available at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/diin

Digital Investigation

Forensic acquisition and analysis of the Random Access Memory of TomTom GPS navigation systems

Onno van Eijk, Mark Roeloffs[*]

Netherlands Forensic Institute, Department of Digital Technology and Biometry, P.O. Box 24044, 2490 AA The Hague, Netherlands

ARTICLE INFO

Article history:
Received 19 October 2009
Received in revised form
19 February 2010
Accepted 22 February 2010

Keywords:
Forensic
Memory acquisition
Memory analysis
Navigation system
GPS
TomTom
RAM
NMEA
JTAG
Linux

ABSTRACT

TomTom navigation systems are widely used, but unfortunately these devices do not store any information that can be used to link time and position on any non-volatile media. However, this kind of information can be found in the volatile Random Access Memory (RAM) of these devices. There are two methods to extract the content stored in RAM. The first method uses the available JTAG signals on the device. For the second method, a small Linux distribution is loaded into the device. Analysis of the extracted content shows some information linking time and position of the device. Although the quantity of information found at this point is limited, the possibility to link the position and time of the device can be invaluable in some cases.

1. Introduction

Driving on the motorway at night, you can observe the faint glow from a navigation system from almost every car; as if nobody knows their way home without one. The combination of the wide acceptance of these devices and the information stored on them, makes them valuable in a forensic context. The possibility of linking a device to a specific place at a specific time or determining the speed of the device at a certain point can provide critical information to forensic teams. Depending on the brand and model of the navigation system, the information required to make these links is or is not stored on any non-volatile media (van der Knijff, 2009).

In Europe, TomTom was the market leader with a market share of almost 50% in 2008 (Luttikhedde, 2008). Unfortunately, their present models do not store any information which can be used to link time and position in a readable format on any non-volatile media (Nutter, 2008). As this information is presented on the screen of the devices during normal use, the information should be available in its volatile Random Access Memory (RAM) while operating. This paper describes two methods for making forensic copies of this memory and some results on its analysis.

After noting the work of others which was used during the research in Section 2 an overview of the TomTom devices and their differences is shown in Section 3. Also, the anatomy of a TomTom navigation device is explained and the circumstances in which system power is retained are given.

In Section 4 the first method for copying the data retained by the RAM is presented; the JTAG method. This method uses

[*] *Corresponding author*. Tel.: +31 70 8886 431; fax: +31 70 8886 559.
E-mail addresses: Onno.van.Eijk@klpd.politie.nl (O. van Eijk), m.roeloffs@holmes.nl (M. Roeloffs).

1742-2876/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.diin.2010.02.005

the JTAG connector found on the mainboard of most TomTom devices to copy the content of the RAM.

Section 5 introduces the second method for reading the RAMs content. This method, called ‘TomCopy’, is based on a small Linux distribution on an SD card.

After copying the content of the RAM, the acquired image should be analysed and decoded. In Section 6 two methods of decoding the RAM image with scripts and tools are presented.

The results of both methods for copying the data retained by the RAM on TomTom navigation systems are discussed in Section 7. Both methods can be improved upon. The ideas for improving these methods are discussed in Section 8.

2. Previous work

As acquisition of the non-volatile media of TomTom navigation systems could be done using common available tools, the analysis of data stored on these was the primary focus of the forensic community and information on this topic was already available from various sources. Siezenga (2008) described the structure of the configuration file stored on the non-volatile media in dept. His description of this file was used during analysis to decode items like favourites and the home location. Nutter (2008) described the process of extracting and decoding individual location records from the non-volatile media even when they are found in any unallocated clusters or slack space.

Little is known on acquisition and analysis of data stored in the volatile memory of these navigation systems. The open source community, centered around the OpenTom wiki (OpenTom, 2008), worked on the development of alternative software for these devices. Their efforts revealed many details on the hard- and software of these devices. Especially the description of the bootloader was very helpful in this research. Breeuwsma (2006) described a method to locate and use JTAG signals found on the printed circuit boards in order to access embedded flash media. This method was used as a basis for one of the methods of creating a copy of the content of the RAM.

3. TomTom hardware overview

There are a lot of different TomTom devices. The TomTom devices are normally placed on the market in series; for example the Go 300, 500, 700 serie, the Go 510, 710, 910 serie and the Go 740, 940 serie. Between the devices in one serie the differences are for example the size of the memory, the availability of bluetooth or whether there is a hard drive or a memory card. For the Go 740 and 940 there even is a possibility for a SIM-card to be inserted in the device for receiving traffic data. Generally speaking the devices in a serie with the highest numbers have the most options and the largest memory.

Although specifications vary between the different models, the basic anatomy of all TomTom devices is the same. Build around an ARM processor and running Linux as its operating system, these devices closely resemble basic Linux based embedded system used as examples in many books (Yaghmour et al., 2008; Hallinan, 2007).

In an embedded system like the TomTom, volatile RAM is used during execution of programs. As the word ‘volatile’ implies, the RAM loses its content when it is no longer powered. The researchers performed tests on a TomTom Go 500 and Go 720 to establish when power to the RAM is lost. As a reference for the system power, the voltage present on the VCC pin of the JTAG connector (Table 2) was measured in relation to ground. When the system was powered, a voltage of 3.3 V was measured on this pin. The tests showed system power was lost on the system when:

- The battery of the device was empty.
- The reset button on the bottom of the device was pressed.

However, the system power was retained when:

- The power button was pressed in order to turn the device on or off.
- The device was externally powered (and reset was not pressed), so the battery was being charged.
- The SD card was removed or inserted.

From the above it was concluded that for a successful examination of the content of the RAM, the TomTom system must be externally powered as soon as it was brought in for investigation.

4. JTAG method

One method for making a copy of the content of the RAM is using JTAG (Breeuwsma, 2006). JTAG stands for Joint Test Action Group and is also referred to as boundary scan. With JTAG it is possible to test and debug an embedded system at various levels of the design and production. Using JTAG it is possible to stop the processor and gain access to the memory space of the device, thus allowing the data retained in RAM to be read without changing the content.

The main processor of a TomTom Go 500 is the Samsung S3C2440A. The datasheet of this processor (Samsung Electronics, 2009) states it uses an ARM9TDMI core. The datasheet also shows JTAG is accessible on the pins of the main processor, providing access to the core of the device. This core can be debugged in JTAG debug mode. The RAM chip is connected directly to the main processor, therefore it is possible to make a copy of the content of the RAM using the JTAG connection. As indicated by the datasheet, the core also provides a Data and Instruction Memory Management Unit (MMU). The MMU is used to translate physical addresses to virtual addresses and vice versa. Since it is easier with JTAG to read physical addresses instead of searching in the virtual address space, the MMU should be disabled.

4.1. *JTAG connector*

A TomTom Go 500 was examined for the availability of a JTAG connector. On the bottom of the device a docking connector was found. The pinout of the connector is shown in Table 1. Some pins on this connector could be used for serial communication with the device, but no JTAG signals were found. On previous TomTom models the JTAG signals were