An Improved Known Plaintext Attack on PKZIP Encryption Algorithm

Kyung Chul Jeong, Dong Hoon Lee, and Daewan Han

The Attached Institute of ETRI,
P.O. Box 1, Yuseong-Gu, Daejeon, Korea
{jeongkc,dlee,dwh}@ensec.re.kr

Abstract. The PKZIP encryption algorithm has been widely used to protect the contents of compressed archives despite the known security weakness. Biham and Kocher proposed a known plaintext attack with the complexity 2^{40} when 12 plaintext bytes are given. Stay suggested a different way to attack and addressed an idea which makes the complexity be reduced if the information of additional files encrypted under the same password is provided. However, the complexity of Stay's attack is quite large when only one file is used.

In this paper, we propose a new attack based on Biham and Kocher's attack. We introduce a method to reduce the complexity using the information of multi-files, so our attack can have the both advantages of previous two attacks. As a result, our attack becomes about $(3.4)^l$ times faster than the attack of Biham and Kocher when l additional files are used. Our experiment shows that ours is at least 10 times faster than Stay's. In addition, our attack can be improved in the chosen ciphertext model. It is about $(21.3)^l$ times faster than Biham and Kocher's attack with chosen plaintext of l additional files.

Keywords: PKZIP encryption, known plaintext attack.

1 Introduction

Compression softwares are used for various reasons: to reduce the size of big files, to unify many files and folders into a single archive, to split a big file into several parts with a small size and to protect the contents of the files by the password-based encryption. Most compression softwares support the ZIP file format among several compression file formats such as ZIP, RAR, ARJ, 7z and etc. As one of their protection algorithms, they also support the *traditional* PKZIP encryption included in the ZIP format specification. Almost all softwares support more strong encryption algorithms like AES in addition.

The traditional PKZIP encryption algorithm (a.k.a standard Zip 2.0 encryption) was designed by Roger Schlafly [4]. It has been widely supported by most compression softwares despite the publicly known security weakness. Biham and Kocher presented a known plaintext attack on the PKZIP encryption in [1]. They described an algorithm which extracts the encryption key (initialized with

H. Kim (Ed): ICISC 2011, LNCS 7259, pp. 235–247, 2012.

the password) with 2^{40} complexity providing 12 plaintext bytes (or 2^{38} complexity providing 13 plaintext bytes). Stay introduced a ciphertext-only attack on the PKZIP encryption for some of the compression softwares [3]. If 5 files in an archive are given, the first 10 plaintext bytes of all files could be derived in some softwares at that time. He proposed a new type of attack to utilize this additional information. His attack has the complexity of 2^{63} when only one file with 12 plaintext bytes is given, but it can be much more efficient using every plaintext of 5 files.

In this paper, we propose an attack which includes a new method to reduce the complexity using the additional file's plaintext. Our attack can be regarded as a generalization of Biham and Kocher's attack in the sense that ours is same as theirs when there is only one file. In the early state of our attack, a portion of key candidates can be filtered out by checking a certain condition induced from the relation between the plaintexts. As a result, we combine the two advantages: smaller complexity of [1] and the efficiency of utilizing multi-files in [3]. In the known plaintext attack, our attack becomes about $(3.4)^l$ times faster than the attack of [1] if plaintexts of l additional files are given. The experiment supports our claim and shows that ours is at least 10 times faster than Stay's attack.

The ratio of the reduced complexity depends on the relation between plaintext values. In the chosen ciphertext attack, we can determine plaintexts to satisfy the optimal relation. As a result, our attack can be improved to become $(21.3)^l$ times faster than the original attack when l additional files are given.

This paper is organized as follows. The preliminary and previous works are briefly described in the next section. It includes the overview of PKZIP encryption algorithm and the sketches of the attacks of [1] and [3]. We explain a new attack in Section 3, 4 and validate our result by comparing with other results by some experiments in Section 5. Finally we conclude in Section 6.

2 Previous Works

In this section, we briefly describe the PKZIP encryption which can be found in [4] and fix the notation. The attack of Biham and Kocher [1] is summarized in 2.2 and the recent attack of Stay [3] is summarized in 2.3.

2.1 The PKZIP Encryption

The PKZIP encryption is a stream cipher which encrypts one byte at a time. Three 32-bit keys $\mathtt{K}^0$, $\mathtt{K}^1$ and $\mathtt{K}^2$ are used as an internal state. One byte information '$\mathtt{B}$' is used to update these 3 keys as follows.

$$\begin{aligned}\texttt{Key\_update(B):}\ \ \mathtt{K}^0 &= \mathrm{CRC32}(\mathtt{K}^0, \mathtt{B})\\ \mathtt{K}^1 &= \{\mathtt{K}^1 + \mathtt{L}(\mathtt{K}^0)\} \times \texttt{0x08088405} + 1\\ \mathtt{K}^2 &= \mathrm{CRC32}(\mathtt{K}^2, \mathtt{M}(\mathtt{K}^1))\end{aligned}$$

The definition of CRC32(,) is described in [1]. $\mathtt{L(X)}$ and $\mathtt{M(X)}$ are the least and the most significant byte of $\mathtt{X}$, respectively. Each bit is numbered from right to