Digital Investigation 17 (2016) 53–65

Contents lists available at ScienceDirect

Digital Investigation

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

Forensic investigation framework for the document store NoSQL DBMS: MongoDB as a case study

Jongseong Yoon [a], Doowon Jeong [a], Chul-hoon Kang [b], Sangjin Lee [a, *]

[a] Center for Information Security Technologies (CIST), Korea University, Anam-Dong, Seongbuk-Gu, Seoul, South Korea
[b] Digital Forensic Center, Supreme Prosecutors' Office, Banpo-daero, Seocho-Gu, Seoul, South Korea

ARTICLE INFO

Article history:
Received 24 December 2015
Received in revised form 16 March 2016
Accepted 19 March 2016
Available online 7 May 2016

Keywords:
Database forensics
Digital forensics
NoSQL DBMS
Document store NoSQL DBMS
MongoDB

ABSTRACT

The NoSQL DBMS provides an efficient means of storing and accessing big data because its servers are more easily horizontally scalable and replicable than relational DBMSs. Its data model lacks a fixed schema, so that users can easily dynamically change the data model of applications. These characteristics of the NoSQL DBMS mean that it is increasingly used in real-time analysis, web services such as SNS, mobile apps and the storage of machine generated data such as logs and IoT (Internet of Things) data. Although the increased usage of the NoSQL DBMS increases the possibility of it becoming a target of crime, there are few papers about forensic investigation of NoSQL DBMS.

In this paper, we propose a forensic investigation framework for the document store NoSQL DBMS. It is difficult to cover all of the NoSQL DBMS, as 'NoSQL' includes several distinct architectures; our forensic investigation framework, however, is focused on the document store NoSQL DBMS. In order to conduct an evaluative case study, we need to apply it to MongoDB, which is, a widely used document store NoSQL DBMS. For this case study, a crime scenario is created in an experimental environment, and then we propose in detail a forensic procedure and technical methods for MongoDB. We suggested many substantial technical investigation methods for MongoDB, including identifying real servers storing evidences in a distributed environment and transaction reconstruction method, using log analysis and recovering deleted data from the MongoDB data file structure.

Introduction

Relational DBMSs (RDBMS) are mostly used for the storage of the important data of organizations, or as the back-end of a web service. However, they are too costly, under-functioning and too complex to be applied to such fields as Big Data Analysis, Log Analysis, Social Network Services and Mobile Applications, which have recently been on the rise (Stonebraker, 2010a). In response to these limitations, various NoSQL DMBSs have increasingly been developed and applied, replacing RDBMSs. NoSQL DBMSs provide a more efficient means to store and access big data because their servers are more easily horizontally scalable and replicable than relational DBMSs. Their data model does not have fixed schema, so users can dynamically change the data model of applications easily. These characteristics of NoSQL DBMSs mean that the systems are increasingly used in real-time analysis, web services such as SNS, mobile apps and for the storage of machine-generated data such as logs and IoT (Internet of Things) data. While relational DBMSs currently dominate the database market, the NoSQL DBMS is a strongly upcoming trend (Andlinger, 2013).

Increased use of the NoSQL DBMSs, also increases the possibility of sensitive personal information being illegally obtained and stored in the NoSQL DMBSs for the purpose of

* Corresponding author. Fax: +82 2 3290 4738.
E-mail addresses: yoonjs53@naver.com (J. Yoon), sangjin@korea.ac.kr (S. Lee).

http://dx.doi.org/10.1016/j.diin.2016.03.003
1742-2876/

identity theft and/or of NoSQL DBMSs being targeted by cyber-attacks. On July 2015, classified documents were leaked from an Italian software manufacturer, indicating that the company had sold RCS (Remote Control System) wiretapping spyware to a number of governments (Weissman, 2015). The RCS hacks into personal devices, such as computers and smartphones, collects information including photographs, documents, keyboard input data and phone call recordings, which stores them using MongoDB, a form of NoSQL DBMS. Recently strong suspicions have been aroused that South Korea's intelligence agency has unlawfully surveilled civilians, and it came to light that an agent in charge then deleted the relevant records (Kwon and Witeman, 2015). It is likely that digital forensic investigation will be conducted on the RCS that contains a MongoDB database.

Digital forensic aspects of relational DBMSs such as oracle, Mysql, and Mssql are actively studied in terms of investigative methods, forensic artifacts, and deleted data recovery. Khanuja et al. (Khanuja and Adane, 2012) proposed a framework for DBMS forensic analysis and analyzed the internal structures and artifacts of Mysql. Fruhwirt et al. (Fruhwirt et al., 2010) studied the data file format of Mysql. Wright (Wright, 2005) developed the forensic tool, LogMiner, to reconstruct Oracle database activities. Pavlou et al. (Pavlou and Snodgrass, 2008) proposed a detection algorithm against database tampering. In addition, a method of recovering Oracle database records is studied by Choi et al. (Choi et al., 2013). However, there is little research about the digital forensic process and methods for NoSQL DBMSs.

Most research on the NoSQL DBMS is related to performance comparisons, differences between relational DBMS and NoSQL DBMS, special features in types of NoSQL DBMS, and their application to other fields (Stonebraker, 2010b) (Li and Manoharan, 2013). We suggested a method for the forensic investigation process and technical investigations for the MongoDB in our preceding research (Yoon et al., 2014). However, in our prior work, we created simple sample data to conduct an experiment, therefore we introduced the structure of MongoDB and artifacts which just provided a basic investigation process. Here we build an environment similar to the real world and extended our preceding research, based on this, we developed the investigation process to reflect the document store NoSQL DBMS's features. Providing a more detailed process to improve the technical analysis method. Especially in the recent work, we studied the new technical analysis methods, which are crucial to forensic investigation of the document store NoSQL Database, which are; identifying physical server storing evidences from distributed servers, transaction reconstruction method, using log analysis and deleted data recovery method from the MongoDB data files. We also demonstrated these technical analysis method in realistic environment.

In this paper, we propose a forensic investigation framework for the document store NoSQL DBMS. The NoSQL DBMS concept includes many different kinds of databases, except SQL DBMS, however, the area of our framework is, the document store NoSQL DBMS such as MongoDB and CouchDB. The document store NoSQL DBMS refers to databases that store their data in the form of documents such as XML, JSON, etc., and they support distributed environments and non-fixed schema (Nayak et al., 2013). We then evaluate the proposed framework by applying it to MongoDB, the most widely used document store NoSQL DBMS. For this evaluation, we build a crime scenario and an experimental environment, and then propose in detail a forensic process and technical methods for MongoDB.

Forensic investigation framework for the document store NoSQL DBMS

A number of digital forensics frameworks have been published. Pollitt (1995) proposed a four phase framework of acquisition, identification, evaluation and admission. Palmer (2001) proposed a six phase framework of identification, preservation, collection, examination, analysis, presentation, and decision in a digital forensic research roadmap. These general digital forensic frameworks are the basis for digital processes for new types of services or devices. However, this is really ordinary framework using traditional media and file systems. Furthermore, we need a specific processing model with a new concept of using contribution systems and a flexible data model i.e. NoSQL DBMS.

Martini and Choo (2012) proposed a four phase digital forensic framework for cloud computing: evidence source identification and preservation; collection; examination and analysis; and reporting and presentation. The evidence source identification and preservation phase in the distributed environment of cloud computing is a concept that can be adapted to a NoSQL DBMS forensic framework. However, this framework not including the analysis for the schema of a database, or database forensic features for example; collecting logs for transaction analysis. Therefore, it's difficult to use the NoSQL DBMS Forensic process.

There are few studies about DBMS forensic frameworks. Khanuja and Adane (2012) proposed a six phase framework for DBMS forensic analysis: identify, collect, analyze, validate, interpret, generate forensic reports and preserve evidence. Fowler (2007) published a whitepaper about the real world forensic analysis case of SQL Server 2005. He conducted forensic analysis of the SQL Server with 7 steps: verification, system description, evidence collection, timeline creation, media analysis, data recovery, and string search. DBMS forensic frameworks, upon which research has long been conducted, better reflect the structures and characteristics of DBMSs than the forensic framework based upon the file system, and cannot easily be applied to NoSQL DBMSs whose structure is dissimilar to RDBMSs.

Digital forensic investigation of NoSQL DBMSs, in addition to the usual requirements of DBMS forensics, can involve a large quantity of data, distribution and replication servers, and non-fixed data models and schema.

Whereas conventional file system forensics searches for evidence with a file as a single unit, and then collects it as disk images or files, investigation of DBMSs requires the analysis of a stored database and its schema, the generation of appropriate queries, and the selective collection of evidence. The transaction logs of DBMSs also need to be collected and analyzed.