

Intelligent Surveillance



About Me

- 9 Years experience developing on Linux based platforms.
- Prior to that, worked as a system admin in hardware and networking.
- Google Summer of Code 2015: Developed RootFS Build Factory for CentOS.
- Presently working with GStreamer for video monitoring & multimedia applications.
-



Surveillance

An extendable framework for CCTV applications.



Talk Contents

- Motivation behind this project. The Problem and a Solution.
- Architecture of a Surveillance based system.
- Description of each of the components in Surveillance.
 - ▶ Surveillance Camera Agent (Sca)
 - ▶ Surveillance Relay (SRelay)
 - ▶ Surveillance Viewer (SView)
- Processing the video stream. Using multiple plugins simultaneously.
- Sending processed information to clients on the network.
- Configuring cameras.
- Extending Surveillance, writing plugins.
- Future Scope & Conclusion.



The Problem:

- Just Surveillance: Monotonous, Tiring, Boring.
- Manual Monitoring Video Screens: A Herculean task. Failure to detect some intrusions.

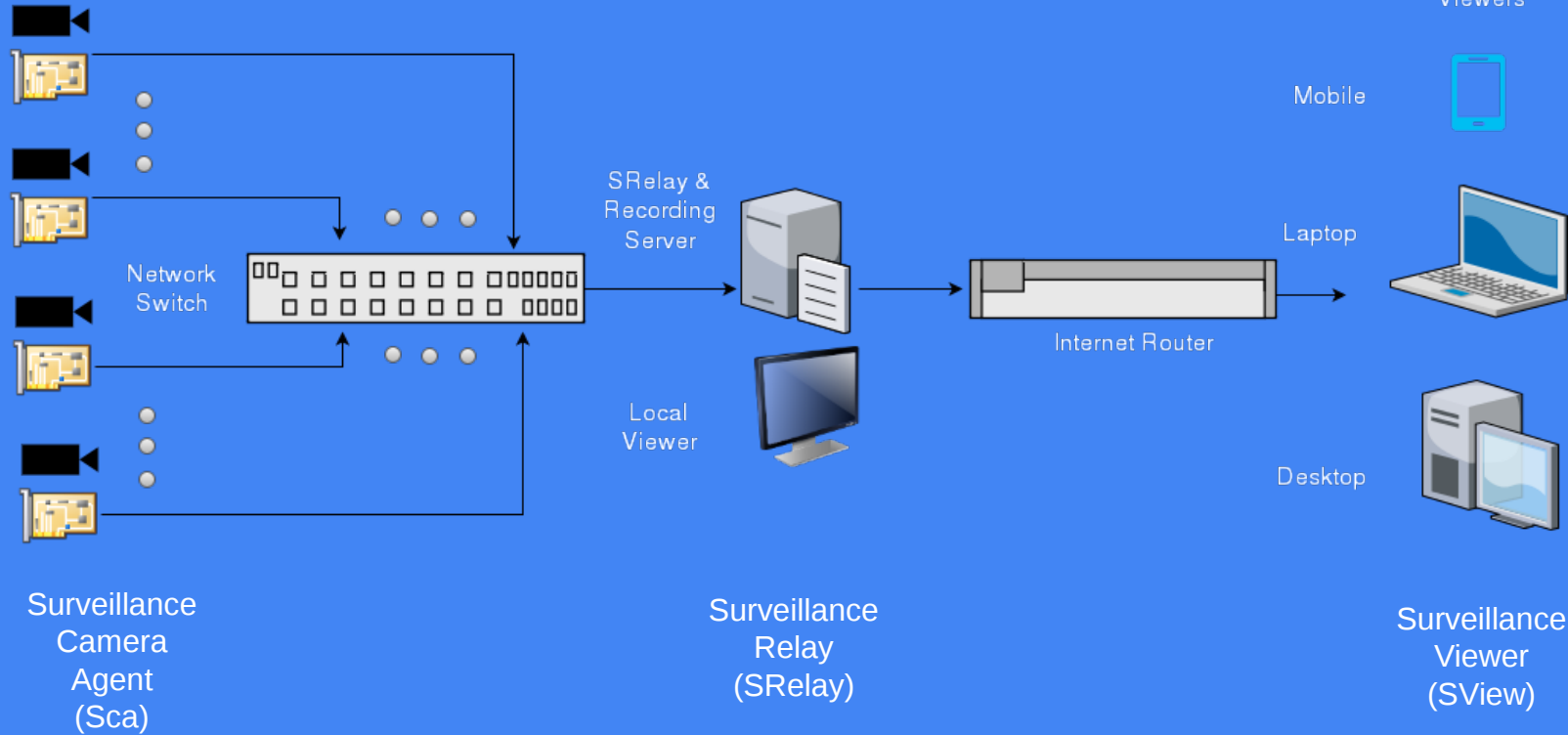
A Solution:

- Make cameras smarter. Get them to do the processing. Have them send processed text information (meta-data) along with the video.
- Covert '**dumb**' cameras to '**intelligent**' cameras.
- Process this information (meta-data), extract Region-Of-Interest (roi) from images. Do your analysis.
- Intelligent Surveillance: Process Video + Alert On Suspicious Event

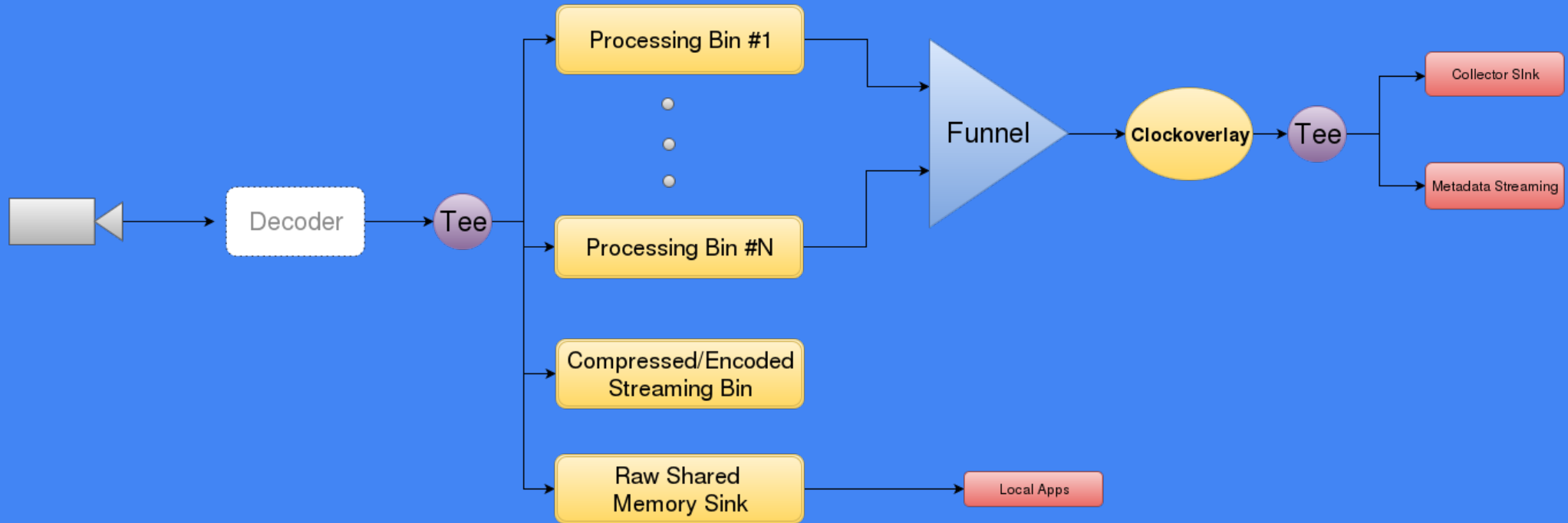


Hardware, Software & Network Structure

Camera Endpoints



Sca: Surveillance Camera Agent



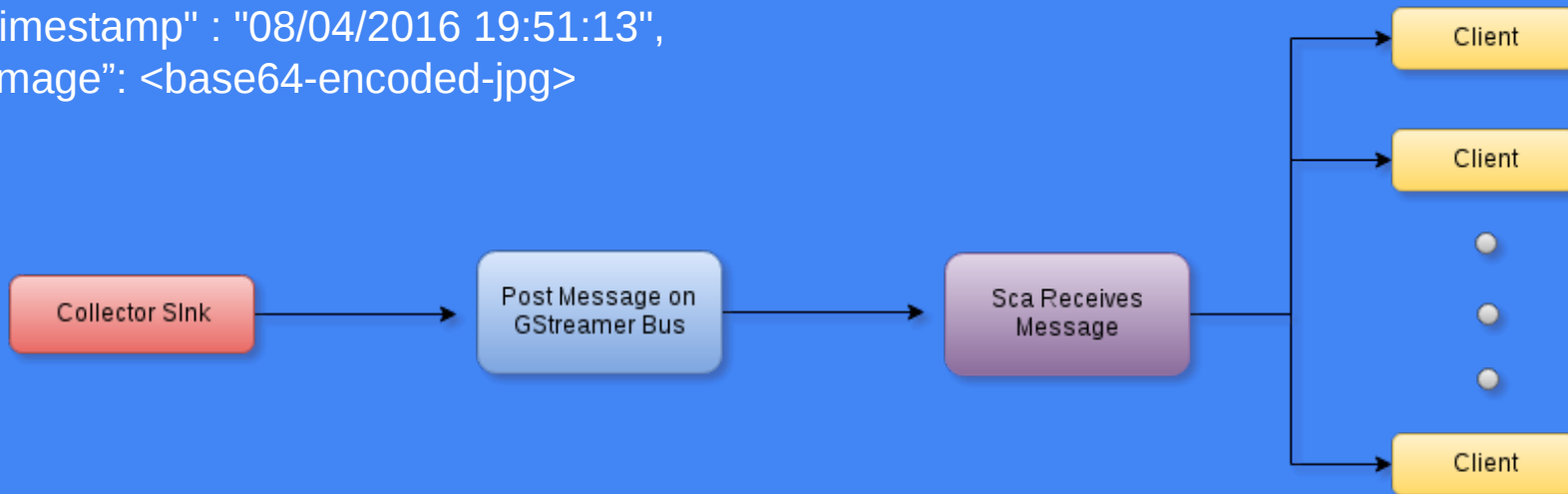
*Pipeline for a single camera on a camera endpoint.
One camera endpoint supports multiple cameras.*



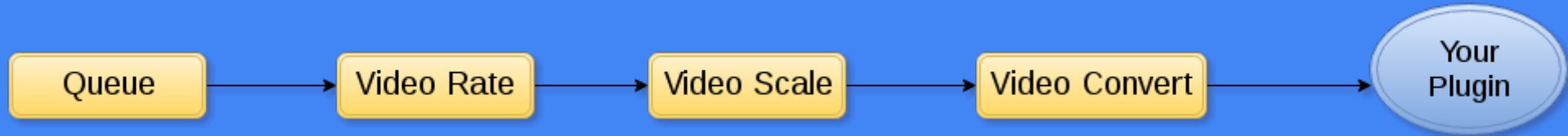
Sca: Collector Sink

Sample Meta-Data in JSON

```
{  
  "command" : "metadata",  
  "camera" : "logitech",  
  "plugin" : "motiondetect",  
  "description" : "Motion Detected",  
  "long_description" : "Some motion was detected in specified regions of interest",  
  "roi" : "0,0,160,120,1",  
  "timestamp" : "08/04/2016 19:51:13",  
  "image": <base64-encoded-jpg>  
}
```



Sca: Processing Bin



- Video Rate: Limit Frames Per Second
- Video Scale: Scale to lower resolution
- (The above have been Implemented keeping in mind low power CPUs)
- Video Convert: Convert raw video into a format you need for processing.
- Then comes your plugin. How to write one is explained later.



Camera Sample Configuration:

```
[camera]
Name=logitech1
Description=Logitech Camera on Desktop
Enabled=true
VideoSource=v4l2src device=/dev/video0 do-timestamp=true
#VideoSource=rtspsrc location=rtsp://192.168.1.202:8554/rpi latency=0
V4L2ExtraControls=c,power_line_frequency=1
VideoFormat=I420
#VideoFormat=MJPEG
FPS=30
StreamingEnabled=true
MetadataStreaming=true
Decoder=avdec_h264
#Encoder=x264enc tune=zerolatency pass=qual quantizer=27 subme=4 threads=2 bitrate=1024 key-int-max=15
#MetadataEncoder=x264enc tune=zerolatency pass=qual quantizer=27 subme=4 threads=2 bitrate=1024 key-int-max=15
Encoder=openh264enc multi-thread=8 enable-frame-skip=true gop-size=30
MetadataEncoder=openh264enc multi-thread=8 enable-frame-skip=true gop-size=1
Payloader=rtph264pay
Depay=rtph264depay
Parser=h264parse
StreamingResolution=640x480
ProcessingResolution=320x240
ProcessingElements=motiondetect;facecount
```

```
[motiondetect]
Name=motiondetect
Format=RGB
FPS=4
Properties=roi=0,0,160,120,1
```

```
[dummy]
Name=dummy
Format=RGB
FPS=4
Properties=
```

```
[facecount]
Name=facecount
```



SRelay: Surveillance Relay

- Provides a single point of access for all cameras on a network.
- Can discover cameras on the network using Multicast.
- Alternatively, camera hosts can be added manually.
- Provides recording for all cameras. Forwards meta-data and re-streams (without transcoding) to all clients.
- Has secure mode equivalent to the one in Sca.
- Has the ability to mix multiple camera streams to produce a single video stream. Uses the compositor element.
- Can be used to provide external access to government authorities & the police over the internet or via a leased line.



Noteworthy Surveillance Features:

1. Drone Mode:

- ▶ If your drone uses a 3G/4G connection, you are likely not going to get a static IP address. It is also unlikely that your cell phone company forwards the required ports to the assigned IP address.
- ▶ To circumvent this problem, there is a 'drone mode' in Surveillance. In this mode, Sca connects to remote server instead of a client connecting to it.
- ▶ The drone has to first send a rtsp_request to SRelay. SRelay then creates a mount point for this request in RTSP RECORD mode and communicates it to the Sca. The drone then sends the video stream to this mountpoint and Srelay makes the video stream available to client's in its network.



Noteworthy Surveillance Features (Continued...):

2. Region Of Interest:

▶ Present in the Meta-Data with field name 'roi'.

Meant to focus the operator's attention to a specific area on the video stream and give the developer a facility to do processing outside of the cameras.

▶ Example:

- The facecount plugin gives the co-ordinates of the detected faces. The plugin developer can extract the face using this information and use it in his face recognition algorithm
- The motiondetect plugin will tell the operator where the suspicious activity occurred.



Noteworthy Surveillance Features (Continued...):

1)3. Video Mixer/Compositor

- ▶ Allows the user to create a single stream out of multiple cameras on the network.
- ▶ Primary motive behind developing this feature is producing a low bandwidth stream of viewers accessing the system over Internet especially for users of mobile phones with a 3G/4G connection.

1)4. Meta-Data encryption

- ▶ When configured in secure mode, uses GnuTLS to encrypt meta-data.
- ▶ Uses mutual authentication.

1)5. Adding more commands to Surveillance

- ▶ Place your shared library in the configured path and have it loaded using `dlopen(...)` to extend the command set of Surveillance.



Developing Plugins:

- A script will be provided to generate the basic structure of a plugin.
- Alternatively, you could use the code in the 'dummy' plugin.
- Process raw video in *_transform.
- Add meta-data using `gst_buffer_add_surveillance_meta (...)`
- Return `GST_FLOW_OK` if you detect a valid event.
- Return `GST_BASE_TRANSFORM_FLOW_DROPPED` to ignore the event / frame.

```
static GstFlowReturn gst_<plugin_name>_transform (...) {  
    <do_processing>  
    <if event_detected>  
        gst_buffer_add_surveillance_meta (...)  
        return GST_FLOW_OK  
    <else>  
        return GST_BASE_TRANSFORM_FLOW_DROPPED  
    }  
}
```



Testing New Plugins:

- Testing the Facecount plugin:
 - ▶ `gst-launch-1.0 v4l2src ! video/x-raw,width=640,height=480 ! Videoconvert ! facecount ! Videoconvert ! xvimagesink sync=false`
 - ▶
- Testing the motiondetect plugin:
 - ▶ `gst-launch-1.0 v4l2src ! video/x-raw,width=640,height=480 ! videoconvert ! motiondetect roi=320,240,320,240,1 ! videoconvert ! xvimagesink sync=false`



Future Scope:

Surveillance can be used for a wide range of applications.

Specialized plugins can be developed for handling any video monitoring requirement.

Examples:

- Intruder Detection: Using Motion Detection plugin by specifying region of interest. However, complex situations involving variations in illumination, movement of trees, bushes, shrubs etc. can be handled by developing specialized plugins.
- Traffic Signal Monitoring. Detect vehicles that jumped the traffic signal, capture number plate of speeding vehicle. Detect traffic congestion and determine if manual intervention is required at a traffic junction.
- Monitoring heat dissipated by your industrial equipment using thermal imaging. Generating alerts when a region gets too hot.



Future Scope (Continued...):

- Monitoring queue at a mall counter: Use facecount plugin to detect number of people waiting in queue at the exit counter. Alert the management when this number exceeds a threshold.
- Another scenario to consider would be one where there is crowding of customers in a particular area in the mall. With analysis done on the camera view, the manager can decide to move some products to another area in the mall.
- Unidentified object detection: Perform background analysis and detect suspicious objects.
-
- Integrate **Surveillance** with wide range of applications. These applications will need specialized plugins. Implementation of such plugins will need development of intensive image processing algorithms requiring either powerful CPUs and/or custom designed **DSP, FPGA** support.



Conclusion:

- Built to be integrated into any environment.
- Ability to establish a TCP (secure/unsecured) connection and a RTSP client to view the video stream.
- Get alerted quickly on detection of a suspicious event. Reporting an event can take as little as 200 to 300 bytes (without image).
- Possible to use this framework in a multi-vendor environment. Save your investment and be worry free while purchasing a Surveillance based CCTV solution.
- With such a framework in place, the possibilities of processing surveillance data are endless and limited only by the imagination of the developer.




Surveillance Video

Surveillance Metadata

sview

Video Metadata




logitech1 2016/09/30 08:55:55

Take Snap logitech1[192.168.1.]

sview

Video Metadata




asus 2016/09/30 08:55:55

Take Snap asus[192.168.1.2]

sview

Video Metadata




logitech1 2016/09/30 08:57:08

Take Snap logitech1[192.168.1.]

sview

Video Metadata



asus 2016/09/30 08:57:14

Take Snap asus[192.168.1.2]

sview

Video Metadata



LogitechOnOdroidC1 2016/09/30 08:55:55

Take Snap LogitechOnOdroidC1

sview

Video Metadata




rpi2 2016/09/30 08:55:55

Take Snap rpi2[192.168.1.202]

sview

Video Metadata




LogitechOnOdroidC1 2016/09/30 08:57:20

Take Snap LogitechOnOdroidC1

sview

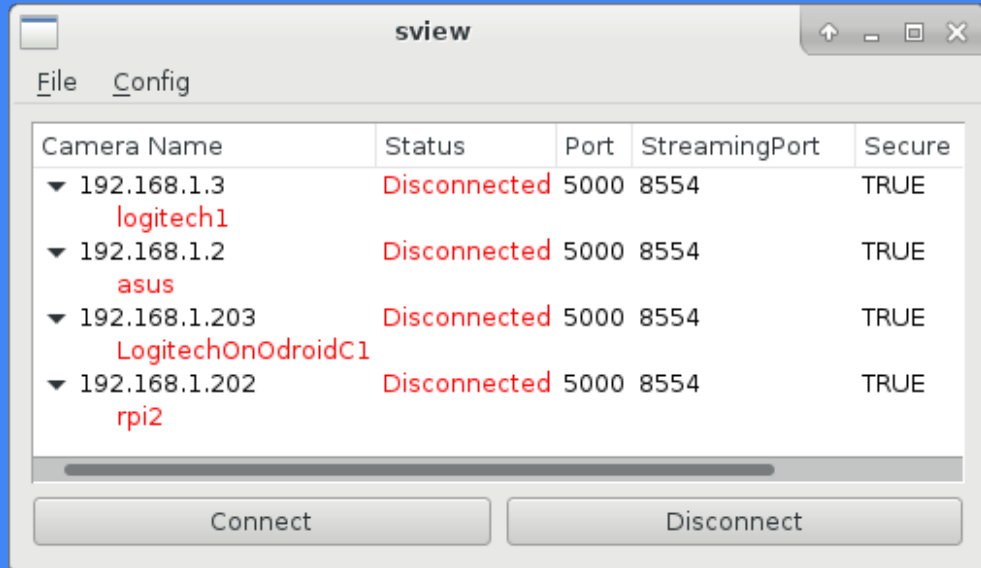
Video Metadata



rpi2 2016/09/30 08:57:20

Take Snap rpi2[192.168.1.202]

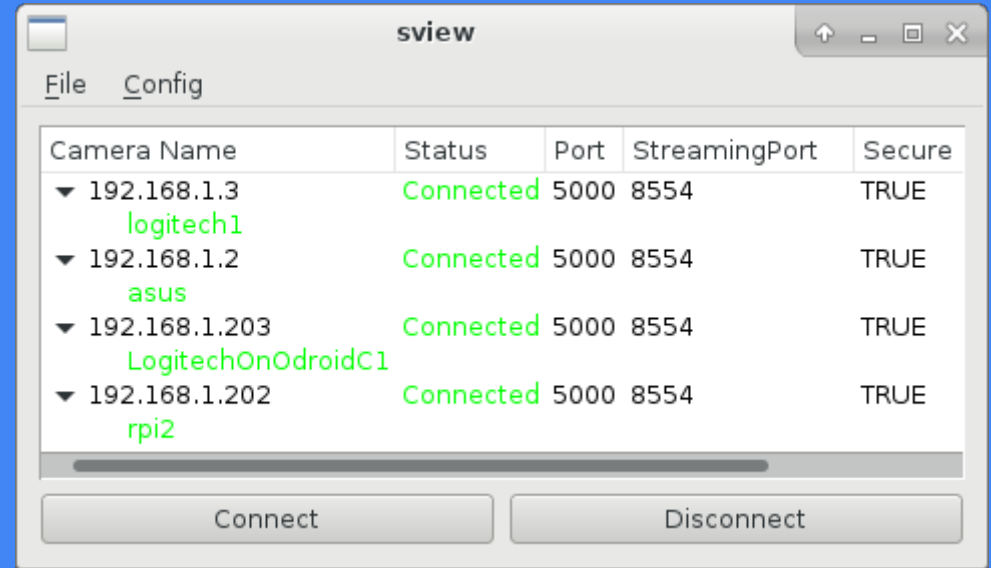
Camera Disconnected



The screenshot shows the 'svview' application window with a menu bar containing 'File' and 'Config'. Below the menu bar is a table with the following columns: Camera Name, Status, Port, StreamingPort, and Secure. The table lists four cameras, all with a 'Disconnected' status. The camera names are 'logitech1', 'asus', 'LogitechOnOdroidC1', and 'rpi2'. At the bottom of the window, there are two buttons: 'Connect' and 'Disconnect'.

Camera Name	Status	Port	StreamingPort	Secure
192.168.1.3 logitech1	Disconnected	5000	8554	TRUE
192.168.1.2 asus	Disconnected	5000	8554	TRUE
192.168.1.203 LogitechOnOdroidC1	Disconnected	5000	8554	TRUE
192.168.1.202 rpi2	Disconnected	5000	8554	TRUE

Camera Connected



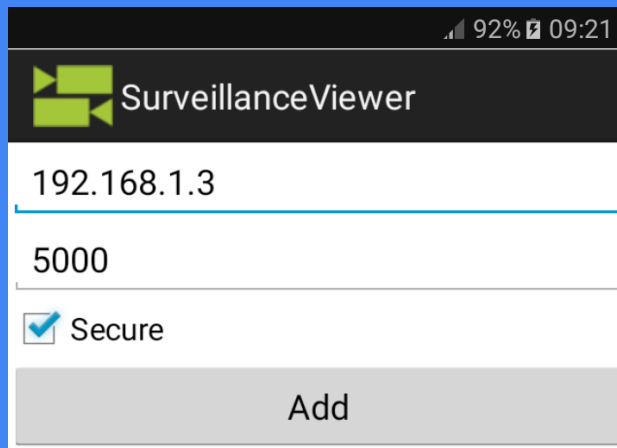
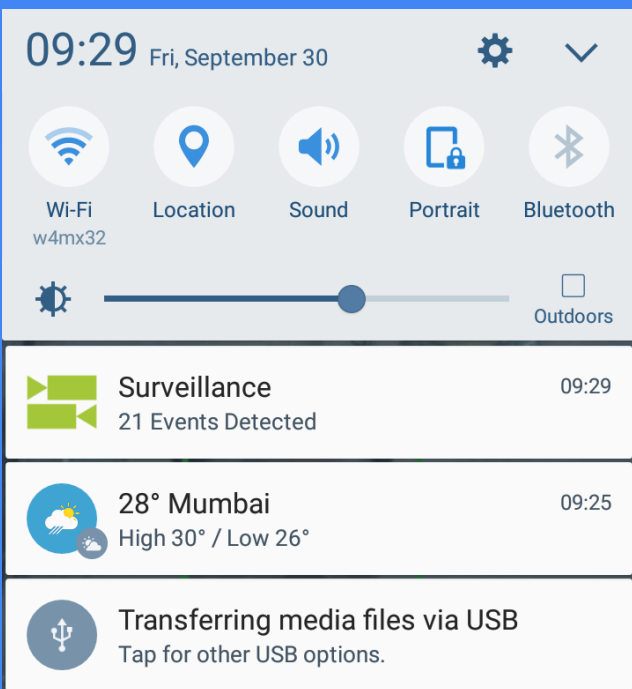
The screenshot shows the 'svview' application window with a menu bar containing 'File' and 'Config'. Below the menu bar is a table with the following columns: Camera Name, Status, Port, StreamingPort, and Secure. The table lists four cameras, all with a 'Connected' status. The camera names are 'logitech1', 'asus', 'LogitechOnOdroidC1', and 'rpi2'. At the bottom of the window, there are two buttons: 'Connect' and 'Disconnect'.

Camera Name	Status	Port	StreamingPort	Secure
192.168.1.3 logitech1	Connected	5000	8554	TRUE
192.168.1.2 asus	Connected	5000	8554	TRUE
192.168.1.203 LogitechOnOdroidC1	Connected	5000	8554	TRUE
192.168.1.202 rpi2	Connected	5000	8554	TRUE

SRelay Mixer Output



Android Application for Surveillance



rpi2 facecount
09/30/2016 10:08:31

logitech1 motiondetect
09/30/2016 10:08:31

LogitechOnOdroidC1 facecount
09/30/2016 10:08:31

logitech1 motiondetect
09/30/2016 10:08:31

Thank You

2

Mandar Joshi

Linux Developer

Phone: +91 9967820381

Email: emailmandar@gmail.com

Name	Mandar Joshi
Phone	+91 9967820381
Email	emailmandar@gmail.com
IRC	mndar
Github	https://github.com/mndar
Linkedin	https://www.linkedin.com/in/mndar