2010 Second International Conference on Information Technology and Computer Science

An automatic Carving Method for RAR File Based on Content and Structure

Yingjie Wei
Computer and Software Institute
Hangzhou Dianzi University
Hangzhou Zhejiang, China
weiyjhz@live.cn

Ning Zheng, Ming Xu
Computer and software Institute
Hangzhou Dianzi University
Hangzhou Zhejiang, China
nzheng@hdu.edu.cn mxu@hdu.edu.cn

***Abstract*—File carving is a digital forensic technique. It aims to reconstitute a file from unstructured data sources with no knowledge of the file system. This paper presents an automatically carving method for RAR files. Since RAR is one of the most popular archive formats，and it is widely used on the digital devices to package data for transport or storage. It is important for forensic investigation to obtain the information of RAR files. We apply mapping function to locate the header and footer of an archived file, utilize the distance between the header and footer of an archived file to determine whether the archived file is fragmented, and apply enumeration to reassemble bi-fragmentation of an archived file. Finally we validate the integrity of archived file and RAR file, repairing RAR files which miss header or footer. Based on artificial data and real world data, experiments show our method can automatically carve continuous and fragmented RAR files. Moreover, the comparative experiments demonstrate that this method is better than other's in accurateness and effectiveness.**

Keywords-file carving; archived file; bi-fragmentation; mapping function; validation

I. INTRODUCTION

File carving technology is a forensics technique that recovers files without any matching file system meta-data. File carving is most often used to recover files from the unallocated space in a drive [1]. In the case of damaged or missing file system structures, File carving will play an important role. Nowadays, most file carving techniques share two important limitations: First of all, these programs can not carve data files fragmented. Second, carvers do not perform effective validation on the files they carve, as a result, these method present many "false positives"(files are presented as intact data, but which in fact contain invalid data or miss valid data so that cannot be displayed).

Since RAR format provides significantly better compression especially in the solid mode, also RAR format is support of multivolume archives and recovery record, RAR almost becomes the most popular archive format. Most general computer users have used this format to store and transport important data. Recovery of RAR files will be significant helpful for computer forensic. In this paper we present a new RAR carving method. The method focus on carving the archived files which embedded in RAR file occur bi-fragmentation. Accordingly, this method can recovers RAR file which occur heavy fragmentation or missing data.

II. RELATED WORK

RAR is a proprietary archive file format, it was developed by Eugene Roshal [2]. RAR files may be created only with commercial software WinRAR. WinRAR [3] is the most popular archiving software for the Windows operating system, supporting not only compression, but also encryption to protect the confidentiality of sensitive files. So far, file carving technology about the RAR files is confined by data continuous and intact.

Header-footer carving is a basic and classical carving technique. It works by locating the file header and file footer then extracted all data between the header and footer. Foremost [4], originally developed by the US Air Force Office of Special Investigation, is one of the first file carvers that implemented Header- Footer carving. Mikus researches many general file formats and extends Foremost [5]. PhotoRec [6] is file data recovery software which ignores the file system and goes after the underlying data, it can carve more intact files than others. Above-mentioned two tools can not exactly carve all files, especially the fragmented and broken files. Simson Garfinkel proposes bifragment gap carving based on a known header and footer can recover file split into two distinct fragments [7]. Cohen presents a theory of fragmentation model and mapping function between the file bytes and the image bytes [8].

III. METHODOLOGY

Because of the strategy of the file system allocation, with the file are added, modified and deleted, most file systems will get fragmented. Fragmentation occurs based on following assumptions.

- Files begin on cluster boundaries.
- Fragmentation can only occur on cluster boundaries.

As we prepare to solve the problem that RAR file is fragmented, next we will discuss a solution to address the bi-fragmentation of an archived file in the local of the RAR file.

978-0-7695-4074-0/10 $26.00 © 2010 IEEE
DOI 10.1109/ITCS.2010.23

A. RAR Format

A RAR file is consisted of invariable length blocks and variable length blocks [9]. These blocks contain marker block, archive header, comment header, file header, authenticity information, sub-block, and end block.

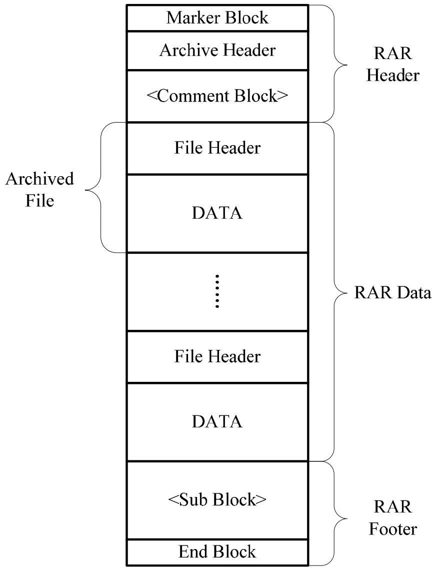

Figure 1. The general structure of RAR file, the contents of angle bracket do not always exist.

Archived files have no significant correlation each other, so if some archived files are corrupted, we can remove these archived file to assure the whole RAR file valid. Moreover, an archived file to be decompressed only need marker block, archive header and end block, which these blocks' structure and content are fixed relatively. So if a RAR file misses these blocks or some intact archived files are found on disk, we can repair these data artificially.

B. Scenario

In this section, we will discuss two scenario of bi-fragmentation, one is sequential, and the other is non-sequential.

Fig.2 is one scenario of bi-fragmentation on disk, File X is an archived file within a RAR file. In this figure, File X is fragmented to two parts. The first part of data with file header locates in the low field, and the second part of data followed by next archived file Y locates in the high field. Between the two fragmentations is several clusters' other data.

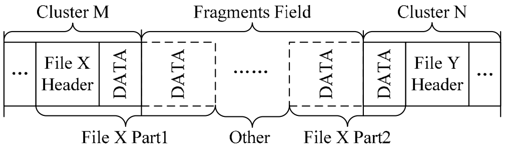

Figure 2. Sequential bi-fragmentation of the archived file

Fig.3 is the other scenario of bi-fragmentation on disk, File Y is the next archived file follows File X, however, because of non-sequential bi-fragmentation, another part of File X is fragmented to the front of the first fragment of File X.

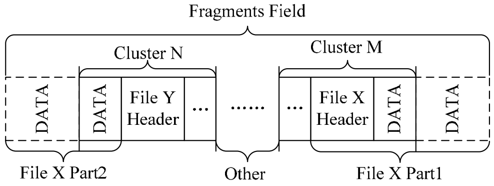

Figure 3. Non-sequential bi-fragmentation of the archived file

C. Framwork

The above analysis describes two scenario of bi-fragmentation for archived file wihin the local of RAR file. The major components of our carver are shown in Fig.4.

- The disk image is first fed into a preprocessing system which extracts information about RAR file being carved.
- This information is then fed to the mapping function. The mapping function can produces possible matching blocks.
- Possible matching block will be fed into discriminator. The discriminator judges these blocks are continuous, fragmentation or loss data.
- Fragmented data is fed to Fragment Processing. The fragment processing tries to splice fragments.
- The validator returns correctness of data. This component is essential for splicing fragments, it can feedback fragments of assembling success or not.

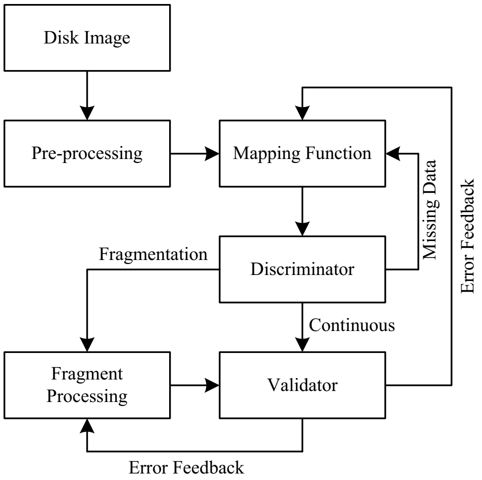

Figure 4. Overview of carving algorithms

1) Pre-processing

We search every block of RAR file in disk image through matching characteristic of each block. At the same time, we also extract information about RAR file from various kinds of block contained in the RAR.

- The start absolute offset of each block by searching: A_i , i is a identifier of the block.
- The relative offset of each block by calculating: R_i , i is a identifier of the block.
- The size of each block header: *headersize* , we use h_i to indicate the value *headersize* of each block, i is a identifier of the block.
- The size of data contained in each block: *datasize* , we use d_i to indicate the value *datasize* of data, i is a identifier of the block.

So we can calculate next block's relative offset through the equation: $R_{i+1} = R_i + h_i + d_i$ .These information are essential for the next components to carve the RAR file.

2) Mapping Function

Cohen has presented the theory of mapping function [8] between the file bytes and the image bytes in 2007. In this section, we elaborate the concept and principle of mapping function from another perspective. Mapping function is a mapping relation between the relative offset of the bytes contained in the file and the absolute offset of the bytes located in the disk.

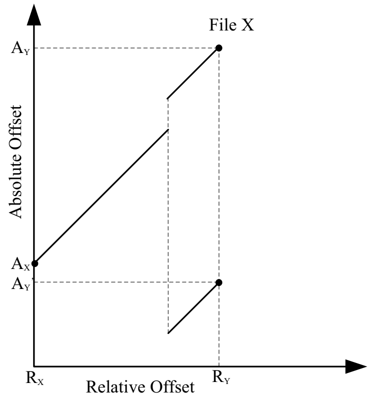

Figure 5. Mapping funcion of archived file

In our carving system, an example of mapping function is shown in Fig.5. Above all, we can conclude the following equation:

$$\mathrm{mod}(R_X, C) = \mathrm{mod}(A_X, C) \quad (1)$$

$$\mathrm{mod}(R_Y, C) = \mathrm{mod}(A_Y, C) \quad (2)$$

$$R_Y = R_X + h_X + d_X \quad (3)$$

According to (1), the following equation must be established:

$$R_X = kC + A_X \quad (4)$$

Where k is an integer, so we can obtain the following equation:

$$\mathrm{mod}\left[(R_X + h_X + d_X), C\right] = \mathrm{mod}(A_Y, C) \quad (5)$$

$$\mathrm{mod}\left[(kC + A_X + h_X + d_X), C\right] = \mathrm{mod}(A_Y, C) \quad (6)$$

Finally, the following equation can be obtained:

$$\mathrm{mod}\left[(A_X + h_X + d_X), C\right] = \mathrm{mod}(A_Y, C) \quad (7)$$

Therefore, we can determine the file follows file X, that is to say we can determine the end of file X (the start of file Y).

3) Discriminator

The first step of processing file fragmentation is to determine whether the file is fragmented. The most common approach is to compare the file size.

Suppose A_X and A_Y are two definitive points shown in Fig.6, which A_X is the start of File X, and A_Y is the start of File Y. Because next to file X is file Y, A_Y also can be regard the end of file X. The *Filesize* in Fig.6 is the actually size of file X.

If file X is continuous and intact, we use F_X to indicate the value *Filesize* of archived file X. There should has following equation: $F_X = A_Y - A_X$. An example of a typical such situation is shown in Fig.6.

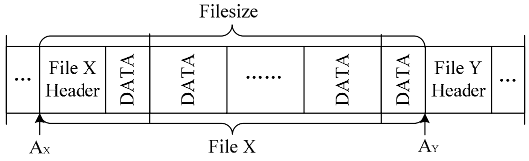

Figure 6. Archived file X is continuous and intact.

If file X occurs sequential bi-fragmentation, but data does not miss, there should has following equation: $F_x < A_Y - A_X$. An example of a typical such situation is shown in Fig.7.

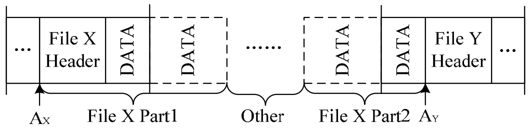

Figure 7. Archived file X is fragmented and the fragments are sequential.

If file X occurs non-sequential bi-fragment, but data does not miss, there should has following equation: $A_Y - A_X < 0$. An example of a typical such situation is shown in Fig.8.

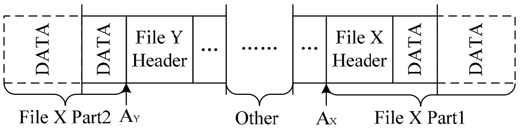

Figure 8. Archived file X is fragmented and the fragments are non-sequential.

If file X miss data, there should has following equation: $0 < A_Y - A_X < F_X$. But if $A_Y < A_X$, we can not use this equation to determine whether File X miss data. The critical component of validator can help to process this problem.

4) Fragment Processor

We use fragment processor to process the problem which Archived file occurs sequential bi-fragmentation or non-sequential bi-fragmentation, an example of sequential bi-fragmentation of archived file X is shown in Fig.9.

- Sequential bi-fragmentation: the premise is that A_X and A_Y have been located, the following equation must be established.

$$\begin{cases} A_X < A_Y \\ A_Y - A_X > F_X \end{cases}$$

Because we can know the size of file X header h_X and the size of data d_X in file X, the size of other data can be defined:

$$O_e - O_s = A_Y - A_X - h_X - d_X \qquad (8)$$

According to the fragmentation rules, the following rules must be hold:

$$\left\{ O_s = mC \mid \lceil A_X / C \rceil \le O_s \le O_e,\ m \text{ is an integer} \right\}$$

$$\left\{ O_e = nC \mid O_s \le O_e \le \lfloor A_Y / C \rfloor,\ n \text{ is an integer} \right\}$$

In Fig.10, the minimum beginning of other data is at P_s ,and the maximum ending of other data is at P_e , so the following equation must be hold:

$$P_s = \lceil A_X / C \rceil \qquad (9)$$

$$P_e = \lfloor A_Y / C \rfloor \qquad (10)$$

Let O_s starts at P_s , moves along the direction of A_Y , the distance between O_s and O_e is hold on the basis of (8). When reach the fragmentation point or $O_e = P_e$, the processing stop.

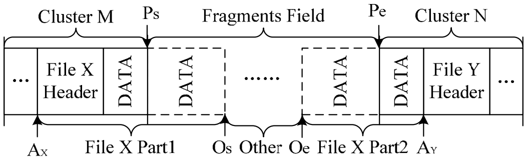

Figure 9. Sequential bi-fragmentation of archived file X.

- Non-sequential bi-fragmentation: the premise is also that A_X and A_Y have been located, and the equation must be established: $A_Y < A_X$. An example of non-sequential bi-fragmentation of archived file X is shown in Fig.10

In order to process this situation convenient, we change (8) to another form:

$$(A_Y - O_e) + (O_s - A_X) = d_X + h_X \qquad (11)$$

The remaining rules are the same as the sequential bi-fragmentation's. Let O_s starts at P_s , moves along the opposite direction of A_Y , O_e moves along the direction of A_Y , and the (11) must be hold . When reach the fragmentation point or $O_e = P_e$, the processing stop.

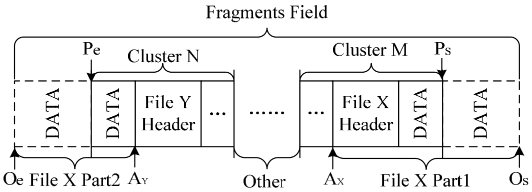

Figure 10. Non-sequential bi-fragmentation of archived file X

5) Validator

Validator plays an important role in our carving system, it determines the correctness of the file assembled.

In each file header, the CRC of decompressed data is stored to provide a verification process after extraction to make sure that it is the same as the original. So our validator uses the CRC and RAR extract module (provided by the official website) to validate the archived file. Let the extract module to decompress the archived file, then compare the CRC of decompressed to the CRC in file header, if the same, indicating that archived file is valid, otherwise, fragmentation processor will try the next possibility or denote the archived file miss data.

IV. RESULT AND DISCUSSION

This section discusses the experiment and the results of applying our carving system. The experiment is consisted of two parts. First, we apply the carving system on artificial data set. Second, we apply carving system on the image of actual disk, and compare our method to Foremost and PhotoRec. Carving time, the number of carved files and accuracy are three evaluation criteria. Accuracy equals to true positive, means the number of files which are carved and valid. The equation is:

$$Accuracy = N_{TP} / N_{Total} \times 100\% \qquad (12)$$

N_{TP} is the number of file that is correctly carved, N_{Total} is the total number of file that is carved.

A. Arificial scenario

Our experiment is done under a 300MB image. The image was initialized by random data. The RAR files needed are found from the public domain on the Internet. This partition contains 11 RAR files, 83 archived files.

TABLE I. SCENARIOS OF OUR DATA SET

File	Description
1	1 archived files, 2 fragments, missing start and end(2,1)
2	5 archived files, 3 fragments, missing start, middle and end(3,1,2)
3	11 archived files, 3 fragments, missing start and end(1,3,2)
4	5 archived files, 4 fragments, missing start and end(4,2,1,3)
5	5 archived files, 3 fragments, missing start and end(1,3,2)
6	9 archived files, 3 fragments, missing start, middle and end(1,2,3)
7	4 archived files, 2 fragments, missing start and end(1,2)
8	27 archived files, 5 fragments, missing start, middle and end(3,4,2,5,1)
9	5 archived files, 3 fragments, missing start and end(3,2,1)
10	9 archived files, 2 fragments, missing start and end(2,1)
11	2 archived files, 2 fragments, missing start and end(2,1)

We create several scenarios of RAR files in our data set, these scenarios include fragmentation and missing data. Table 1 shows the detail of these scenarios, where the number of intact archived files is 55.

TABLE II. RESULT OF OUR METHOD

Method	Time	Result	Accuracy
The new RAR carving method	5min43sec	18 RAR files, 48 archived files	100%

In the artificial scene, our carving system performance well, the successful rate is 87.27%. It can recover all intact or bi-fragmented archived files regardless of the RAR files whether have header or footer. The time consuming depends on the distance between two matching fragments. 18 RAR files are caused by some RAR files miss middle data or archived file occurs heavy fragmentation, our method removes those archived file which miss data so that RAR file may be separated to two or more RAR files.

B. *Actual scenario*

Our experiment is also done under the image of actual disk. The test object is the image of system partition, in this partition, frequency of write and erase file is high, so the situation of file storage is complicated, file fragmentation often occurs. The size of image is 15GB. We apply Foremost, PhotoRec and our new RAR carving system to this image.

TABLE III. RESULT OF THREE METHODS

Method	Time	Result	Accuracy
The new RAR carving method	18min57sec	57 RAR files, 57 correct RAR files, 205 archived files	100%
Foremost	13min 43sec	39 RAR files, 3 correct RAR files, 70 archived files	7.69%
PhotoRec	1h7min12sec	9 RAR files, 3 correct RAR files, 70 archived files	33.33%

- Carving time: Foremost is the fastest, only runs 13min 43sec. Our RAR carving system runs 18min 57sec due to the validator component is time-consuming, but the processing speed is still close to Foremost. The slowest is PhotoRec.
- Carving result: The new RAR carving system carves 57 correct RAR files contain 205 archived files, and accuracy is 100% owing to the validator component, validator verifies each archived file. Foremost carves 39 RAR files but only 3 RAR files contain 70 files are valid. PhotoRec carves 9 RAR files where 3 files contain 70 files are correct.
- Accuracy: the (12) is use to calculate the value of accuracy. From table 3, we can know the accuracy of new RAR carving system is the highest 100%, owing to the validator component, validator verifies each archived file. PhotoRec is 33.33% and foremost is the lowest 7.69%.

Through the above mentioned two experiments, we think that the new RAR carving method is more useful. It can identify the integrity of archived file contained in a RAR file, successfully reassembles the bi-fragment of archived file, thereby, it can recover RAR files despite heavy fragmentation under certain conditions. The new RAR caving system also can accurately recover RAR files missing header or footer. The whole process does not need any manual intervention, and has high accuracy.

V. CONCLUSION AND FUTURE WORK

In this paper, we present a new RAR file carving method include five critical components. The algorithm based on the content characters and the internal structure of RAR file. Finally, we apply the new RAR carving system on the artificial data set and the image of actual disk, compare our method to another two carving methods. The experiment's result shows our carving method is better.

In the Future work, we will improve the speed of the new RAR file carving method, focus on exploring the new method of seeking fragmentation point and reassembling fragmentation.

ACKNOWLEDGMENT

This work is supported by the Natural Science Foundation of Zhejiang Province of China under Grant No. Y1090114.

REFERENCES

[1] Pal, A. and N. Memon, "The evolution of file carving", Signal Processing Magazine. Vol. 26, Issues 2, pp. 59-71, 2009.

[2] The RAR Archiver EULA(End user license agreement). http://www.rarlabs.com/download.htm, 2009.

[3] RARlab, "WinRAR-What's new in the latest version". http://www.rarlabs.com/rarnew.htm, 2009.

[4] Foremost 1.56. http://foremost.sourceforge.net/.

[5] Nichoals Mikus, "An analysis of disc carving techniques". Master's thesis. Monterey: Naval Postgraduate School, 2005.

[6] PhotoRec, http://www.cgsecurity.org/wiki/PhotoRec, 2009

[7] Simson L.Garfinkel, "Carving contiguous and fragmented files with fast object validation", Digital Investigation. Vol. 4, Supplement 1, pp. 2-12, 2007.

[8] Michael Cohen, "Advanced carving techniques", Digital Investigation. Vol.4, Issues 3-4, pp.119-12, 2007.

[9] TechNote.http://www.rarlabs.com/download.htm,2009