看雪-2014 APP应用攻防竞赛第二阶段第1题（攻击篇）解析

转载

weixin_33754065 于 2014-10-30 14:55:00 发布 44 收藏
文章标签： php 嵌入式 移动开发
原文链接：https://my.oschina.net/auo/blog/338868

0x0

29号晚上才知道看雪有个比赛，分析出来已经过了提交答案的时间了。还是记录到博客里来吧。

0x1

链接：http://bbs.pediy.com/showthread.php?t=193755

平台：Android

类型：CrackMe

0x2

Java层代码很简单，输入name和passwd后，转到so中的验证函数进行处理，APK也没有做反调试保护，主要考的是动态库调试的基本能力和arm汇编知识。

简单的说下验证逻辑：

1、对name和passwd的长度进行限制（实际上由后面具体的逻辑判断，name长度应该固定为9）：

6 <= strlen(name) <= 14

12 <= strlen(passwd) <= 30

2、并且passwd[3] = passwd[7] = passwd[11] = “-”，也就是说输入的passwd是类似序列号的形式，每3个用“-”分割；

3、初始化0x100大小的表，表中数据固定；

4、将passwd的值作为下标，查询表中对应数据，并进行一些移位异或处理；

5、上一步得出的结果即为passwd对应的name，然后与我们输入的name做对比，相同即验证成功、否则验证失败；

0x3

具体的逻辑这里不在记录，下面给出注册机，可能有些问题，时间有限就没有继续分析了：

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

void initKeyTab(unsigned char *keyTab)
{
 memset(keyTab, 0x80, 0x100);
```

```
 int j;
 int i;

 j = 0x41;
 for(i = 0x0; i < 0x1A; ++i)
 {
  keyTab[j + 1] = i;
  ++j;
 }

 j = 0x62;
 for(i = 0x1A; i < 0x34; ++i)
 {
  keyTab[j] = i;
  i = i << 0x18;
  i = i >> 0x18;
  ++j;
 }

 j = 0x31;
 for(i = 0x34; i < 0x3e; ++i)
 {
  keyTab[j] = i;
  i = i << 0x18;
  i = i >> 0x18;
  ++j;
 }

 keyTab[0x2C] = 0x3E;
 keyTab[0x30] = 0x3F;
 keyTab[0x3E] = 0x0;
 keyTab[0x0] = 0x1;

 for(i = 0; i < 0x100; ++i)
 {
  printf("%02x ", keyTab[i]);
  if((i+1) % 16 == 0)
  {
   printf("\n");
  }
 }
 printf("\n");
}

void getPasswd(unsigned char *name, int name_length, unsigned char *keyTab, unsigned char *passwd)
{
 int i;
 int k = 0;
 char mid[4];
 for(i = 0; i < name_length; i += 3)
 {
  int j;

  mid[0] = name[i] >> 0x2;
  mid[1] = ((name[i] << 0x4) & 0x3F) ^ (name[i + 1] >> 0x4);
  mid[2] = ((name[i + 1] & 0x0F) << 0x2) ^ (name[i + 2] >> 0x6);
  mid[3] = name[i + 2] & 0x3F;

  for(j = 0; j <= 0x100; ++j)
```

```
  {
   if(keyTab[j] == mid[0])
   {
    passwd[k] = j - 1;
   }
   if(keyTab[j] == mid[1])
   {
    passwd[k + 1] = j - 1;
   }
   if(keyTab[j] == mid[2])
   {
    passwd[k + 2] = j - 1;
   }
   if(keyTab[j] == mid[3])
   {
    passwd[k + 3] = j - 1;
   }
  }

  k += 4;
 }
 printf("passwd: %s\n", passwd);
}

void main()
{
 unsigned char name[] = "123456789";
 int name_length = strlen(name);
 unsigned char passwd[12];
 unsigned char keyTab[0x100];
 initKeyTab(keyTab);
 getPasswd(name, name_length, keyTab, passwd);
}
```

另外给出几组可用的：

123456789　MTI-zND-U2N-zg5

abcdefghi　YWJ-jZG-VmZ-2hp

转载于:https://my.oschina.net/auo/blog/338868