Digital Investigation 11 (2014) 336–348

Contents lists available at ScienceDirect

Digital Investigation

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

Towards a forensic-aware database solution: Using a secured database replication protocol and transaction management for digital investigations

Peter Frühwirt*, Peter Kieseberg, Katharina Krombholz, Edgar Weippl

SBA Research gGmbH, Favoritenstraße 16, 1040 Vienna, Austria

ARTICLE INFO

Article history:
Received 24 July 2014
Received in revised form 31 July 2014
Accepted 5 September 2014
Available online 11 October 2014

Keywords:
MySQL
InnoDB
Digital forensics
Databases
Data tempering
Replication
Transaction management

ABSTRACT

Databases contain an enormous amount of structured data. While the use of forensic analysis on the file system level for creating (partial) timelines, recovering deleted data and revealing concealed activities is very popular and multiple forensic toolsets exist, the systematic analysis of database management systems has only recently begun. Databases contain a large amount of temporary data files and metadata which are used by internal mechanisms. These data structures are maintained in order to ensure transaction authenticity, to perform rollbacks, or to set back the database to a predefined earlier state in case of e.g. an inconsistent state or a hardware failure. However, these data structures are intended to be used by the internal system methods only and are in general not human-readable.

In this work we present a novel approach for a forensic-aware database management system using transaction- and replication sources. We use these internal data structures as a vital baseline to reconstruct evidence during a forensic investigation. The overall benefit of our method is that no additional logs (such as administrator logs) are needed. Furthermore, our approach is invariant to retroactive malicious modifications by an attacker. This assures the authenticity of the evidence and strengthens the chain of custody. To evaluate our approach, we present a formal description, a prototype implementation in MySQL alongside and a comprehensive security evaluation with respect to the most relevant attack scenarios.

Introduction

Common ACID-compliant *Database Management Systems (DBMS)* provide mechanisms to ensure system integrity and to recover the database from inconsistent states or failures. Therefore they contain a large amount of internal data structures and protocols. Their main purpose is to provide basic functionality like rollbacks, crash recovery and transaction management, as well as more advanced techniques like replication or supporting cluster architectures. They are solely intended to be used by internal methods of the system to ensure the integrity of the system.

Since databases are typically used to store structured data, most complex systems make use of at least basic database techniques for forensic analysis. Thus, when investigating an arbitrary system, standardized forensic techniques targeting the underlying database allow an investigator to retrieve fundamental information without having to analyze the (probably proprietary) application layer. Database forensics support efficient forensic

* Corresponding author.
E-mail addresses: pfruehwirt@sba-research.org (P. Frühwirt), pkieseberg@sba-research.org (P. Kieseberg), kkrombholz@sba-research.org (K. Krombholz), eweippl@sba-research.org (E. Weippl).

http://dx.doi.org/10.1016/j.diin.2014.09.003
1742-2876/

investigations in order to e.g. detect acts of fraud or data manipulation. However, little attention has been paid on the enormous value of internal data structures to reconstruct evidence during a forensic investigation.

To illustrate the need for guaranteeing that a database is unaltered, the following questions may be useful in the course of some digital investigations:

- Was a data record changed in a certain period of time and at what exact moment?
- Was data manipulated in the underlying file system by bypassing the SQL-interface?
- What statements were issued against the database in a given time frame?
- How have manipulated data records been changed with respect to the time line?
- What transactions have been rolled back in the past?

In this paper, we propose a novel forensic-aware database solution. Our approach is based on internal data structures for replication and transaction that are used by the database for crash recovery. They are in general not human-readable and intended to be read and used only by internal methods of the system. The overall benefit of our method is that no log files such as administrator logs are needed in order to create an entire audit trail as a pre-incident security mechanism. Furthermore it can be used as a post-incident method to locate unauthorized modifications. Our approach is feasible for all ACID-compliant DBMSs, because it solely relies on the standard replication and transaction mechanisms. In addition to that, our approach aims at securing these data structures against retroactive malicious modifications in case of an attack scenario to guarantee the authenticity and integrity of the reconstructed forensic evidence. To demonstrate the feasibility of our approach we provide a formal description and present a prototype implementation in MySQL. Furthermore, we provide a comprehensive security evaluation to demonstrate the benefits for system security and the integrity of the forensic evidence.

The remainder of this paper is structured as follows: Section 2 discusses related work. Section 3 provides a description of our approach. In Section 4 we present a showcase implementation of our approach based on MySQL. In Section 5 we evaluate our solution with respect to security and applicability. Finally, Section 6 concludes our work.

Related work

Due to the ever rising importance of incorporating computer systems and equipment in investigations, computer forensics is an emerging field in IT-Security (Casey, 2011). In this section, we present related scientific work with respect to database forensics and secure logging.

Database forensics

In digital forensics, log files on the operation system level have been used as a vital source to collect evidence in the last decades. Still, as several authors demonstrated in the past, the database layer is unpopular when it comes to forensic exploitation, even though it constitutes an integral part of enterprise assets.

In 2009 Martin Olivier provided a thorough review on the then current state of the art in research on database forensics (Olivier, 2009) and compared it to the then state of the art of file forensics. Five years later the situation has not changed much.

However, the amount of literature in the area has increased in recent years compared to what was available back then. The topic has been featured at the IFIP WG 11.9 meetings in recent years. The papers featured covered a range of topics relating to database forensics. Beyers et al. discussed the creation of a method to separate the different layers of data and metadata to prepare a database management system for forensic examination (Beyers et al., 2011). In a similar vein, Fasan et al. demonstrated how a database reconstruction algorithm can be utilized to reconstruct a database allowing an examination to be performed (Fasan and Olivier, 2012a, b). Pieterse et al. discussed the various techniques that can be used to hide data within a database caused by the complexity of databases and the lack of forensic tools with which to examine databases (Pieterse and Olivier, 2012). Lalla et al. described a model for investigating computer networks through network log files and how the examination of said files could reveal concealed activity (Lalla et al., 2012).

The Digital Investigation Journal has published two papers on database forensics in the last five years: Martin Olivier describes the pertinent differences between file systems and databases and how file system Forensic techniques could possibly be applied to database forensics. The paper also attempts to highlight potential areas of research within database forensics (Olivier, 2009). In 2012 Fasan et al. published an extended version of the respective IFIP publication (Fasan and Olivier, 2012a, b).

Additionally, in (Khanuja and Adane, 2011) the authors discussed research challenges in database forensics and regret the lack of attention to this field until now. Their main concern lied in the absence of practical tools available for forensic analysts. In a very recent work (Adedayo and Olivier, 2013), the authors aimed at developing practical techniques to exploit the database layer. One of their techniques aims at providing a better reconstruction method for changed data, another method focuses on providing confirmative evidence on data stored in a database. Furthermore, the authors emphasized the absence of in-depth research regarding database forensics.

Using database internals for forensic investigations

In a student work in 2005 (Stahlberg, 2005) it was shown that data stays persistent in the files system layer when using MySQL or PostgreSQL. This work was extended in two subsequent papers (Stahlberg et al., 2007; Miklau et al., 2007), both focusing on privacy aspects in database systems. In these papers they pointed out where data is preserved inside the internal structure of the DBMS: Table storage, the transaction log, indexes and other system components.

In a series of practical resources (Litchfield part 1, 2007; Litchfield part 2, 2007; Litchfield part 3, 2007; Litchfield part