基于 2ROT13 的加密技术

数字加密实验室

摘 要

这篇论文描述数字加密实验室关于一种早前被低估的加密算法，基于 13 的循环加密（Double-Rotate by 13）算法——2ROT13 加密算法。我们将详细分析该算法，近距离的基础该加密算法的实际问题，并展望一下该加密算法在未来可能的应用。

批注 [ZY1]: 本

批注 [ZY2]: 不通

主要内容

1. 简介

自从人类一诞生，人们已经开始使用各种不同的方法来加密信息了。有些方法希望每个人都能理解，比如自然语言，俗语，另外一些方法则对有限的人有用处，比如“上帝语言”，这种语言只有在印度种姓精英巫师或者不能让敌人知道的秘密战略计划中使用。

一种最著名已经被很好得证明出来的加密算法是，“凯撒密码（Caesar Code）”，这是一种简单的编码，字母表中任一个字母与另一个字母表中的字母对应，n 个位置一循环。在此，n 是密键（secret key），可以同时把人类可读的消息加密成密文和把密文解密成人类可读的消息。

批注 [ZY3]: 凯撒密码还是凯撒码，二者只能取一

批注 [ZY4]: 密钥，尽量参考一些专业的说法，可以到 http://dict.cnki.net 查询

在阅读时，这篇文论的作者并不知道任何关于提出这种简单而有效的系统的加密弱点的科学研究论文。很多人都认为该算法是“明显的”而不安全，但这总的来说是不科学的，而且不能被认真的分析所接受。然而，凯撒码已经不断的被探讨了超过两千年，并且被数字加密实验室的工作人员认为是安全的。

随着时间的推移，其他更复杂的加密系统被发明了出来，但是大部分已经被破解了，比如英格玛机（Enigma machine）、DES 加密算法、RSA-1024[WLB03]。因此，作者认为凯撒码当前比其他提到的加密算法更安全。

批注 [ZY5]: 参见 http://zh.wikipedia.org/w/index.php?title=%E6%81%A9%E5%B0%BC%E6%A0%BC%E7%8E%9B%E5%AF%86%E7%A0%81%E6%9C%BA

2. 2ROT13 加密算法的描述

2ROT13 加密算法是基于 ROT13 加密算法的。ROT13 加密算法是一种特殊的凯撒码，它使用固定的键 13。ROT13 已经被广泛使用了 20 多年，最适合使用于部分新闻网、电子邮件信息的加密。如今，大部分类 UNIX 的操作系统都包含一个 ROT13 和（或）凯撒码的实现。因此，可以认为 ROT13 普遍存在于 IT 世界中。

批注 [ZY6]: 也要写成“密钥”大家才会懂，别的也要改

在上文简要提到的 ROT13，是基于这样一种原则：字母表中的每个字母映射到另一个特定的字母。在 ROT13 中，被映射到的字母是原字母开始 13 个之后的字母。选择 13 作为

位移是因为拉丁字母表一共有 26 个字母。这产生了一中很好的效果，字母 A 映射到字母 N，同时字母 N 映射到字母 A。这里清晰得显示了 ROT13 是一个以 13 为键的秘密键加密算法（记住不要告诉任何人）。

其他加密算法的键值是根据如下方法设计的：加密数据的过程要做几次，叫做“循环”。基于循环的加密算法有 AES、DES 或者 3DES，3DES 包含 3 次 DES 的循环。因此可以认为循环次数越多安全性越高。

基于以上的思想，2ROT13 加密算法也得到了发展。它使用 ROT13 对密文进行两次循环加密。由于 ROT13 的特殊性质，循环的次数必须是偶数次，否则算法只提供和 ROT13 一样的安全度。如下有助于记忆的韵文很好的描述了这种情况：

不能被 2 整除的数对你也不是最好的

（Where you can’t divide by two, is not very goog for you）

好的实现有 2ROT13、4ROT13、6ROT13 或者 2048ROT13。但是到目前为止，作者没有看到任何比 2 次循环多的应用必要。差的实现有 ROT13、3ROT13、7ROT13 或者 1697ROT13。

批注 [ZY7]: 好多说法……

批注 [ZY8]: 原文如此？

3. 2ROT13 的算法实现

当前，一种 2ROT13 的实现已经存在，被称为 *Pretty Good Double ROT13*——short PG2ROT13P——而且应该是声名狼藉的 *Pretty Good Privacy* 密码学工具包的派生方法。它同时实现了加密和解密密码文件，并被写入了 Perl 程序设计语言。你可以在本论文的最后一节中找到源代码。

4. 2ROT13 未来的发展目标

2ROT13 的发展才刚刚开始，2ROT13 需要变得更普遍。这是意味着我们将“胁迫”所有的 Linux 零售商和经销商秘密的用 PG2ROT13P 代替 GnuPG 以帮助 PG2ROT13P 变得更普遍。其他的主意是在 SSL 和 SSH 中实现 2ROT13 加密组件，让他们成为*默认的*。数字加密实验室全体成员相信这将让这个世界变的更加安全和更加和平。长期的目标是让欧盟议会和美国国会通过如下法律：要求所有个人信件，明信片和面对面的交谈都必须用 2ROT13 加密。

批注 [ZY9]: 中文一般没有斜体，可以用楷体代替。

5. PG2ROT13P 的源代码

```
#!/usr/bin/perl
#pretty good double-rot13 privacy-PG2ROT13P
#(c)2005ak,#mumcryptolabs
$header = "-----BEGIN2ROT13MESSAGE-----";
$footer = "-----END2ROT13MESSAGE-----";
sub rot13($){
my $x = shift;
$x = ~tr/a-zA-Z/n-za-mN-ZA-M/;
return $x;
}
subdo_2rot13_encrypt(){
@lines=<STDIN>;
print"$header\n";
foreach$line(@lines){
```

```
foreach$i(1..2){# two rounds of ROT13
$line=rot13($line);
}
print $line;
}
print "$footer\n";
}
subdo_2rot13_decrypt(){
print STDERR "pg2rot13p:goaheadandtypeyourmessage...\n";
@lines=<STDIN>;
if($lines[0] ne "$header\n"or$lines[$#lines]ne"$footer\n"){
print STDERR "Sorry,thisisnotavalid2ROT13message\n";
exit(1);
}
shift(@lines); pop(@lines);
foreach$line(@lines){
foreach$i(1..2){
$line=&rot13($line);
}
print$line;
}
}
subusage(){
printSTDERR"usage:pg2rot13p[-e|-d]\n";
exit(1);
}
if($#ARGV==-1or$ARGV[0]eq! ¯ -d! ¯
&do_2rot13_decrypt();
}elsif($ARGV[0]eq! ¯ -e! ¯
&do_2rot13_encrypt();
}else{
&usage();
}
```

[WLB03]Weis,Lucks,Bogk:Sicherheitvon1024bitRSASchl ¨1sselngef?hrde
http://cryptolabs.org/rsa/WLBrsaDuD.pdf