Supporting Secure File Deletion in Page Mapping

Ilhoon Shin

NowonGu GongleungDong, Seoul National University of Science and Technology,
Building 3, 207, Seoul 139-743, South Korea
ilhoon.shin@snut.ac.kr

Abstract. The amount of digitalized data is rapidly increasing, and it is important to completely remove the confidential or private data. There are several secure file deletion tools, and however they may not properly work in NAND-based block devices because of the existence of flash translation layer. Among various flash translation schemes, we focus on page mapping scheme and designs the method to support secure file deletion.

Keywords: secure file deletion, page mapping scheme, flash translation layer, NAND flash memory

1 Introduction

Recently, various kinds of data such as pictures, personal videos, contacting information, social security number, and so on are digitalized and stored in computing devices, which results in raising security concerns. For example, we need to guarantee that the files are completely removed by the deletion operation, which is called secure file deletion. There are lots of secure file deletion tools for hard disk drives. They distort the original files by over-writing them with randomly generated data repeatedly [1]. However, these secure file deletion tools may not work properly in the NAND-based block devices because of the existence of flash translation layer (FTL).

NAND flash memory consists of blocks and pages. A page is a unit of a read/write operation. A block that consists of multiple pages is a unit of an erase operation. NAND flash memory does not support an over-write operation. Once a page is written, it cannot be re-written before the block that the page belongs to is erased. NAND-based block devices therefore perform an out-of-place update, which writes new data to a new location (a clean page). The original page is not modified. It is marked as invalid. Emulating the over-write feature with the out-of-place update is a main function of FTL. Note that the original data are not modified even with the repetitive over-write requests, and they can be recovered by directly scanning NAND flash memory.

There are several FTL schemes according to the mapping unit size of logical sector, and the page mapping scheme [2] is known to deliver a good performance. Thus, we focus on the page mapping scheme and discuss the method to support secure file deletion.

In the page mapping scheme, FTL searches for a clean page on a write request, and the new data are written in the found page. The old page is marked as invalid. If the number of clean page becomes lower than the threshold, the garbage collection process is initiated. The garbage collection process selects a victim block which has the most invalid pages [3]. The valid pages of the victim block are moved to other clean block, and finally the victim block is erased. Consequently, the secure contents are restorable in spite of the repetitive over-write requests, until the block that the secure contents belong to is selected as victim and erased by the garbage collection process.

In order to solve the problem, we maintain the over-write count per each logical page. If the over-write count exceeds the pre-defined threshold, we erase all the invalidated pages of the logical page. Thus, the repeatedly over-written pages cannot be recovered.

2 The Presented Scheme

In order to implement the secure file deletion, it should be guaranteed that the invalidated pages are all erased if the over-write requests to the same logical page occur at least n times repeatedly. In order to satisfy this condition, we maintain the number of the over-write request and the physical blocks that have the invalidated data for each logical page.

If the number of over-write request to a logical page exceeds the predefined over-write threshold, FTL finds the blocks that have the invalidated pages. Then, it reclaims the blocks with the erasure operation. Of course, before erasing them, the valid pages should be moved to other clean block, which is an overhead of the presented scheme.

3 Conclusion

In this work, we discussed the reason that the existing secure file deletion tools do not work properly in NAND-based block devices, and the method to support the secure file deletion in the page mapping scheme. The presented scheme maintained the physical locations of the invalidated pages and the over-write count per each logical page. If the over-write count exceeded the threshold, the presented scheme found all the invalidated pages and removed them by the block erasure operation. Before erasing the blocks, the valid pages of the blocks were moved to other clean block, which was an overhead of the presented scheme. In other words, to support the secure file deletion causes performance degradation, which should be discussed more in the future work.

Acknowledgments This work was supported by Seoul National University of Science and Technology.

References

1. Gutmann, P.: Secure deletion of data from magnetic and solid-state memory. In proceedings of USENIX Security Symposium (1996)
2. Ban, A.: Flash file system. United States Patent, No. 5,404,485 (1995)
3. Kawaguchi, A., Nishioka, S., and Motoda, H.: A Flash-Memory based File System. In proceedings of USENIX Technical Conference (1995)