2016 alictf Timer writeup

转载

weixin_30514745 于 2017-07-01 23:12:00 发布 82 收藏
文章标签： 移动开发
原文链接：http://www.cnblogs.com/zhaijiahui/p/7103776.html

Timer-smali逆向

参考文档：http://blog.csdn.net/qq_29343201/article/details/51649962

题目链接： https://pan.baidu.com/s/1jlNx7Fo （在里面找相应的名字就行）

题目描述： 每秒触发一次计算，共有200000秒，答案参与计算，不可能等待下去。

使用工具：Android Killer，jadx-gui

解题方法有多种，我参照网上的一种方法。通过对native层，代码的还原，计算出200000秒后的关键变量k，传入主调用，得到答案

解题过程：

System.currentTimeMillis()相当于是毫秒为单位，获取当前时间

```
public class MainActivity extends AppCompatActivity {
    int beg = (((int) (System.currentTimeMillis() / 1000)) + 200000);
    int k = 0;
    int now;
    long t = 0;
```

先是把单位变成秒，然后加上200000秒

第二块逻辑，用于筛选秒数的自定义函数，可以忽略，之后按代码逻辑就行

```
    public static boolean is2(int n) {
        if (n <= 3) {
            if (n > 1) {
                return true;
            }
            return false;
        } else if (n % 2 == 0 || n % 3 == 0) {
            return false;
        } else {
            int i = 5;
            while (i * i <= n) {
                if (n % i == 0 || n % (i + 2) == 0) {
                    return false;
                }
                i += 6;
            }
            return true;
        }
    }
```

第三块逻辑，主程序段，包含最后打印flag值的代码段

```
36      protected void onCreate(Bundle savedInstanceState) {
37          super.onCreate(savedInstanceState);
38          setContentView((int) R.layout.activity_main);
39          final TextView tv1 = (TextView) findViewById(R.id.textView2);
40          final TextView tv2 = (TextView) findViewById(R.id.textView3);
41          final Handler handler = new Handler();
68          handler.postDelayed(new Runnable() {
46              public void run() {
47                  MainActivity.this.t = System.currentTimeMillis();
48                  MainActivity.this.now = (int) (MainActivity.this.t / 1000);
49                  MainActivity.this.t = 1500 - (MainActivity.this.t % 1000);
50                  tv2.setText("AliCTF");
52                  if (MainActivity.this.beg - MainActivity.this.now <= 0) {
53                      tv1.setText("The flag is:");
54                      tv2.setText("alictf{" + MainActivity.this.stringFromJNI2(MainActivity.this.k) + "}");
                    }
                    MainActivity mainActivity;
56                  if (MainActivity.is2(MainActivity.this.beg - MainActivity.this.now)) {
57                      mainActivity = MainActivity.this;
                        mainActivity.k += 100;
                    } else {
60                      mainActivity = MainActivity.this;
                        mainActivity.k--;
                    }
62                  tv1.setText("Time Remaining(s):" + (MainActivity.this.beg - MainActivity.this.now));
65                  handler.postDelayed(this, MainActivity.this.t);
                }
            }, 0);
        }
```

分成三段来看：

第一段：super继承，调用页面设计框架，实例化handler函数

第二段：t的作用在于将时间一点一点减少，beg的值在前面，代表一开始时间，now代表现在的时间

第三段：当差值为0或小于打印flag，否则调用is2函数判断，true，k+100，false，k-1

所以有了下面的解密程序

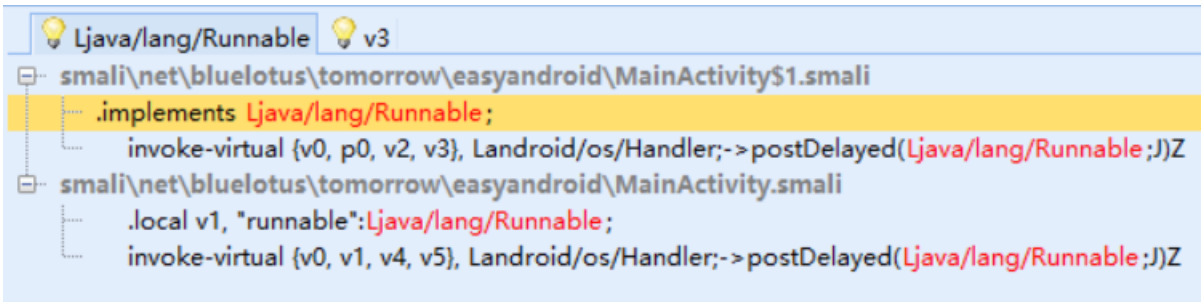
```python
def is2(n):
    if (n <= 3):
        if (n > 1):
            return True
        return False
    elif (n % 2 == 0 or n % 3 == 0):
        return False
    else:
        i = 5
        while (i * i <= n):
            if (n % i == 0 or n % (i + 2) == 0):
                return False
            i += 6;
        return True;

def main():
    time = 200000
    k = 0
    while time > 0:
        if is2(time):
            k += 100
        else:
            k -= 1
        time -= 1
    print(k)


if __name__ == '__main__':
    main()
```

将得到的k值传给native代码，此时的k是经过200000运算的k，传进去直接返回结果

在传入native的时候要注意，首先要确定run函数的位置，这一块方法结束了，没找到逻辑，

```
190
191     .line 68
192     .local v1, "runnable":Ljava/lang/Runnable;
193     const-wide/16 v4, 0x0
194
195     invoke-virtual {v0, v1, v4, v5}, Landroid/os/Handler;->postDe
196
197     .line 74
198     return-void
199 .end method
```

通过字符串搜索找到run函数的位置，在MainActivity$1.smali中



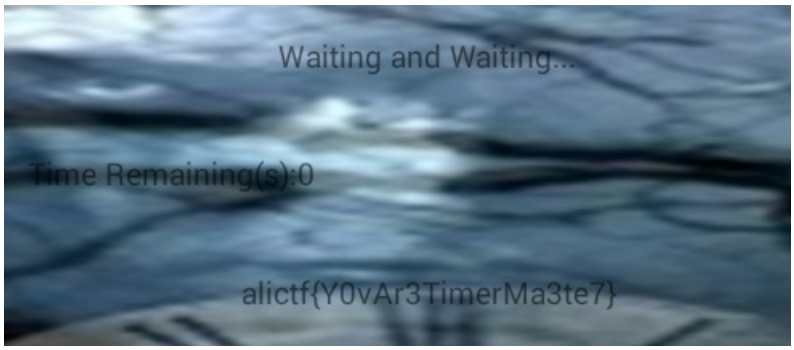
直接将k传入不用运行200000次，需要更改"The flag is:"之前的关键跳转

if-gtz v0, :cond_0 改为 if-ltz v0, :cond_0

根据分析，**k**的值是**v3**的值，所以要在获取**v3**之后，修改**v3**的值

```
139     iget v3, v3, Lnet/bluelotus/tomorrow/easyandroid/MainActivity;->k:I
140
141     const v3,0x18aa00
142
143     invoke-virtual {v2, v3}, Lnet/bluelotus/tomorrow/easyandroid/MainActivity;->stringFromJNI2(I)Ljava/lang/String;
144
```

重打包，得到flag



嘿嘿，又做出来一道。

转载于:https://www.cnblogs.com/zhaijiahui/p/7103776.html