UNIVERSITY OF AMSTERDAM
SYSTEM AND NETWORK ENGINEERING

The forensic reliability of recorded video images on digital video recorders by Dahua Technology

Iwan Hoogendoorn
iwan.hoogendoorn@os3.nl

Rory Breuk
rory.breuk@os3.nl

May 15, 2012

Contents

1 Introduction to camera security systems and digital video recording technology 5

1.1 Camera security - the past . . . 5

1.2 Camera security - the present . . . 5

1.3 Camera technology/methods . . . 5

1.4 Digital video recording technology/methods . . . 5

2 Research 6

2.1 Research Question . . . 6

2.2 Method/approach . . . 7

3 Setup & data access 8

3.1 Setup . . . 8

3.2 Data Access . . . 8

3.2.1 Direct access . . . 9

3.2.2 Client software access . . . 10

3.2.3 Web browser access . . . 11

3.2.4 Telnet access . . . 11

3.3 Extended digital video recorder specifications . . . 12

3.3.1 Operating system . . . 12

3.3.2 Processor . . . 13

3.3.3 Memory . . . 14

3.3.4 Chips . . . 14

4 Data sources & data analysis 17

4.1 Data sources . . . 17

4.1.1 Flash memory . . . 17

4.1.2 Hard disk . . . 18

4.2 Data . . . 18

4.2.1 Recorded video files . . . 18

4.2.2 Configuration . . . 19

4.2.3 Logs . . . 21

4.2.4 Time stamps . . . 22

5 Forensic Analysis **24**
5.1 General manipulation of recorded video data 24
5.2 Risks 24
5.2.1 Manipulation risks 24
5.3 Reliability of recorded images of the DH-DVR0404LE-AS 26
5.3.1 Getting rid of the evidence 26

6 Conclusion **27**
6.1 Answers to the research question 27
6.1.1 Research question 27
6.2 Improvement 28
6.3 Further research 29

A Turning the digital video recorder off through Telnet **30**

B Log database outtakes **31**
B.0.1 User table 31
B.0.2 Mode table 31
B.0.3 Operate table 32
B.0.4 Log table 32

Abstract

This paper will give some background information on digital video recording technology used in the past and present. It describes the digital video recording hardware that was used to investigate if manipulation of recorded video data is possible, and if this can be detected. The investigation of the digital video recorder resulted in a few interesting aspects which included open shell access with the Telnet protocol. Also notable is that manipulating log files is fairly easy and the system time can be adjusted before and after editing system files, which makes time stamping basically unreliable. Besides the unreliability of the time stamping, the digital video recorder uses proprietary protocols that are unencrypted. Our main conclusion is that although the digital video recorder has unsafe vectors, manipulation is only possible by deleting the full hard disk or by switching off the system.

Introduction

Forensic video image analysis is commonly used for the purpose of identification and comparison of objects displayed, including suspect faces, guns, weapons, clothing, vehicle descriptions, license plates and anything else of relevance in criminal cases. Frequently, forensic video image analysis is used to identify or compare suspects captured on video, usually in the process of committing a crime or in the background of a video or surveillance image.

Some projects in the past focused on the possibilities to extract useful video footage from its storage media without access to the manufacturer's application software[**related_research_1**]. Other projects focused on the reverse engineering a specific digital video recorder *Samsung Digital Video Recorder* in order to find or recover "lost" video data and find explanations why video images does not reside on the hard disk anymore[**related_research_2**]. While these projects are related to digital video recording, we did not find publications about the reliability of stored recordings on the storage media of the recording device.

With the use of recorded video image files (that are analyzed by forensics) used as proof in courtrooms worldwide, it is important to have a waterproof procedure and analysis techniques to determine the reliability of recorded video images.

Goal

This research is determined to find, and test methods in order to verify the reliability of digital recorded video images. In order to execute our test methods, we will use hardware that is provided by a Dutch camera security company named *Camera Installatie*[1]. The methods described, and tested, will be applicable to do research on other digital video recorder brands and models.

Motivation

Because digital video images can prove guilt or innocence in the courtroom, we find it of importance that the reliability of these digital video images is guaranteed.

This research was done in a period of four weeks using two full days per week. With this limited time frame, it was not possible to do a full analysis on video data itself in order create techniques for manipulation of the data. We only focussed on specific methods that can be used on the hardware and software level, to determine the reliability of recorded video files that are retrieved from the digital video recorder.

[1] `http://www.camerainstallatie.nl`

Scope

The scope of this research lies on investigating a specific type of digital video recording device. We will try to determine how reliable the recorded video files are, if it is possible to manipulate this data and put it back on the storage media of the digital video recorder. We will also try to find methods to detect manipulation.

Before we can investigate the reliability of the recorded video data, we will need to know how this specific digital recording device works, what the specifications are, how and where the recorded video data is stored and how this device works low-level. The recorded video files will be investigated partly and we will look at the recorded format, timestamps and other properties that can be of use to discover manipulation.

Chapters

After this introduction chapter, there are three main sections in this report:

Part 1 THE RESEARCH Chapter 1 *Introduction to camera security systems and digital video recording technology* provides a general introduction to the recording for security purpose and is followed by chapter 2 *Research* that will define the research question, sub questions and the method/approach we used for our research. Chapter 3 *Setup & data access* is the part where we describe our base setup, investigate the hardware and look at possible data access methods. Chapter 4 *Data sources & Data Analysis* will provide details about the storage devices that the digital video recorder uses to store the operating system, log files and the recorded video files. This chapter will not only explore the storage devices but will also describe the properties of the log files and the recorded video files.

Part 2 A FORENSIC VIEW Chapter 5 *Forensic risk analysis* describes the risks that we found during this research. We also explain in what conditions a certain risk applies and how this will impact a court case. We also discuss mitigation of these possible risks.

Part 3 OUR CONCLUSION The last chapter, chapter 6 *Conclusion* talks about the conclusions that we drew after doing this research, what the answers are to the research question and its sub questions asked in chapter 2, and where there is room for further research.

Chapter 1

Introduction to camera security systems and digital video recording technology

1.1 Camera security - the past

In the past, analog cameras were placed at building entrances and other high-security locations. These cameras sent analog video signals over a coaxial cable to a VCR (Video Cassette Recorder) that recorded the images onto VHS (Video Home System) tapes.

Managing tapes was labor intensive and prone to human error. For every VCR in operation, companies needed to store several tapes (one for each day of the month). It is not hard to imagine that a month of video data from all security cameras would already require an enormous amount of tapes.

Security guards had to physically visit each building daily to verify that the recorders were operating and to replace the old tape with a fresh one. Forgetting to press the record button meant a day of not recorded video, and the risk that an organization would have no video evidence to investigate a possible incident.

1.2 Camera security - the present

Various types of camera security systems are available. There is the traditional CCTV (Closed Circuit Television) where the images are captured by the camera and transferred trough a coaxial cable towards devices capable of displaying the live video images (computer monitors or televisions) and digital recording devices. Besides the CCTV systems there are also IP based camera security systems, where camera images are transferred trough computer networks.

1.3 Camera technology/methods

Whether a camera is a CCTV or a network camera, a number of options can be available. Camera resolution, zoom, pan and tilt functionality, all belong to the most common options in order to achieve a safe and secure camera surveillance system.

1.4 Digital video recording technology/methods

As for digital video recorders, the more advanced the software, the more functionality it offers, but also the more expensive it is. Software options like "only record when there is movement" or "only record when an alarm signal is triggered" are often standard options. Besides these options, other options can be available like external storage, which can be used to store data for a longer period of time.

Chapter 2

Research

2.1 Research Question

As already became clear in the previous chapter, this research will be about investigating the reliability of digitally recorded video files that are retrieved from a digital video recorder. Therefore, the following research question therefore was defined:

> *In what way is it possible to detect manipulation of video data used in modern digital video recording security systems?*

Sub questions

To help answering this question, a few sub questions have been defined below.

What is the low level operation of the digital video recorder during and after recording video files?

Before the reliability of recorded video files can be determined it is important to know how the digital video recorder works.

What options are available to retrieve digital recorded video files?

After we determined the low-level operation of the digital video recorder, and discovered how recorded video files are stored, and in what way these files are stored, we need to retrieve the files in order to investigate if manipulation is possible and therefore loses reliability .

What methods are there to upload manipulated digital recorded video files?

When recorded video files are retrieved successfully, we will verify the options to put them back on the storage media of the digital video device.

What are methods to verify if recorded video files are manipulated, and can or cannot be used as evidence in the court room?

Regarding to whether it is possible to upload "manipulated" video files as if they were recorded by the digital video recorder itself, we still need to know how it can be determined if manipulation occurred. This can be done by looking at the digital video recorder itself (operating system level), but also by looking for elements within the recorded video files that can be used to prove or disprove manipulation.

2.2 Method/approach

The approach we took for conducting this research is described below.

(a) Select a video camera and a digital video recorder to conduct research with.

(b) Investigate the specifications and operation of this digital video recorder.

(c) Investigate data access methods that are required to watch and/or retrieve recorded video data.

(d) Investigate data sources where digital video recorded data is stored on.

(e) Investigate digital video recorded data itself.

(f) Investigate logging possibilities on the digital video recorder.

(g) Preform a risk analysis and risk classification on possible hardware vulnerabilities found and see what is applicable during a court case.

(h) Preform a risk analysis and risk classification on possible software vulnerabilities found and see what is applicable during a court case.

(i) Draw conclusions and advise on hardware and software improvements of the digital video recorder in order to improve reliability of the recorded video files.

Chapter 3

Setup & data access

3.1 Setup

The digital video recorder we used for this research is of the brand Dahua[1] and the type is DH-DVR0404LE-AS[**hardware_used_** The analog video camera that we used is of the same brand and the type is DH-CA-DW480DP[**hardware_used_2**]. No manual is available for both devices, except for the data sheets.

All possible connections to and from the DH-DVR0404LE-AS can be seen in figure 3.1.

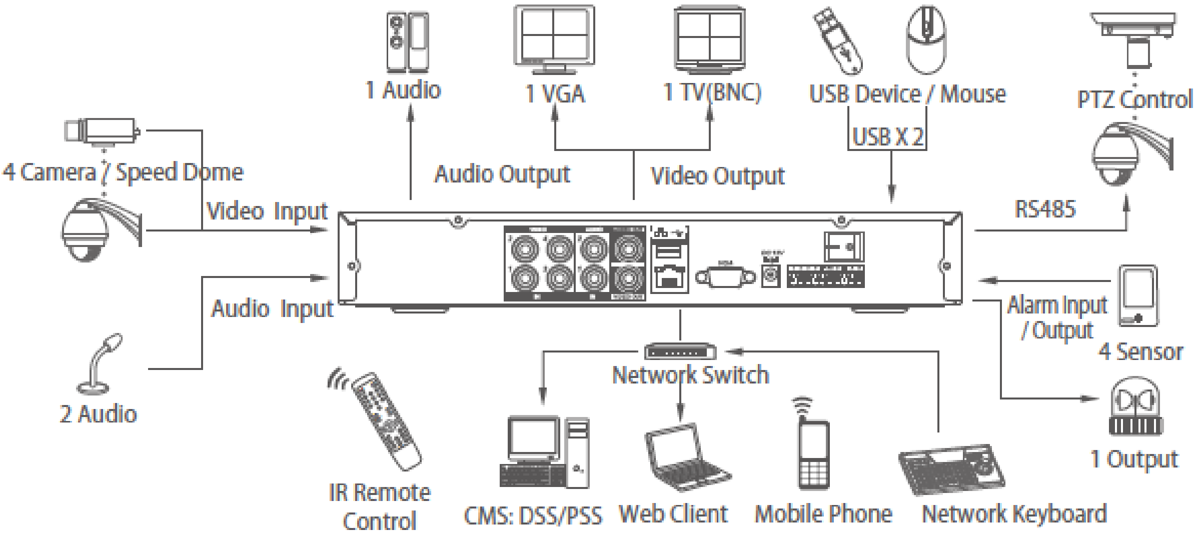

Figure 3.1: DH-DVR0404LE-AS connections[**hardware_used_1**].

We connected the camera with the use of an Bayonet-Neill-Concelman (BNC) connector[**bnc_connection**] and a coaxial cable to one of the four available inputs on the digital video recorder (video input). The power for the camera was supplied by a PoE (Power over Ethernet) adapter. In order to see the live video image that was captured by the camera, we used a simple VGA flat screen TFT computer screen (video output 1 VGA). A mouse was used to provide access to the graphical user interface when we directly access the digital video recorder (USB Device/Mouse).

3.2 Data Access

The digital video recorder offers several ways to interact with it. There is direct access, web browser access, telnet access and access with client software that the digital video recorder is shipped with.

[1] http://www.dahuasecurity.com

3.2.1 Direct access

With the connected mouse and VGA flat screen we were able to log in to the digital video recorder. As we did not have the login credentials, we guessed them. The software that provided us the login screen offered the last entered usernames. One of these usernames was "888888". When we selected this username we tried several passwords like "password", "admin", "Admin" and "888888". The last entered password was successful and we had access to the system. With access to the system we started our investigation to explore the possibilities.

We soon discovered that we had full administrative rights and we listed the most important settings below.

(a) View the recorded video images.

(b) View and delete the logging messages.

(c) View, modify, create and delete the system user accounts.

(d) View and modify the recording options (alarm, detection parameters and schedule parameters).

(e) View and modify the system network settings (Network Time Protocol settings (NTP) , IP address settings, Domain Name System settings (DNS), email notification settings, external storage settings like File Transfer Protocol storage (FTP)).

(f) View and modify the encoding settings (how the captured camera images are recorded).

(g) View and modify the system date and time.

(h) View and modify the disk options.

(i) Manage the hard disk (erase all data from the disk).

(j) View and modify the backup options.

(k) Reboot or switch off the digital video recorder.

We created an extra system account, which can be used for external access. Once logged out of the console, we could only view the live images that the video camera was recording.

Recording modes

During our console access session we discovered that the digital video recorder has four "recording modes".

1. "Regular" - continuous recording.
2. "MD" - only record when motion is detected.
3. "ALARM" - only record when an alarm signal is provided.
4. "MD & ALARM" - only record when motion is detected and/or an alarm signal is provided.

Recorded video files

When we wanted to watch a specific recorded video file, a list was presented with these video files. The file properties shown on the screen indicate that each video file has a length of sixty minutes. This means that a full day of video recording consisting of 24 hours was split into 24 separate video files of each sixty minutes.

3.2.2 Client software access

Another way of interacting with the digital video recorder was to install the client software that was supplied with the recording device. This software named *PSS* or *Pro Surveillance System* offered to choose our own login credentials during the installation. These credentials were only to gain access to the client software on the computer where it was installed on, and not to gain access to the digital video recording system itself. The username that we were forced to take was "admin". When we executed this application it became clear that we needed to add our digital video recorder, give it a name, add the IP address, port number and add the login credentials in order to connect to the digital video recorder. Once connection was established, we tried some of the most important and relevant features and settings that we listed below.

(a) View the live recorded video images.

(b) View and download the recorded video images in a specific .dav file format.

(c) Convert the .dav file format images to an .avi file format and download this.

(d) Modify the password for the PSS application.

It became clear that this software is purposed to view live recorded images, view old recorded images and retrieve them in a digital format to either backup or distribute these.

Recorded video file retrieval

As mentioned before, the recorded video files can be retrieved using the PSS client software. They can be retrieved in separate files with a length of sixty minutes just as this was described in the *Direct access* subsection of this chapter. The first file we retrieved had the filename *DOWNLOAD_20120305140802_1001_1.Dav*. A logical explanation for long number in the file name could be:

- "2012" - **the start year**.
- "03" - **the start month**.
- "05" - **the start day**.
- "14" - **the start hour**.
- "08" - **the start minute**.
- "02" - **the start second**.

It is unclear what the purpose of "1001_1" is.

More details will be given about the retrieved `.dav` video format in section 4.2 in this paper.

If we choose to download the recorded file in `.avi` the PSS client software will convert the `.dav` file on the fly and store this on our local hard drive.

Communication protocol

To learn more about the communication protocol that the client software is using to connect to the digital video recorder, we used Wireshark to capture the IP packets that are sent and received. One interesting packet we've captured is the login packet, which is shown below:

```
0040        a0 00 00 28 00 00  00 00 61 64 6d 69 6e 00     ...(.. ..admin.
0050  00 00 73 6e 65 62 6f 79  00 00 04 01 00 00 00 00   ..sneboy ........
0060  a1 aa                                              ..
```

The captured packet above shows that the username and password are sent unencrypted over the wire. Because we needed to add our recording device in order to investigate the capabilities of the application, we needed to enter the credentials of our digital video recorder. We decided to enter the credentials of the account we created ourselves; username *"admin"* and password *"sneboy"*.

The digital video recorder was able to communicate with the external software using a proprietary protocol. This protocol runs on the IP TCP and UDP port 37777.

The rest of the data that we captured was protocol data between our recording device and the PSS software client. Due to time constraints, the complexity of the captured data, and not knowing how to interpret this data we decided not to reverse engineer this.

Other software

After we investigated the possibilities of the included client software, we did a search on other client software that was able to use the same protocol and do the same as PSS. This resulted in that we found a couple of software clients for various platforms that were able to use this proprietary Digital Video Recording-IP protocol.

We tested one of these applications (Linux based) named *TaniDVR*[2] that uses the command line to access our tested digital video recorder over TCP and UDP port 37777. The program can only retrieve current video streams, which can be piped to `mplayer`.

We also found a mobile application that we downloaded from the Apple App store named *DMSS*[3] and tested this with our iPhone. What became clear is that the developer of this iPhone application is *ZHEJIANG DAHUA TECHNOLOGY CO.,LTD.* the same manufacturer as the digital video recorder we are testing.

TaniDVR however is an open source application. This means that the developers of TanIDVR reverse engineered this protocol and made it possible interact with the digital video recorder. After looking at their source code we still found it difficult to understand the protocol used so we send an email for clarification to the developers, but unfortunately we did not receive any response back. Due to time constraints we decided not to focus on this, but on other parts of the digital video recording technology.

3.2.3 Web browser access

During our investigation using the direct access, we discovered an option to enable or disable web management. This was enabled by default and the default port entered was TCP port 1024. When we entered the IP address of our recording device together with the port number "1024" in our Firefox browser but nothing much happened and only a part of the web interface was loaded. We decided to try a few other browsers and discovered that Microsoft Internet Explorer offered to install an Active X plugin. Once this was installed, we were able to log in with the same credentials as we created with direct access and which we used within PSS in order to connect to our digital video recorder.

The web client offered the same possibilities as we had with the PSS client software.

Looking at Wireshark captures during the session, we noticed that the protocol that is used for communication between the PSS client software and the digital video recorder, was the same as the ActiveX plugin uses to communicate with the digital video recorder.

3.2.4 Telnet access

The digital video recorder also responded to Telnet that provided us user shell access to the recording device. The credentials were not documented, but simply by trial and error we found out that a user can log in with the username "root" and a blank password. We listed the most important things that we were able to achieve below.

(a) Reboot and switch off the digital video recorder.

(b) Change configuration of the digital video recorder.

(c) View, delete and modify the logging messages.

(d) Create new user accounts with administrative privileges.

[2] `http://tanidvr.sourceforge.net/`

[3] `http://itunes.apple.com/nl/app/dmss/id355287559?mt=8`

The next section shows a number of possibilities of the Telnet session. In the next chapter *Data sources & data analysis* we will explain in detail how data can be altered through Telnet access.

3.3 Extended digital video recorder specifications

Because the data sheet that we found on the internet did not really gave us an in-depth overview of the digital video recorders specifications we decided to use two extra possibilities to discover more about the recording device. As we now have command line access through the Telnet protocol we were able to execute some Linux based commands to discover more about the OS (Operating System), processor and memory, where the video files are stored, where the logging files are stored and any other information that is useful for our research. The second method we decided to use, is to open the digital video recorder and look at the chips used, look the hard disk, and see if there is anything else useful for our research.

To see what ports the digital video recorder has opened, we did a quick port scan with Nmap[4] (`nmap <ip>`). This showed that the digital video recorder has the following ports opened: 23, 80, 1024, 37777, 37778 and UDP port 37777, 37778.

3.3.1 Operating system

The recording device runs a Linux distribution, *HiLinux*, which runs BusyBox as a toolset. This information was given in a *message of the day* banner when we entered the Telnet interface and was confirmed with the execution of the Linux command *uname -a*.

```
(none) login: root
Password:
warning: cannot change to home directory
root login  on ‘pts/0’


BusyBox v1.1.2 (2010.05.26-05:32+0000) Built-in shell (ash)
Enter ’help’ for a list of built-in commands.

Welcome to HiLinux.
```

Information about the Linux kernel version is shown below.

```
/ $ cat /proc/version
Linux version 2.6.24-rt1-hi3515v100 (nils@localhost.localdomain)
    (gcc version 3.4.3 (release)
(CodeSourcery ARM Q3cvs 2004)) #157 Wed May 26 12:58:03 CST 2010
/ $
```

BusyBox

BusyBox combines tiny versions of many common UNIX utilities into a single small executable[**busybox**]. It provides replacements for most of the utilities usually found in *GNU fileutils*[5] and *shellutils*[6]. The utilities in BusyBox generally have fewer options than their full-featured GNU cousins. However, the options that are included provide the expected functionality and behave very much like their GNU counterparts.

BusyBox has been written with size-optimization and limited resources in mind. It is modular, which makes it easy to include or exclude commands (or features) at compile time. Therefore, BusyBox is often used to provide a default toolset in embedded systems.

Utilities like `OpenSSH` and `FTP` are excluded from the BusyBox version on the digital video recorder. This makes it harder to transfer files between a computer and the digital video recorder. Mounting a Network File System (NFS)

[4] http://nmap.org

[5] http://www.gnu.org/software/fileutils

[6] http://www.gnu.org/software/shellutils/shellutils.html

turned out to be supported. We used this to mount a folder of our local computer on the digital video recorder with the use of the Telnet session.

Power Control

It is possible to control power of the recording device in a number of ways. The system logs will only consider power control through the digital video recorders appliance software as normal (warm).

(a) Remove the power plug and plug it back in.

(b) Reboot, trough the regular digital video recorder's appliance software menu.

(c) Shutdown, trough the digital video recorder's appliance software menu.

(d) Reboot trough Telnet with the `reboot` command.

(e) Shutdown trough Telnet with the `halt` command.

When the digital video recorder is turned off trough Telnet, the video camera will still output the signal to a connected VGA monitor, meaning that the video circuit is outside the recorder circuit. This means it is possible to watch live images, without noticing that the system is not recording. It can still be spotted, because the overlay information (camera number, recording symbol, timestamp) will disappear when this happens.The difference between the images can be found in Appendix A.

3.3.2 Processor

As there is no information available about the processor by just reading the data sheet we discovered this by executing this command on the command line via the Telnet session:

```
/ $ cat proc/cpuinfo
Processor       : ARM926EJ-S rev 5 (v5l)
BogoMIPS        : 229.37
Features        : swp half thumb fastmult edsp java
CPU implementer : 0x41
CPU architecture: 5TEJ
CPU variant     : 0x0
CPU part        : 0x926
CPU revision    : 5
Cache type      : write-back
Cache clean     : cp15 c7 ops
Cache lockdown  : format C
Cache format    : Harvard
I size          : 16384
I assoc         : 4
I line length   : 32
I sets          : 128
D size          : 16384
D assoc         : 4
D line length   : 32
D sets          : 128

Hardware        : hi3515v100
Revision        : 0000
Serial          : 0000000000000000
/ $
```

The processor that the digital video recorder is equipped with is an ARM ARM926EJ-S processor[**dvr_processor**]. The ARM processor is capable to support several OSs like Linux, Windows CE and Symbian. This capability was already confirmed in the last subsection.

3.3.3 Memory

The total amount of RAM memory that the digital video recorder is using is 52 MiB.
To discover this we executed this command on the command line:

```
/ $ cat proc/meminfo
MemTotal:        53320 kB
MemFree:          2692 kB
Buffers:          6084 kB
Cached:           8748 kB
SwapCached:          0 kB
Active:          34892 kB
Inactive:         8024 kB
SwapTotal:           0 kB
SwapFree:            0 kB
Dirty:               0 kB
Writeback:           0 kB
AnonPages:       28088 kB
Mapped:           4572 kB
Slab:             3240 kB
SReclaimable:      464 kB
SUnreclaim:       2776 kB
PageTables:        408 kB
NFS_Unstable:        0 kB
Bounce:              0 kB
CommitLimit:     26660 kB
Committed_AS:    39752 kB
VmallocTotal:   458752 kB
VmallocUsed:    124492 kB
VmallocChunk:   329396 kB
/ $
```

3.3.4 Chips

When we physically opened the digital video recorder we could immediately see some chips soldered on the motherboard.

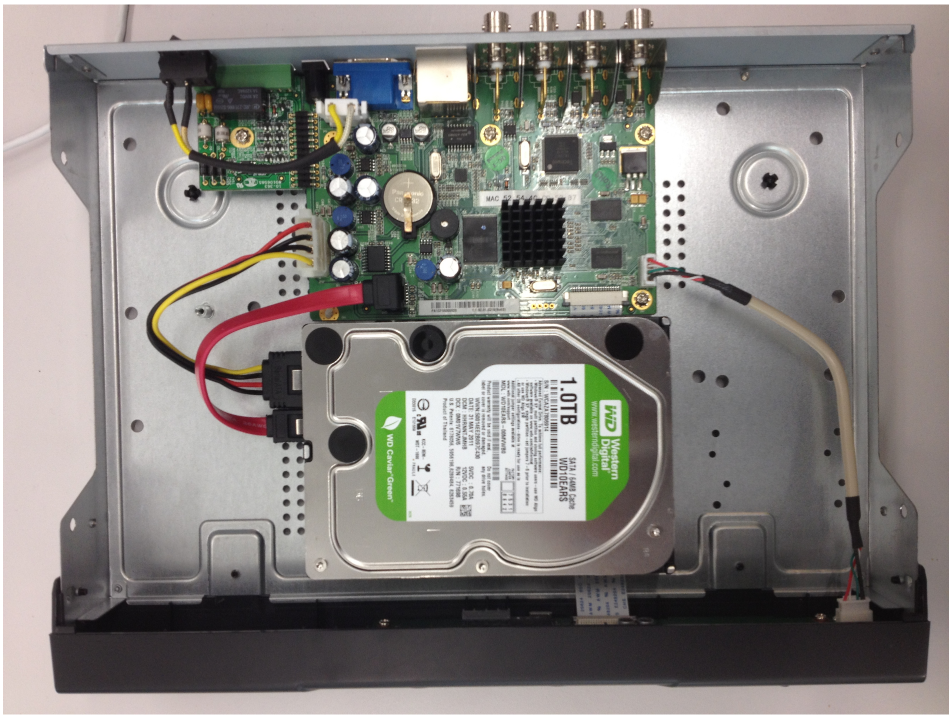

Figure 3.2: Inside the digital video recorder 1.

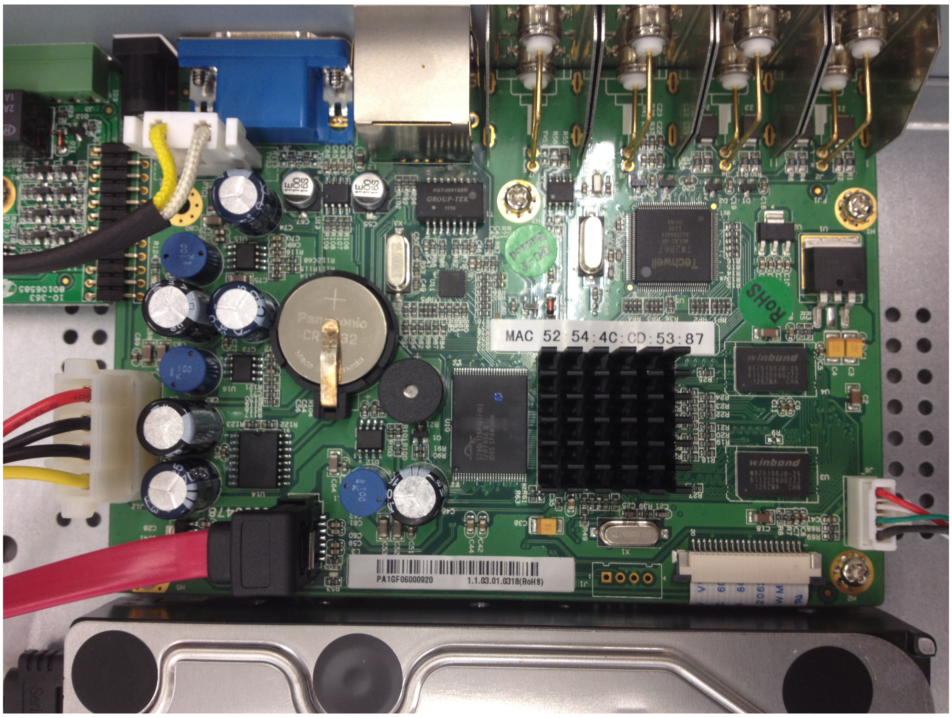

Figure 3.3: Inside the digital video recorder 2.

Spansion

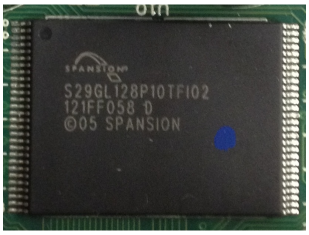

Figure 3.4: Spansion Chip.

This chip is a flash memory chip. It is available in 1 Gb, 512 Mb, 256 Mb, but as the number on the chip contains an "128", this hints that the 128 Mb version of the S29GL-P family[**chip1**] is used within our recording device.

Techwell

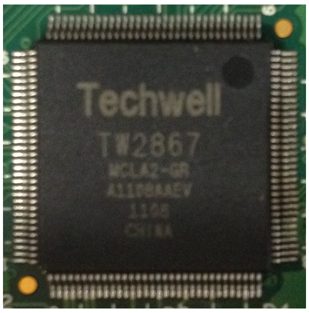

Figure 3.5: Techwell Chip.

The TW2867 is Techwell's advanced 4-in-1 video decoder and audio codec for security surveillance applications. This chip includes four high quality NTSC/PAL/SECAM video decoders that converts analog composite video signals to digital component YCbCR[7] data[**chip2**]. This is the same number of inputs for video cameras that is provided by our recording device as we have seen before in the *Setup* section.

Winbond

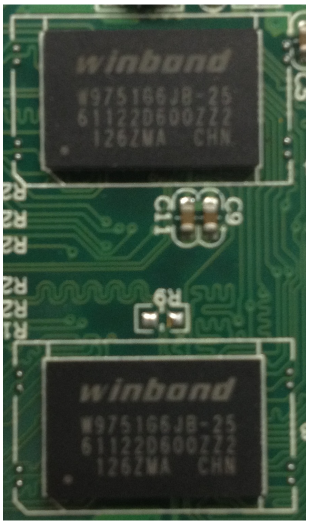

Figure 3.6: Winbond Chip.

The Winbond W9751G6JB chips are the actual RAM memory. If we take a closer look on the chip itself we see the number "-25". In the data sheet of this chip we find the information that this number "-25" is an DDR2-800, 512M Bits SDRAM[**chip3**] module.

[7] `http://en.wikipedia.org/wiki/YCbCr`

Chapter 4

Data sources & data analysis

4.1 Data sources

The digital video recorder has a number of physical sources in which it stores data. Running `mount` through a Telnet connection shows the file systems mounted on the recorder's OS.

```
$ mount
/dev/root on / type cramfs (ro)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
tmpfs on /dev type tmpfs (rw)
devpts on /dev/pts type devpts (rw)
/dev/mtdblock2 on /usr type cramfs (ro)
/dev/mtdblock3 on /mnt/web type cramfs (ro)
/dev/mtdblock4 on /mnt/custom type cramfs (ro)
/dev/mtdblock5 on /mnt/logo type cramfs (ro)
/dev/mtdblock6 on /mnt/mtd type jffs2 (rw,sync)
```

The OS has a number of standard pseudo and virtual file systems mounted: proc; sysfs; tmpfs; devpts. The recorder also has `/dev/mtdblock2`,`/dev/mtdblock3`,`/dev/mtdblock4`,`/dev/mtdblock5`,`/dev/mtdblock6` mounted in a number of directories. These are device nodes created by FTL (Flash Translation Layer) for a Memory Technology Device, which emulate block devices over MTDs[**mtdblock**]. The `/dev/root` device is a symbolic link to `/dev/mtdblock1`:

```
$ ls -l /dev | grep mtdblock1
brw-rw----    1 root     disk      31,   1 Mar 18 15:20 mtdblock1
lrwxrwxrwx    1 root     root            9 Mar 18 15:20 root -> mtdblock1
```

The digital video recorder also contains the `/dev/sda` device, which is the only registered mass-storage device:

```
$ fdisk -l

Disk /dev/sda: 1000.2 GB, 1000204886016 bytes
255 heads, 63 sectors/track, 121601 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

Disk /dev/sda doesn't contain a valid partition table
```

4.1.1 Flash memory

The only MTD we found physically on the digital video recorder was the Spansion Chip, so we assume the system's OS is stored on this flash memory chip. Except for `/dev/mtdblock6`, all the flash chip's device nodes are mounted read only and formatted as cramfs. This is a Linux file system designed to be simple, small, and to compress things well. It is used on a number of embedded systems and small devices[**cramfs**]. The /dev/mtdblock6 is mounted as readable and writable. It is formatted as a jffs2 file system, which is a log-structured file system designed for use on flash devices in embedded systems[**jffs2**].

Because only data on the /mnt/mtd mount point is writable, the system software is relatively secure from modification. Section 4.2 shows how some non-static data is prone to unwanted modification.

4.1.2 Hard disk

The recording device contains a hard disk drive (HDD), which seems to be the /dev/sda device. When the digital video recorder is booted with the HDD removed, recording is not possible and the /dev/sda device is also removed. The device's file system is not recognized by the recorder's OS, so the recording software directly communicates to the HDD.

A hex editor shows the following in the first few bytes of the HDD:

```
0000000  44  48  46  53  34  2e  31  00  00  00  00  00  00  00  00  00
          D   H   F   S   4   .   1
```

DHFS4.1 is a proprietary file system, created by Dahua[**dhfs**]. We assume this means the HDD is formatted in this format. Drivers for this file system are not publicly available, so manually mounting the drive is difficult.

We compared .dav and .avi video files, downloaded from the digital video recorder, to data on the disk (still containing the original video files). Full parts of the .dav video files could be found on the disk using the hex editor in AccessData FTK Imager. This means the file system is not encrypted. With the proper tools it could be possible to tamper data on the HDD. However, the digital video recorder OS does not provide any tools (like dd) to directly write to block devices.

Finding a way to read and write to the HDD should be possible, but can be very difficult and lies beyond the time constraints of this project.

4.2 Data

The physical data sources on the digital video recorder contain data which can be of forensic importance in a case. In the following sections we will describe this data, depict how it can be used in a forensic case and analyse how the data can be altered.

4.2.1 Recorded video files

Video files are the most important files on the digital video recorder. Video recordings of certain events can often be used as evidence in criminal cases. Timestamps on the video data can be used to depict on what time recorder events have happened.

It is very import that the video data is reliable. Video files should be properly secured, so manually changing the video files is not possible. Only when this can be assured, video files should be regarded as sound proof.

In our recording device, files can be downloaded in either .avi format or .dav format. As we showed in section 4.1.2, the digital video recorder stores the files in the .dav format. When video files are downloaded in the .avi format, the corresponding .dav file is first transcoded to .avi.

.dav

The file type is a modified version MPEG File Format[**dahua4**]. The appliance software of the video recorder is responsible for creating these .dav files. Documentation of the files are not publicly available, and the files can only be played or transcoded through proprietary software. We investigated a .dav file to find out more about possible encryption of the files.

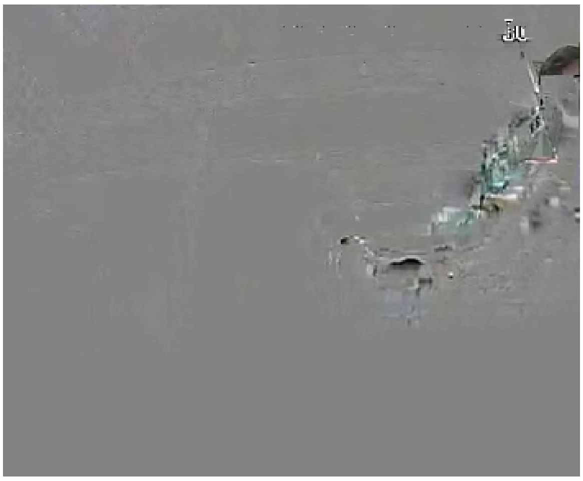

Figure 4.1: Screenshot of forced playback of the video.

Entropy

An important method for investigation encryption in files is examine the entropy of files. Proper encryption maximizes the entropy. A first easy entropy test is to compress the file. If the entropy is very high, compression will not lead to a smaller file size, the file size might even increase. Using `WinRar` we compressed the file in `rar` format. A `.dav` file of 332,431,360 bytes can be compressed to 315,920,093 bytes. The level of compression is 5.0%, which is high, if the file would be properly encrypted.

Using Cryptool, we computed the entropy of the file. The result is: 7.83 of maximum 8.00. This is also a fairly low entropy for an encrypted file. These findings indicate that the files are probably only encoded, or have bad encryption (for instance, ECB mode).

Forced playing the file as MPEG with `mplayer` generally shows a grey picture. Every now and then, especially when there's movement in the image, recognizable formations can be seen in the image. An example for this can be seen in figure 4.1. The image clearly shows a part of the timestamp of the frame and the contours of a person's head. These findings supports our suspicion about the encryption of the video files.

We also expect the system's weak processor will be unable to record high resolution video streams on four different channels and encrypt the images, all on a continuous data stream.

Timestamping on .dav files

All frames in the recorded video files have a timestamp placed on the picture. This timestamp can be very important in proving when a certain recorded event has happened. The timestamp is taken from the system time of the digital video recorder. Manually changing the system time will, after a few seconds, change the time on the recordings. Besides changing the system time through the digital video recorder software, it is also possible to change the system time through a Telnet session by using the `date` command.

Figure 4.2 shows a screenshot of the recorded video, before changing the system time and after changing the system time.

4.2.2 Configuration

The digital video recorder's configuration is not of direct importance as evidence for a criminal case. However, they might open doors for an attacker to obfuscate an event. Changing settings may give the possibility for an attacker to prevent the system from recording at certain moment, alter camera settings to change the direction of cameras and change user settings in order to increase control of the recorder.

It is of great importance to try to find out whether configuration files have been altered. This can be through unexpected timestamps (as we will describe in section 4.2.4), events in log files (as we will describe in section 4.2.3), or shadowing the settings outside the digital video recorder. With the latter option, expected settings can be compared to the recorder's configuration after a criminal event, in order to depict unwanted changes to the system. Because settings can also be changed to its original, this method can only show whether configuration has changed, but it cannot prove that it has not.

Settings of the recorder can either be accessed through the methods described in the previous chapter, or directly with a Telnet session. Configuration files are stored in `/mnt/mtd/Config`. As we showed in section 4.1, the file system mounted in `/mnt/mtd/` is writable. A directory listing is shown below:

```
$ ls -l /mnt/mtd/Config/
-rw-r--r--    1 root     root          8408 Mar 26 17:35 Config1
-rw-r--r--    1 root     root          8408 Mar 26 10:54 Config2
---------x    1 root     root             4 Mar 26 17:35 HDOutPutFormatConfig
-r--------    1 root     root             4 Mar 26 17:35 VideoStandardConfig
-rw-------    1 root     root            48 Mar 26 10:54 dhcp_cfg_cli
-rw-------    1 root     root           547 Oct 19 18:27 group
-rw-r--r--    1 root     root           141 Mar 26 17:35 network
-rw-------    1 root     root          1032 Mar 26 11:12 passwd
drwxr-xr-x    2 root     root             0 Jan  1  1970 ppp
-rw-r--r--    1 root     root            38 Mar 26 17:35 resolv.conf
```

All files in this directory are writable. They have the following purpose:

- Both `Config1` as `Config2` are gzip compressed files and contain all configuration options which are accessible through the recorder's appliance software.
- The files `HDOutPutFormatConfig` and `VideoStandardConfig` are, as the name suggests, used for some video configuration settings. These files are encoded, so they cannot easily be altered.
- The file `group` contains authorization information for different user groups.
- The file `network` contains network settings for the recorder.
- In `passwd`, user credentials and authorization settings for software is stored. This file also stores authorization information for each user.
- `ppp` is a directory, containing 4 files which can be used for configuring Point-to-Point Protocol settings.
- Finally, `resolv.conf` contains IP addresses of name servers used by the recorder.

Changing user credentials

User credentials and authorization settings are stored in `/mnt/mtd/Config/passwd`. A record in this file contains the following fields:

id A unique ID for each user.

name The username of the user.

passwd A hash of the user's password.

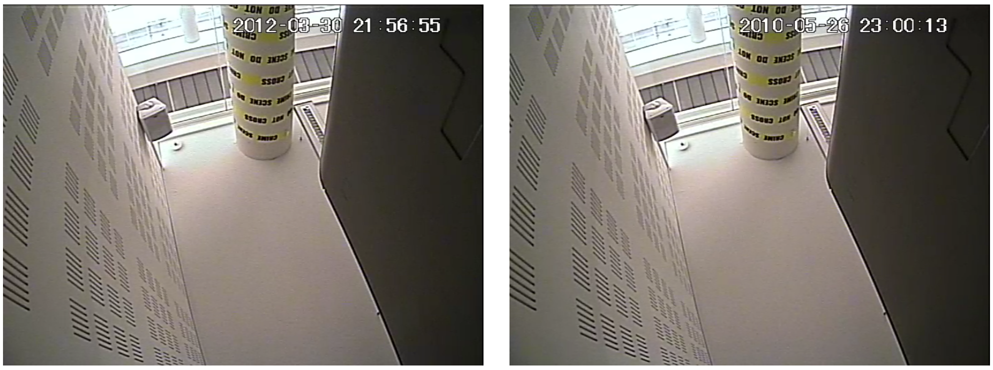

Figure 4.2: Screenshot of the video stream before and after changing the time.

groupid The ID of a group (defined in /mnt/mtd/Config/group).

authority A comma separated list of authority of the user. All values stand for certain access levels withing the software for the user.

memo A memo about the user.

share Either 0 or 1, stating whether the user settings can be duplicated for a new user.

The length of all passwords is constrained to be six characters or less. Allowed characters are: [0-9][a-z][A-Z]. This means that the amount of possible passwords is only:

$$\sum_{i=1}^{6} 62^i = 57,731,386,986$$

Password hashes are stored as 8 characters, and contain the following characters: [0-9][a-z][A-Z]. This format seems to indicate that the hashes are stored using a Base64 encoding, which is often used in order to store hashes in a human readable format. Creating multiple users with the same password leads to the same password hashes, so the hashes are not (properly) salted.

We obtained the hash of the word "test", by creating a user with "test" as a password. Using a python script called Hash Identifier[1], we tried to identify the hasing algorithm used for storing the passwords. We used the script to compare the hash and the Base64 decoded value of the hash to the hashes of the same password for a number of commonly used hash algorithms. These include Adler, CRC and XOR. The tests did not give any result.

The very small password size and the small amount of possible hashes due to the hash size show a big security issue. The password can easily be brute force cracked and the search time might even decrease by the high chance of hash collisions due to the small hash size.

Engineering the exact hash algorithm used by the recording device is out of the scope of this paper, however knowing the hash type is not necessary to add users. We only need to know the hash of one known word. We found that the hash for the word "test" is S2fGqNFs. By copying the admin user and changing the necessary fields, we added a user with full authority to the file. Note has to be taken that the file is stored in memory. When the file is changed, a reboot is required before the newly created user can actually be used.

Below, an example can be found of a user that used the password test. This snippet can be manually added to the passwd file.

```
5:Example:S2fGqNFs:1:CtrPanel, ShutDown, Monitor_01, Monitor_02,
    Monitor_03, Monitor_04, Replay_01, Replay_02, Replay_03,
    Replay_04, Record, Backup, MHardisk, MPTZ, Account, Sysinfo,
    Alarm, Config, QueryLog, DelLog, SysUpdate, Control, AutoMaintain,
    GeneralConf, EncodeConf, RecordConf, ComConf, NetConf, AlarmConf,
    VideoConfig, PtzConfig, OutputConfig, DefaultConfig,
    DataFormat:Example account:1:4
```

4.2.3 Logs

Logs on the digital video recorder can be used to keep track of a number of events on the system. In the field of forensics they can be of great use in depicting and explaining unexpected events. Access of users, changes in configuration and interruption during recording are examples of events which can be intercepted in logs. A list of all events stored in the logs is given later in this section.

Care has to be taken in defining expected events and unexpected events. It is advisable to manually keep tracks of events on the recorder, so that it is easier to find key issues in the log files. For instance, manually keeping track of known interruptions in recordings can

Besides the ways described in chapter 3, log files can be accessed directly through Telnet. They are stored in /mnt/mtd/Log and are writable. The log files are:

- LogDB,

[1] http://code.google.com/p/hash-identifier/

- `LogDB_backup`

Both log files are exactly the same and are stored in the SQLite format. They store up to 1024 records of events happened in the past. Events on all access methods of the digital video recorder are logged, except for Telnet events. Events are constructed with the following data fields:

id A unique number for that specific record.

timep The timestamp for that specific record.

data Contains additional information of the specific event.

oid Operation ID is a number that corresponds with a record stored in the table "operate".

mid Mode ID is a number that corresponds with a record stored in the table "mode".

uid User ID is a number that corresponds with a record stored in the table "user".

Actions that trigger an event (or create a record in the logging database) can be found in the "operate" table. These actions are described in further detail below:

name	Meaning
Log_DelConfig	Reset configuration to default
Log_ShutDown	System is shut down
Log_Reboot	System is rebooted. Data is 0x00 for a warm reboot, 0x01 for a cold reboot.
Log_SaveConfig	Configuration is changed and saved
StartHddInfo	Information on the HDD is requested
Log_Login	A user logged in directly on the digital video recorder
Log_NetLogin	A user logged in through a network interface (Telnet is excluded)
Log_EventStart	Start of any user triggered event not in this list
Log_EventEnd	End of any user triggered event not in this list
Log_Logout	A user logged out of the system
Log_RecSearch	Search for a record is performed
Log_RecPlayback	An old record is played
Log_AddUser	A user is added to the system
Log_ModGroup	A group is modified
Log_ModUser	A user is modified
Log_DelUser	A user has been deleted
Log_ClearDir	The HDD in the recorder is cleared
Log_ManualRec	A recording is manually started
Log_ManualStop	A recording is manually stopped
Log_AutoRec	A recording is automatically started
Log_RecDownload	A recording is downloaded

The *mode* table contains possible type of log records and the *user* table contains all usernames and user IDs on the recording device. Snippets of the tables in the log files can be found in Appendix B.

By loading the log files into any SQLite program, it is easy to remove, add or change events to the logs. Although SQLite software is not available on the recorder itself, using the file transfer method described in section 3.3.1 makes it possible to prepare altered log files locally and transfer them back to the recorder.

4.2.4 Time stamps

Time stamps on the file system are the lowest level of logging on the digital video recorder. The amount of information time stamps can give is limited, but can be very valuable in forensic research. Unwanted changes to files on the recorder can be detected by comparison of expected timestamps to the real time stamp of the file. For instance, the modification time (`mtime`) of a log file is expected to be (near to) the time of the latest logged event. When this is not true, the chance that someone removed certain events in the file has to be put into consideration.

To effectively use time stamps, it is important to know when file access and changes to files are expected. As described before, this is easily determined for changes to log files. Configuration changes should be properly tracked, so expected timestamps of configuration files can be determined.

Time stamps on flash memory

The operating system of the digital video recording uses regular UNIX time stamping of files on the flash memory. This means that access time (`atime`), modification time (`mtime`) and change time (`ctime`) can be retrieved for these files. These timestamps can only be accessed through the Telnet connection.

Although inconsistency between a file's true time stamp and its expected time stamp can be used to prove the system has been tampered with, correct time stamps do not prove nothing happened. The command `touch` is enabled in Busybox, but changing the timestamp through `touch` has been disabled. As we stated before, the `date` command can be used to change the system time. When one wants to change files, it is possible to change the system time first (back in time), change log or configuration files and then change the system time back again, in order to cover up this action.

Time stamps on hard disk

We are unable to mount the hard disk drive. Looking at the HDD with a hex editor shows no clear signs of stored hashes. That's why we can't see whether the file system `DHFS4.1` stores time stamps and how these could be used in forensic investigation.

Chapter 5

Forensic Analysis

5.1 General manipulation of recorded video data

The manipulation of recorded video data in general, can be done in several ways. Below we have listed those.

(a) Retrieve files from the storage media of the recording device and modify this data and restore the manipulated data back.

- With this, all possible traces of logging need to be altered as well, including file system timestamps and possible timestamps on the recorded video files.

(b) Delete recorded files from the storage media of the digital video recorder.

(c) Break or power off the digital video recorder and eliminate the possibility to make recording possible.

All of these manipulation techniques (especially the first one) are very important for law enforcement agencies and forensic analysts to investigate. It speaks for itself that if recorded video data can be manipulated and uploaded back to the digital video recorder as it was "originally recorded" by the system this would lose all of its reliability.

5.2 Risks

The most important risk that we found, which may be of great impact on the reliability of the data, is the "open" Telnet access. Another risk is the unencrypted protocol used between client software and the digital video recorder. With this it is possible to retrieve login credentials used to manage the digital video recorder. Weak password policy As we found in section 4.2, videos are probably stored unencrypted, which will make is fairly easy to create modified video files. The most important vectors to "secure" the system are the proprietary file system *DHFS*, which makes it difficult to mount and read/write to the file system (as compared to a FAT or NTFS file system) and the lack of native support of writing to the disk through any of the interfaces.

5.2.1 Manipulation risks

We saw in earlier chapters that, with some basic computer and network knowledge, it was very easy to **(a)** retrieve the login credentials and use these to erase the complete storage media where the recorded video images are stored on and **(b)** that the logging that was stored on another storage medium could be altered as well. We could also **(c)** easily turn the digital video recorder off, causing that no images are recorded anymore. Another risk was **(d)** that the system time could be changed. As the digital video recorder by default creates separate recording files that have a length of sixty minutes, we can detect based on the filename the digital video recorder if we have a time gap. Whenever we have a time gap, for example that the digital video recorder stopped recording on the first of January 2012, and suddenly started to record on the fifth of January 2012 again, this could indicate that the digital video recorder either malfunctioned or that the power was cut off by either a scheduled or unscheduled power down.

We have created a table with all the possible risks digital video recording systems in general could be exposed to, what the possible risk levels are, and how these risks can be mitigated.

#	Risk	Description	Access Level	Risk mitigation	Forensic impact
1	Management recording device not secured with secure login mechanism.	Before management on the device can be done proper security measurements should be taken in the software.	local/remote	Make sure to select a system that uses secure login methods.	HIGH
2	Communication protocol between client/server not secure.	When client software is used for real time display, retrieve old recorded files or even remote management the communication protocol should be secure so that is not possible for everyone to listen in.	remote	Make sure to select a system that uses secure client/server communication between client software and recording device.	HIGH
3	Operating system used is commercial and not proprietary.	With the use of commercial operating systems it will be easier to find known vulnerabilities to attack the system.	local/remote	Only choose video recorders with proprietary operating systems or make sure the operating system is always patched with latest updates.	MEDIUM
4	Operating system of recording device not patched with latest updates.	Not patched operating systems will always be vulnerable for attacks.	local/remote	Always patch operating system with the latest updates.	MEDIUM
5	No encryption used for log files, user database, recorded video files.	With strong file encryption manipulation will be harder.	local	Choose video recorders that uses proper file encryption.	MEDIUM
6	Change the system date and time.	The system time is used for timestamps in the file system, and may be used in encodings, or as overlay on recorded video images	local/remote.	Make sure management access is secured, unnecessary ports are closed, operating systems are patched.	HIGH
7	Control the power of the recording device.	With the possibility to reboot or power off the system one can make sure no video evidence can be collected.	local/remote	Make sure management access is secured, unnecessary ports are closed, operating systems are patched and that the recording device is physically secured.	HIGH
8	Write & delete files to and from storage media.	When it is possible to write and delete files, recorded video files in particular, reliability cannot be guaranteed at all	local/remote.	Make sure the storage media where log files, recorded video files stored are "read only".	HIGH
9	Network connectivity to open ports.	With open ports that are not relevant and used for any specific purpose the risk level will only be increased.	remote	Scan the recording device for open ports, close these and/or use devices like devices like intrusion detectors and/or firewalls within the network.	MEDIUM / HIGH
10	Not using digital signatures (from the camera).	The use of digital signatures coming from a video camera can proof that the recorded images are recorded with that specific camera.	local	make sure to select a camera system that uses digital signatures.	LOW
11	Camera resolution not high enough.	The camera resolution should be high enough to have good quality video images.	local	Choose a camera system that capable of recording high resolution video like High Definition (HD) video quality.	HIGH
12	No email notification when storage media is running full.	When the storage media becomes full, the recording device will either stop recording, or overwrite old recordings.	local/remote	Always enable email notification for certain types of logging messages if the option is provided by the system software.	MEDIUM

Table 5.1: Risks & risk mitigation.

The items classified as HIGH are the ones that can make recorded video images very unreliable, the MEDIUM and LOW items are also very important to take into account but are no direct threat for the recorded file reliability.

5.3 Reliability of recorded images of the DH-DVR0404LE-AS

With most of the important risks having lined out we can conclude that the security measurements that Dahua took for the DH-DVR0404LE-AS digital video recorder is good enough to guarantee reliability on files recorded by this device. We can conclude this because of the following reasons:

(a) It is not possible to write to the digital video recorder's storage media where the digital video recordings are stored.

(b) It is not possible to delete (specific) selected recorded video files (of sixty minutes) on the digital video recorder's storage media where the digital video recordings are stored (it is either delete nothing or delete everything).

With this having said we can also conclude that if there is a court case where video images were recorded on the Dahua DH-DVR0404LE-AS digital video recorder, these recorded images of course retrieved by law enforcement agencies or forensic analysts are reliable enough for usage in this case.

It also has to be stated that the video files do have security issues. Although at the point of writing this paper, no methods for writing to the recording device's HDD are available, in the future new attacks might change our current view on the reliability of the video files.

There is another item, which could be an issue. With the possibility that the system date and time can be changed with the open Telnet access, the time stamping of recorded video files on the file system and that is used as overlay (in the right upper corner) on all recorded video files, basically becomes unreliable. This does not mean that the recorded video file itself is unreliable but the date/time that the system says the specific files are recorded becomes unreliable. It is possible that this could cause issues in the courtroom, but we leave the decision to the judges and jury to decide on this, whether to conclude if a crime was, or was not, committed, and if the time stamping is important enough with the knowledge this can be changed (if the digital video recorded was of the same type we tested and if this device was connected to the network) to speak a certain verdict.

We believe that if people know about this specific digital video recording system, they would be able to turn the system off or delete the storage media in order to get rid of the evidence (which is a crime as well).

5.3.1 Getting rid of the evidence

To get rid of the evidence, one could use the "open" Telnet access mechanism create a user with the proper administrative rights and use this account to use client software of choice to delete the hard disk.

With our research done and lined out what this means for forensics we will answer the research questions in the next chapter. We will also provide some advice to increase the reliability of our tested recording device and discuss possibilities for further research.

Chapter 6

Conclusion

This research was destined to investigate the reliability of digital recorded video images that were created by the Dahua DH-DVR0404LE-AS digital video recorder. In order to investigate this we needed to think of methods to do a full investigation in terms of hardware and software on this specific digital video recorder to draw any conclusions about the reliability of the recorded video files. We first looked at the information available by reading the data sheet and looked for articles on the internet about this digital video recorder. We then started to investigate the possible ways to interact with this device by locally and remotely connecting to it and ended with investigating the storage media and the recorded video files itself. This resulted in some interesting findings and conclusions.

6.1 Answers to the research question

6.1.1 Research question

In what way is it possible to detect manipulation of video data used in modern digital video recording security systems?

After investigating the possibilities (with connecting locally and remotely) on the digital video recorder, we did not manage to add video data to the storage media that the digital video recorder is using to store its recorded video files on `/dev/sda` .

The hard disk uses a proprietary file system (Dahua File System - DHFS) which makes it difficult to mount and read/write like we are used to with a FAT or NTFS file system.

A form of manipulation can be achieved using weaknesses like the open Telnet connection to either turning off the digital video recorder by issuing the `halt` command or manually gain admin privileges by manipulating the `passwd` file.

The detection of this form of manipulation could only be done by searching for time gaps between the separate recorded files by looking at the file name. When the hard disk where the recorded images are stored on is suddenly empty or no data can be found before a certain point in time this could indicate that the disk has either been wiped, or that the system settings allowed the files on the hard disk to be overwritten when the disk is full.

Sub questions

What is the low level operation of the digital video recorder during and after recording video files?

The digital video recorder will initially start with continuous recording. Settings are provided to change continuous recording to only start recording where there is movement, when an alarm is triggered or when one of those two applies. The digital video recorder is equipped with an ARM processor with 52 MB of RAM memory, and uses 128 MB of flash storage where the operating system *HiLinux* together with the toolset *BusyBox* is stored on. Within this operating system drivers are installed in order to mount a hard disk where the recorded video files are stored on. This hard disk uses a proprietary file system *DHFS*.

What options are available to retrieve digital recorded video files?

Digital video files can be retrieved with the use of the provided client software named *PSS* or *Pro Surveillance System*, or with the use of other software that is able to communicate with the digital video recorder over a proprietary network protocol on TCP/UDP port 37777. Retrieval can only be done with proper login credentials and can be done on a file by file basis. Each file has a length of sixty minutes.

What methods are there to upload manipulated digital recorded video files?

We did not find ways to upload recorded video images to the digital video recorder, however the use of an unencrypted file system and the lack of proper encryption on file level show vulnerabilities.

What are methods to verify if recorded video files are manipulated, and can or cannot be used as evidence in the court room?

Possible methods that we can use to look at file manipulation from the digital video recorders point of view are:

(a) Look at file timestamps of the file system.

(b) Look at file timestamps encoded within the file.

(c) Look for encrypted signatures used between the video camera and the digital video recorder.

(d) Look at the timestamp that is recorded as an overlay on images stored on the digital video recorder.

(e) Look at logging messages provided on the system level.

The methods mentioned above are general methods. In our case we applied them all, but we are not able to use them as:

(a) **time could be changed remotely causing the time stamp to be unreliable.**

(b) **we did not do low-level investigation on the encoding.**

(c) **we did not do low-level investigation on the encoding.**

(d) **time could be changed remotely causing the time stamp to be unreliable.**

(e) **logging could be changed remotely causing the logging messages unreliable.**

6.2 Improvement

We listed some of the items that we think could increase the reliability of recorded video files with our tested recording device below.

(a) Not connect the digital video recorder to the network and only use it standalone.

(b) Secure the proprietary protocol used to communicate between software clients and the digital video recorder.

(c) Disable Telnet connectivity.

(d) Use Secure Shell (SSH) connectivity mechanisms with strong passwords.

6.3 Further research

This paper focussed on possibilities to manipulate digital recorded video files that impact the reliability of these files on the Dahua DH-DVR0404LE-AS digital video recorder.

Although the conclusion was that manipulating the digital recorded images from the digital video recorder is not possible, there is still room for further research.

More research can be done on the proprietary protocol used between the digital video recorder and its client software by reverse engineering this. Other possibilities are to reverse engineer the proprietary file system *DHFS* that the hard disk is using to store the digital recorded video images.

The high level method that we used by (a) looking at all possible access possibilities to interact with the digital video recorder, (b) investigate the data sources and (c) investigate the data itself, can be used to investigate the reliability of other digital video recording systems as well.

Appendix A

Turning the digital video recorder off through Telnet

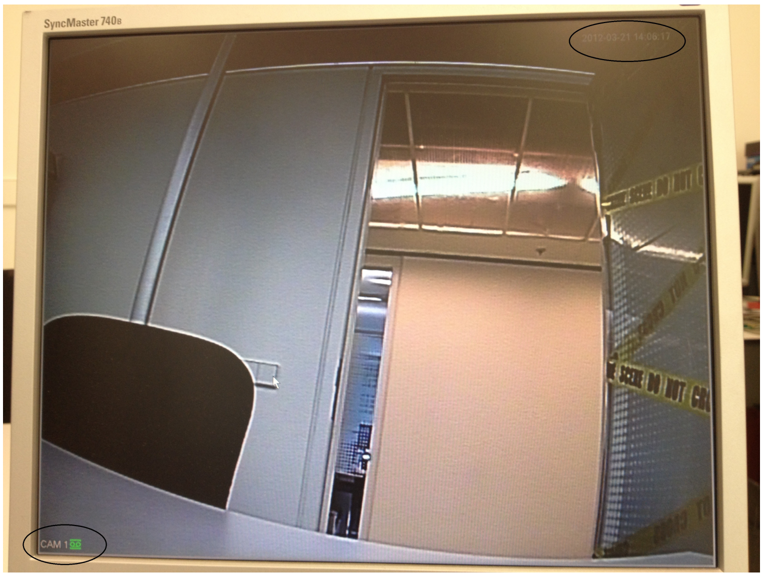

Figure A.1: Output image before digital video recorder is turned off trough Telnet.

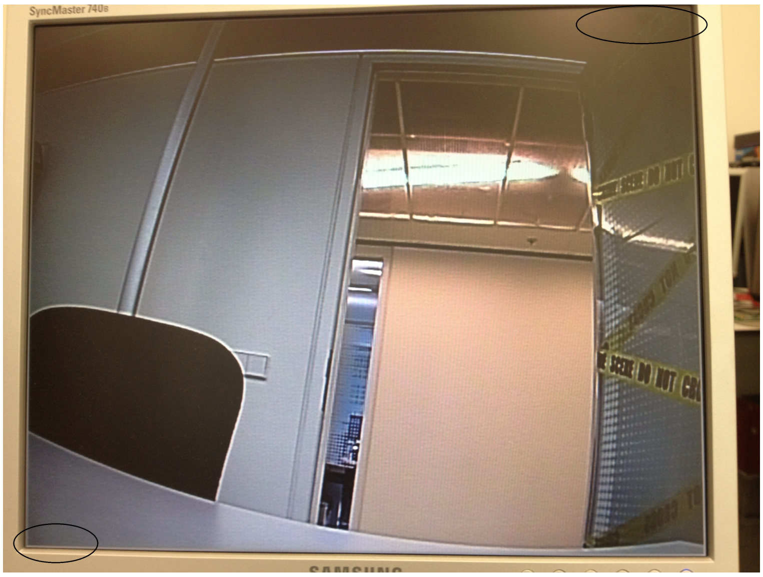

Figure A.2: Output image after digital video recorder is turned off trough Telnet.

Appendix B

Log database outtakes

B.0.1 User table

id	name
1	system
2	default
3	888888
4	admin

B.0.2 Mode table

id	name
1	Log_System
2	Log_Config
3	Log_Storage
4	Log_Event
5	Log_Recctrl
6	Log_Ugm
7	Log_Clear
8	Log_RecOpr

B.0.3 Operate table

id	name
1	Log_DelConfig
2	Log_ShutDown
3	Log_Reboot
4	Log_SaveConfig
5	StartHddInfo
6	Log_Login
7	Log_NetLogin
8	Log_EventStart
9	Log_EventEnd
10	Log_Logout
11	Log_RecSearch
12	Log_RecPlayback
13	Log_AddUser
14	Log_ModGroup
15	Log_ModUser
16	Log_DelUser
17	Log_ClearDir
18	Log_ManualRec
19	Log_ManualStop
20	Log_AutoRec
21	Log_RecDownload

B.0.4 Log table

id	timep	data	oid	mid	uid
394343	1330959001	"PlayFile" : [["Log_Index" : 1, "Recfile_Time" : ["2012-03-05 00:00:-3"], "conf_tim.rectype" : 15], ["Log_Index" : 2, "Recfile_Time" : ["2012-03-05 00:00:-3"], "conf_tim.rectype" : 15], ["Log_Index" : 3, "Recfile_Time" : ["2012-03-05 00:00:-3"], "conf_tim.rectype" : 15], ["Log_Index" : 4, "Recfile_Time" : ["2012-03-05 00:00:-3"], "conf_tim.rectype" : 15]], "RecOpType" : "Log_RecSearch", "RecPlayIP_Address" : "ugm_main.userSt_LoginLocal", "RecPlay_User" : "admin"	11	8	4
394344	1330959001	"Log_Time" : "12-03-05 14:49:57", "RecOpType" : "RecSearchStop"	11	8	4
394345	1330959091	"Log_ShutTime" : "12-03-05 14:50:54"	2	1	2
394346	1330959091	"Log_RebootFlag" : 0	3	1	2
394347	1330959096	"Log_Code" : "appEventVideoLoss", "Log_Index" : [2, 771], "Log_LinkRec" : [1]	8	4	2
394348	1330959096	"Log_Code" : "appEventVideoLoss", "Log_Index" : [3, 771], "Log_LinkRec" : [2]	8	4	2
394349	1330959096	"Log_Code" : "appEventVideoLoss", "Log_Index" : [4, 771], "Log_LinkRec" : [3]	8	4	2
394350	1330959096	"info_log.logHddInfoone" : "<1>", "info_log.logHddInfotwo" : "<1>"	5	3	2