect.

Output Parameters Table

Parameter	Description
Name	Specify the unique name for the stream.
Type	Specify the output type. Select File Output .
Input	
Input Name	Select the stream which has been previously configured as an Input in the Feeder.
Program	If you have created a filter, you can select the relevant program.
Filter	If you have created a filter, you can select the relevant filter.
Properties	

Performing Actions on Input Streams in Feeder

Adding Outputs in Feeder

Group

If your deployment includes a DNS Load Balancer, it is possible to group multiple outputs into a single Output Group. All the outputs created in Zixi Feeder within the same group and with the same host name, will be resolved into the same IP address. You can create a new output group or use an existing group. See [OUTPUT GROUPS](#).

URL

Specify the URL of the target.

Creating Outputs in Feeder

Zixi Broadcaster supports the following output types:

- **UDP Output** – Outputs the stream in UDP or RTP format over unicast or multicast.
- **File Output** – Zixi Feeder will output the stream as a file.
- **Push to Broadcaster** – Push streams that are Zixi protocol streams to a Zixi Broadcaster.
- **HTTP Push** - Pushes a Transport Stream over HTTP to a target URL.

Output Grouping: For deployments that include a DNS Load Balancer, it is possible to group multiple outputs into a single Output Group. All the outputs created in Zixi Feeder within the same group and with the same host name, will be resolved into the same IP address (i.e. the same Zixi Broadcaster). This feature can be used to ensure all adaptive streams will be delivered to the same machine (i.e. resolved to the same destination IP address). For more information, see [OUTPUT GROUPS](#).

Network Bonding: Zixi's Network Bonding feature enables you to divide a stream into several network channels by utilizing multiple NICs and links and then subsequently reuniting them at the Zixi Broadcaster side.

Creating a File Output in Feeder

The File Output will output the stream as a file.

➔ To Create a File Output:

1. In the **Outputs** page of Zixi Feeder, click the **+ New Output** button in the menu bar. The **New output** window is displayed:

The screenshot shows the 'New output' dialog box. The 'Name' field is empty. The 'Type' dropdown is set to 'File Output'. Under the 'Input' section, the 'Input name' dropdown is set to 'Stream1 - File: C:\Users\epielet\Downloads\ChubChubs.ts', 'Program' is set to 'Original', and 'Filter' is set to 'Original'. Under the 'Properties' section, the 'Group' dropdown is set to 'No group' with a '+' button next to it, and the 'Filename' field is empty. 'Ok' and 'Cancel' buttons are at the bottom right.

2. Type the unique identifier for this output in the **Name** field.
3. In the **Type** field, select **File Output**.
4. In the **Input Name** drop-down list, select the desired input stream.

Creating Outputs in Feeder

Creating a File Output in Feeder

5. If the stream is a *Multi Program Transport Stream* and you want to stream a specific program, select the relevant **Program** from the drop-down list. Select **Original** to stream all the programs in the stream.
6. Within the selected program you can stream specific PIDs by creating a Filter (see: [Adding Filters to Input Streams \(PID Filtering\)](#) section) and then selecting it from the **Filter** drop-down list. Select **Original** to stream all PIDs within the program.
7. In the **Group** field, verify that *No group*, which is the default setting, is selected.
8. Enter the **Filename** for saving the output file.
9. Click **OK**.

The new File Output with its settings is created and now appears in the list of outputs.

Output Parameters Table

Parameter	Description
Name	Specify the unique name for the stream.
Type	Specify the output type. Select File Output
Input	
Input Name	Select the Input which has been previously configured as Input in the Feeder.
Program	If the stream is a Multi Program Transport Stream and you want to stream a specific program, select the relevant Program from the drop-down list. Select Original to stream all the programs in the stream.
Filter	Within the selected program you can stream specific PIDs by creating a Filter (see: Adding Filters to Input Streams (PID Filtering) section) and then selecting it from the Filter drop-down list. Select Original to stream all PIDs within the program.
Properties	
Group	Not applicable.
Filename	Specify the location in which the file will be saved.

Creating a Push to Broadcaster Output

The Push to Broadcaster output pushes the stream from the Zixi Feeder to a Zixi Broadcaster, while using the Zixi protocol. It is possible to divide a stream into several network channels by utilizing multiple NICs and links and then subsequently reuniting them at the Zixi Broadcaster side. For more information see [Network Bonding](#).

➔ To Create a Push to Broadcaster Output:

1. In the **Outputs** page of Zixi Feeder, click the **+ New Output** button in the menu bar. The **New Output** window is displayed:

The screenshot shows the 'New output' configuration window. The 'Name' field is empty, and the 'Type' is set to 'Push to Broadcaster'. Under the 'Input' section, the 'Input name' is 'bbb - udp: 4554@Localhost', 'Program' is 'Original', and 'Filter' is 'Original'. The 'Properties' section includes 'Group' (No group), 'Stream ID', 'Password', 'Max bitrate (kbps)' (8000), 'Max latency (ms)' (6000), and 'Ignore TLS certificate' (unchecked). A table for 'Host', 'Port', and 'IP' shows '2088' and 'Any'. 'Bond Links' and 'Show all settings' are also unchecked. 'Ok' and 'Cancel' buttons are at the bottom.

2. Enter the unique **Name** that you want to configure on the Feeder.
3. Select **Push to Broadcaster** as the **Type** of output.
4. Select the **Input name** from the drop-down list.



The drop-down list consists of all previously configured inputs in the Feeder.

Creating Outputs in Feeder

Creating a Push to Broadcaster Output

5. If the stream is a *Multi Program Transport Stream* and you want to stream a specific program, select the relevant **Program** from the drop-down list. Select **Original** to stream all the programs in the stream.
6. Within the selected program you can stream specific PIDs by creating a Filter (see: [Adding Filters to Input Streams \(PID Filtering\)](#) section) and then selecting it from the **Filter** drop-down list. Select **Original** to stream all PIDs within the program.
7. If you would like to add this stream to an Output Group, select an existing **Group** from the dropdown list OR create a new group by clicking (see Output Parameters Table below for additional details).
8. Enter the **Properties** (see table below for additional parameters).
9. If desired, enter the **Advanced Properties** (see Output Parameters Table below for additional parameters).
10. If desired, enter the **Remote RTMP Output** (see Output Parameters Table below for additional parameters).
11. Click **OK**.

The **New Output** with its new settings takes effect.

Output Parameters Table

Parameter	Description
Name	Specify the unique name of the stream.
Type	Specify the output type. Select Push to Broadcaster .
Input	
Input Name	Select the Input which has been previously configured as Input in the Feeder.
Program	If the stream is a Multi Program Transport Stream and you want to stream a specific program, select the relevant Program from the drop-down list. Select Original to stream all the programs in the stream.
Filter	Within the selected program you can stream specific PIDs by creating a Filter (see: Adding Filters to Input Streams (PID Filtering) section) and then selecting it from the Filter drop-down list. Select Original to stream all PIDs within the program.
Properties	
Group	If your deployment includes a DNS Load Balancer, it is possible to group multiple outputs into a single Output Group. All the outputs created in Zixi Feeder within the same group and with the same host name, will be resolved into the same IP address. You can create a new output group or use an existing group. See OUTPUT GROUPS .

Creating Outputs in Feeder

Creating a Push to Broadcaster Output

Parameter	Description
Stream ID	Specify the unique Stream ID that the Broadcaster expects. Must be identical to the one configured on the Broadcaster.
Password	Optional. Specify a password string that can be used to authenticate that the Feeder is allowed to push this stream to the configured Zixi Broadcaster. Only a Feeder who has the Password and the Stream ID of the input stream that was configured on the Broadcaster can stream into it.
Max bitrate [kbps]	Specify the maximum bitrate of the stream. Default: 8000. Note: Will be used for internal memory allocation and will not affect the transmitted bitrate. Recommended: double the bitrate of the stream.
Max Latency [ms]	Specify the maximum number of milliseconds for which the Zixi Broadcaster should buffer the stream. Default: 6000
Ignore TLS certificate	Select this option when streaming to a trusted server that does not have a valid certificate. When this option is selected, Zixi Feeder will ignore the TLS certificate errors and transmit the stream to its destination.
Host	Specify the destination host for the push stream. Note: additional failover destinations can be added by clicking the adjacent “+” button.
Port	Specify the destination port (to listen on for Push streams). Default: 2088.
IP	Specify the local IP address. The default enables Any IP address.
Backup	Select Backup to define the destination as a backup destination. In this case the destination will be active only if the non-backup destinations are not available. Additional destinations that are not defined as backups will receive the stream simultaneously, allowing hitless failover in case the other fails.
Bond Links	Select this checkbox to split the stream into multiple links, which will be forwarded to a single Zixi Broadcaster.
Max Bitrate (for Bond Links enabled)	Relevant for Bond Links option. Specify the max bitrate in Kbps for the specific link.
Backup (for Bond Links enabled)	Relevant for Bond Links option. Select this checkbox to define the link as a backup. The backup will be used only if the accumulated bandwidth of the non-backup links is not sufficient.

Creating Outputs in Feeder

Creating a Push to Broadcaster Output

Parameter	Description
Advanced Properties	
PCR Smoothing delay (ms)	Enables transmission of the output at the correct rate (pacing). Required when the receiving device is sensitive and can't lock on to the stream (in such a case, the recommended latency is 100 - 1000 ms).
Encryption Type	Specify the type of Encryption. Options are: None (default), AES 128, AES 192, AES 256 or Automatic. Selection of Automatic will generate the encryption and exchange it securely between the applications automatically.
Key (If encryption was enabled)	Click Generate and the encryption key for the input stream will appear in the adjacent text box. Send it to the end-user to decipher the received encoded stream.
DSCP	Specify the DSCP decimal value. This setting is relevant for networks that support Differentiated Services, a mechanism for classifying and managing network traffic and providing quality of service (QoS) on modern IP networks.
ECN	Zixi Feeder supports Explicit Congestion Notification. To enable ECN, select one of the following options that represent codepoint encoding: ECT0 - ECN Capable Transport (10) ECT1 - ECN Capable Transport (01) ECN allows end-to-end notification of network congestion without dropping packets. If the underlying network infrastructure supports ECN, this optional feature may be used between two ECN-enabled endpoints to allow notifications of network congestions without dropping packets. Conventionally, TCP/IP networks signal congestion by dropping packets. When ECN is successfully negotiated, an ECN-aware router may set a mark in the IP header instead of dropping a packet in order to signal impending congestion. The receiver of the packet echoes the congestion indication to the sender, which reduces its transmission rate as if it detected a dropped packet.
RTMP input feedback	For RTMP streams only. Select this checkbox to adjust the Zixi Feeder reading speed of the RTMP input according to the network connection speed (between the Zixi Feeder and Zixi Broadcaster). This can be useful if the RTMP source (encoder) can adjust its bitrate. When enabled, the Zixi Broadcaster will send a message through the Zixi Feeder the encoder that specifies whether to decrease or increase the bitrate.
Encoder feedback script	Select this checkbox to receive reports back to the encoder about the current available bitrate according to Zixi's network condition analysis. Supporting encoders will adjust their output bitrate

Creating Outputs in Feeder

Creating a UDP Output in Feeder

Parameter	Description
	accordingly. Note: for supporting encoders only.
	This option requires installation of a special script. For more information, see Configuring Encoder Feedback Script .
Minimum bitrate [kbps]	If Encoder feedback script option is enabled, specify the minimal bitrate for the stream. This will be the lowest bitrate, which can be recommended by the Zixi Feeder to the encoder.
Encoder address	If Encoder feedback script option is enabled, specify the IP address of the source encoder.
Remote RTMP Output	This setting allows sending a Zixi-protected stream to Zixi Broadcaster and then Zixi Broadcaster automatically creates an RTMP output to the remote RTMP Server in the specified destination. Note: Allow automatic RTMP Output option must be enabled on the Zixi Broadcaster.
URL	Specify the URL of the remote RTMP server.
Backup URL	Specify a backup URL (similar to the primary URL) in case the primary RTMP server is faulty, then the Broadcaster will switch and stream to the backup one.
Stream	Specify the unique name for the stream that the RTMP server expects at the other end of the connection.
Username	Specify the Username to authenticate on the remote RTMP server.
Password	Specify the string that is used for authorization on the remote RTMP server.
Hot Backup	Click this checkbox to set the Broadcaster to output to the two RTMP servers (Primary & Backup) in parallel.

Creating a UDP Output in Feeder

The UDP output outputs the stream in UDP or RTP format over unicast or multicast.

Creating Outputs in Feeder

Creating a UDP Output in Feeder

➔ To Create a UDP Output:

1. In the **Outputs** screen of Zixi Feeder, click the **+ New Output** button in the menu bar. The **New output** window is displayed:

The screenshot shows the 'New output' dialog box. The 'Name' field is empty. The 'Type' dropdown is set to 'UDP Output'. Under the 'Input' section, the 'Input name' dropdown is set to 'bbb_1080p_File - File: bbb_1080p.ts', 'Program' is set to 'Original', and 'Filter' is set to 'Original'. Under the 'Properties' section, 'Group' is set to 'No group' with a '+' button next to it. The 'Host', 'Port', and 'Smoothing' fields are empty. 'NIC' is set to 'Any' and 'TTL' is empty. 'Ok' and 'Cancel' buttons are at the bottom.

2. Type in the unique identifier for this output in the **Name** field.
3. From the Type drop-down list, select **UDP Output**.
4. From the **Input Name** drop-down list, select the desired input stream.
5. If the stream is a *Multi Program Transport Stream* and you want to stream a specific program, select the relevant **Program** from the drop-down list. Select **Original** to stream all the programs in the stream.
6. Within the selected program you can stream specific PIDs by creating a Filter (see: [Adding Filters to Input Streams \(PID Filtering\)](#) section) and then selecting it from the **Filter** drop-down list. Select **Original** to stream all PIDs within the program.
7. Specify the **Properties** (see Output Parameters Table below for additional parameters).
8. If you would like to add this stream to an **Output Group**, select an existing **Group** from the dropdown list OR create a new group by clicking . (see Output Parameters Table below for additional details).
9. In the **Host** field, specify the IP address of the stream destination.
10. In the **Port** field, specify the port of the destination.
11. In the **NIC** field, select the network card from which the stream will be transmitted.
12. Click **OK**.

The new UDP Output with its settings is created and now appears on the Outputs screen.

Output Parameters Table

Parameter	Description
-----------	-------------

Creating Outputs in Feeder

Creating a UDP Output in Feeder

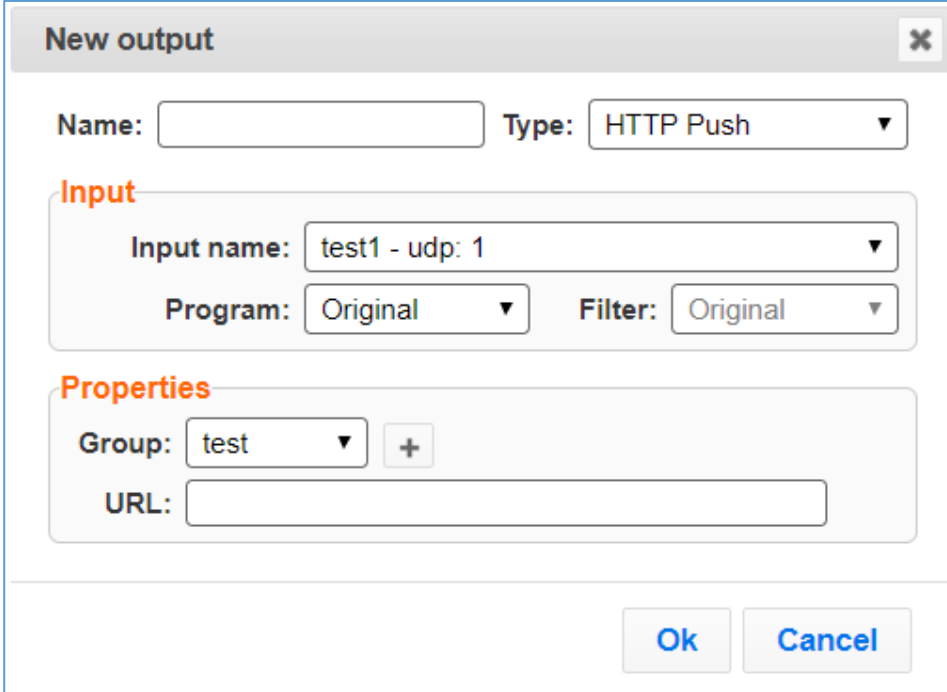
Name	Specify the unique name of the stream.
Type	Specify what kind of activity is being performed on the stream.
Input	
Input Name	Select the Input which has been previously configured as Input in the Feeder.
Program	If the stream is a Multi Program Transport Stream and you want to stream a specific program, select the relevant Program from the drop-down list. Select Original to stream all the programs in the stream.
Filter	Within the selected program you can stream specific PIDs by creating a Filter (see: Adding Filters to Input Streams (PID Filtering) section) and then selecting it from the Filter drop-down list. Select Original to stream all PIDs within the program.
Properties	
Group	If your deployment includes a DNS Load Balancer, it is possible to group multiple outputs into a single Output Group. All the outputs created in Zixi Feeder within the same group and with the same host name, will be resolved into the same IP address. You can create a new output group or use an existing group. See OUTPUT GROUPS .
Host	Specify the IP address of the stream destination.
Port	Specify the ports to listen on for the output streams.
Smoothing	Enables transmission of the output at the correct rate. Required when the receiving device is sensitive and can't lock on to the stream (in such a case, the recommended latency is 100 - 1000 ms).
NIC	Specify the Network card used.
TTL	Specify the network Time-to-Live parameter.

Creating an HTTP Push Output in Feeder

The HTTP Push Output will output the stream as a Transport Stream over HTTP to a target URL.

➔ To Create an HTTP Push Output:

1. In the **Outputs** page of Zixi Feeder, click the **+ New Output** button in the menu bar. The **New output** window is displayed:



2. Type the unique identifier for this output in the **Name** field.
3. In the **Type** field, select **HTTP Push**.
4. In the **Input Name** drop-down list, select the desired input stream.
5. If the stream is a *Multi Program Transport Stream* and you want to stream a specific program, select the relevant **Program** from the drop-down list. Select **Original** to stream all the programs in the stream.
6. Within the selected program you can stream specific PIDs by creating a Filter (see: [Adding Filters to Input Streams \(PID Filtering\)](#) section) and then selecting it from the **Filter** drop-down list. Select **Original** to stream all PIDs within the program.
7. If you would like to add this stream to an **Output Group**, select an existing **Group** from the dropdown list OR create a new group by clicking **+**. (see Output Parameters Table below for additional details).
8. Enter the **URL** of the target.
9. Click **OK**.
The new HTTP Push Output with its settings is created and now appears in the list of outputs.

Output Parameters Table

Parameter	Description
Name	Specify the unique name for the stream.
Type	Specify the output type. Select HTTP Push .
Input	
Input Name	Select the Input which has been previously configured as an Input in the Feeder.
Program	If the stream is a Multi Program Transport Stream and you want to stream a specific program, select the relevant Program from the drop-down list. Select Original to stream all the programs in the stream.
Filter	Within the selected program you can stream specific PIDs by creating a Filter (see: Adding Filters to Input Streams (PID Filtering) section) and then selecting it from the Filter drop-down list. Select Original to stream all PIDs within the program.
Properties	
Group	If your deployment includes a DNS Load Balancer, it is possible to group multiple outputs into a single Output Group. All the outputs created in Zixi Feeder within the same group and with the same host name, will be resolved into the same IP address. You can create a new output group or use an existing group. See OUTPUT GROUPS .
URL	Specify the URL of the target.

Output Groups

For deployments that include a DNS Load Balancer, it is possible to group multiple outputs into a single *Output Group*. All the outputs created in Zixi Feeder within the same group and with the same host name, will be resolved into the same IP address (i.e. the same Zixi Broadcaster). This feature can be used to ensure all adaptive streams will be delivered to the same machine (i.e. resolved to the same destination IP address).

Creating a New Output Group

The definition of the output group is part of the **New Output** process. Once you have created the output group, you can select it from the **Group** field in the **New Output** window.


➔ To Create a new Output Group:

1. In the **Outputs** page of Zixi Feeder, click the **+ New Output** button.

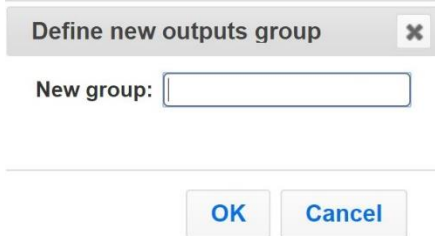
Creating Outputs in Feeder

Network Bonding

The **New output** window is displayed.

2. In the **Group** field click the  button.

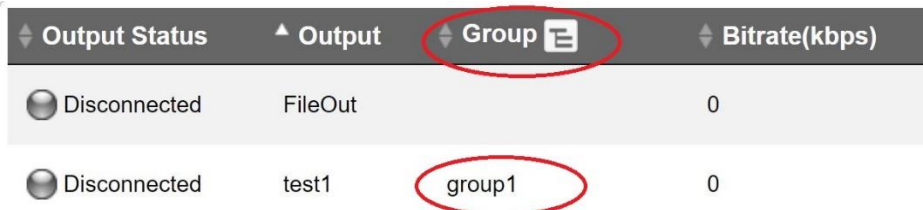
The **Define new output group** dialog opens:



The dialog box titled "Define new outputs group" has a close button (X) in the top right corner. Below the title is a text input field labeled "New group:" which is currently empty. At the bottom of the dialog are two buttons: "OK" and "Cancel".

3. In the **New Group** field, type a name for the new group.
4. Click **OK**.

The new output will appear in the Outputs list with the newly created group name:



Output Status	Output	Group	Bitrate(kbps)
Disconnected	FileOut		0
Disconnected	test1	group1	0

Additional outputs that need to be resolved into the same IP address by the DNS Load Balancer, should use the same group name (see [Using an Existing Output Group](#)).

Using an Existing Output Group

If you have already created an Output Group and would like to add another output to this output group, follow the instructions below:

➔ To Add an Output to an Existing Output Group:

1. In the **Outputs** page of Zixi Feeder, click the **+ New Output** button.
The **New output** window is displayed.
2. In the **Group** field, select the relevant group from the drop-down menu.



The "Properties" dialog box has three fields: "Group:", "Stream ID:", and "Password:". The "Group:" field is a dropdown menu with "No group" selected and a plus button to its right. The "Stream ID:" field is a text input with "group1" entered. The "Password:" field is an empty text input.

3. Complete the Properties fields, while ensuring the Host is the same as the other output in the group.
4. Click **OK**.

Network Bonding

Zixi's Network Bonding feature enables you to divide a stream into several network channels by utilizing multiple NICs and links and then subsequently reuniting them at the Zixi Broadcaster

Creating Outputs in Feeder

Network Bonding

side. For example, a 5 Mbps stream can be streamed using two links, one 2 Mbps link that uses a mobile network and another 3 Mbps Wifi network. The stream is automatically bonded at the Zixi Broadcaster side without any special configuration (as long as the Push input is defined and the two NICs are setup in the machine running the Zixi Broadcaster).

Network Bonding can be used for load balancing between multiple links in the following configurations:

- **Zixi Feeder with Multiple NICs to a single Zixi Broadcaster NIC** - the stream is divided at the Zixi Feeder end and then bonded at the Zixi Broadcaster end. This configuration requires configuring Network Bonding in the Zixi Feeder (see [Feeder Output with Network Bonding](#))
- **Zixi Feeder with Multiple NICs to multiple Zixi Broadcaster NICs** - the stream is divided at the Zixi Feeder end, sent to multiple IP addresses (each Zixi Broadcaster NIC has a separate IP address), and then bonded at the Zixi Broadcaster end. This configuration requires configuring Network Bonding in the Zixi Feeder. Each destination (i.e. each Broadcaster NIC is configured in a separate network bonding link by using a different Host IP address). (see [Feeder Output with Network Bonding](#))
- **Zixi Feeder with single NIC to Multiple Zixi Broadcaster NICs** - the stream is sent to multiple IP addresses (each Zixi Broadcaster NIC has a separate IP address), and then bonded at the Zixi Broadcaster end. This configuration requires configuring Network Bonding in the Zixi Feeder. Each destination (i.e. each Broadcaster NIC is configured in a separate network bonding link by using a different Host IP address). (see [Feeder Output with Network Bonding](#))
- **Zixi Broadcaster with multiple NICs to another Zixi Broadcaster with a single/multiple NICs** - a Zixi Broadcaster can transfer a stream to another Zixi Broadcaster through multiple NICs at the source Broadcaster and single/multiple NICs at the destination Broadcaster. This configuration requires configuring Network Bonding in the Zixi Broadcaster (see Network Bonding in the Zixi Broadcaster User Guide).

Using Network Bonding for Hitless Failover

The Network Bonding feature can be used to ensure high availability of the stream. The feature ensures hitless failover between multiple networks and network cards (i.e. network and hardware protection). As part of the configuration of the network bonding feature it is possible to set a NIC (which may be connected to a separate network) as a "Backup". In this case, the backup link will only be utilized if the stream has used all other available capacity. For example, a stream can be configured to be transferred on a Wi-Fi network as the main link and a mobile network as a backup. If the Wi-Fi link is compromised (either completely unavailable or limited bandwidth), the stream will be transferred, in full or partially through the backup mobile link and then bonded at the Zixi Broadcaster end, enabling uninterrupted streaming and high availability. Network Bonding is available in **Push to Broadcaster** outputs.

Feeder Output with Network Bonding

➔ To Add Network Bonding to a Feeder Output:

In the **Outputs** page of Zixi Feeder, click **+ New Output** in the menu bar.

The **New Output** window is displayed:

New output

Name: Type: **Push to Broadcaster** ▼

Input

Input name: **bbb - udp: 4554@Localhost** ▼

Program: **Original** ▼ Filter: **Original** ▼

Properties

Group: **No group** ▼ +

Stream ID:

Password:

Max bitrate (kbps):

Max latency (ms):

Ignore TLS certificate:

Host	Port	IP	
<input type="text"/>	<input type="text" value="2088"/>	Any ▼	+ -

Bond Links:

Show all settings:

Ok **Cancel**

5. Enter the unique **Name** that you want to configure on the Feeder.

6. Select **Push to Broadcaster** as the Type of output.

7. In the **Properties** section, enter the parameters as described in the

Creating Outputs in Feeder

Network Bonding

7-9. [Creating a Push to Broadcaster Output](#) procedure.

8-10. Click the **Bonded Link** checkbox.

The bonded link section is displayed:

New output

Name: Type: Push to Broadcaster ▾

Input

Input name: Stream 1 - File: C:\Users\lepielet\Downloads\ChubChubs.ts ▾

Program: Original ▾ Filter: Original ▾

Properties

Group: No group ▾ +

Stream ID:

Password:

Max bitrate (kbps):

Max latency (ms):

Ignore TLS certificate:

Host	Port	IP	Max. Bitrate [kbps]	Backup	+
<input type="text"/>	<input type="text" value="2088"/>	ALL ▾	<input type="text" value="Unlimited"/>	<input type="checkbox"/>	<input type="checkbox"/>

Bond Links:

Show all settings:

Ok Cancel

9-11. In the boded link section, click the + button.

A new row of destination fields is displayed:

Host	Port	IP	Max. Bitrate [kbps]	Backup	+
<input type="text"/>	<input type="text" value="2088"/>	ALL ▾	<input type="text" value="Unlimited"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="text"/>	<input type="text" value="2088"/>	ALL ▾	<input type="text" value="Unlimited"/>	<input type="checkbox"/>	<input type="checkbox"/>

Bond Links:

10-12. In the **Host** field, type the IP address of the destination Zixi Broadcaster.

11-13. In the **Port** field, type the destination port number of the Zixi Broadcaster to which you want to stream. Default: 2088

12-14. In the **IP** field, do one of the following:

- If you want to set a specific IP address, select the IP address of the Zixi Feeder that you want to use
- If you want to use all available IPs, select **ALL**.

13-15. In the **Bitrate Limit** field, do one of the following:

- If you want to limit the bitrate that will be used in this link, enter the limit value in Kbps.
- If you DO NOT want to limit the bitrate, leave the default value - *Unlimited*.

14-16. In the **Backup** field, do one of the following:

- If you want to use this link only if there is not enough bandwidth in the other links, select the **Backup** checkbox.

Creating Outputs in Feeder

Encrypting an Output Stream in Zixi Feeder

- If you want to use this link in parallel to other links, DO NOT select the **Backup** checkbox.

~~15.17.~~ Click **OK**.

~~16.18.~~ Repeat the steps above to add additional links.

Encrypting an Output Stream in Zixi Feeder

Zixi Feeder can encrypt outputs using a fixed key (AES 128, AES 192, AES 256) which can be entered into the output settings or automatically generated. In addition, selection of Automatic mode will generate the encryption and exchange it securely between the applications automatically.

➔ To Encrypt an Output Stream in Zixi Feeder:

1. In **Outputs** screen, click **+ New Output**.
2. In the **Type** field select **Push to Broadcaster**.
3. Fill in the output details.
4. At the bottom, select the **Show all Settings** checkbox.
5. Select the **Advanced Properties** checkbox.
6. In the **Encryption Type** field, select AES 128, AES 192, AES 256 for fixed key or Automatic for automatically system-generated encryption.
7. If you have selected one of the fixed key encryptions, enter your own encryption key into the **Key** field or click **Generate** to generate a key by the system. The exact key should be used to decrypt the stream on the other end.

Creating Outputs in Feeder


Searching for Outputs in Feeder

The screenshot shows the 'New output' configuration window. The 'Advanced Properties' section is highlighted with a red box, showing the 'Encryption type' set to 'AES 128' and the 'Key' field containing the hexadecimal string '07bc1d0f5fb9c88 12b695497bef9af a4'. Other settings include 'PCR smoothing delay (ms)' at 0, 'DSCP' at 0, and 'ECN' set to 'Non-ECT'. The 'Input' section shows 'Input name' as 'bbb - udp: 4554@Localhost' and 'Program' as 'Original'. The 'Properties' section shows 'Max bitrate (kbps)' as 8000 and 'Max latency (ms)' as 6000. The 'Name' field is empty, and the 'Type' is 'Push to Broadcaster'. The 'Ok' and 'Cancel' buttons are at the bottom right.

8. Click **OK**.

Searching for Outputs in Feeder

➔ To Search for an Output:

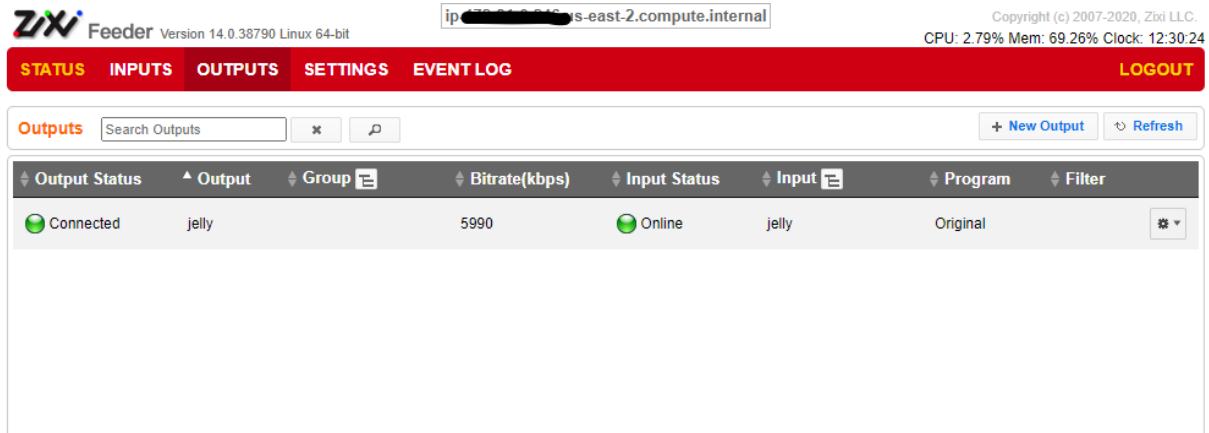
- In the **Outputs** page of Zixi Feeder, enter the specific output name in the search text box in the menu bar and click the **Search**  button. The **Output** screen displays the requested output stream(s).


Viewing Existing Outputs in Feeder

The **Outputs** screen displays all the output streams that have been created. At the bottom of the screen you can also view cumulative utilization statistics for the selected connected streams. Hovering the cursor on the status indicator of a stream shows the stream configuration information and basic utilization statistics, where applicable.

Creating Outputs in Feeder

Viewing Existing Outputs in Feeder



The outputs can be grouped according to Output Groups or according to Inputs by clicking on the  icon next to the Group or Input headings.

To ungroup the outputs, click the  icon, located in the toolbar.

Any of the columns in the following table can be sorted in ascending/descending order by clicking on its heading.

Field	Description
Output Status	Displays the status of output stream. The following colored indicators are displayed: <ul style="list-style-type: none">• Green = Connected• Yellow = Connecting• Red = Disconnected with an error condition• Blue = Offline with no connection attempts• Grey = Offline and stopped
Output	Displays the name of the output stream.
Group	Displays the name of the output group with which the output is associated.
Bitrate [kbps]	Displays bitrate of the connected output stream.
Input Status	Displays the status of input stream. The following colored indicators are displayed: <ul style="list-style-type: none">• Green = Online. The input is connected and streaming.• Yellow = Connecting• Red = Offline. The input is not connected.• Blue = Offline with no connection attempts• Grey = Offline and stopped

Creating Outputs in Feeder

Viewing Output Statistics in Feeder

Input	Displays the name of the related input stream.
Program	Displays the program name in the input stream.
Filter	Displays the name of the filter that was selected.
Actions	Provides access to various actions to be performed on this output stream (see PERFORMING ACTIONS ON OUTPUTS IN FEEDER below).

Viewing Output Statistics in Feeder

You can view all of the cumulative statistics for a connected output in real-time.

➔ To View Output Statistics:

- In the **Outputs** page of Zixi Feeder, click the desired output for which you want to view statistics.
The stream's statistics are displayed on the bottom of the screen.

Field	Description
In Bitrate (kbps)	Displays the current bitrate of the input stream.
Out Bitrate (kbps)	Displays the current bitrate for the output.
Up Time	Displays the amount of time the stream has been active.
RTT (ms)	Displays the Round Trip Time (RTT) from the source (in milliseconds)
Jitter (ms)	Displays the current Jitter, measured in milliseconds.
Total Packets	Displays the total number of packets that have been transmitted.
Packet Rate	Displays the current number of packets per second.
Dropped Packets	Displays the total number of packets that were dropped since the beginning of the stream.
Recovered Packets	Displays the total number of dropped packets that have been recovered since the beginning of the stream.
Not Recovered Packets	Displays the number of non-recovered packets.
Failed Sends	Displays the number of failed attempts to send the stream.
FEC Packets	Displays the number of FEC packets transmitted.
FEC Recovered	Displays the number of FEC packets recovered.

Creating Outputs in Feeder

Configuring Encoder Feedback Script

ARQ Requests	Displays the number of requests for transmission of dropped packets made with ARQ.
ARQ Recovered	Displays the number of dropped packets recovered via ARQ.
ARQ Duplicates	Displays the number of duplicate recovery packets received via ARQ.
Overflows	Displays the number of packets lost due to buffer overflow. Note: overflow is usually the result of the Max Bitrate being too low.

Configuring Encoder Feedback Script

Zixi Feeder features an option to provide feedback to the encoder regarding the current available bitrate according to Zixi's network condition analysis. This means that based on its network condition analysis (on the path between Zixi Feeder and the Zixi Broadcaster), Zixi Feeder may recommend lowering or raising the stream's bitrate. This feedback is sent back to the encoder, and the encoder will thereby adjust its bitrate.

To implement this feature, you will need to create a special script and place it in the designated folder location of the Zixi Feeder installation, as described below.

➔ To Configure the Encoder Feedback Script:

1. Create a bash/shell script with the following parameters:

```
<bitrate_bps> <iframe> <name> <encoder_address>
```

Parameter	Description
bitrate_bps	Total desired bitrate. This includes the audio + video + overhead.
iframe	Specifies if an i-frame should be introduced or not. Use when GOPs are large (multiple seconds).
name	The name of the input.
encoder_adderss	Optional parameter with the encoder address


2. Place the script in the following location and using the following filename:
 - For Linux: /usr/local/bin/feedback.sh
 - For Windows: <Feeder program installation folder>\feedback.bat

Performing Actions on Outputs in Feeder

From the Outputs screen you have a centralized control on the output streams to perform the following actions:

Stopping/Starting Outputs in Feeder


➔ To Stop/Start an Output:

1. In the Zixi Feeder **Outputs** page, click the **Actions**  button for the output that you want to start or stop.
2. In the drop-down menu, click **Start/Stop** (depending on the current status). The output in that row is either started or stopped.
To continue the stream, click again.

Editing Output Settings in Feeder


You can edit an output which has been previously configured.

➔ To Edit Output Settings:

1. In the Zixi Feeder **Outputs** page, click the **Actions**  button for the output stream that you want to edit.
2. In the drop-down menu, click **Edit**.
The stream details window of the selected output is displayed.
3. Make the necessary modifications and click **OK**.
The newly modified output takes effect.

Deleting Outputs from Feeder

➔ To Delete Outputs from Feeder:

1. In the Zixi Feeder **Outputs** page, click the **Actions**  button for the output that you want to delete.
2. In the drop-down menu, click **Delete**.
The output in that row is deleted.


Duplicating Outputs in Feeder

In the event that you want to create outputs with similar configurations, you can duplicate the configuration and make minor modifications.

Performing Actions on Outputs in Feeder

Duplicating Outputs in Feeder

➔ To Duplicate the Configuration of an Output:

1. In the Zixi Feeder **Outputs** page, click the menu button  for the output that you want to duplicate.
2. In the drop-down menu, click **Duplicate**.
3. The output details window opens with similar parameters to the original stream.
4. Type the new Output Name, and make any additional modifications that you want.
5. Click **OK**.
The duplicate output stream with its new settings is created.

Feeder Settings

Zixi Feeder is pre-configured with default settings, so generally there is no need to change the server settings. However, some users may want to change specific settings to meet their specific needs.

The server settings enable the configuration of settings that can be used to implement various use cases or applications such as broadcast delivery, monitoring, OTT delivery, enterprise distribution and more.

Feeder General Settings

The General Settings screen includes basic configuration parameters for the server including the server's ID, access ports, authorizations, security, and storage.



If you make changes to the settings, you must click **Apply** to save the changes before navigating away from the page.

The Parameters shown in the General Settings are described in the table below:

Field	Description
ID	The unique (alphanumeric) name of the Zixi Feeder server. Typically: host name.
Web Server Port	The port number of the web-based administration. You can configure this parameter to any legal port number which is supported by your Firewall. Default: 4200
NIC	The server's Network Card ID. Default: Any
Password	The password of the administrator.

Feeder Settings

Feeder ZEN Master Settings

Root Folder	Displays the configured root folder. You can modify the root folder in the demuxer-private-settings.xml configuration file, which is located in the Feeder installation folder.
Enable HTTPS	Enables secure HTTPS connection to the server. Enabling HTTPS requires the relevant certificates and private keys.
HTTPS certificate uploaded	Indicates if a certificate has been uploaded. Click Upload to upload the HTTPS certificate.
HTTPS private key uploaded	Indicates if the private key has been uploaded. Click Upload to upload the private key.
HTTPS private key passphrase	Enter the private key passphrase.
TR101 analyzer mode	Monitoring options: <ul style="list-style-type: none">• None – no monitoring• ATSC• DVB

Feeder ZEN Master Settings

Once you have configured a Feeder in ZEN Master, you can automatically connect it through the ZEN Master tab in the Feeder UI.

➔ To connect a Feeder to ZEN Master automatically:

1. Go to the **ZEN Master** tab in the main navigation.
2. The ZEN Master login fields are displayed.

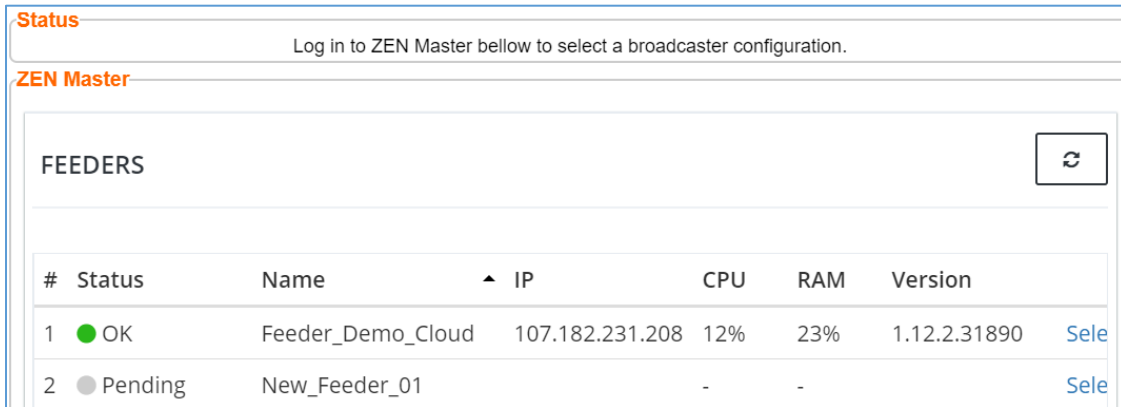
3. In the **Account ID** field, enter the customer domain of your ZEN Master account (e.g. if you login to ZEN at *demo.zen.zixi.com* then your Account ID is 'demo').
4. In the **Email** field, enter the username for your ZEN Master account.

Feeder Settings

Feeder SSH Tunnels

5. In the **Password** field, enter the password for your ZEN Master account.
6. Click **Login**.

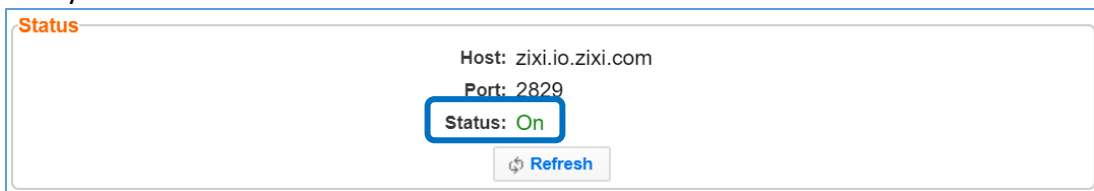
A list of Feeders that are configured in ZEN Master is displayed.



The screenshot shows the ZEN Master interface. At the top, there is a 'Status' section with the text 'Log in to ZEN Master bellow to select a broadcaster configuration.' Below this is the 'ZEN Master' section, which contains a 'FEEDERS' table. The table has columns for '#', 'Status', 'Name', 'IP', 'CPU', 'RAM', and 'Version'. There are two rows of data: one for 'Feeder_Demo_Cloud' with status 'OK' and one for 'New_Feeder_01' with status 'Pending'. A 'Refresh' button is located in the top right corner of the table area.

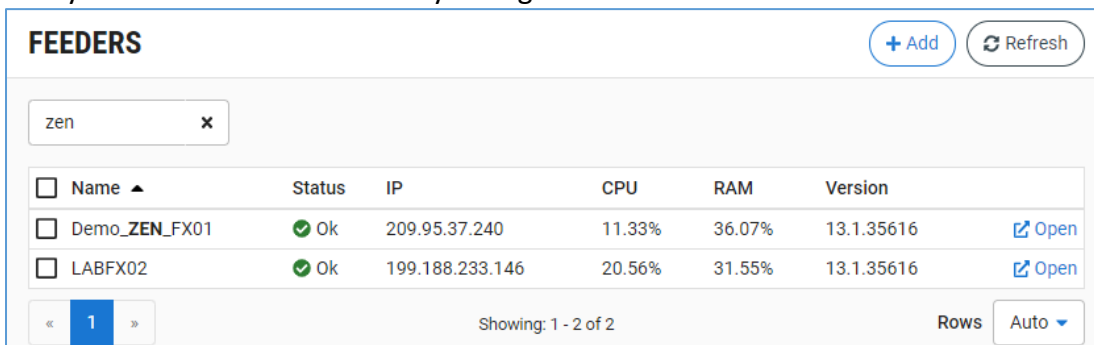
#	Status	Name	IP	CPU	RAM	Version
1	OK	Feeder_Demo_Cloud	107.182.231.208	12%	23%	1.12.2.31890
2	Pending	New_Feeder_01		-	-	

7. For the Feeder that you are adding, which should be in Pending mode, click **Select**. Details of the ZEN Master connection are shown in the **Status** section.
8. Click **Refresh**. The connection is configured automatically.
9. Verify that the Status is shown as **On**.



The screenshot shows the 'Status' section of the ZEN Master interface. It displays the following information: Host: zixi.io.zixi.com, Port: 2829, and Status: On. A 'Refresh' button is located below the status information.

10. In the **ZEN Master UI**, in the main navigation, click **Feeders**.
11. Verify that the status of the newly configured Feeder is **OK**.



The screenshot shows the 'FEEDERS' table in the ZEN Master UI. The table has columns for 'Name', 'Status', 'IP', 'CPU', 'RAM', and 'Version'. There are two rows of data: one for 'Demo_ZEN_FX01' with status 'Ok' and one for 'LABFX02' with status 'Ok'. A search bar at the top left contains the text 'zen'. A '+ Add' button and a 'Refresh' button are located in the top right corner. The table is currently showing 1 of 2 rows.

Name	Status	IP	CPU	RAM	Version
Demo_ZEN_FX01	Ok	209.95.37.240	11.33%	36.07%	13.1.35616
LABFX02	Ok	199.188.233.146	20.56%	31.55%	13.1.35616

Feeder SSH Tunnels

SSH (Secure Shell) is a network protocol that allows a secure access over an encrypted connection. Reverse SSH is a technique that can be used to access systems (that are behind a firewall) from the outside world.

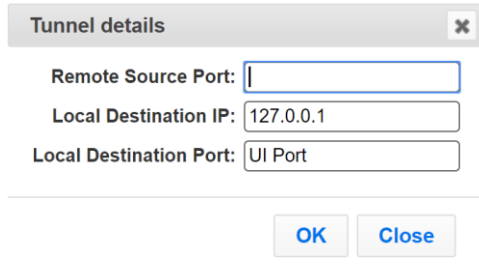
Feeder Settings

Feeder SSH Tunnels

➔ To add a Reverse SSH Tunnel connection:

1. Fill in the details in the Server Connection section (see table below) to configure a connection to the server.
2. Click **Apply**.
3. In the Reverse Tunnels section, click **Add**.

The Tunnel Details dialog appears:



Tunnel details ✕

Remote Source Port:

Local Destination IP:

Local Destination Port:

4. Type the **Remote Source Port** that will be used to connect.
5. In the **Local Destination IP** field, type the IP address of the Zixi Feeder.
6. In the **Local Destination Port**, type the port number in the Zixi Feeder for the specific reverse SSH tunnel.
7. Click **OK**.

The reverse SSH tunnel will be added to the list of reverse tunnels at the bottom.

Feeder Settings

RTMP Server

Field	Description
<i>Server connection</i>	
Host	Specify the Host name or IP address of the remote server for the SSH connection.
SSH Port	Specify the port for the SSH connection. Default:22
Username	Specify the username required for SSH connection. Default: the username on the remote computer is the same as that on the local one.
Key File	Indicates if a key file has been uploaded. Click Upload Key to upload a key file to the server.
<i>Reverse tunnels</i>	
Remote Source Port	Displays the remote source port.
Local Destination IP	Displays the IP address of the local destination.
Local Destination Port	Displays the port number of the local destination.
Status	Displays the status of the SSH tunnel.

RTMP Server

This screen enables the configuration of RTMP input streams. If you wish to add RTMP input streams, you must first enable and configure RTMP in this screen and then click **Apply**.

Field	Description
Enable RTMP Server	Select this checkbox to allow RTMP input streams.
RTMP Server Port	Specify the port to listen to the RTMP input streams.

Feeder Logging

The Logging screen defines a log for debugging purposes.

Field	Description
<i>Logging</i>	

Feeder Settings

Export/Import Settings

Logs Path	Specify the log file name and path where the collected event information will be stored.
Log Level	The Severity Level that will be collected in the log, from Info , Warning , Error , to Fatal . The default setting is None , since Logging is resource-intensive and only enabled when necessary.
Send Crash Reports	Select this checkbox to automatically send an email to Zixi Support when the Feeder crashes containing the call stack of the event. This simplifies the support process and will eliminate the need to connect remotely to the system or send out log reports. The report is anonymous and does not include any sensitive data (e.g. configuration data, etc.).
Events	Select this checkbox to capture events in the log.

Export/Import Settings

You can export your current settings configuration as an XML document and then re-apply those settings later by uploading the file.

➔ To Export Settings to a File:

1. In the **Settings** page, click **Export** in the menu bar.
2. Select the destination location.
3. Click **Save**.
Your settings are saved as an XML document.

➔ To Import a Settings File:

1. On the **Settings** page, click **Import** in the menu bar.
A prompt window opens.
2. Click **Upload configuration file**.
3. Navigate to the desired file and click **Open**.
4. In the prompt windows, click **OK**.
The program will restart with the new settings applied.

Feeder Network Settings

The Network Settings screen includes configuration parameters for the network interfaces in the physical server. There are slight differences in the settings between Linux and Window systems.

Field	Description
-eth0/-eth1/-eth2/-eth3	NIC interface configurations. To configure the interface, click the Interface name to open additional settings.

Feeder Settings

Feeder Network Settings

DNS servers	
Use DHCP DNS servers	Select this checkbox to enable DHCP. You can also configure the IP addresses manually.
DNS1	Specifies the address of the DNS1 server. Available only if the Use DHCP DNS server option is not selected.
DNS2	Specifies the address of the secondary DNS2 server. Available only if the Use DHCP DNS server option is not selected.

➔ To Display the Network Statistics:

1. From the **Settings** page of the Zixi Feeder, click **Network** on the left hand side. The various Network settings are displayed.
2. Click [Statistics](#) on the right-hand side. The UDP statistics window is displayed.

Field	Description
####Packets received	Number of packets on network received.
####packets to unknown port received	Number of packets to unknown port on network received.
####packet receive errors	Number of packets on network which received errors.
####packets sent	Number of packets on network sent.
####SndbufErrors	Number of Sent buffer errors.

3. **Refresh** button will update the data, while **Close** will close the window.

Viewing Feeder Events Log

The Event Log screen displays all the events (errors, connections, configuration changes, analysis events etc.) that occurred in the Zixi Feeder. The logs can be exported as a .csv file.

➔ To Clear the Log:

- In the **Event Log** page, click **Clear** in the menu bar.
The events on the log screen are deleted.

➔ To Save the Log as a File:

- In the **Event Log** page, click **Save** in the menu bar.
A .csv file is created and saved to the local destination.

➔ To Refresh the Screen:

- From the **Event Log** page, click **Refresh** in the menu bar.
The data on the screen is updated.

Viewing Feeder Events Log

Feeder Network Settings

- License for OpenSSL

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit

(<http://www.openssl.org/>)

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com). This product includes software written by Tim

Hudson (tjh@cryptsoft.com).

Copyright (c) 1998-2017 The OpenSSL Project. All rights reserved.

- License for Libssh2

Copyright (c) 2004-2007 Sara Golemon <sarag@libssh2.org>

Copyright (c) 2005,2006 Mikhail Gusarov <dottedmag@dottedmag.net>

Copyright (c) 2006-2007 The Written Word, Inc.

Copyright (c) 2007 Eli Fant <elifantu@mail.ru>

Copyright (c) 2009-2014 Daniel Stenberg

Copyright (c) 2008, 2009 Simon Josefsson

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the copyright holder nor the names of any other contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

- License for DekTec's Linux SDK. Version 1, August 2012

Copyright (C) 2000-2015 DekTec Digital Video B.V.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary format must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation.

THIS SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL DEKTEC DIGITAL VIDEO BV, ITS AGENTS OR ITS EMPLOYEES BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, INCIDENTAL, OR OTHER DAMAGES (INCLUDING DAMAGES FOR THE LOSS OF USE, INFORMATION, GOODWILL, PROFIT, WORK STOPPAGE, DATA, BUSINESS OR REVENUE) UNDER ANY CIRCUMSTANCES, OR UNDER ANY LEGAL THEORY, WHETHER IN CONTRACT, IN TORT, IN NEGLIGENCE, OR OTHERWISE, ARISING FROM THE USE OF, OR INABILITY TO USE THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.