



# Research Repository UCD

|                                     |  |
|-------------------------------------|--|
| <b>Title</b>                        | An Analytical Approach to the Recovery of Data from 3rd Party Proprietary CCTV File Systems  |
| <b>Authors(s)</b>                   | Gomm, Richard, Le-Khac, Nhien-An, Scanlon, Mark, Kechadi, Tahar  |
| <b>Publication date</b>             | 2016-07-08   |
| <b>Publication information</b>      | Gomm, Richard, Nhien-An Le-Khac, Mark Scanlon, and Tahar Kechadi. "An Analytical Approach to the Recovery of Data from 3rd Party Proprietary CCTV File Systems," 2016. |
| <b>Conference details</b>           | 15th European Conference on Cyber Warfare and Security (ECCWS-16), Munich, Germany, 7-8 July 2016  |
| <b>Item record/more information</b> | <a href="http://hdl.handle.net/10197/7893">http://hdl.handle.net/10197/7893</a>  |

Downloaded 2024-04-18 19:00:57

The UCD community has made this article openly available. Please share how this access benefits you. Your story matters! (@ucd\_oa)



© Some rights reserved. For more information

## **An Analytical Approach to the Recovery of Data from 3rd Party Proprietary CCTV File Systems**

Richard Gomm, Nhien-An Le-Khac, Mark Scanlon and M-Tahar Kechadi  
School of Computer Science, University College Dublin, Dublin 4, Ireland

[richard.gomm@gmail.com](mailto:richard.gomm@gmail.com)

[an.lekhac@ucd.ie](mailto:an.lekhac@ucd.ie)

[mark.scanlon@ucd.ie](mailto:mark.scanlon@ucd.ie)

[tahar.kechadi@ucd.ie](mailto:tahar.kechadi@ucd.ie)

**Abstract:** According to recent predictions, the global video surveillance market is expected to reach \$42.06 billion annually by 2020. The market is extremely fragmented with only around 40% of the market being accounted for by the 15 top video surveillance equipment suppliers as in an annual report issued by IMS Research. The remaining market share was split amongst the numerous other smaller companies who provide CCTV solutions, usually at lower prices than their brand name counterparts. This cost cutting generally results in a lower specification of components. Recently, an investigation was undertaken in relation to a serious criminal offence, of which significant video footage had been captured on a CCTV Digital Video Recorder (DVR). The unit was setup to save the last 31 days of footage to an internal hard drive. However, despite the referenced footage being within this timeframe, it could not be located. The DVR unit was submitted for forensic examination and data retrieval of specified video footage which, according to the proprietary video backup application, was not retrievable. In this paper, we present the process and method of the forensic retrieval of video footage from a DVR. The objective of this method is to retrieve the oldest video footage possible from a proprietary designed file storage system. We also evaluate our approach with a Ganz CCTV DVR system model C-MPDVR-16 to show that the file system of a DVR has been reversed engineering with no initial knowledge, application or documentation available.

**Keywords:** CCTV forensics, CCTV-DVR file systems analysis, video file carving, reverse-engineering

### **1. Introduction**

The closed-circuit television (CCTV) is a video surveillance system that can be used for any type of monitoring. In the 2012 annual report issued by IMS Research, an independent supplier of market research, it estimated that the video surveillance equipment market was worth over \$9 billion in 2010. Yet only around 40% of the market was accounted for by the 15 top video surveillance equipment suppliers (IMS Research 2012). The remaining market share was split amongst the numerous other smaller companies who provide CCTV solutions, usually at lower prices resulting from a lower specification of components. The advanced forms of CCTV normally use Digital Video Recorder (DVR) that allows CCTV video images are recorded and archived continuously from all cameras for 90 days or more, with a variety of quality. The CCTV-DVR devices are widely used for surveillance in areas that may need monitoring such as banks, airports, military installations, stores, etc.

Recently, an investigation was undertaken in relation to a serious criminal offence, of which significant video footage had been captured on a Ganz CCTV Digital Video Recorder (DVR) model C-MPDVR-16. The unit was setup to save the last 31 days of footage to an internal hard drive, however despite the referenced footage being within this timeframe it could not be located. The Ganz DVR unit was submitted for forensic examination and retrieval of specified video footage which, according to the proprietary video backup application, was not retrievable. Initial examination of the Ganz DVR unit, under forensic conditions, revealed a proprietary file system i.e. a file storage system not recognised as an industry standard. In that fashion the standard tools like EnCase (<https://www.guidancesoftware.com/>), X-Ways (<https://www.x-ways.net/>) etc. were unable to recover any video footage. In fact, DVR's come from a multitude of manufacturers, each using their own unique variants of both equipment and software. In some case this presents as an industry standard operating system, such as Linux, which presents as an easy forensic retrieval using standard tools. However in the majority of cases the forensic investigator will encounter a proprietary or drastically cobbled-together operating system which provides a significant challenge to decode. The latter is the focus of this paper.

The topic of this paper are the CCTV systems described in the previous paragraph and in particular the forensic examination of a Ganz CCTV DVR model C-MPDVR-16. The research problem was established to reverse engineer (Poole et al. 2008, Zeltser 2010) the proprietary file system and establish the details of the oldest recorded video footage available for retrieval. A secondary objective was set in being able to carve video footage from the Ganz DVR unit's internal hard disk drive into a playable format.

Our paper is set out as follows: Section 2 shows related work of CCTV-DVR forensics. We discuss on forensic techniques applied for CCTV-DVR investigations as well as forensic challenges in Section 3. We describe and discuss a case study of forensic acquisition and analysis of Ganz CCTV DVR model C-MPDVR-16 in Section 4. Finally, we conclude and discuss on future work in Section 5.

## **2. Related work**

Ariffin et al (2013) describe a forensic technique to carve video files with timestamps. Within this paper, authors also proposed an extension to the digital forensic framework established by (McKemmish, R. 1999), which had four steps: (i) Identification, (ii) preservation, (iii) analysis and (iv) presentation. It is within Step 3: Analysis that authors specifically reference the issues with proprietary file systems. Due to the amount of variables and unknown systems available it would not be feasible to produce a technical guide to reverse engineering a proprietary file system. Instead a detailed analysis of the required steps was given; in summary the following was established: (i) First the byte storage method must be determined (little / big endian); (ii) File signatures can then be derived and used to correlate each file signature to the channel video that captured the scene with timestamps and (iii) Video coded must be located and installed. In this paper, we provides details of the analysis of the Ganz DVR unit conducted in accordance with the framework outlined by (Ariffin et al 2013). Another research was performed by Dongen (2008) on a Samsung CCTV system. This research focuses however on a well-known file system: *ext3*. In Wang (2009), author focused on Digital Video Forensics that could apply in the context of CCTV forensics.

Han et al (2015) present the forensic analysis of a CCTV-DVR of which the hard disk using a HIKVISION file system. Authors also mentioned it is an unknown file system. In this research, they identify the structure and mechanism of a HIKVISION file system. Authors show moreover that the procedure in the case analysis can be useful to counter anti-forensic activities. This paper only however focuses on HIKVISION file system that is not popular in video surveillance devices in western countries.

Recently, Tobin et al. (2014) conducted a CCTV-DVR forensics. Authors produced a short report into his examination of the file system used on the Ganz internal hard disk drive. This report was non-technical and aimed for the legal investigative function. Authors found that the Ganz internal hard disk drive was split into three regions, each of which had a specific action: (i) Region 1 – date 1 block, (ii) Region 2 – date 2 block, (iii) Region 3 – video data. To date there is any technical paper that has been published by authors on how the findings were reached; it is believed they used a software tool to watch the disk access whilst using the AvTech proprietary application, DiskTools.exe, to access a copy of the Ganz internal hard disk drive. This process has a distinct disadvantage as you are reliant on the DiskTools.exe programming, which is proven later in this paper to not provide full details of the oldest video footage available from the Ganz DVR internal hard disk drive.

## **3. CCTV-DVR forensics**

Following the literature survey in Section 2, we notice that there is very little information available on the proprietary file system used by the Ganz DVR unit, or the specified AvTech hard disk drive system. Whilst CCTV footage can and should be obtained through the proprietary application shipped with each unit, this leaves very little room for forensic examination and leaves the investigator at the mercy of proprietary programmers. Additionally there may remain remnants of video footage which are not complete and therefore not available through any such proprietary system.

By reverse engineering the AvTech file system the investigator obtains full access to all information on the DVR hard disk drive. Additionally this can be done in a forensic manner and without the need to rely on the ability of an unknown 3rd party programmer. Following the forensic acquisition a sample of video footage must be extracted from the DVR hard disk drive and converted for playback. This will require codec, headers and encoding identification.

Before looking at a case study on forensic acquisition and analysis of Ganz CCTV-DVR in the next section, we discuss on the identification of file systems of internal hard drive of Ganz CCTV-DVR (Ganz internal hard disk drive). In fact, file systems such as FAT16, FAT32, NTFS, HFS, Ext2 are some examples of industry standard file system. These file systems are used to keep track of where data is located on a disk, providing a tree like structure comprising of files and folders. Each of the industry standard file systems referenced above are

required to identify themselves with a unique hexadecimal code in the Master Boot Record at the beginning of each hard disk drive. This unique hexadecimal code is commonly referred to as a 'Magic Marker' or 'Magic Byte' (Haider et al. 2012). However, forensic examination of Sector 0, the Master Boot Record, of the internal hard disk drive from the Ganz CCTV-DVR contained no Magic Marker and therefore did not contain one of the industry recognised file systems. In order to progress it would be a requirement to reverse engineer the unknown file system used on the internal hard drive. In order to do this you must first review how a file system stores data. In general before a file system can be created a partition is required to specify how much of the hard drive is to be used. These partitions equate out to a start point (cylinder) on the hard drive and an end point (cylinder). Once a partition is identified it is possible to establish the content of each area and how they relate to each other, building up a picture of communication as they progress.

Despite Master Boot Record area of the Ganz internal hard disk drive contains no magic marker to assist in the identification of the file system, there is other data which gave a starting point for research. The Master Boot Record of the Ganz internal hard disk drive was examined in X-Ways Forensic tool, it displayed the following (Figure 1):

| Offset     | 0  | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9  | A  | B  | C  | D  | E  | F  |                  |
|------------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|------------------|
| 0000000000 | 00 | 00 | 44 | A9 | 4E | 39 | 00 | 00 | FB | FF | FF | 39 | 00 | 00 | 00 | FF | D@N9 ãÿÿ9 ŷ      |
| 0000000010 | 00 | B0 | 4B | BA | 01 | 00 | 00 | 00 | 00 | 00 | 58 | 6D | 01 | 00 | 00 | 00 | *K² Xm           |
| 0000000020 | 30 | 7E | F7 | 00 | 00 | 00 | 00 | 00 | 60 | 6F | F7 | 00 | 00 | 00 | 00 | 00 | 0~÷ `o÷          |
| 0000000030 | 55 | 41 | 56 | 54 | 45 | 43 | 48 | AA | 46 | 53 | 53 | 31 | 36 | 41 | 00 | 55 | UAVTECH³FSS16A U |
| 0000000040 | 00 | 3C | AD | BA | 02 | 00 | 00 | 00 | 00 | 30 | 5E | 38 | 3A | 00 | 00 | 00 | <-² 0^8:         |
| 0000000050 | 00 | 00 | 72 | 74 | 00 | 00 | 00 | 00 | 00 | FF | AC | BA | 02 | 00 | 00 | 00 | rt ŷ-²           |
| 0000000060 | 00 | 02 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | FF | 71 | 74 | 00 | 00 | 00 | 00 | ŷqt              |
| 0000000070 | 00 | 00 | AD | BA | 02 | 00 | 00 | 00 | 00 | FF | AC | BA | 02 | 00 | 00 | 00 | -² ŷ-²           |
| 0000000080 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 |                  |

**Figure 1:** Ganz internal hard disk drive – MBR (Sector 0)

The ASCII text AVTECH and FSS16A became relevant when it was established that the internal components of the Ganz DVR unit were manufactured with the brand AvTech. We conduct a search on both the branded company, Ganz and also the maker of the internal components AvTech. We realise that Ganz is a product brand of CBC America Corp. It would appear that the Ganz DVR unit model C-MPDVR-16 is an old product and as such there was very little information available on the unit. The only notable information is that the unit records with a MPEG format ([http://en.cbc-cctv.com/uploads/tx\\_n21products/05\\_tab\\_088\\_01.gif](http://en.cbc-cctv.com/uploads/tx_n21products/05_tab_088_01.gif)).

Of particular importance to this paper is a thread on the CCTVforums.com website (CCTVforums.com), which discusses retrieving data from AvTech based units. This thread identified that AvTech devices use a proprietary file system which has not been decoded, however AvTech had released a program to assist in the interrogation of such DVR hard drives. The program is called Disk Tools.exe and a download link was provided. We also use this tool to compare forensic results with our approach.

Significantly the AvTech website also provides a warning that data may be destroyed if the DVR internal hard disk drive was connected to a PC and accessed. This would likely be due to Microsoft Windows trying to write a new recognised Master Boot Record to the hard disk drive on its initialization. In fact, our paper relates to a forensic examination with the use of a disk image created through a read only device. The actual Ganz DVR internal hard disk drive was never connected directly to any computer.

#### 4. Forensic acquisition and analysis of Ganz CCTV: A Case study

##### 4.1 Acquisition

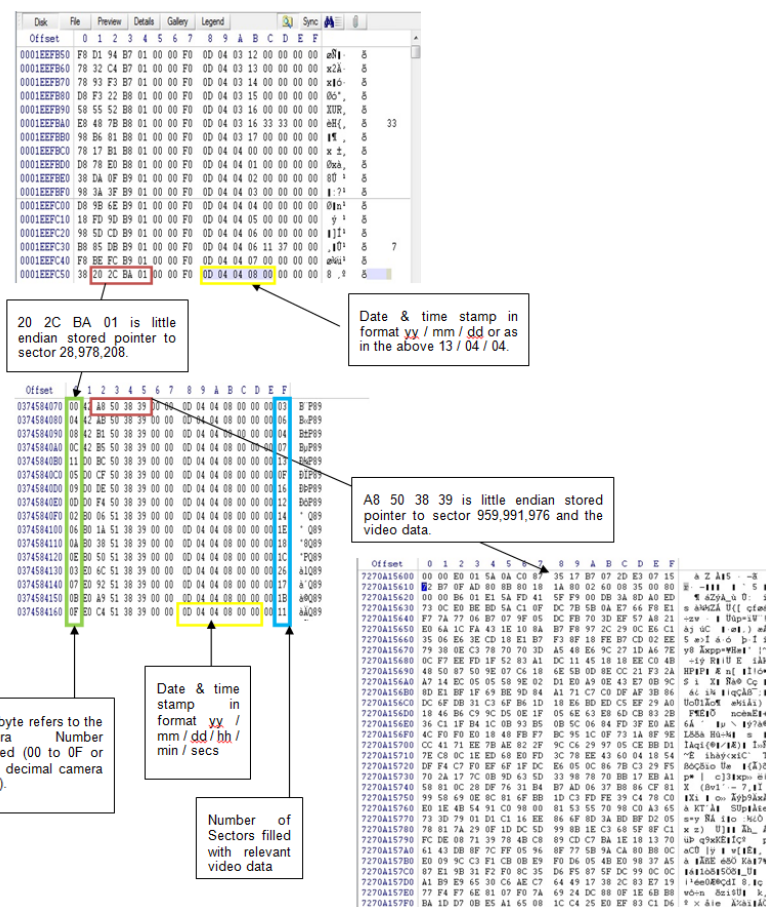
The forensic process we used in this section is adopted from the Ariffin's model (Ariffin et al 2013). In our experiment, we perform forensic acquisition and analysis of a Ganz DVR Unit. The first step is identification where a Ganz C-MPDVR-16 DVR was photographed. We also take the photos at various stages of the examination. Figure 2 shows the internal view of this Ganz DVR Unit.

The next step is preservation. Within the law enforcement environment it is imperative that any data is recovered in a forensically-sound method. The general accepted criterion is that no changes are made to the original data source, and that any copies made are identical to the original data source. In our experiment, the Ganz DVR unit was seized from the working environment by law enforcement agents on the 4<sup>th</sup> of April 2013

shortly after 08:00hrs. A strict chain of custody from seizure to examination was in place and no access to the Ganz DVR Unit was permitted. For this examination the DVRs internal hard disk drive was removed and connected to a forensic device known as a write blocker. Next a bit copy was made of the hard disk drive. This is a method to ensure that an exact copy of the entire hard disk drive is produced; including all used and unused space.



**Figure 2:** Internal view of the Ganz C-MPDVR-16 DVR unit.



**Figure 3:** Overview of the AvTech Proprietary File System

## 4.2 Analysis

The analysis took place on the clone hard disk drive, examining the information at the storage level to identify any recognised file signature. The key challenge with a proprietary file format is to locate the video streams without the assistance of predefined storage methods (FAT, NTFS etc.)

### 4.2.1 Initial Actions

The cloned hard disk drive was opened within X-Ways Forensics, EnCase and WinHex none of which were able to recognise the file system or any video file. So, we examine the first sector of a hard disk drive (Master Boot

Record). The only identifiable data in the sector 0 was the label UAvTech and FSS16A. No technical information was obtainable, other than FSS16A which is a proprietary file system of AvTech.

#### 4.2.2 Identifying the data

Further examination was conducted to review the type of data visible. We found there are totally three distinct areas on the hard drive; the first contains entries which provide a general date and time for recorded footage and a pointer to the second area. The second area provides separate entries containing refined date and time stamps for each of the individual 16 cameras video footage recorded, and a pointer to the video footage. The third area contains the actual video footage (Figure 3). Due to the limitation of the paper size, we do not present in details the forensic process we used to locate these three areas. Besides, the main objective of our paper is to show how to recover and playback the recovered video data.

#### 4.2.3 Retrieving data

In our experiment, one second period timeframe retrieval was attempted, 04/04/2013 08:00:00 hrs to 08:00:01 hrs. The data of this period is located in sector 28,978,208 / offset 0374584070 thru to sector 28,978,211 / offset 0374584720. By analysis of the headers in those sectors we notice that the video footage was captured at 6 frames per second. This carved data also revealed duplicate entries for cameras 0C, 0D, 0E, 0F on the 2<sup>nd</sup> and 5<sup>th</sup> frames, represented by bytes 1 of each entry showing FF. This pattern continued through all data and is possibly recording an error. In order to recover one second of video footage for camera 00 (6 frames per second) data required carving from the following sectors:

```
00 42 A8 50 38 39 00 00 0D 04 04 08 00 00 00 03
00 50 D5 51 38 39 00 00 0D 04 04 08 00 00 00 1A
00 51 8D 52 38 39 00 00 0D 04 04 08 00 00 00 05
00 52 E0 52 38 39 00 00 0D 04 04 08 00 00 00 03
00 53 28 53 38 39 00 00 0D 04 04 08 00 00 00 03
00 54 80 53 38 39 00 00 0D 04 04 08 00 00 00 03
```

|                       |
|-----------------------|
| Sectors 959991976 +3  |
| Sectors 959992277 +26 |
| Sectors 959992461 +5  |
| Sectors 959992544 +3  |
| Sectors 959992616 +3  |
| Sectors 959992704 +3  |

All data was carved and compiled into one file; it totalled 21.5k in size.

#### 4.2.4 Carved Data Identification

In order to identify the data within the carved file, we tried a number of industry standard applications including MediaInfo 0.7.69, GSpot v2.70a, VideoInspector 2.6.0.129. However, none of the above software was able to detect the video format contained in the carved file, indicating that it was likely a proprietary encoding. The Ganz DVR system was supplied with proprietary viewing software called Video Player MFC, version 1.1.6.1. The application stated it played the following proprietary file formats: .VS4, .VSE, .DVR, .AVC, .DV4. So, we made five copies of the carved data file and each provided with one of the five file extensions above. These were then loaded into Video Player MFC. The only successful access within the video player was the copy of the carved data given the .DVR extension. Thus the video player identified the clip as being from the 04<sup>th</sup> April 2013 at 0800hrs as expected, it also displayed that the footage was recorded in the 720 x 576 pixel format (Figure 4).

Based on our investigation, the Ganz DVR is listed as recording in MPEG4 format. A review of the carved file used previously showed that each segment of video data started with the hex: 00 00 E0 01. According to the above MPEG file header format (<http://mpeg.chiariglione.org/>) a standard MPEG file would start with: 00 00 01 ?? . With the ?? being replaced by the relevant bit above, which for a video stream would be E0 to EF depending on the stream. So, if we place the two sections of hex codes together there is a clear similarity:

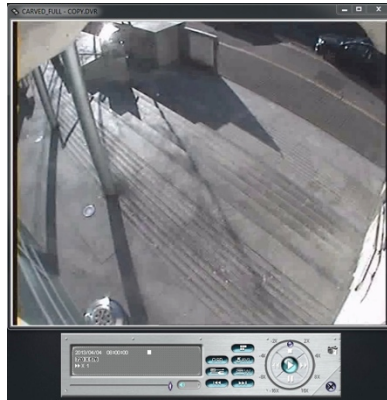
Carved data file - 00 00 00 E0 01

MPEG Format - 00 00 00 01 E0 (first video stream)

It was possible that the Ganz DVR simply transposed bytes 3 and 4. In order to test this theory, we used a hex editor to modify a copy of the carved data file swapping bytes 3 and 4 in each of the stream headers. This modified data was saved as carved.mpg file and attempted to be opened in a number of media players. However the carved.mpg file would not play. The carved.mpg was further analysed with the previously referenced application MediaInfo. On this occasion MediaInfo recognised the carved.mpg file as a MPEG-PS stream, but it refused to provide any details of codec used for encoding. Indeed, it appeared that the Ganz



DVR uses a proprietary codec for encoding and storing the video streams in the DVR file type. Further research was required into the actual data streams, it was elected to analyse the first 32 bytes of the start of each video stream for two specific channels (channel 1 & 2) at a specific time (08:00hrs) over a 4 day period spanning a month change to provide a reference guide to decoding the file header.



**Figure 4:** Video Player MFC showing carved data file

#### 4.2.5 Video File Header Investigation

In this experiment, we extract the first 32 bytes of the video stream for channel 1 & 2 at 0800hrs on the 30/03/2013, 31/03/2013, 01/04/2013 and 02/04/2013 (Figure 5). In order to establish whether offsets 30 and 31 (the last two bytes: 00 80) provide a time stamp, we retrieve data from the camera 1 on the 30<sup>th</sup> of March 2013 for each hour recorded between 0800hrs on the 30<sup>th</sup> until 0000hrs on the 31<sup>st</sup>. We then find that Offset 30 and 31 was linked to the time, it was clear that the Ganz DVR recorded the hour directly until it reached 1600hrs. At that point it resets offset 31 to 00 but increased offset 29 by 1. In this first test, we only focus on hours: 08:00, 09:00, 10:00hrs etc. There was no provision for minutes or seconds to be accounted. In order to locate the minute change additional data was retrieved splitting out the 08:00hrs timeframe for the 30<sup>th</sup> of March 2013. In reviewing the results we notices that offset 31 did not remain at 80 as expected if it was to indicate 08:00hrs. Neither was offset 30 remaining constant with the time stamp for minutes. Rather offset 30 appeared to initially be recording seconds within the first minute 08:00 – 08:01hrs. This appeared to be in direct hex notation, i.e. 00 – 3B (Dec 00 to 59). However at the beginning of 08:01hrs, offset 30 became hex 40. Then at the beginning of 08:02hrs, offset 30 became hex 80. Then at the beginning of 08:03hrs, offset 30 returned to hex C0. Finally at 08:04hrs offset 30 reset to 00. It was noted that offset 31 increased by 1, from 80 to 81. In visual form offset 30 represented:

Hex 00 – 3B = 0 to 59 Sec, Minute 0  
 Hex 40 – 7B = 0 to 59 Sec, Minute 1  
 Hex 80 – BB = 0 to 59 Sec, Minute 2  
 Hex C0 – FB = 0 to 59 Sec, Minute 3

The analysing of headers is shown in Figure 6. The difference between the starts of each cycle was noted as decimal 64. (64, 128, 192). The value in seconds could be established from the hex code in the following manner:

Example: offset 30 = hex FB = Dec 251  
 Dec 251 / 64 (cycle) = 192 r 59  
 Dec 192 / 64 = 3  
 Therefore Hex FB = 3<sup>rd</sup> Minute cycle, 59 seconds

Offset 31 was clearly recording two separate portions of data. Bit 1 was the direct hex value of the hour. Bit 2 was a record keeper of how many cycles offset 30 had completed within that hour.

Example: offset 31 = hex 8E  
 Bit 1 reads hex 8 = dec 8, therefore 0800hrs. Bit 2 reads hex E = dec 14, there 14 x 4 minute cycles = 56  
 Time stamp would read: 08:56hrs + value from offset 31

30<sup>th</sup> March 2013 – 0800hrs

|          |  |
|----------|--|
| Camera 1 | 00 00 E1 01 9A 09 C0 87 3B 17 0D D2 BD A6 D2 1B<br>36 0D 0F 3D 80 8B A0 5F 19 80 0A A0 FC 34 00 80 |
| Camera 2 | 00 00 E2 01 DA 09 C0 87 3B 17 0F D2 5D 33 D2 1B<br>C2 0D 0F DD 80 8B 00 4C 19 80 0A E0 FC 34 00 80 |

31<sup>st</sup> March 2013 – 0800hrs

|          |  |
|----------|--|
| Camera 1 | 00 00 E1 01 3A 25 C0 87 3B 17 03 10 AD 98 10 1B<br>28 03 0F 2D 80 8B 80 5D 4D 80 09 40 FE 34 00 80 |
| Camera 2 | 00 00 E2 01 BA 29 C0 87 3B 17 03 10 CD B4 10 1B<br>44 03 0F 4D 80 8B 40 4D 59 80 09 C0 FE 34 00 80 |

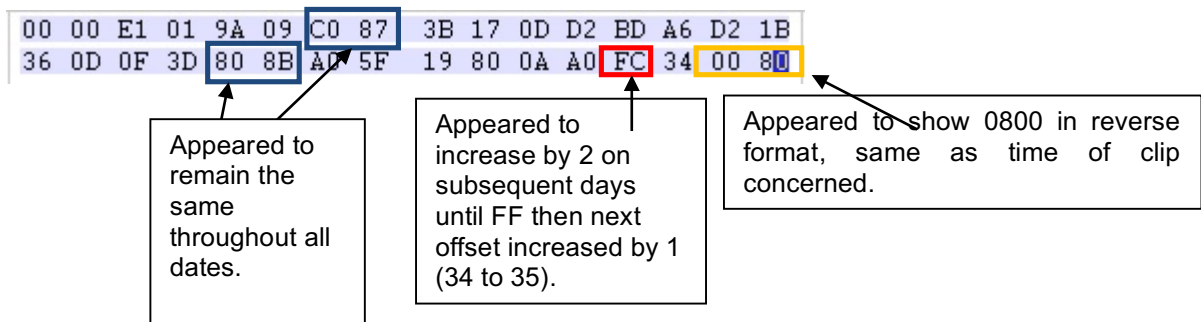
1<sup>st</sup> April 2013 – 0800hrs

|          |  |
|----------|--|
| Camera 1 | 00 00 E1 01 5A 23 C0 87 39 17 F7 4D BD A8 4D 19<br>38 F7 0F 3D 80 8B E0 69 4E 80 09 60 02 35 00 80 |
| Camera 2 | 00 00 E2 01 BA 29 C0 87 39 17 F7 4D DD C4 4D 19<br>54 F7 0F 5D 80 8B 60 4B 59 80 09 C0 02 35 00 80 |

2<sup>nd</sup> April 2013 – 0800hrs

|          |  |
|----------|--|
| Camera 1 | 00 00 E1 01 FA 24 C0 87 37 17 ED 8B F9 96 8B 17<br>26 ED 0F 79 80 8B E0 0D 4D 80 09 00 04 35 00 80 |
| Camera 2 | 00 00 E2 01 DA 29 C0 87 37 17 ED 8B 19 B3 8B 17<br>42 ED 0F 99 80 8B 00 4D 59 80 09 E0 04 35 00 80 |

**Figure 5.** The first 32 bytes of the video stream for channel 1 & 2



**Figure 6.** Video stream – Header Analysis

Next, looking at an overall example where we extract a video header from Sector 696,662,102 (Figure 7). If the above is to be proven then the video footage in Sector 696,662,102 should relate to footage taken at 08:32:42hrs. So we reverse the lookup method for sectors from date. We have sector 696,662,102 = Hex 29 86 38 56, transposed into little endian for search = 56 38 86 29 produced (Figure 8).

00 00 E0 01 5A 03 C0 87 3B 17 27 FC DD F9 FC 1B  
89 27 0F 5D 80 8B C0 0B 0B 80 02 60 FC 34 2A 88

offset 30 shows:  
Hex 2A = 42  
42 / 64 = 0 r 42  
Therefore hex 2A = 0 minutes and 42 seconds

offset 31 shows:  
Bit 1 – hex 8 = Dec 08  
Bit 2 – hex 8 = Dec 08

Therefore time stamp is: 0800hrs + 8x4 minute cycles  
= 08:32hrs + 42 seconds (value from offset 30)  
= 08:32:42hrs

**Figure 7.** Video header from sector 696,662,102



034820E7F0 00 BD 56 38 86 29 00 00 0D 03 1E 08 20 2A 00 02

↑ ↑ ↑ ↑ ↑

13 03 30 08 32 42

**Figure 8.** Footage Analysis

This result established that the footage in sector 696,662,102 was for the 30/03/2013 at 08:32:42hrs as per the example workings above and demonstrates the time function has been correctly deciphered. Next, we are looking at the date stamp artefacts. Based on the time stamp analysis it was evident that the date stamp was somehow linked with offsets 28 and 29, and that 16:00hrs played a crucial role in increasing the offset 28 counter. In order to establish the system used for the date stamp, we take the video footage from Camera 1 at 0000hrs and 1600hrs on the 30<sup>th</sup> March, 31<sup>st</sup> March, 1<sup>st</sup> of April and 2<sup>nd</sup> of April. This selection would provide four day changes and one month change. (Figure 9)

|                                       |  |
|---------------------------------------|--|
| 30 <sup>th</sup> March 2013 - 0000hrs | 00 00 E0 01 3A 1D C0 87 37 17 13 68 B9 B0 68 17<br>40 13 0F 3D 80 8B 40 4A 3D 80 05 40 FC 34 00 00 |
| 30 <sup>th</sup> March 2013 - 1600hrs | 00 00 E0 01 3A 36 C0 87 31 17 0B 3C 09 97 3C 11<br>26 0B 0F 89 80 8B 20 6D 6E 80 01 40 FD 34 00 00 |
| 31 <sup>st</sup> March 2013 - 0000hrs | 00 00 E0 01 7A 0A C0 87 35 17 07 A6 35 C2 A6 15<br>51 07 0F B5 80 8B 00 4A 1A 80 02 80 FE 34 00 00 |
| 31 <sup>st</sup> March 2013 - 1600hrs | 00 00 E0 01 DA 35 C0 87 3F 17 FD 79 9D C6 79 1F<br>56 FD 0F 1D 80 8B 20 8B 6D 80 01 E0 FF 34 00 00 |
| 1 <sup>st</sup> April 2013 – 0000hrs  | 00 00 E0 01 1A 09 C0 87 33 17 F9 E3 91 F1 E3 13<br>81 F9 0F 11 80 8B C0 0F 19 80 0A 20 02 35 00 00 |
| 1 <sup>st</sup> April 2013 – 1600hrs  | 00 00 E0 01 3A 0A C0 87 3D 17 F7 B7 D1 05 B7 1D<br>95 F5 0F 51 80 8B 20 1B 1A 80 02 40 03 35 00 00 |
| 2 <sup>nd</sup> April 2013 – 0000hrs  | 00 00 E0 01 9A 1D C0 87 33 17 F1 21 91 C0 21 13<br>50 F1 0F 11 80 8B 60 4A 3D 80 05 A0 04 35 00 00 |
| 2 <sup>nd</sup> April 2013 – 1600hrs  | 00 00 E0 01 1A 06 C0 87 3B 17 EB F5 8D 84 F5 1B<br>14 EB 0F 0D 80 8B 60 0E 0E 80 02 20 05 35 00 00 |

**Figure 9.** Hex dump of Video footage of four days

We notice that the hexadecimal value present in offset 28 increases at 0000hrs and 1600hrs of each day. However the data of offsets 28, 29 and 30 do not appear to match any of the industry date stamp formats (UNIX, MS-DOS etc.) and is likely a time counter from some point set by the manufacturer. Further examination would be required to establish the date stamp encoding method.

#### 4.3 Proprietary Application vs Forensic Examination

In this section, we use DiskTools.exe to examine the Ganz DVR unit internal hard disk drive to compare with our forensic results. The DiskTools.exe application stated that the earliest available video footage recoverable was from the 18<sup>th</sup> of March 2013 at 00:48:09 hrs. (Figure 10)

| Event Start Time    | Event End Time      | Event Attributes | Channel No |
|---------------------|---------------------|------------------|------------|
| 2013/03/18 00:48:09 | 2013/03/18 00:48:09 | SYSTEM           | ALL        |
| 2013/03/18 01:00:00 | 2013/03/18 02:00:00 | SYSTEM           | ALL        |
| 2013/03/18 02:00:00 | 2013/03/18 03:00:00 | SYSTEM           | ALL        |
| 2013/03/18 02:28:12 | 2013/03/18 03:28:12 | SYSTEM           | ALL        |
| 2013/03/18 03:00:00 | 2013/03/18 04:00:00 | SYSTEM           | ALL        |
| 2013/03/18 04:00:00 | 2013/03/18 05:00:00 | SYSTEM           | ALL        |
| 2013/03/18 04:28:45 | 2013/03/18 05:28:45 | SYSTEM           | ALL        |
| 2013/03/18 05:00:00 | 2013/03/18 06:00:00 | SYSTEM           | ALL        |
| 2013/03/18 06:00:00 | 2013/03/18 07:00:00 | SYSTEM           | ALL        |
| 2013/03/18 07:00:00 | 2013/03/18 08:00:00 | SYSTEM           | ALL        |
| 2013/03/18 08:00:00 | 2013/03/18 09:00:00 | SYSTEM           | ALL        |
| 2013/03/18 09:00:00 | 2013/03/18 10:00:00 | SYSTEM           | ALL        |
| 2013/03/18 10:00:00 | 2013/03/18 11:00:00 | SYSTEM           | ALL        |
| 2013/03/18 11:00:00 | 2013/03/18 12:00:00 | SYSTEM           | ALL        |
| 2013/03/18 12:00:00 | 2013/03/18 13:00:00 | SYSTEM           | ALL        |
| 2013/03/18 12:08:25 | 2013/03/18 13:08:25 | SYSTEM           | ALL        |

**Figure 10:** DiskTools.exe

So our forensic analysis was conducted for the date / time stamp 0D 03 12 00 30 09. The first reference appeared at Sector 63,343 which pointed to Sector 23,943,068 and finally to Sector 973,078,543 where the video data was located. The hex code for the time stamp was equated as: 0000hrs + (12x4)mins + 9secs = 00:48:09 hrs, which was identical to the timestamp expected. The footage suggested by DiskTools.exe was verified. In order to test the DiskTools.exe accuracy the previous timestamp from Sector 63,343 was taken and the pointers followed:

```
F8 95 31 6D 01 00 00 F0 0D 03 12 00 00 00 00
```

This pointed to Sector 23,933,333:

```
0C FF 8F EB E4 39 00 00 0D 03 12 00 00 00 05
```

This pointed to Sector 971,303,982

```
00 00 E0 01 1A 1F C0 87 39 17 BB 85 49 F3 85 19
82 BB 0F C9 80 8B 00 49 3F 80 01 20 E4 34 00 00
```

This had video footage present, with a timestamp reflecting 00:00hrs. In order to view whether the footage was the correct footage the data was extracted and converted using the methods within this paper. The sectors concerned were: Sector 971303982 - (10 sectors), Sector 971304138 - (6 sectors), Sector 971304220 - (6 sectors), Sector 971304320 - (5 sectors), Sector 971304398 - (5 sectors) and Sector 971304475 - (5 sectors).

The carved data was saved to the file 0000hrs.DVR and opened in the Video Player.



**Figure 11.** Carved data playback

The Video Player displayed footage from the 18<sup>th</sup> March 2013 at 00:00hrs. This footage was not retrievable through the DiskTools.exe application. In order to establish the oldest video footage available a trial and error system was deployed, starting with the attempt to retrieve footage from 2300hrs on the 17<sup>th</sup> March 2013.

Sector 23921205 contained the pointer for 0D 03 11 17 00 00 or in normal terms 17<sup>th</sup> March 2013 at 2300hrs. Sector 23921205 pointed to Sector 969083922. Sector 969083922 contained video data with the header of:

```
00 00 E0 01 FA 0A C0 87 39 17 7D 38 0D 4A 38 19
D9 7B 0F 8D 80 8B 40 4A 1B 80 0A 00 E3 34 00 70
```

This header indicates that the footage is for 23:00hrs (00 70 with the +16hrs from E3). The next trial and error search was conducted for the 16<sup>th</sup> of March 2013 at 2300hrs. Sector 23630109 contained the pointer for 0D 03 10 17 00 00 and pointed towards Section 916459161. However Sector 916459161 contained video data without a header:

```
E6 05 F0 00 3B 2F 35 DE 82 81 8C 1E 4D 79 DC BB
07 1E 35 5D B7 DB EB 22 06 1C 38 24 CC 0C 3E 74
```

This data is clearly not a start of a video stream as the 00 00 E0 01 starting bytes are missing. Further reading showed that Sector 916459161 contained video data from a stream that started in Sector 916459157, which contained a timestamp for 12:53:51 hrs. Continuing the trial and error method established that the oldest footage available was from 20:00hrs on the 17<sup>th</sup> of March 2013. Therefore an additional 4hrs and 48 minutes of footage was available through manual examination of the Ganz DVR hard drive, compared to the official DiskTools.exe application. In criminal investigations any additional time recoverable may result in crucial evidence, which if reliant on the official application would not have been recovered. Why the official tool missed these 4hrs and 48 minutes was concerning so further investigation was conducted.

## 5. Conclusion and Future work

In this paper we proposed a new method of reverse engineering the proprietary file system of a CCTV Digital Video Recorder. By conducting the examination as shown in this paper the file system has been reversed engineered with no initial knowledge, applications or directions available. Further it has been reversed engineered to a sufficient degree to allow for the identification and retrieval of video footage from any specified camera for any specified date recorded without the use of any proprietary applications. Whilst this paper was unable to evidence how to decode the date stamp from within the actual video footage data, this is a redundant step as the date and time stamps are available from the first two locators as referenced in Section 4. Further research into the actual video data stream may reveal how the date stamp is recorded, this would assist when presented with a raw partial data stream was provided for examination i.e. no locator data is available due to damaged hard drive etc. Overall the results are directly transferable to any CCTV Digital Video Recorder system that uses the AvTech file system. With minor alterations the process contained within this paper can be utilised with any proprietary file system. We are also looking at combining similar approaches in Vehicle Forensics (Jacobs 2016) and Mobile Device Forensics (Faheem 2015, Sgaras 2015) to improve our method.

## References

- Ariffin, A., Slay, J., Choo, K-K. (2013), Data Recovery from Proprietary Formatted CCTV Hard Disks Digital Forensics, Chapter in Advances in Digital Forensics IX, Volume 410 of the series IFIP Advances in Information and Communication Technology pp. 213-223
- Poole, N.R., Zhou, Q. and Abatis, P. (2008), Analysis of CCTV digital video recorder hard disk storage system, Digital Investigation, vol.5, no.1, pp. 85-92, May 2008.
- McKemmish R. (1999) What is forensic computing? Trends & Issues in Crime and Criminal Justice 1999;118:1-6.
- Dongen, W. S. V (2008) Case Study: Forensic Analysis of a Samsung digital video recorder, Journal of Digital Investigation, vol. 5, pp. 19-28, 2008.
- Wang, W. (2009), Digital Video Forensics, PhD. Thesis, Dartmouth College, New Hampshire, USA, June 2009
- Han, J., Jeong, D. and Lee, S. (2015) Analysis of the HIKVISION DVR File System, Digital Forensics and Cyber Crime: 7th International Conference, ICDF2C 2015
- Tobin, L., Shosha, A., Gladyshev, P., (2014) Reverse engineering a CCTV system, a case study, Digital Investigation vol.11(3) pp. 179-186
- Haider, Dr. , Al-Khateeb M., (2012) Analyzing the Master Boot Record, Webspaces, 2012  
CCTVforums.com <http://www.cctvforum.com/viewtopic.php?f=56&t=24717>
- IMS Research (2012), Trends for 2012
- Zeltser, L. (2001) "Reverse Engineering Malware" <https://zeltser.com/reverse-engineering-malware-methodology/>
- Jacobs (2016) Jacobs, D., Le-Khac, N-A., Vehicle Entertainment System Forensics: A Case Study of Volkswagen Automobile, Twelfth Annual IFIP WG 11.9 International Conference on Digital Forensics, New Delhi, India, January 2016
- Faheem (2015) Faheem, M., Kechadi M., Le-Khac, N-A., The State of the Art Forensic Techniques in Mobile Cloud Environment: A Survey, Challenges and Current Trends, International Journal of Digital Crime and Forensics (IJDCF), Vol 7(2) p.1-19
- Sgaras (2015), Sgaras C., Kechadi, M-T., Le-Khac, N-A. Forensics Acquisition and Analysis of Instant Messaging and VoIP Applications, Computational Forensics, Springer International Publishing, 2015 p.188-199