Interoperability Of IP-Based Cameras

Faythe C. Maston
fcmaston@mix.wvu.edu

Follow this and additional works at: https://researchrepository.wvu.edu/etd

Part of the Electrical and Electronics Commons

Recommended Citation
Maston, Faythe C., "Interoperability Of IP-Based Cameras" (2020). Graduate Theses, Dissertations, and Problem Reports. 7702.
https://researchrepository.wvu.edu/etd/7702
Interoperability Of IP-Based Cameras

Faythe Maston

Problem Report submitted to the Benjamin M. Statler College of Engineering and Mineral Resources at West Virginia University in partial fulfilment of the requirements for the degree of

Master of Science
In Electrical Engineering

Roy Nutter, Ph. D., Chair
Powsiri Klinkhachorn, Ph.D.
Kevin Bandura, Ph.D.

Lane Department Of Computer Science And Electrical Engineering

Morgantown, West Virginia
2020

Keywords: security, IP-based, interoperability

Copyright 2020 Faythe Maston
Abstract
Interoperability Of IP-Based Cameras

Faythe Maston

In this day and age of advancing technology and increasing crime, more and more citizens are investing in technology to increase their personal security. One such technology is the use of home-based IP security systems. These systems are comprised of one or more IP-based security cameras with owners preferring to have a way to view all camera feeds at once. Since not all security cameras are made to interact with each other, it is necessary to find a program that allows a user to view all security cameras at once, regardless of what brand of cameras they use. After researching, iSpy seems to be the most adaptable program that can be used in monitoring all security feeds, with most programs having larger limitations on what cameras can be used or the amount in use. While programs, such as Anycam or VLC, met some of the goals and others met all of the goals, but had limited functionality, like RTSP Mosaic Viewer, iSpy was found to be the ideal program to match the goal of this thesis.

Keywords: security, IP-based, interoperability
Table of Contents

Abstract ................................................................................................................................. ii
List Of Tables ........................................................................................................................ iv
List Of Figures ........................................................................................................................ iv
1. Problem Statement ................................................................................................................ 1
2. Introduction .......................................................................................................................... 1
3. Background .......................................................................................................................... 1
   3.1 Why Interoperability? ....................................................................................................... 1
   3.2 Why Not Interoperability? .............................................................................................. 2
4. Relevant Protocol Review .................................................................................................... 3
   4.1 RTSP Protocol ............................................................................................................... 3
   4.2 ONVIF Protocol ............................................................................................................. 3
5. ONVIF Profiles .................................................................................................................... 4
6. Tested Players ....................................................................................................................... 5
   6.1 RTSP Mosaic VLC Player Version 0.1.4.2 ..................................................................... 5
   6.2 iSpy By iSpyConnect ...................................................................................................... 8
   6.3 IP Camera Viewer By Deskshare .................................................................................. 11
   6.4 Security Monitor Pro ..................................................................................................... 14
   6.5 Anycam .......................................................................................................................... 14
   6.6 VideoLAN and VLC ....................................................................................................... 15
   6.7 PMPlayer ....................................................................................................................... 18
7. Comparison ......................................................................................................................... 20
8. Evaluation ............................................................................................................................. 20
9. Conclusion ............................................................................................................................ 21
References .............................................................................................................................. 22
List Of Tables
Table 1: A table listing a couple quick positive and negative points about the Mosaic Player….7
Table 2: A table listing a couple quick positive and negative points about the iSpy camera…..11
Table 3: A table listing one of the recognizable brands for each letter of the alphabet that is…12
Table 4: A table listing a couple quick positive and negative points about the IP Camera……..13
Table 5: A table listing a couple quick positive and negative points about the Anycam........15
Table 6: A table listing a couple quick positive and negative points about the VLC player…..18
Table 7: A table listing a couple quick positive and negative points about the iSpy camera….19
Table 8: A table comparing the features of the tested camera viewers in order to find the……20

List Of Figures
Figure 1: A figure displaying the "Options" pane of RTSP Mosaic VLC Player and how it......5
Figure 2: A screenshot showing the information needed under the "Sources" pane to add.......6
Figure 3: A screenshot showing iSpyConnect's URL Generator in practice, generating a........7
Figure 4: A screenshot displaying iSpy's "IP Wizard" generating a list of possible URLs that..8
Figure 5: A screenshot displaying iSpy’s "Grid" control and the options allowed to the user...9
Figure 6: A screenshot displaying iSpy's grid view for cameras and microphones, along……10
Figure 7: A screenshot showing IP Camera Viewer's interface to add IP cameras to the........12
Figure 8: A screenshot of IP Camera Viewer in use, displaying feeds from two camera........13
Figure 9: A screenshot showing the camera settings available for a feed in Anycam..........15
Figure 10: Screenshots showing the initial RTSP feed connection and authentication..........17
Figure 11: A screenshot showing a saved playlist that can be used to toggle between IP……18
Figure 12: A screenshot of PMPlayer in use with a live camera feed and the ability to load…19
1. Problem Statement

In today’s world of advancing technology, more and more people are attempting to integrate technology into their homes. One of the most common advancements to be integrated is the use of IP security camera systems. These are a system of camera(s) that are placed in common security locations such as doors, windows, garages, etc that can be used to view the array remotely. These systems are usually set up to also record said arrays and save the footage using video recorders. The most common need for home owners is a program that will allow users to view all of their cameras at once, preferably something that is open-source to the public and free of charge. The aim of this research program is find and research potential solutions that allow multiple cameras to be viewed at once in the same window, is open source, and has no monetary cost.

2. Introduction

With the increase in technology available, multiple companies will always attempt to make their own version of the newest product. One such industry that does this is the field of IP-based security cameras. Thousands of companies have created their own cameras and viewing programs that are available for both personal and commercial use. However, with the influx of these different brands comes multiple different features and means of communication. In order for companies and owners to build the best possible system for their use, they may begin to use and buy cameras from multiple companies. The struggle then becomes having a coherent way for these different systems to interact and be usable together. The key to solving this problem is the creation of interoperability software that allows cameras from different manufacturers to be viewed at the same time and possibly control the cameras in question.

3. Background

3.1 Why Interoperability?

The question for most is why interoperability for IP security devices is so important. Interoperability is defined by the “ability of a system (such as a weapons system) to work with or use the parts or equipment of another system”\(^\text{10}\). Interoperability in this case means the ability to use IP-based cameras. A point of question is why interoperability is needed in this field. The most common cause is the desire to set up a security network comprised of multiple IP cameras
from multiple brands. In the case of most homeowners or companies, different IP cameras may be chosen due to features such as camera clarity, degrees of motion, or powering ability. This issue of interoperability comes with the fact that depending on the setup of the software and hardware multiple pieces of software may be needed to properly view the feeds on said cameras. Another issue that can be solved by manufactures including interoperability is the issue of time. As time goes on, so does technology and hardware. If a new software is developed, it could mean that older software or hardware is no longer supported and would not be useable. By creating interoperability between software and hardware, these products are considered “future-proof”.

3.2 Why Not Interoperability?

While the concept of interoperability is considered a boon for most users, there are also reasons for systems not setting up interoperability as part of the systems. Interoperability encourages and allows for users to create “mix-and-match” security systems, comprised of multiple brand cameras and models. These style of security systems tend to be favored due to three main reasons: buying cameras for specific features needed in a location, affordable options, and survivability over time. By being able to mix and match cameras in a system, this allows a user more freedom in choosing affordable cameras that meet the needs of the location.

In contrast, by creating a security system that is not interoperable, users must usually purchase IP cameras and software only from said manufacturer, promising more business as long as the products are considered worth-while. Another reason for a lack of interoperability is the danger of using a publicly known protocol. By not needing to meet a well-known or publicly known protocol, a manufacturer can focus on their innate protocols and security. If hackers know the protocol, it can be easily exploited and possible for the attackers to learn how to manipulate and abuse the protocols to attack or exploit IP cameras. If users know this, users may feel more comfortable in buying a non-compatible camera. An example of this thought process comes from the early desire to own Apple-based products due to the belief that they are more secure and cannot be hacked. This is not true, but the in-built security and lowered amount of threats still encourage some users to continuously buy Apple products.
4. Relevant Protocol Review

4.1 RTSP Protocol

Real Time Streaming Protocol or RTSP is an application-level network protocol that is designed to control and manage the use of real-time data. This data can be comprised of audio and video, for functions such as streaming. This is considered to be one of the traditional streaming protocols, as it was drafted by Real Networks and Netscape in the 90s. The protocol functions by working as a network-remote-control for continuous media streams, synchronized by time. It is a common protocol used in IP-based cameras to transmit video and is often transmitted using Port 554. As this is an established protocol as well highly compatible, it is often used in creating interoperability between streaming devices.

4.2 ONVIF Protocol

ONVIF is an acronym representing the Open Network Video Interface Forum. This is a global and open industry protocol whose original design goal was to be the open global standard for the interface of physical IP-based security devices. Its primary use has been in the creation of a standard for how IP products, primarily IP security cameras. By creating a standard, it creates an easy means of communication between IP cameras of different brands, creating easy interconnectivity. ONVIF memberships is open to a wealth of users such as manufactures, software developers, consultants, system integrators, end users, and interest groups. Since the goal is interconnectivity, it is in everyone’s interest to allow as many interested parties as possible to integrate ONVIF into a wealth of products. According to ONVIF, there cornerstones of design revolve around creating a standard of communication between IP-based devices, interoperability regardless of the brand, and open communication to all parties.

The use of an open standard provides a wealth of benefits when operating IP-based devices. The most notable is the interoperability available by allowing devices from multiple manufacturers by using the same protocol for communication. By setting an open standard for companies/creators to follow, a measure of quality is expected from users, creating a respected user base. The open standard also allows for more flexibility in designing solutions for viewing and storing data from devices. Lastly, by creating and using an open standard allows for IP-devices to be essentially “future-proof”. Regardless of if a certain company discontinues a
product or service, there will always be interoperable products that can interact with a device, both new and old.

5. ONVIF Profiles

ONVIF also has developed a subset of specifications called profiles. These profiles have specifications in place that can ensure the interoperability of certain features between conforming devices. Profile S has a focus on addressing most common functionalities of IP video systems, including features such as video/audio streaming, PTZ control, and relay activation. Profile C’s specifications are aimed at addressing the common functionalities expected of IP access control systems, including state and control, credential management, and event handling. Profile G’s guidelines are setup to handle video storage, recording, searching, and retrieval. Profile Q is designed to handle device discovery and configuration along with the management of TLS certificates. Profile A places an emphasis on retrieving information, status, and events. It also handles the specifications and configurations of a Physical Access Control Systems (PACS) including access rules, credentials, and schedules. Lastly, Profile T is designed around the interoperability of video streaming features, including the use of H.264/H.265 encoding formats, imaging settings, and alarm events, such as motion and tampering detection. When designing clients and devices, they can fall under more than one ONVIF profile, increasing their interoperability with other devices, clients, and systems. The main issue with these profiles tends to lay with the compliance of these regulations, which is outside the scope of ONVIF. It is up the creators/integrators to check on requirements, while still creating a solid product/system that has the proper amount of security level implemented for the necessary use cases.

In general, the use of ONVIF serves as the easiest means of creating an interoperable system. The ONVIF website provides a list of all products that have been marked as ONVIF protocol compliant and the Profiles that it complies to. This gives users greater customization when designing IP-based security systems. In opposition, the interoperability of ONVIF means that any IP device that does not use ONVIF, in comparison, is much harder to connect with in a mixed system. ONVIF seems to have become a standard practice for most smaller or independent companies who are starting in the field in order to take advantage of currently existing software and devices. In comparison, most well-known or larger IP-based systems, such as Amcrest, are not all ONVIF-compatible. From a business standpoint, the decision to not be
ONVIF-compatible can make sense. By not allowing the interoperability of multiple cameras from different manufacturers, this allows the company to encourage users to only use and purchase their brand of product, whether it be software or hardware, in order to create a unified system. By not building around the ONVIF-protocol this also gives manufacturers more leeway with designing their system, whether for security or for multiple features across multiple devices.

6. Tested Players

6.1 RTSP Mosaic VLC Player Version 0.1.4.2

RTSP Mosaic VLC Player Version 0.1.4.2 is a program created by Grigory Lobkov that is an open source viewer that, when combined with VLC 2.1.5, can be used to create a custom grid and database of IP cameras. The program starts with a basic 2 x2 grid to display cameras in along with an open menu with two options, “Sources” and “Options”. “Options” allows a user to change the size of the grid by entering the proper grid size and selecting “ok” to change size.

![Figure 13: A figure displaying the "Options" pane of RTSP Mosaic VLC Player and how it can control the grid view of security cameras.](image)

“Sources” is used to be able to add and edit the IP cameras saved by the program. When selecting “New Camera” after right-clicking the panel, the user will be prompted for two necessary pieces of information “Camera Name”, which will be used to reference the camera, and an “RTSP connect string”. An RTSP Connect String is a URL that can be used to access a certain camera, containing information such as the IP address, username, and password, all depending on the brand and model. By entering the proper RTSP URL, the user can save the
camera and drop it into the corresponding grid location that they want the image to appear through, if it properly connects\textsuperscript{11}.

Figure 14: A screenshot showing the information needed under the "Sources" pane to add cameras to RTSP Mosaic VLC Player.

The main issue with this software is the knowledge of the RTSP connect string. Depending on the manufacturer or brand, you may or may not be told a basic RTSP URL format, since these can differ. The two main options for discovering said RTSP string are to attempt to contact the camera’s vendor or use a Third-Party generator, such as iSpyConnect. iSpyConnect is a company that has a wealth of information regarding the various RTSP URLs in use depending on factors such as manufacturer, model, and type of connection (JPEG, FFmpeg, VLC, etc.). By clicking one of the offered links, the user is greeted by a generator that will prompt the user for the camera’s IP, authentication information, and channel to generate the RTSP URL to access said camera\textsuperscript{7}. 
Figure 15: A screenshot showing iSpyConnect's URL Generator in practice, generating a RTSP URL to connect to a test camera.

As a note, a VLC player is needed in order to work in conjunction with the Mosaic player. There are also newer versions of this program available through Github, however, I could test their usefulness through my equipment, as I was unable to get a higher version to test and run on my equipment. Lastly, it is important to note that this is a Windows-based application. Table 1 provides a quick display of some obvious positive and negative features of the Mosaic Player.

Table 1: A table listing a couple quick positive and negative points about the Mosaic Player.

<table>
<thead>
<tr>
<th>Positives</th>
<th>Allows view of multiple cameras at once</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Open source</td>
</tr>
<tr>
<td>Negatives</td>
<td>Challenging installation and use, needs third-party software</td>
</tr>
</tbody>
</table>
6.2. iSpy By iSpyConnect

iSpyConnect is a company that started in 2007 and has continued to expand its databases and programs as technology has advanced. iSpy is their standalone program that can be used to connect to IP cameras if you know the inherent RTSP URL, the iSpy Wizard to connect with using URLs generated by iSpyConnect, the camera is connected locally, or through ONVIF protocol by going to “Add”. For clarity, the iSpyWizard begins by having the user select the make and model from a list before asking for any camera authentication and IP address. Once this information is entered, a number of potential URLs will be generated and the user can select the URL that best meets their needs for the camera.

Figure 16: A screenshot displaying iSpy's "IP Wizard" generating a list of possible URLs that can be used to connect to iSpy in a variety of ways.
Once cameras have been properly added to the system, selecting the “Grid” option will give you the ability to create and set the features of a video grid. These features include name, rows and columns, location, and more to create a user’s ideal setup.

![Grid View Manager](image)

Figure 17: A screenshot displaying iSpy’s "Grid" control and the options allowed to the user for customization.

After entering a “Grid” window, the user can select any of the available tiles to add cameras to, be able to select from a list of all cameras or microphones to add, even adding duplicates as necessary.
Outside of the three preliminary features that were the focus of research, iSpy also supports a number of other features that make it more useful for owners. When editing an IP camera, the user is given the option to use P/Z/T, or Pan/Zoom/Tilt, controls that can be used to move camera views if the camera is setup to take said commands. In general, the main issue with this program is that it is only available for Windows platforms.

The interface also displays how much CPU and RAM the program is currently using, allowing users to check how the program is functioning and how feeds are influencing the system. Another feature of this program is the ability to add “Plugins”, programs that can work with iSpy to increase its use. The three most endorsed by iSpy are: License Plate Recognition and Face Recognition. License Plate Recognition is designed to detect and recognize any license plates that match or do not match any license plates from a user-created list. Face recognition will inform the user if a human face is recognized in the video feed, but will ignore all other movement in order to reduce false alerts. Table 2 provides a quick display of some obvious positive and negative features of the iSpy camera program.
Table 2: A table listing a couple quick positive and negative points about the iSpy camera program.

<table>
<thead>
<tr>
<th>Positives</th>
<th>Negatives</th>
</tr>
</thead>
<tbody>
<tr>
<td>Easy installation and use</td>
<td>Windows-based only</td>
</tr>
<tr>
<td>Able to download plugin programs for increased usage</td>
<td></td>
</tr>
<tr>
<td>Large range of possible cameras and view screens at once</td>
<td></td>
</tr>
</tbody>
</table>

6.3 IP Camera Viewer By Deskshare

DeskShare was a company created in 2000 that developed products for the Windows platform. The company originally focused on the development and market of Web-based, subscription software services, such as video sharing www.videodesk.net. In 2001, due to the market demand for video editing software, they developed their first independent product called Video Edit Magic. At the current time, products are being developed in the following fields, FTP Client, Audio and Video Conversion, PC Desktop Recorders, and Video Surveillance. One of the video surveillance software available is IP Camera Viewer³.

IP Camera Viewer is a Windows-based application that is free to download and use. This software is designed to view live video feed from IP or USB cameras to make them viewable on a PC with limited regard for camera model or manufacturer. This program allows for the user to add devices to the program and be able to view four camera feeds at once in one window.

When setting up a new camera, the program will scan for any discoverable cameras, which can then be selected and will have the information for setup entered. Otherwise, the program will prompt the user for IP address, port, and authentication. Following this information, there are two options to finalize the remaining information. By selecting “Test Connection”, the program will attempt to connect to the camera and, if it is recognized, will fill out the brand and model of the camera before finishing initialization. Otherwise, the user can select a brand from the drop-down list and be prompted to then select from that brand’s recognized models before initializing⁶.
Figure 19: A screenshot showing IP Camera Viewer's interface to add IP cameras to the program.

The main issue of this program is that it requires the camera to be part of a set number of recognized camera manufacturer and brand. A list of the manufacturers is in Table 3.

Table 3: A table listing one of the recognizable brands for each letter of the alphabet that is recognized by IP Camera Viewer.

<table>
<thead>
<tr>
<th>Recognizable IP Camera Manufacturer For Each Letter Of The Alphabet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amcrest</td>
</tr>
<tr>
<td>Foscam</td>
</tr>
<tr>
<td>Keebox</td>
</tr>
<tr>
<td>Pixord</td>
</tr>
<tr>
<td>UBIQUITI</td>
</tr>
<tr>
<td>ZModo</td>
</tr>
</tbody>
</table>
For any unrecognized brand, the developers are open to contact and adding the brand/model in future updates, if possible. The main function of the program is the ability to control and view up to 4 camera feeds simultaneously, while held in the same window. The program can easily shift the layout of the camera feeds to suit the user’s preferences.

Figure 20: A screenshot of IP Camera Viewer in use, displaying feeds from two camera feeds of the four it can show at maximum.

For user customization, the program sports features that can adjust camera image and video properties in case of factors such as limited CPU or bandwidth. Users also have the ability to set or alter image properties from the feeds received. Lastly, this program supports a multitude of PTZ-enabled IP cameras, allowing the user to adjust the orientation of adjustable cameras. Even for those devices without a zoom function, IP Camera Viewer provides an artificial/digital zoom that can be applied to the feeds. Table 4 provides a quick display of some obvious positive and negative features of the IP Camera Viewer.

Table 4: A table listing a couple quick positive and negative points about the IP Camera Viewer program.

<table>
<thead>
<tr>
<th>Positives</th>
<th>Easy installation and use</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Searches for connected cameras to connect automatically</td>
</tr>
<tr>
<td>Negatives</td>
<td>Limited to 4 cameras only</td>
</tr>
</tbody>
</table>
6.4. Security Monitor Pro

Security Monitor Pro is an advanced version of the IP Camera Viewer from DeskShare that is available for a cost of $89.95. This system supports all the features of IP Camera Viewer, but either enhances previously existing features or includes new features. Compared to the original IP Camera Viewer, which could handle four IP cameras, Security Monitor Pro can handle up to 32 cameras feeds simultaneously. This advanced version also includes the ability to record camera feeds in multiple resolutions, which is not available in IP Camera Viewer. This program also allows for a variety of plug-ins, which have features such as motion-triggered recordings depending on the camera and motion detection on recorded feeds.

6.5. Anycam

Anycam is a manufacturer that produced the AnyCam.io IP Camera Software. This Windows-based software is marketed as being able to provide multiple IP camera feeds. By entering a camera’s IP and authentication, the program scans through to locate open ports for communication and commands to interact with said camera. For multiple displays, one feed is displayed in a large format, while all other camera feeds are displayed as a tile, which can be selected and viewed as the main feed. These feeds can be recorded to a file path of the user’s choosing for a set amount of time, as well as adjust the quality of feed and its effect on CPU usage.
This program features a free version and three paid versions, with the difference being due to the number of cameras that can be displayed. The free version is only applicable for adding and viewing one camera at a time. The paid versions cost $29, $37, and $49 and could handle 4, 8, and 32 cameras, respectively. Overall, while the program is useful and meets the goals of the thesis when using the paid model, the free version is only suitable for one stream. Table 5 provides a quick display of some obvious positive and negative features of the Anycam camera program.

Table 5: A table listing a couple quick positive and negative points about the Anycam camera program.

<table>
<thead>
<tr>
<th>Positives</th>
<th>Easy installation and use</th>
<th>Adjustable viewing settings for limited hardware</th>
</tr>
</thead>
<tbody>
<tr>
<td>Negatives</td>
<td>Free (demo) version is limited to only one camera</td>
<td></td>
</tr>
</tbody>
</table>

6.6. VideoLAN and VLC

VideoLAN was a piece of software that originated out of an academic project started in 1996. VideoLAN Client was abbreviated to “VLC” when VLC was a client of the original
VideoLAN project, even though VLC is no longer a client of VideoLAN. VideoLAN’s original plan was to consist of a client and a server, which could be used to stream videos from satellite dishes across a campus network, developed by students at the Ecole Centrale Paris. The code was rewritten from scratch in 1998 and released on February 1, 2001 under the GNU General Public License with the authorization of the headmaster. In this build, the original VideoLan Server (VLS) had been integrated into VLC and the project was changed to the VLC media player due to removal of the client/server infrastructure from the initial product.

After being developed for 13 years, the first version of VLC media player was released on July 7, 2009. In 2010, development of VLC for Android platform began and is now available on Android and Google Play stores as of 2011. An independent company, “Applidium”, started work on an iOS port of VLC in September 2010, with the VLC project’s endorsement. In 2011, VLC was withdrawn from the Apple store following a licensing conflict between VLC’s GPLv2 and the App store’s policies. A version for Windows Store was released on March 13, 2014 with support for Windows RT, Windows Phone, and Xbox One being added at a later date. Version 3.0 started development in June of 2016 for the macOS, Linux, and Windows operating system and was released in February 2018.

One of the perks of VLC is the modular design of the program, allowing for the ability to easily include modules for new file formats, codecs, interfaces, or streaming methods. VLC supports all audio and video formats that are supported by the libavcodec and libavformat libraries. Inherently, this means that VLD can play back H.264, MPEG-4 Part 2, FLV, or MXF file formats. Alternative video/audio may be supported by the use of modules that are not based on the use of FFmpeg’s libraries. At the current time, VLC media player has available versions for Android, iOS, iPadOS, Windows Phone, OS/2, Syllable, BeOS, tvOS, Windows, Chrome OS, macOS, and Linux. These diverse systems allows for an increase in compatibility regardless of the viewing apparatus in place.

Part of VLC’s set features is an ability to open/save network streams. For the purpose of most IP cameras, RTSP protocol is used to transmit the physical video footage collected from the IP camera. In this case, selecting “Open Network Streams” allows for the user to input a protocol and IP that they want to interact. By inputting the information “rtsp://x.x.x.x/”, with the x’s replaced by the octals of the IP address of the camera in question. If the VLC can successfully
connect, the user may then be prompted for the username and password combination assigned to the device during its initial setup.

![Screenshot of VLC interface with RTSP feed connection and authentication messages]

Figure 22: Screenshots showing the initial RTSP feed connection and authentication messages present in VLC.

Afterwards, if all initial information is input correctly and no disconnect happens, the visual and audio from said IP camera will be displayed in VLC media players. Multiple RTSP streams can be added or saved to a playlist and can be easily switched between the audio and visuals.
Figure 23: A screenshot showing a saved playlist that can be used to toggle between IP camera feeds in VLC, along with a minimized feed of one such camera.

However, there is no option to currently display multiple videos in the same window at the same time. In conclusion, this program works as a partial solution to the issue of IP camera interoperability. Another issue that may come with using this program is that the program is not capable of recording the RTSP streams of cameras. Table 6 provides a quick display of some obvious positive and negative features of the VLC player.

Table 6: A table listing a couple quick positive and negative points about the VLC player.

<table>
<thead>
<tr>
<th>Positives</th>
<th>Negatives</th>
</tr>
</thead>
<tbody>
<tr>
<td>Able to download plugin programs for increased usage</td>
<td>Can only view one camera at a time (addresses can be saved)</td>
</tr>
<tr>
<td>Available for Windows and Mac</td>
<td></td>
</tr>
</tbody>
</table>

6.7. PMPlayer

The Picomixer Media Player or PMPlayer is another free software that can be used to view the visual and audio feeds of IP cameras. Picomixer as a company that started in 2005 and is a multinational group in the software industry. The main goal of this company is in the development and implementation of video-based systems in multimedia projects or in analytical studies. This is another program that has interface that allows for the ability to view IP cameras.
This program allows the user to view live feeds, record feeds (while the window is open), and potentially send commands to your IP camera (based on his infrastructure). The program prewrites in the “rts://” part of the command needed to connect to the common IP camera, requiring only the IP camera’s IP address and gives an access point to enter the username and password for the camera, if it has one. Once an address and username/password combination have been entered, this camera’s information can be saved and easily accessed upon reentry into the program.

Figure 24: A screenshot of PMPlayer in use with a live camera feed and the ability to load a URL from a previous entry visible.

The UI for the program tends to be more intuitive for newer users as instructions are clear and have a graphical image to represent setting up the connection. However, the program seems to have issues with switching between cameras, while another feed is being viewed. In practice, this led to multiple stalls during the testing procedure. Table 7 provides a quick display of some obvious positive and negative features of the PMPlayer.

Table 7: A table listing a couple quick positive and negative points about the iSpy camera program.

<table>
<thead>
<tr>
<th>Positives</th>
<th>Easy installation and use, can potentially use P/T/Z controls</th>
</tr>
</thead>
<tbody>
<tr>
<td>Negatives</td>
<td>Can only view one camera at a time, crash issues with changing camera</td>
</tr>
</tbody>
</table>
7. Comparison

Table 8 provides a comparison view of the tested players according to the goals listed in the problem report, along with additional features.

Table 8: A table comparing the features of the tested camera viewers in order to find the best product available for the requirements of the problem report.

<table>
<thead>
<tr>
<th>Asset</th>
<th>Mosaic</th>
<th>iSpy</th>
<th>IP</th>
<th>Security</th>
<th>Anycam</th>
<th>VLC</th>
<th>PMP</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 Or More Cameras Viewable At Once</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No(1)</td>
<td>No(1)</td>
<td>No(1)</td>
</tr>
<tr>
<td>Free</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Demo</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Open-Source</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Ease Of Download And Setup</td>
<td>Hard</td>
<td>Easy</td>
<td>Easy</td>
<td>Easy</td>
<td>Easy</td>
<td>Medium</td>
<td>Easy</td>
</tr>
</tbody>
</table>

8. Evaluation

After reviewing the comparison, two programs meet the three ideal standards placed at the start of this research. These standards were that the program should be open-source, free to any user, and have the ability to view multiple cameras of different types at a time. The two that meet these guidelines are the Mosaic VLC Viewer and iSpy. When comparing these two products, however, iSpy comes across as the superior product. iSpy has an easy installation and is fairly straightforward for users of varying experience to use due to a proper user interface and built-in features for adding cameras. As well, iSpy has a number of additional plug-ins that can be added and applied to security feeds to increase their usability. In comparison, Mosaic has multiple issues that require secondary software to solve, such as finding the addresses to properly interact with the camera. Mosaic also suffers from an unclear user interface and issues such as
delay. iSpy also features the secondary software needed to make Mosaic work built-in, helping with its preference. With these features in mind, iSpy was found to be the product that best fit the goals of problem report and goes beyond them.

9. Conclusion

As a conclusion, each program was judged on whether it was open-source, monetarily free, and the ability to view multiple cameras at once in the same window. Overall, the program that was found to best meet these standards was iSpy. While this program met all three requirements, it also features additional plugins and features that can maximize usability, such as the ability to use P/T/Z controls, storage options, and CPU/RAM usage reports. Therefore, iSpy is the program that most closely meets the goal of the thesis and has more features on top of those.
References

1  Anycam — IP camera software, easy to setup, easy to use. 2020. *Anycam - IP Camera Software, Easy To Install, Easy To Use*. [online] Available at: <https://anycam.io/pricing/>.


