

# 实验报告：使用 John the Ripper 还原 ZIP 压缩包密码

## 1. 实验目的

通过本实验，学习如何在 Kali Linux 中使用 John the Ripper 进行 ZIP 压缩包的密码破解。John the Ripper 是一个广泛应用于密码破解的工具，能够通过多种方法（如字典攻击和暴力破解）来破解多种类型的加密文件，包括 ZIP 和 RAR 文件。

### John the Ripper 的工作流程

- 提取哈希：**John 无法直接对 ZIP 文件进行破解，因此必须先使用 zip2john 提取加密的密码哈希。哈希值表示了文件加密后的状态，但无法直接推导出密码。
- 破解密码：**John 通过一系列可能的密码（使用字典或逐个尝试的方式）计算出它们对应的哈希值，并与提取出的哈希值进行比对。当计算出的哈希值与提取出的哈希值匹配时，John 就破解出了密码。
- 结果展示：**John 完成破解后，会将正确的密码输出，用户可以使用该密码解锁 ZIP 文件。

### 破解过程涉及的具体原理包括：

- 哈希函数：**提取出的哈希值是密码经过加密算法的输出，破解密码的目标就是找到能够生成相同哈希值的原始密码。
- 暴力破解和字典攻击的区别：**暴力破解适用于任何复杂性，但时间较长；字典攻击基于已知的常见密码列表，因此速度较快，但仅适用于较弱的密码。
- 多线程支持：**John the Ripper 支持多线程，能够同时使用多个 CPU 核心进行密码计算，提高破解速度。

## 2. 实验环境

操作系统：Kali Linux 2023.1

破解工具：John the Ripper Jumbo 版本

支持工具：zip2john

实验对象：加密的 ZIP 文件

### 3. 实验步骤及输出示例

#### 步骤 1：准备 Kali Linux 环境

检查 John the Ripper 是否已安装：

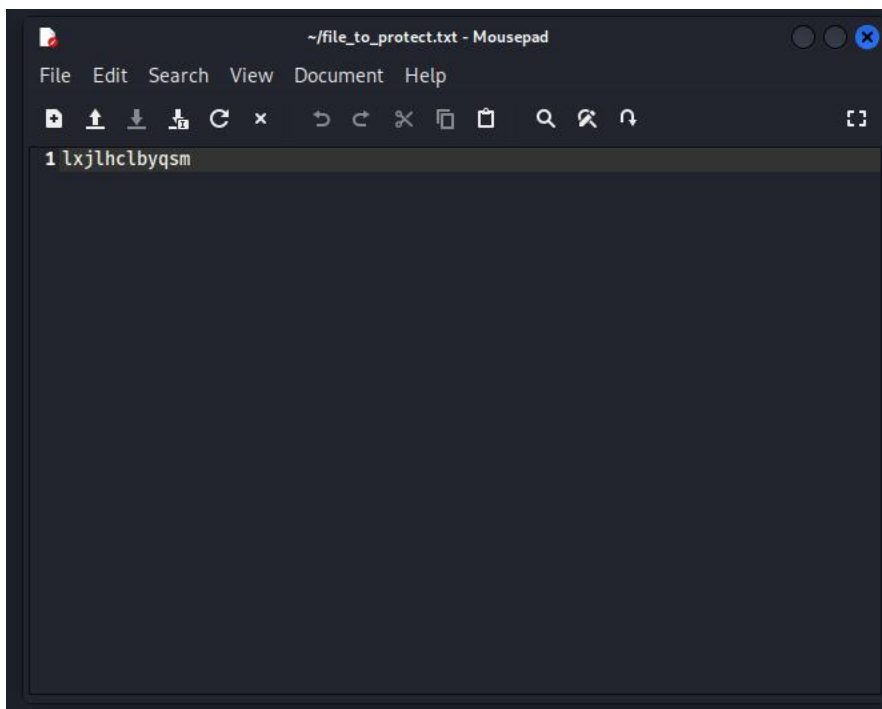
*john*

输出示例：

John the Ripper 1.9.0-jumbo-1+bleeding-<hash> [linux-gnu 64-bit x86\_64 AVX AC]

```
(kali㉿kali)-[~]  
└─$ john  
John the Ripper 1.9.0-jumbo-1+bleeding-aec1328d6c 2021-11-02 10:45:52 +0100 OMP [linux-gnu 64-bit x86_64 AVX AC]  
Copyright (c) 1996-2021 by Solar Designer and others  
Homepage: https://www.openwall.com/john/  
Usage: john [OPTIONS] [PASSWORD-FILES]  
Use --help to list all available options.
```

注意：创建完成后，要在文件中加一些内容，否则文件太小可能导致 zip2john 无法提取密码哈希



#### 步骤 2：创建加密 ZIP 文件

创建一个加密的 ZIP 文件：

*zip -e protected.zip /home/kali/file\_to\_protect.txt*

输出示例：

Enter password: 12345  
Verify password:12345  
adding: file\_to\_protect.txt (stored 0%)

```
(kali@kali)-[~]  
$ zip -e protected.zip /home/kali/file_to_protect.txt  
  
Enter password:  
Verify password:  
adding: home/kali/file_to_protect.txt (stored 0%)
```

### 步骤 3 : 提取 ZIP 文件的密码哈希

使用 zip2john 提取 ZIP 文件的哈希:

*sudo zip2john protected.zip > zip\_hash.txt*

输出示例:

protected.zip->file\_to\_protect.txt PKZIP Encr: 2b chk, TS\_chk, cmplen=38, decmplen=22, crc=XXXXXX

```
(kali@kali)-[~]  
$ zip2john protected.zip > zip_hash.txt  
  
ver 1.0 efh 5455 efh 7875 Scanning for EOD... FOUND Extended local header  
protected.zip/home/kali/file_to_protect.txt PKZIP Encr: 2b chk, TS_chk, cmplen=12, decmplen=0, crc=00000000 ts=5A29 cs=5a29 type=0  
Skipping short file home/kali/file_to_protect.txt
```

### 步骤 4 : 使用 John the Ripper 破解密码

开始使用 John the Ripper 破解密码:

*john zip\_hash.txt*

输出示例:

Using default input encoding: UTF-8  
Loaded 1 password hash (PKZIP [32/64])  
Will run 4 OpenMP threads  
Press 'q' or Ctrl-C to abort, almost any other key for status  
0g 0:00:00:01 DONE (2024-10-22) 0g/s 6571Kp/s 6571Kc/s 6571KC/s 123456..nightmare  
Session completed

```
(kali@kali)-[~]
$ john zip_hash.txt

Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
12345 (protected.zip/home/kali/file_to_protect.txt)
1g 0:00:00:00 DONE 2/3 (2024-10-22 11:38) 12.50g/s 982962p/s 982962c/s 982962C/s 123456..ferrises
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

## 步骤 5：查看破解结果

查看破解出的密码：

*Sudo john --show zip\_hash.txt*

输出示例：

protected.zip:file\_to\_protect.txt:examplepassword

1 password hash cracked, 0 left

```
(kali@kali)-[~]
$ john --show zip_hash.txt

protected.zip/home/kali/file_to_protect.txt:12345 home/kali/file_to_protect.txt:protected.zip::protected.zip
1 password hash cracked, 0 left
```

## 步骤 6：使用自定义字典进行密码破解（可选）

如果使用自定义字典进行破解：

bash

复制代码

*sudo john --wordlist=/path/to/wordlist.txt zip\_hash.txt*

输出示例：

bash

复制代码

Using default input encoding: UTF-8

Loaded 1 password hash (PKZIP [32/64])

Will run 4 OpenMP threads

Press 'q' or Ctrl-C to abort, almost any other key for status

examplepassword (file\_to\_protect.txt)

Session completed

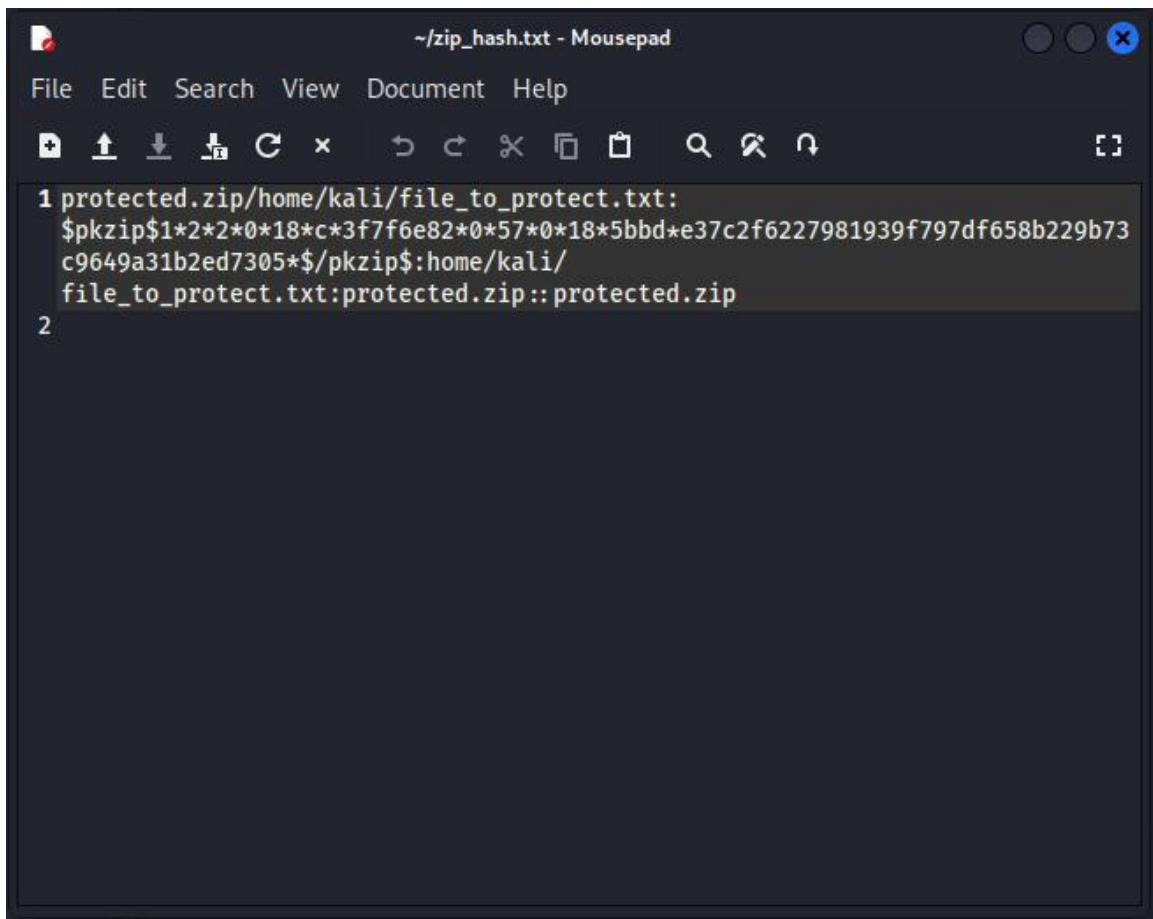
## 4. 实验结果

通过实验过程，John the Ripper 成功破解了 ZIP 文件的密码，密码为 12345。文件解锁后可以正常访问压缩文件中的内容。

## 5. 结果分析

破解速度取决于密码的复杂性和破解方法。在本次实验中，使用默认字典成功破解了较为简单的密码。

查看 zip\_hash.txt 文件：



```
1 protected.zip/home/kali/file_to_protect.txt:
  $pkzip$1*2*2*0*18*c*3f7f6e82*0*57*0*18*5bbd*e37c2f6227981939f797df658b229b73
  c9649a31b2ed7305*$ /pkzip$:home/kali/
  file_to_protect.txt:protected.zip::protected.zip
2
```

其中重要部分的含义如下：

1. 文件名部分：

- `protected.zip/home/kali/file_to_protect.txt`: 是压缩包的路径和文件名，表示 ZIP 文件中包含的文件。

2. 哈希信息部分：

- `$pkzip$1*2*0*18*c3f7f6e82...` 是实际的密码哈希值，`john` 工具通过解析这部分内容来尝试破解密码。它包含了 ZIP 文件加密的关键信息，用于匹配密码。

3. 文件路径部分：

- `:home/kali/file_to_protect.txt:protected.zip::protected.zip`: 这是文件路径信息，用于记录 ZIP 文件及其内容的元数据。

**总结：**`zip_hash.txt` 文件的内容用于存储 ZIP 文件的密码哈希信息，`john` 工具通过读取这些信息并结合字典或穷举法来找到正确的密码。你可以使用该哈希文件继续运行 `john` 工具进行破解。

## 6. 防御策略

为了提高文件的安全性，建议使用复杂密码（包含字母、数字、特殊字符）并定期更换密码。

## 7. 实验总结

本实验展示了如何使用 John the Ripper 在 Kali Linux 中破解 ZIP 文件的密码，并通过该过程提升了对密码强度与安全性的理解。

时间估计：

- **简单密码**（如 6 位以内、只包含数字或字母的密码）：可能几分钟到几十分钟内完成。
- **中等复杂度密码**（8 位左右、混合大小写、数字）：可能需要数小时。
- **复杂密码**（10 位以上、包含符号、大小写和数字）：可能需要数天甚至更长。

