[9] Рязанов В.В., Челноков Ф.Б., О склеивании нейросетевых и комбинаторно-логических подходов в задачах распознавания, Доклады 10-й Всероссийской конференции "Математические методы распознавания образов (ММРО-10)", Москва, 2001,115-118.

[10] Christopher J.C. Burges. A Tutorial on Support Vector Machines for Pattern Recognition, Appeared in: Data Mining and Knowledge Discovery 2, 1998, 121-167.

[11] Yu.I. Zhuravlev, V.V. Ryazanov, O.V. Sen'ko, A.S. Biryukov, D.P. Vetrov, A.A. Dokukin, N.N. Katerinochkina, A.S. Obukhov, M.Yu. Romanov, I.V. Ryazanov, F.B. Chelnokov. The Program System for Data Analysis "Recognition" (LOREG). The 6th German-Russian Workshop "Pattern Recognition and Image Understanding". Proceedings. Novosibirsk, Russia, 2003, pp. 255-258.

[12] Ganster H., Gelautz M., Pinz A., Binder M., Pehamberger H., Bammer M., Krocza J. Initial Results of Automated Melanoma Recognition //Proceedings of the 9th Scandinavian Conference on Image Analysis, Uppsala, Sweden, June 1995, Vol.1, pp. 209-218.

[13] Dyukova Ye.V., Ryazanov V.V., The Solution of Applied Problems by Recognition Algorithms Based on the Voting Principle. VTs Akad. Nauk S.S.S.R., Moscow, 1986.

Authors' Information

Alexander A. Dokukin – Dorodnicyn Computing Centre of the Russian Academy of Sciences, Vavilov st., 40, Moscow GSP-1, 119991, Russia; e-mail: dalex@ccas.ru

Oleg V. Senko – Dorodnicyn Computing Centre of the Russian Academy of Sciences, Vavilov st., 40, Moscow GSP-1, 119991, Russia; e-mail: senkoov@mail.ru

ANALYSIS OF SECURITY IN ARCHIVING

Dimitrina Polimirova–Nickolova

Abstract: Some basic types of archiving programs are described in the paper in addition to their advantages and disadvantages with respect to the analysis of security in archiving. Analysis and appraisal are performed on the results obtained during the described experiments.

Keywords: Web Security, Mail Security, Information Security, Archive Programs, Compressed Objects, Methods Of Encryption.

The Present Situation

In the development of the computer science the creation and the use of archived objects is a classical research problem, which has found different resolutions for decades past. Nowadays the availability of several dozens of methods and their varieties represent an excellent demonstration of the ambitions of the information systems' programmers and designers for a real high-speed and high-effective compression of information flows.

The following basic types of archiving programs could be defined with respect to the information security of compressed objects, obtained after examination of more than 320 archiving programs, known by now:

1) E-mail archiving programs – in this kind of archiving programs the relative homogeneity of the information flow (e-mail traffic) is used and the most suitable methods of compression are selected. There are some differences among the basic existing e-mail clients (MS Outlook, MS Outlook Express, Netscape Mail, Opera Mail, Eudora Mail, Pegasus Mail etc.), which make possible the applying of different realizations of the compressing process. The advantages consist in the multiple reduction of the saved e-mail folders' volume and in the high degree of security against unauthorized access (viruses, worms, spyware, malware etc.). The disadvantages above all are related to the consumption of computing resources to realize the right and the reverse transformation.

The basic 6 extensions and their corresponding applications that are characteristic for this type of archiving program are: DBX (Outlook Express Email Folder), IDX (Outlook Express Mailbox Index), PCE (Eudora Mailbox Name Map), MSG (Pegasus Mail Stored Messages to Be Sent), SNM (Netscape Mail Email Message File), BOE (Outlook Express Backup File).

2) Converting archiving programs – these are archiving programs, that have the possibility to transform objects compressed by a given method in objects compressed by another method. Two variants exist with regard to this transformation: a) without a restoration of the object in its initial appearance; b) with a restoration of the object in its initial appearance. Their advantages consist in the use of a compression method which is optimal for a given type of information (e.g.: .jpg, .gif, .doc, .xls, .ppt etc.). Their disadvantages reside in the high complexity of operating environment.

The basic 6 extensions and their corresponding applications that are characteristic for this type of archiving program are: ACE (WinAce Compressed File), RAR (WinRAR Compressed Archive (RarLab)), ZIP (Compressed Archive File), AIN (AIN Compressed Archive), GZIP (GNU Zip Compressed Archive), UC2 (Compressed File).

3) Multiple archiving programs – these are programs which perform some successive kinds of archiving processing on the object for compression by using several methods of compression differing by their characteristics. In this manner a different (fully optimized) method of compression could be applied for the different parts of the object. The advantages lay in the very high flexibility, functionality and adaptivity to the different parts of the compressed objects which differ by their internal structure. The disadvantages are connected to the high initial expenditure needed for the creation of library of modules for similar methods of compression and the realization of a relevant environment, suitable for analysis of the separate parts of the objects.

The basic 6 extensions and their corresponding applications that are characteristic for this type of archiving program are: ARJ (Robert Jung ARJ Compressed Archive), JAR (JAR Archive (ARJ Software, Inc.)), TAR (Tape Archive File), AI (Ai Archiver Archive), LHA (Compressed Archive File), ZOO (ZOO Compressed Archive File).

4) Image archiving programs – this is an extremely live problem in the present-day real-time processing of video and image web-objects. The predominating trend in this processing is the obligatory compressing of the object immediately after its creation. The transmission and the processing of the object is fully realized in a compressed state to the last moment of its reproduction on the relevant media. The advantages consist in the significant reduction of the objects' dimension and the time needed for transmission, retransmission and processing. The disadvantages are connected to the high expenditure for the hardware components, which realize the compression partly or fully. A reasonable compromise in this respect are the combined (software-hardware) methods of compression.

The basic 6 extensions and their corresponding applications that are characteristic for this type of archiving program are: AIS (ACDSee Image Sequence File), B&W (Image Lab), BIL (ArcView Image File (ESRI)), BIN (Micrografx Designer 7 Project Image), CPT (Corel Photo-Paint Image (Corel)), PDB (PhotoDeluxe Image (Adobe)).

5) Data archiving programs – these are programs specialized in the creation, the processing and the use of compressed objects which result from information flows owning "data" characteristics. In the different platforms and operating systems the notion "data" has a different sense. In this instance we are concerned only by the fact, that the data in the different phases of their existence pass in a compressed form, exist for a fixed time in this form and a little time before to be "processed" the compressed objects are decompressed. The advantages lay in the reasonable degree of the optimal use of the resources. The disadvantages consist in the "superfluous" operations for compression and decompression.

The basic 6 extensions and their corresponding applications that are characteristic for this type of archiving program are: DOC (Word Document (Microsoft)), PDF (Acrobat Portable Document Format (Adobe)), TXT (Text File), XLS (Excel Worksheet (Microsoft)), XML (Extensible Markup Language File), PPT (Power Point Presentation (Microsoft)).

6) Executable archiving programs – the aim of these programs is to accomplish some specificity of the compression, connected with the possibilities for running the compressed objects. These are active objects which

own the capability for algorithmic branching of events depending on the used scenario. The advantages are connected to the extremely precise use of computing resources and the very high degree of protection against "reverse engineering". The disadvantages consist in the dependence from the platform, the operating system, the applications on use and the human factor.

The basic 6 extensions and their corresponding applications that are characteristic for this type of archiving program are: EXE (Executable File (Microsoft)), PE (Portable Executable File), PL (Linux Shell Executable Binary), FOX (FoxBase/FoxProt Executable File), FMX (Oracle Executable Form (FRM)), XXY (SPARC Executable Script File).

The Problem

Protection of information is accomplished by data encryption. Data encryption is a process in which the contents of a message or a file is tangled to such extent that it becomes unintelligible to anybody. To enable the message decoding or the file reversal to its initial state, it is necessary to own some key or access code. This concept is similar to the one for data compression. Thus, two different goals could actually be achieved by using the same approach:

1) Size reduction, which is accomplished by data compression via encoding.
2) Making data unreadable, which encoding performs in the case of encryption.

The results of the experiments which were carried out will be shortly revealed to facilitate better achievements in enhancing security of objects, and especially for compressed objects. The goal of these experiments was to examine and analyze the combination of data compression and data encryption [1].

The first study analyzes the SPEED of the encoding process. In this regard, the following four tasks were defined:

1) Evaluation of the resulting files, compressed with popular compressing programs. Particular experiments were made for all 18 extensions. Their file size was 1 Mb, and all of them were compressed with the most popular compressing programs. The results for all 18 extensions used during the study can be seen in Figures 1a,b,c,d,e,f.

Fig. 1a

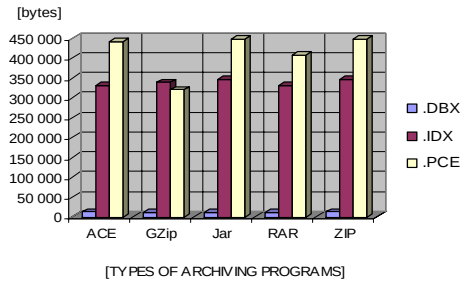


Fig. 1b

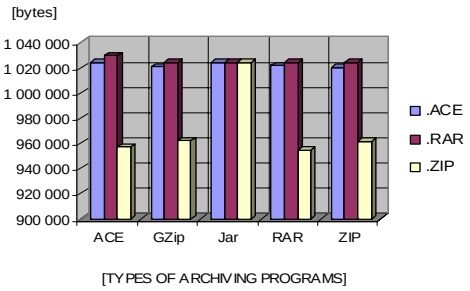


Fig. 1c

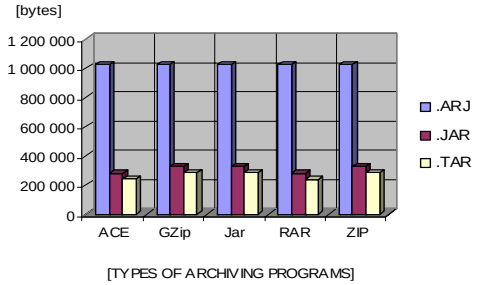


Fig. 1d

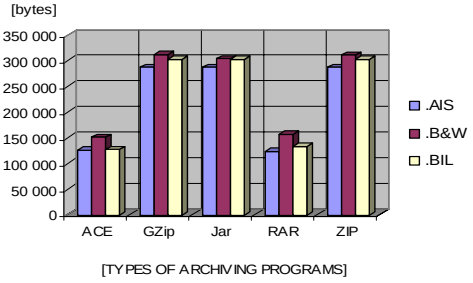


Fig. 1e

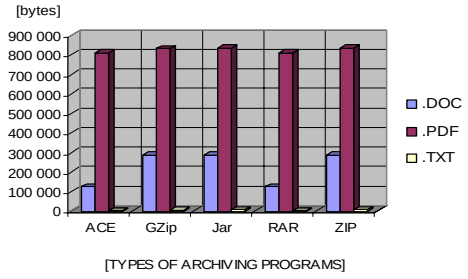
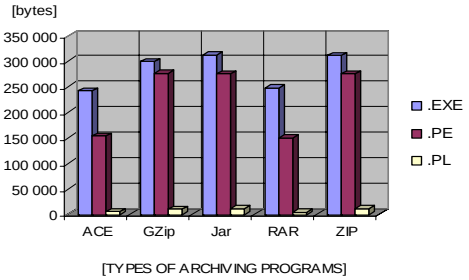


Fig. 1f



2) Encryption of the objects before compression. The same 18 extensions (3 for each type of archiving program) with original file size 1 Mb were used. The experiments which were made examined the file size of the encrypted 18 original extensions (Figure 2) and the time period needed for encryption of those 18 file extensions (Figure 3).

Fig. 2. File sizes of encrypted extensions

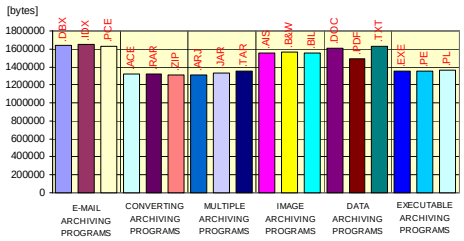
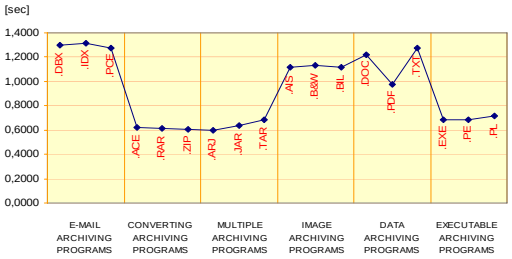


Fig. 3. Time period needed for encryption



3) Encryption of the objects after compression. The results used were the same as the ones obtained in the first task during this study. All compressed 18 extensions were encrypted, and their file sizes (Figure 4) and time periods (Figure 5) needed for encryption after compression were examined.

Fig. 4. File sizes of the encrypted after the compression extensions

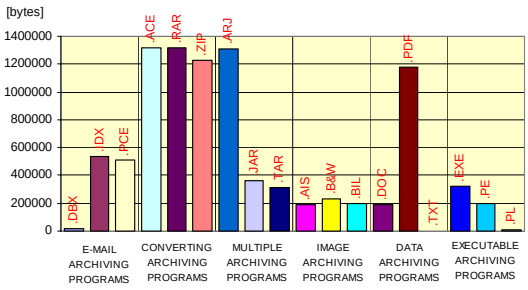
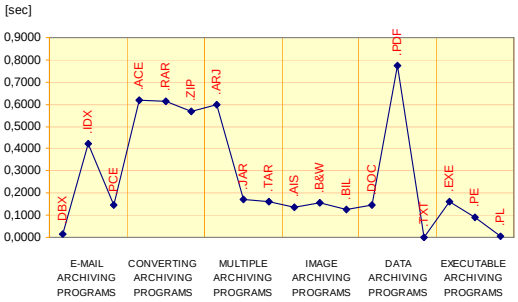
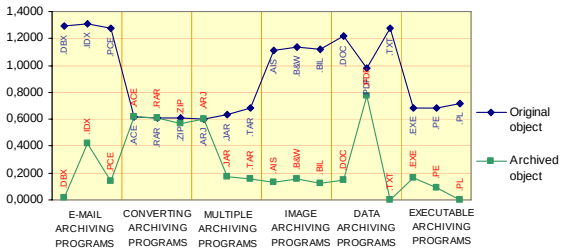


Fig. 5. Time periods needed for encryption after the compression



4) Comparison of the time periods needed for encryption of the objects before and after compression (Figure 6).

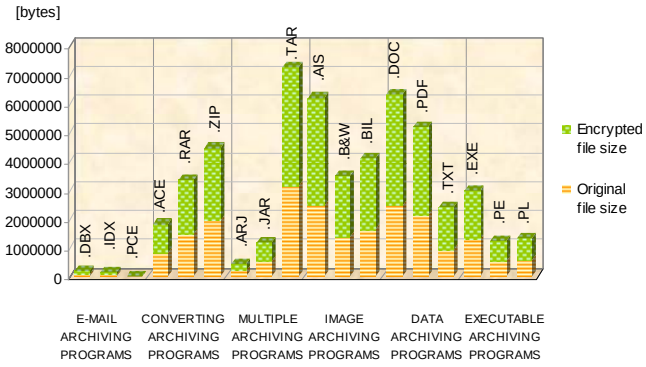
Fig. 6.



The second study analyzes the SIZE of the object created after the encryption. In this regard, the following three tasks were defined:

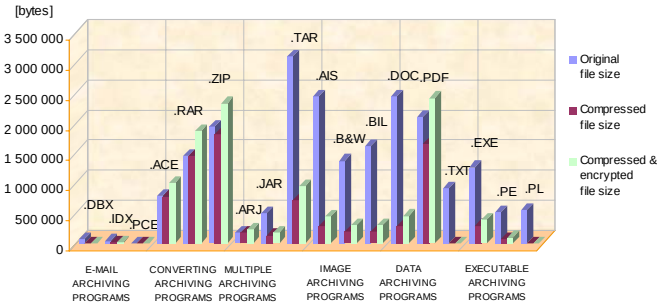
1) Encryption of objects with different file formats. Displayed in Figure 7 are the original file sizes, which are different for the 18 used extensions, and the file sizes obtained after the encryption of the original files.

Fig. 7.



2) Encryption of the files compressed with different compressing programs. Displayed in Figure 8 is the resulting comparison of the sizes of the original files and the sizes of the files encrypted after compression.

Fig. 8.



3) Comparison of the sizes of the original files and the sizes of the encrypted files. Encrypted original file sizes and encrypted-after-compression file sizes were examined, and the results for all 18 extensions are displayed in Figures 9a, b, c, d, e, f.

Fig 9a.

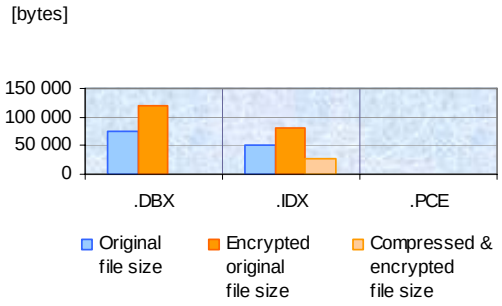


Fig. 9b.

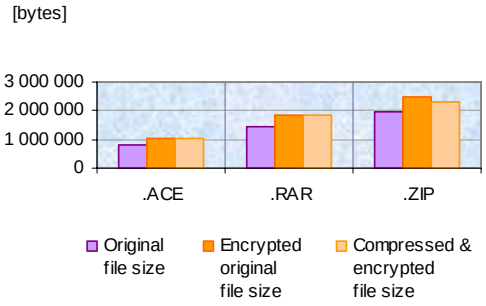


Fig 9c.

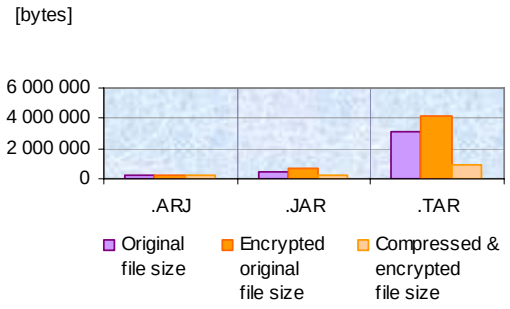


Fig. 9d.

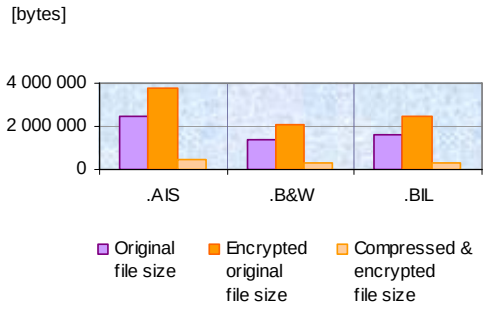


Fig 9e.

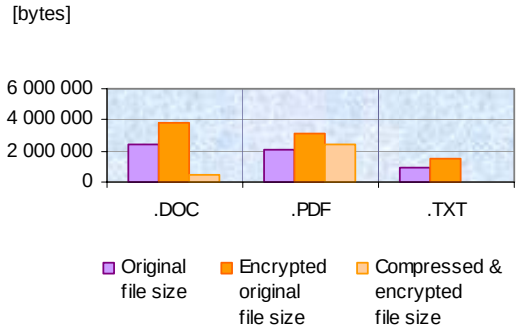
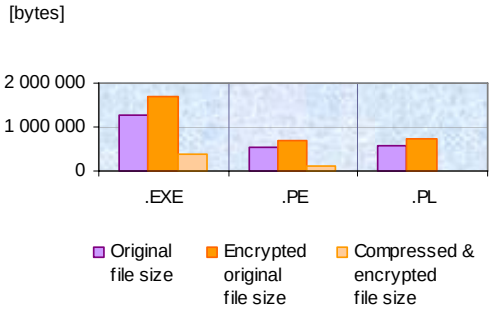


Fig. 9f.



The following assessments could be made from the experiments which were carried out:
1) The speed of the encoding process is higher if the object has been compressed before the encoding. This is due to the decrease of the amount of information for encoding after the compression.
2) The size of the resulting file decreases, if it is first compressed and then encrypted. In many cases, if the object is encoded without compression, its size is increased.
3) Some future investigations could be made in connection with the size of the password used in the encryption process and the effect of passwords on the compression process [2, 3, 4].

Conclusions

A thorough examination of the influence of some chosen parameters of information security on the methods of compression of objects is required.

It is also necessary to create a set of criteria for appraisal of the various commercial compressing and archiving programs in connection with the information security.

Bibliography

[1] Alistair Moffat, Andrew Turpin, Compression and Coding Algorithms, Kluwer Academic Publishers, 2002

[2] Ed Skoudis, Counter Hack: A Step-by-Step Guide to Computer Attacks and Effective Defenses, Prentice-Hall PTR, 2002.

[3] K. Jallad, J. Katz, and B. Schneier, Implementation of Chosen-Ciphertext Attacks against PGP and GnuPG, Information Security Conference 2002 Proceedings.

[4] Rob Shimonski, Introduction to Password Cracking, IBM Developer Works Hacking Techniques article, July 2002.

Author Information

Dimitrina Polimirova–Nickolova – National Laboratory of Computer Virology – BAS

1113 Sofia, Acad. G. Bonchev Str., Block 8, Office 104; poly@nlcv.bas.bg