



Intelligent AV Distribution

Stream Deck and ZMP

How to Guide for MAC

Updated March 2020



Table of Contents

Introduction	3
ZyPer System Description	3
Management Platform	4
MAC Instructions	5
Installing Python	5
MAC Automator	5
Example Python Script for Channel Up command	7
Testing Python script in Automator	9
Running the application from Stream Deck	9

Introduction

This document provides detailed information on how to configure a Stream Deck product to issue commands to the ZyPer Management Platform API. MAC installation is described.

Note: Stream Deck is not a ZeeVee product and that these instructions are provided as-is. Limited additional support is available from ZeeVee.

ZyPer System Description

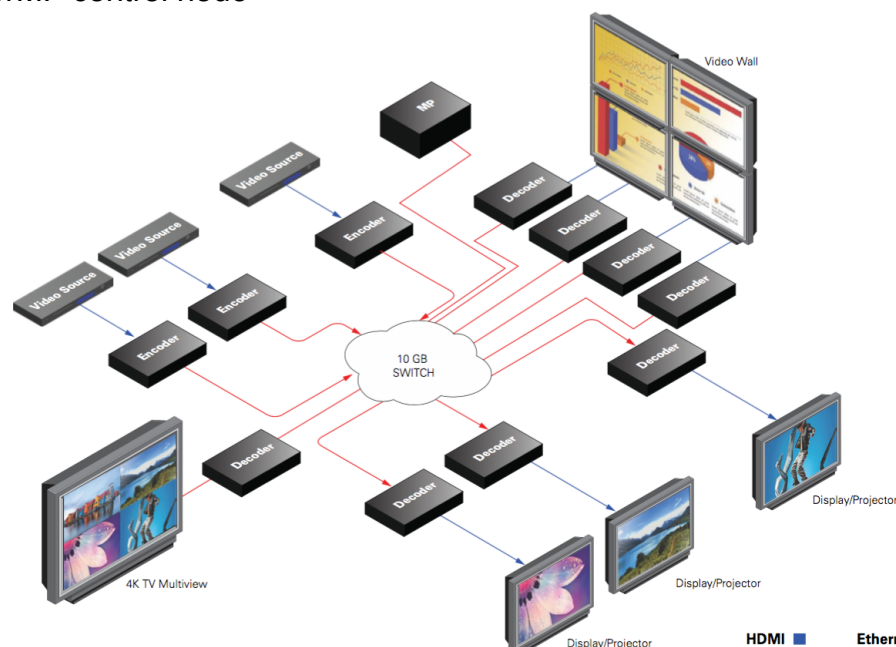
The ZyPer series of products from ZeeVee provide a means for transmitting and receiving up to 4K audio-visual information across a standard IP network with outstanding quality at very low latencies.

A minimal system consists of:

- Video Source (Up to 4K)
- ZyPer Encoder device
- CatX Cable or Fiber cables with SFP modules
- ZyPer Decoder device
- Display device

A fully realized system consists of:

- (N) Video Sources (HDMI 2.0 support with HDR)
- (N) ZyPer Encoder devices
- Ethernet switching Infrastructure (> (N+M) ports of non-blocking capacity)
- (M) ZyPer Decoder devices
- (M) Display devices
- ZyPerMP control node



Management Platform



Installations that want to make use of the Stream Deck button panel will require a ZyPer Management Platform (ZyPerMP). This device controls the operation and manages the connectivity between endpoints. The management platform is a hardened controller node that must reside on the same logical network as the ZyPer Encoders and Decoders. It runs a ZeeVee application as its sole function. The application presents an API to potential third party management platforms as well as hosting its own ZMP management application.

The purpose of the Management Platform is to interface user operational requests into the specific device-level control needed to effect connectivity changes, screen management, interface management, and distribute software and control data to all the endpoints of the overall system. The ZyPerMP is the entity that ensures all the endpoints, are accounted for, operating properly, and performing the correct operation at the correct time.

Further, the ZyPerMP is the element that guarantees security, manages access to the overall system and prevents rogue listeners or observers from snooping or spoofing unwanted content.

Multiple ZyPerMP devices may be deployed in a redundant configuration to provide for fault tolerant control of the greater ZyPer overall system.

Operation of the ZyPerMP, the ZMP application, and the ZyPer-API are beyond the scope of this manual and are documented separately in far greater detail in the *ZyPer Management Platform User Manual*. Documentation can be downloaded from the ZeeVee website.

<https://www.zeevee.com/documentation-all-products>

MAC Instructions

Installing Python



In order to use Stream Deck on a MAC you will need to create short Python scripts. To do this you must have Python installed on the MAC. This is accomplished by first installing a tool called Homebrew. (Internet access required)

To [install Homebrew](#), open Terminal or your favorite OS X terminal emulator and run

```
$ ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

The script will explain what changes it will make and prompt you before the installation begins.

Now, we can install Python:

```
$ brew install python
```

MAC Automator

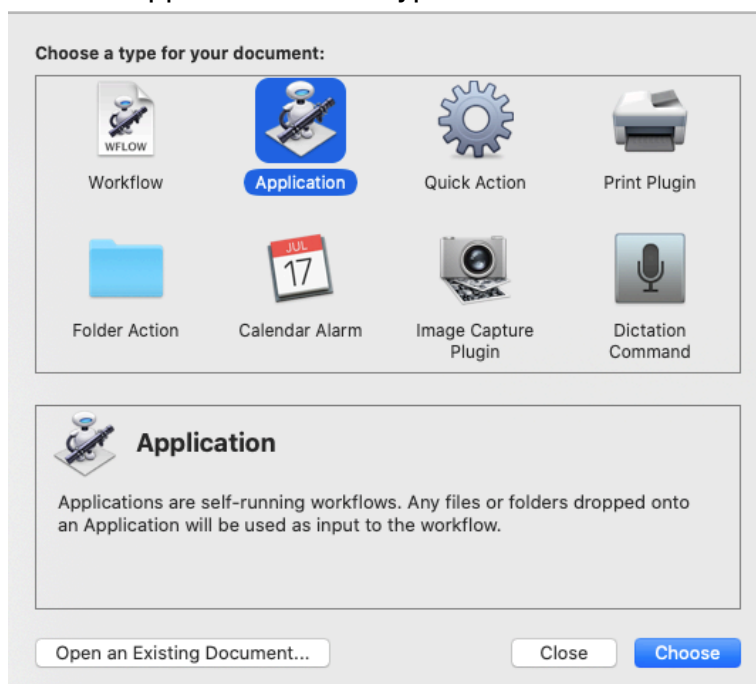


MAC Automator is a tool that comes pre-installed on the MAC. This tool will be used to generate a small python application file for the Stream Deck to call and execute.

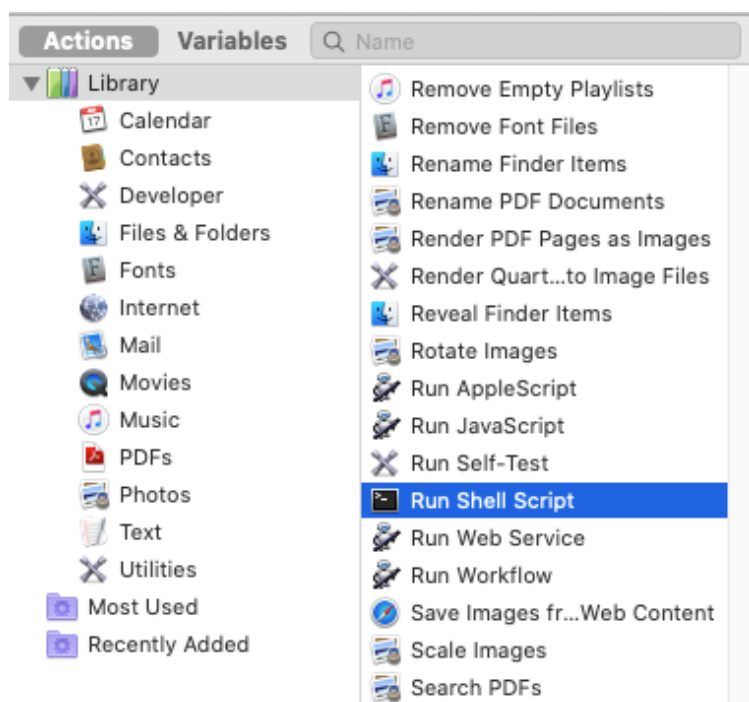
Steps to create a sample Automator application:

1. Open MAC Automator

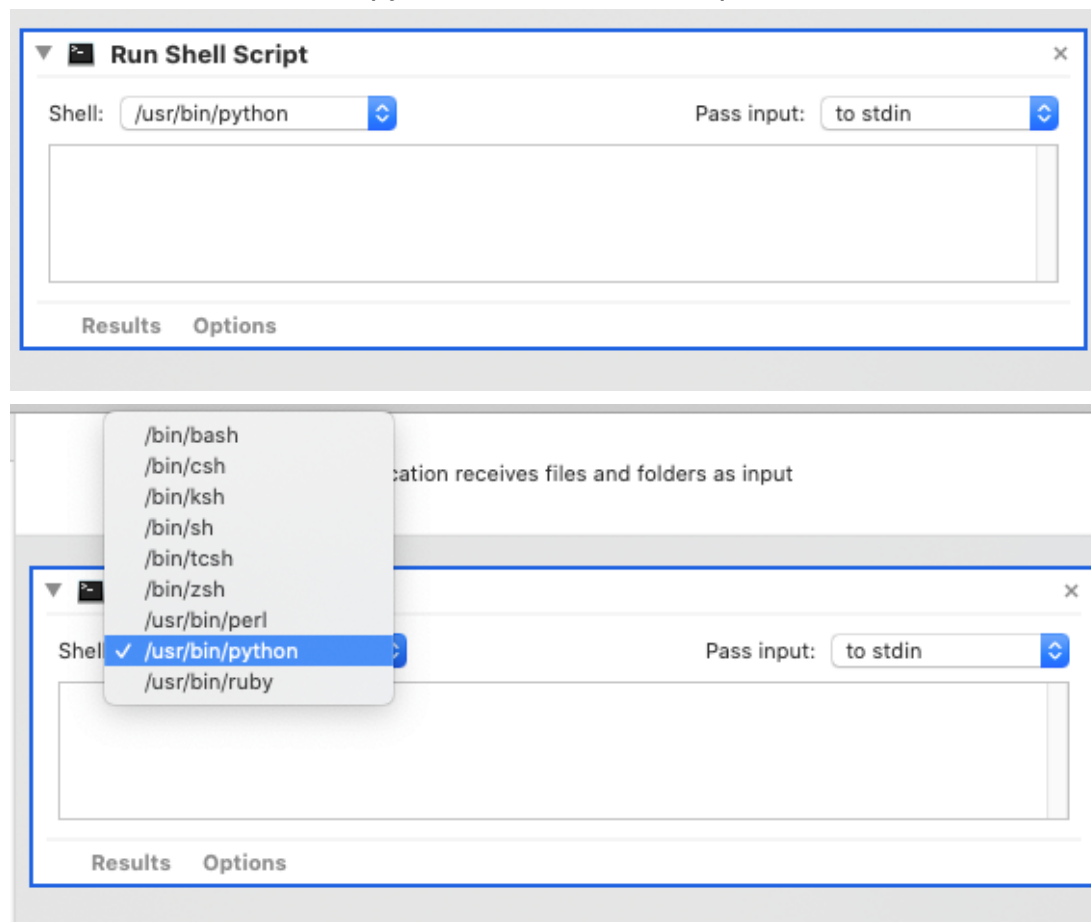
2. Choose Application as the type of document



3. Under Actions select Run Shell Script



4. Set the Shell to: “/usr/bin/python” Set Pass input to: “to stdin”



5. Enter the Python script in the large box. Below is an example Python script to execute a Channel Up command.

Example Python Script for Channel Up command

```
host = '192.168.0.78'      #IP Address of the ZMP in the system
```

```
port = 23                      #Port of the ZMP. Do not change
```

```
import sys,os,socket,time,logging,datetime,string,re,csv
```

```
def api_open_connection(host, port): # Opens a socket connection to a API server  
defined by "host" and "port"
```

```
    global clientsocket
```

```

clientsocket = -1;

logging.info('Opening Socket to API Server...')

clientsocket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

clientsocket.connect((host,port))

if clientsocket >= 0:

    logging.info('Socket opened: '+str(clientsocket))

else:

    logging.info('Error opening socket with host: '+host+' and port: '+str(port))

return clientsocket

```

```

def api_send_command(cmd): # Sends a text command (telnet) to the API Server

    logging.info('cmd: '+cmd)

    try:

        cmd = cmd + '\r\n'

        clientsocket.send(cmd)

        time.sleep(2)

        result = clientsocket.recv(buf)

        logging.info(result)

    except:

        logging.info('Socket Error sending command to API!')

    return True

```

```

def mode_channelup():

    global mode,video_rotation,audio_rotation

    clientsocket = api_open_connection(host, port)

```


if clientsocket:

```
api_send_command ('channel up Decoder_Name') #Decoder_Name you are using
```

```
    return
```

```
mode_channelup()
```

Testing Python script in Automator

Once the script is entered into Automator you can click on Workflow-Run to test the script. Be sure to enter the correct IP address for the ZyPer Management Platform into the script and specify a valid Decoder_Name.

Note: For the Channel Up command to work, all encoders must have a unique number after their name. Examples: Encoder_1, Encoder_2, Encoder_3

The Channel Up command will cycle the target Decoder thru the encoders until the highest number Encoder is reached. It will then cycle back to the lowest numbered Encoder and repeat the process.

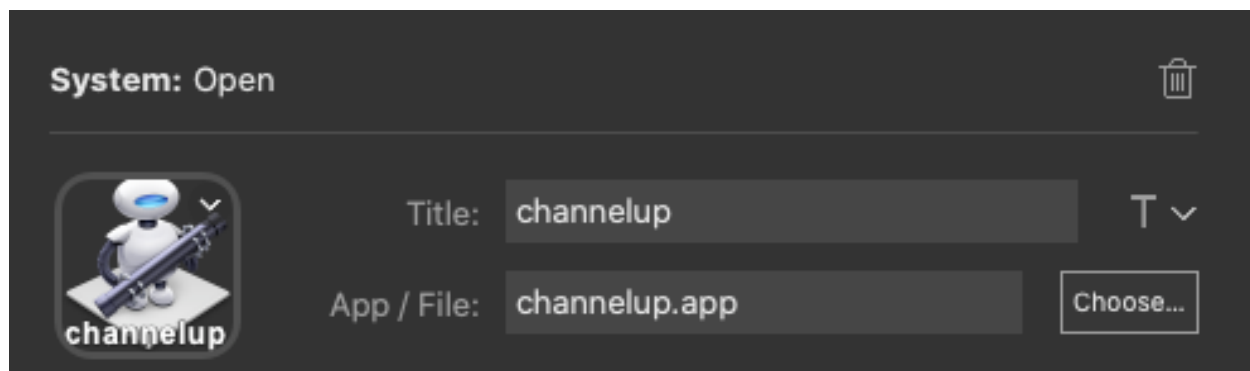
Once the script is tested and working; save it with a unique name. Example: Channel_Up.app

Running the application from Stream Deck

Assuming the Stream Deck unit is connected and the Stream Deck application is installed, open the Stream Deck application. You will want to drag and drop the appropriate action from the right side onto the desired Stream Deck button. In this case find the “Open” action located under the “System” option. (Looks like a small rocket)



Next to the App / File option select “Choose..” This is where you navigate to the Automator.app file you created above.



This will temporarily change the Icon from a Rocket to the Icon for Automator.

The Title and Icon can be changed to anything you like. Below is an example of the Stream Deck Mini button panel with Channel Up and Channel Down commands.

Clicking on the Stream Deck button itself will execute the command. Note there is about a 2 second delay before you can press the button again.

