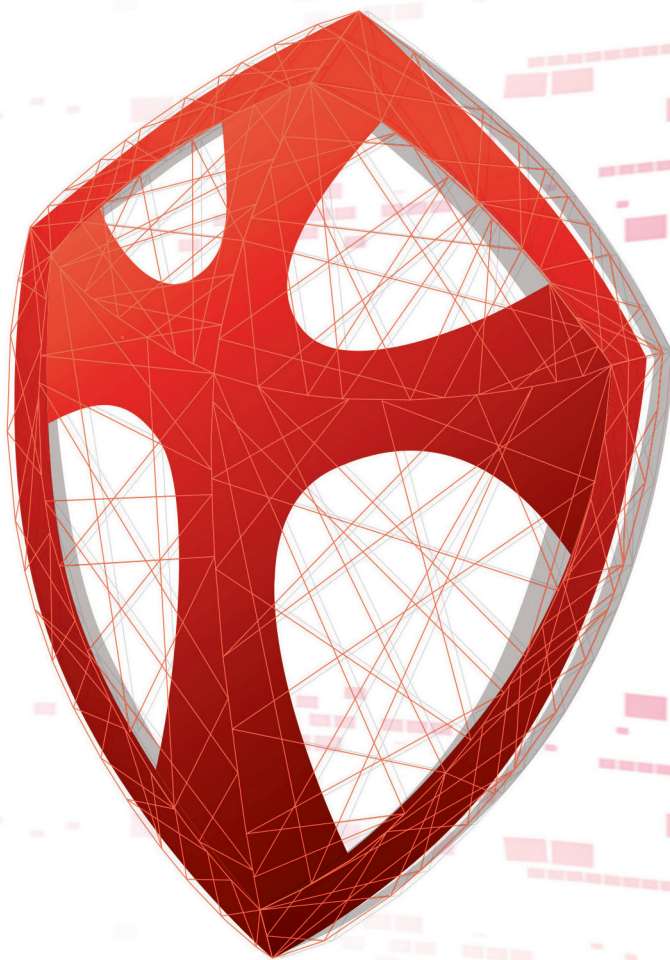


问道安全

2023年度专刊

勒索病毒
Ransomware



序言

数字化信息的快速发展和广泛应用催生了一系列网络安全风险，勒索组织更是乘着数字化转型的东风，不断更新其技术手段，迅速成为网络安全威胁中影响最为恶劣、坚持对抗升级的一种高级网络犯罪手段。近年来，勒索攻击一直占据网络攻击类型的榜首，数量占比高达 20 % 以上。我们也看到，勒索组织针对的对象，从个人到组织、从大型企业到关键基础设施、从内网到云端，在持续调整。其技术和模式也在持续升级，包括新型技战法、Oday 的使用，以及从 RaaS 到类 APT 的转变。可以说，勒索软件，已经发展为网络安全最严重的威胁之一。

通过新华三攻防实验室历年对勒索组织的持续跟踪、观察、分析，勒索软件的发展趋势，我们可以总结为以下三大特点：

一、勒索呈现运营模式清晰化、分工精细化的趋势

RaaS 的出现，使勒索攻击走向商业化，可以通过会员、订阅等方式，向其他攻击者售卖勒索攻击相关服务。同时 RaaS 运营模式有着明确的组织分工，软件开发、渗透攻击、赎金谈判等等都有专人负责。正由于 RaaS 带来的专业化、精细化的分工，大大降低了勒索攻击的准入门槛，增加了勒索攻击的防护难度，使得勒索攻击势力得以迅速蔓延。

二、勒索软件呈现高度定制化、目标高度定向化的趋势

回顾勒索软件的演化历程，它由早期作为一种简单的木马或一段恶意代码已经发展为一种经过精心设计用于获取非法利益的高度复杂的工具，重点聚焦在传播感染、敏感信息窃取、加密算法设计、防御规避，勒索兑现等方面。在全球范围内人工智能和机器学习迅猛发展的背景下，勒索软件开始利用这些新的技术、新型漏洞或者难以检测的攻击向量进行更加智能化的攻击，这意味着攻击者可以根据受害者的行为模式和信息进行定制化攻击，针对性地选择特定的行业、组织或个人等作为攻击目标，加大勒索成功几率。

三、勒索软件带来的恶性影响呈现逐年上升趋势

随着勒索攻击的不断发展，其带来的恶性影响已经从个人、企业层面的数据风险逐渐向其客户和合作伙伴蔓延。这种由勒索软件引发整个产业生态的涟漪效应不仅损害企业的声誉，更会侵蚀着客户对数字化环境的安全感和信任力。另外，勒索软件攻击不仅针对用户、企业，更是发展到关键基础设施，如医疗机构、能源供应系统和交通系统，直接影响到社会运作和公众的生命安全。如今，勒索攻击已经从早期网络滋扰发展成为危害经济稳定、公众安全和社会发展并不断升级的安全新挑战。

新华三安全攻防实验室将持续跟进世界活跃勒索组织，对其技战法和运营模式的迭代进行深度解读，通过各种分析报告的形式，提供专业观点和数据参考。同时，也会秉承“以攻促防”的核心理念，积极研究勒索软件的对抗思路，迭代防御水平，并依托云能力中心，为新华三安全产品和解决方案能力持续赋能。

梁力文

新华三集团安全研发副总裁、安全攻防实验室主任

2023 年 6 月

目录

CONTENTS

01 第一章 / 业内发展趋势分析

勒索攻击现状与趋势 文 / 孙秋男02

07 第二章 / 关键技术

Hive 勒索组织分析 文 / 秦旭08

LockBit 勒索组织分析 文 / 鲍旭丹17

Revil 勒索组织分析 文 / 汪林28

40 第三章 / 应用实践方案

云 - 网 - 端 - 管融合, 构筑勒索防护新防线 文 / 张宇41

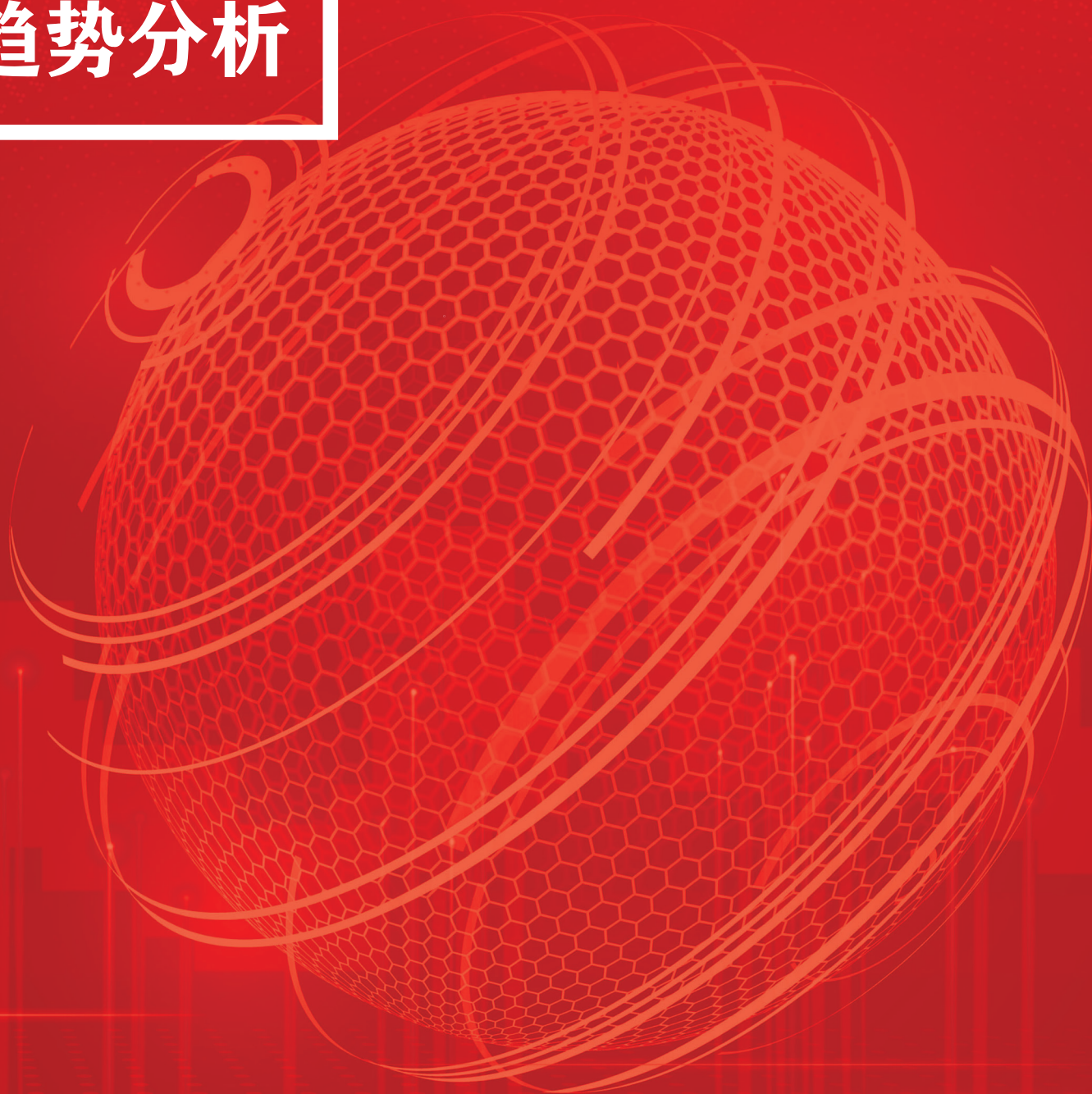
44 第四章 / 测论

勒索病毒测试之道 文 / 霍龙45

Chapter One

第一章

业内发展
趋势分析



勒索攻击现状与趋势

文 / 孙秋男

1 导语

勒索攻击是指网络攻击者通过对目标数据进行加密，导致核心业务停摆，以此要挟受害者支付赎金进行解密的一种网络攻击方式。随着全球进入数字化时代，互联网流量、接口暴增，数据的价值日益凸显，勒索攻击应运而生。起初勒索攻击发展比较缓慢，而随着加密数字货币的兴起，其交易速度快、交易成本低且具有高度匿名性的特点，使之成为了交付赎金的理想途径，促使勒索攻击发展愈演愈烈。而后 RaaS 模式的面世，大大降低了技术人员参与到勒索牟利行业的门槛，勒索攻击势力得以迅速蔓延，已然成为网络安全中的最大威胁之一。

2 勒索攻击现状

近年来，全球各地针对性、极具破坏性的勒索攻击事件频频发生，攻击手段层出不穷、赎金规模也是屡创新高。如今，勒索攻击已从早期网络滋扰发展成为危害社会运作、经济稳定和公共安全并不断升级的全球新挑战。

据 IBM 安全团队发布的《X-Force 威胁情报指数》统计，近年来，勒索攻击一直占据网络安全中主要攻击类型的榜首。2021 年的占比为 21%，比 2020 年的 23% 稍有下降，但勒索攻击总数与 2020 年同比保持稳定，如图 1 所示。

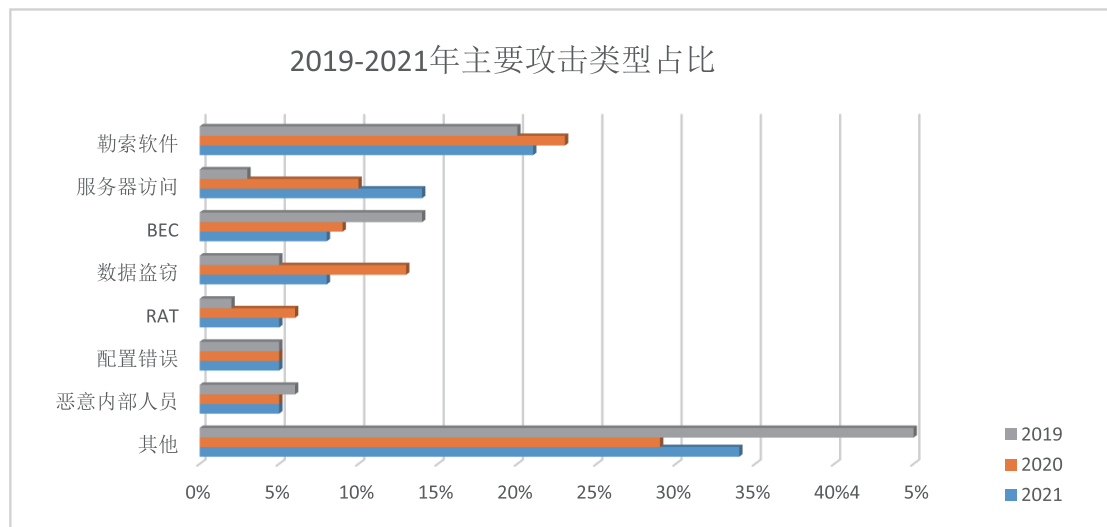


图1 2019-2021年主要网络攻击类型占比（数据来源：X-Force威胁情报报告）

2.1 行业分布情况

从图2勒索攻击行业分布上来看,制造、服务、金融贸易三个行业所受的勒索攻击最为严重。究其原因,主要在于这三个行业的实体数量巨大、现金流充裕、与人们日常生活关系更为紧密、交付赎金的可能性更高,因此更受勒索攻击者的青睐。

2022年勒索攻击事件行业分布

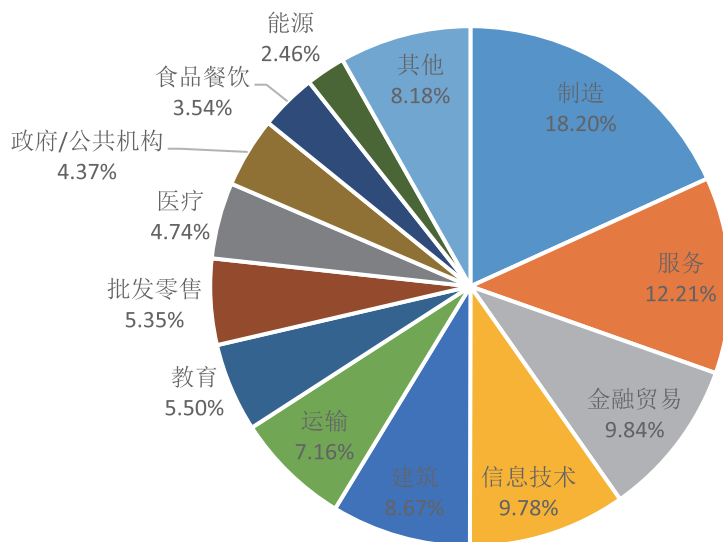


图2 2022年勒索攻击事件行业分布（数据来源：darktracer）

2.2 地域分布情况

从图3地域分布数据统计中可以看出,美国为受勒索攻击影响最为严重的国家,占比近四成,远超其他国家或地区。从整体区域分布上来看,攻击者更倾向于经济发达地区和人口密集地区。原因可能为经济发达地区更具有赎金交付能力,而人口密集地区数据被破坏后可能造成更严重的社会负面影响,所以这些地区更具有赎金交付意愿。

2022年勒索攻击事件发生所在区域TOP10

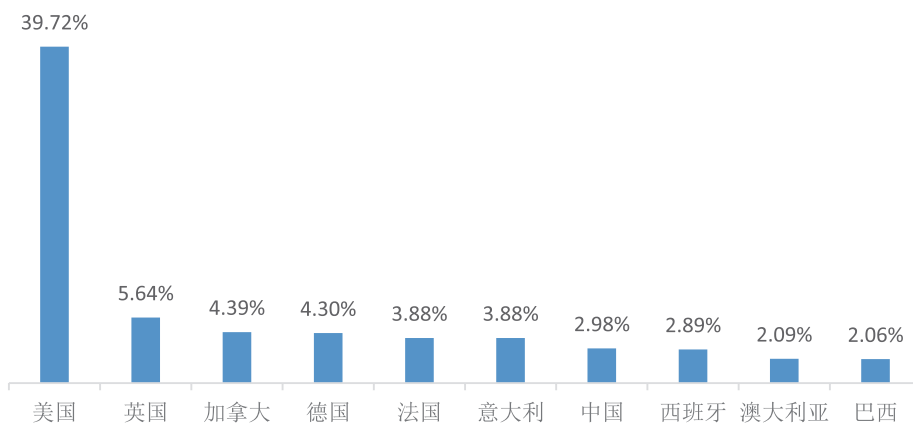


图3 2022年勒索攻击事件区域分布TOP 10（数据来源：darktracer）

2.3 家族分布情况

图4统计了2022年勒索病毒家族占比，从家族分布上来看，LockBit家族为今年最活跃家族，占比高达三成。该家族于2019年9月面世，历经2次大版本更新，于2022年6月推出LockBit 3.0版本。该家族凭借自身较为迅速的版本迭代和完善的运营模式，始终位于近些年的活跃勒索家族行列。BlackCat家族于2021年11月被首次披露，是首个基于跨平台语言Rust编写的新兴勒索家族，并因自身高度定制化的特点迅速赢得市场。而臭名昭著的Conti勒索家族，由于内部数据泄露、执法机构打击等多重原因，于2022年5月份关闭了数据泄露网站等基础设施，因此由其发起的勒索攻击事件同比大大减少。

2022年勒索病毒家族占比

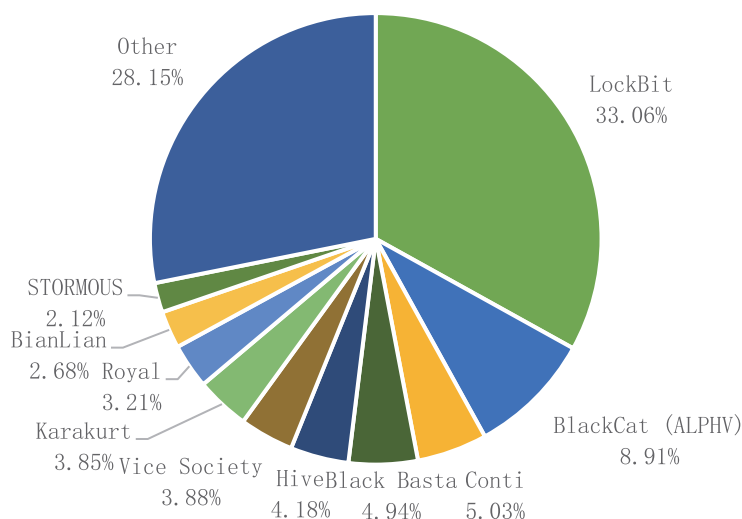


图4 2022年勒索攻击事件家族分布（数据来源：darktracer）

3 勒索攻击发展趋势

3.1 勒索攻击“商业化”发展

RaaS 是勒索软件运营商与附属机构之间的一种商业模式，这种模式使得附属机构通过付费即可获得勒索软件并发起攻击，大幅降低了勒索软件攻击的技术门槛并扩大了攻击范围。而随着 RaaS 模式的逐渐成熟，勒索组织也有了明确的分工，开发运营、渗透攻击、赎金谈判等都有专人负责跟进。RaaS 模式“成熟”的运营、“专业”的分工，是近年来勒索攻击事件泛滥的重要原因之一，如图5展示了 RaaS 勒索组织的工作模式。

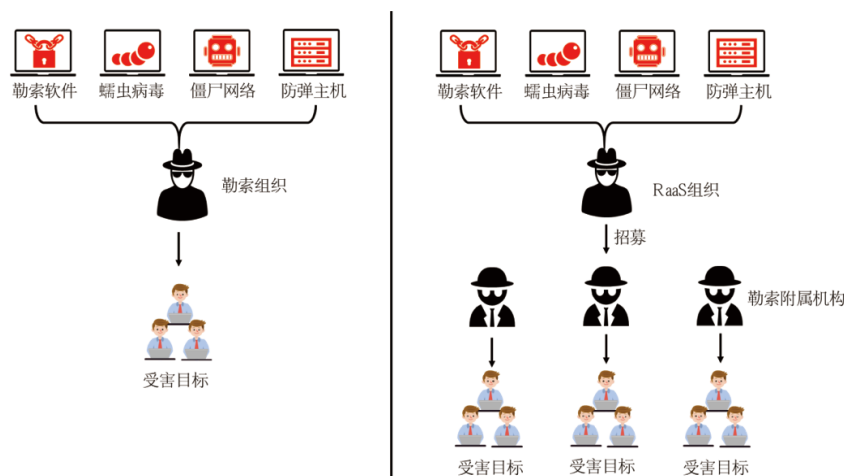


图5 RaaS化勒索组织工作模式

3.2 多重勒索手段叠加利用

传统意义上，勒索攻击最基本的手段是通过向目标系统部署勒索软件，对重要数据文件进行加密，直接破坏数据的可用性，从而威胁受害者缴纳赎金换取解密密钥。随着勒索软件的流行，大多数企业会通过异地备份或云端备份重要数据的形式来应对勒索攻击。然而传统的加密手段并不总能奏效，很多的勒索组织开始采用双重甚至多重勒索策略来对受害者施压，提高勒索成功率。目前最常见的是双重勒索策略，勒索组织专门建立托管被盗数据的站点，用于对外展示所窃取的高价值、高机密的数据。如果受害者拒绝支付赎金，攻击者则会在站点上公开受害者被窃数据，向其竞争对手、合作伙伴、客户、媒体等提供访问权限，甚至会在匿名网络中组织拍卖，这无疑会让受害者承受更大的压力，大大增加支付赎金的可能性。新兴勒索组织 Babuk 建立勒索站点后，甚至公开宣称要成为纯粹的窃密勒索者，如图 6 所示。

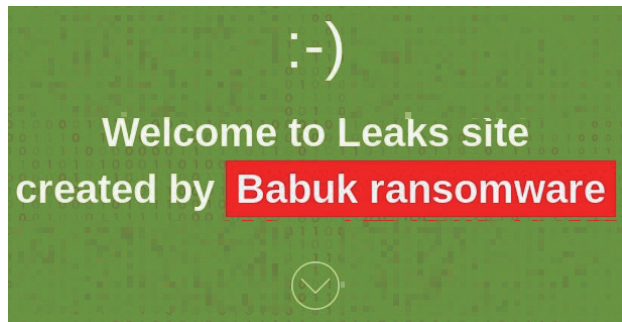


图6 勒索组织Babuk泄露站点界面（图源网络）

在双重勒索基础上，部分攻击者还会采用其他方式对未交付赎金的受害者进一步施压，例如向受害者网站发起 DDoS 攻击等。2021 年初，Avaddon 曾在他们的数据泄露站点发表声明，如图 7，某受害者网站目前正处于 DDoS 攻击威胁中，将持续攻击直到受害者主动联系他们，同时该组织表示还有多种手段给受害企业制造其他问题。

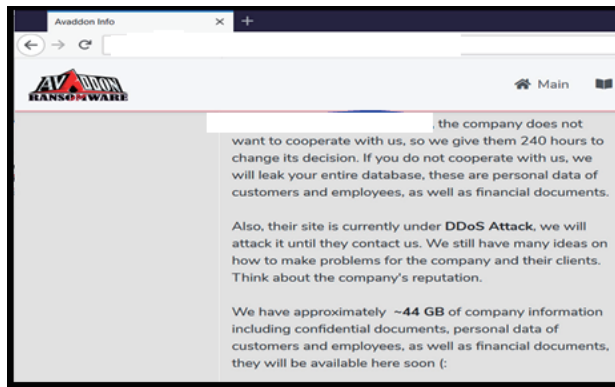
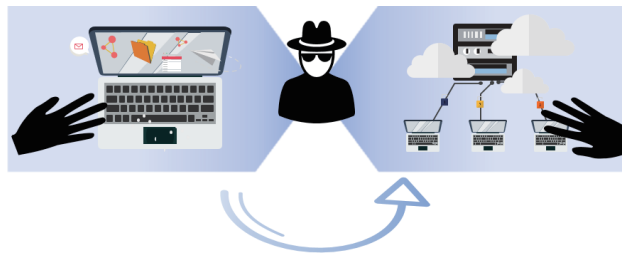


图7 勒索组织Avaddon威胁声明（图源网络）

3.3 攻击目标转向云上业务

企业上云是数字化发展的重要途径，随着全球数字化进程加速，已经有越来越多的业务和数据都部署在云端。在此背景下，勒索组织为了扩大攻击范围，也已经将黑手伸向云端，由此产生一个新术语——Ransomcloud。目前，活跃勒索组织 Babuk、BlackMatter 已经衍生出 Linux 版本的勒索软件，能够针对 ESXi 等虚拟化管理平台进行加密勒索。与此同时，勒索组织针对云上新技术（如 SDN、NFV 等）的安全设计缺陷或漏洞的攻击也愈加频繁。据统计，在 2021 年所有危害云环境系统的攻击案例中，勒索软件与挖矿软件占 50% 以上。



3.4 利用加密货币混合服务躲避赎金追踪

加密数字货币因其高度匿名性、交易速度快、交易成本低等特点已成为勒索攻击的主要赎金支付方式。但当勒索犯罪者获得数字货币支付的赎金后，执法部门和研究人员可以轻松追踪到赎金流入哪一个加密货币钱包，从而进行拦截和封禁。美国燃油管道运营商 Colonial Pipeline 向攻击者支付 75 枚比特币后，被执法人员追回 63.7 枚比

特币，约占总支付金额的 85%。因此，攻击者开始转向求助于加密货币混合服务，又称“混币服务”。该服务中所有加密货币交易都进行拆分、混淆处理，直到发送到指定地址的加密货币总数额达到发送者要求，而执法机构难以追踪赎金的最终流向。因此，网络犯罪分子已经开始普遍采用混合服务来寻求级别更高的隐私保护，并减缓调查人员的追踪速度。

3.5 供应链攻击将成为勒索攻击的重要手段

供应链攻击是指在供应关系中，攻击者针对上游提供服务或软件的供应商发起攻击，从而影响到下游用户的一种攻击手段。作为一种新型威胁，供应链攻击具有突破口多、破坏力强、波及面广、隐蔽性高等特点，已成为最难防御的网络攻击手段之一。正是由于这些特点，勒索组织也已开始使用此种攻击手段。2021 年 7 月，MSP 解决方案提供商 Kaseya 的 VSA 远程管理和监控产品遭受勒索组织 REvil 0day 的漏洞攻击，并被利用向其客户推送恶意的软件更新。瑞典最大连锁超市 Coop 的供应商 Visma Esscom 受到这次攻击的影响，致使 Coop 连锁超市收银系统故障，被迫关闭了近 800 家门店。随后 REvil 在其暗网主页中称，他们在此次事件中影响了超过百万的系统，并要求为其通用解密器支付价值 7000 万美元的比特币，可见此次供应链攻击影响之大。未来，供应链攻击将成为勒索攻击的重要手段之一。

4 结语

随着勒索软件的不断发展，其危害程度逐步加深，攻击广度也在不断扩大。近年来发生的大量造成严重社会影响的勒索攻击事件，不仅给安全研究人员带来新的考验，也给互联网中海量的潜在受害者敲响了一次警钟。可以预见，在暴利的驱使下，即使要面临执法机构的打击，勒索攻击短期内也并不会消失，所以针对勒索攻击的防护已经是网络安全建设中的重要一环。而作为安全厂商，需要把握勒索攻击的最新态势，实时更新最佳解决方案，建立一套健全、持续、可靠的安全防护体系，保障广大用户的网络资产安全。



Chapter Two

第二章

关键技术





HiveRansomware

Hive 勒索组织分析

文 / 秦旭

1 组织概述

1.1 背景

Hive 勒索组织于 2021 年 6 月被发现，该组织以 RaaS 模式运营，使新手也能发起勒索攻击。勒索时采用双重勒索，即窃取数据后再加密文件索要赎金，如果不支付赎金则在泄露数据网站公开数据，图 1 为 Hive 勒索组织的泄露信息网站，被攻击者如果在指定时间内未能支付赎金，被窃取的数据就会被公开。从攻击目标看，Hive 和其他勒索组织有一个明显区别，部分组织明确声称不会攻击医院、学校、养老院等机构，以避免对社会造成巨大伤害，从而减少执法部门的关注，而 Hive 勒索组织的攻击目标选择则完全没有这样的限制，根据 FBI 消息，截至 2022 年 11 月，Hive 勒索组织攻击了全球范围内约 1300 家公司，已收到赎金约 1 亿美元。

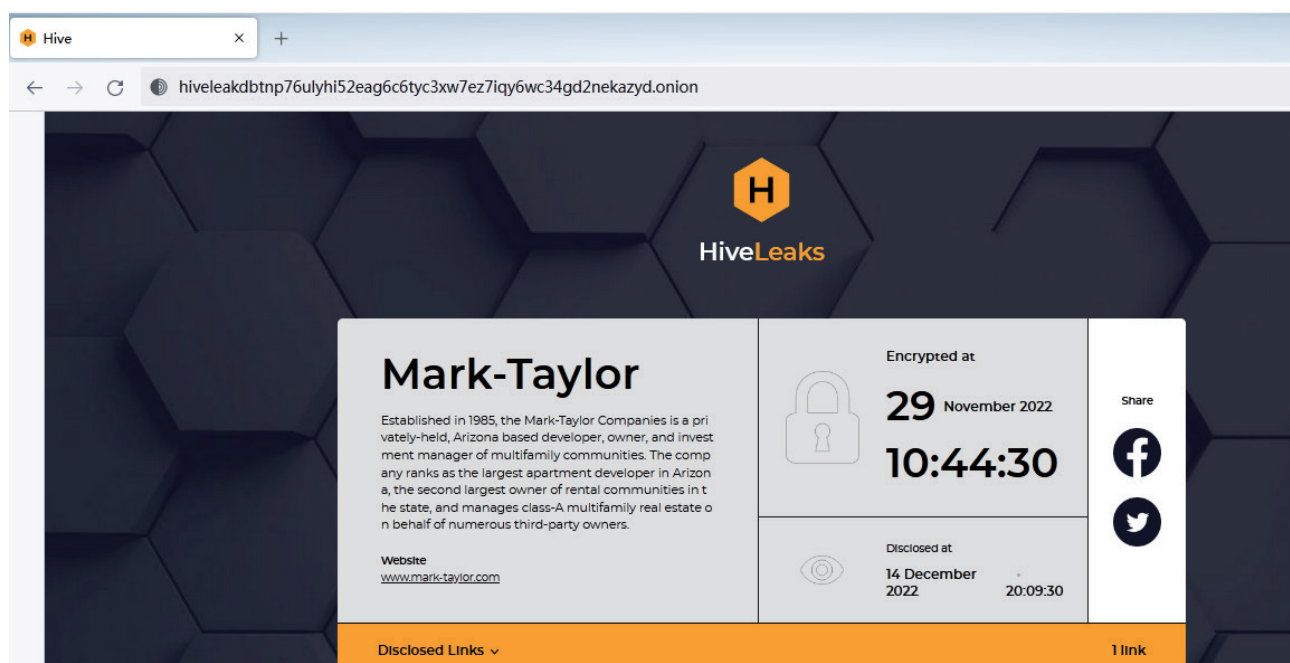


图1 Hive泄露信息网站

1.2 著名事件

2021 年 8 月 15 日星期一，Hive 勒索组织攻击了一个非营利性的综合卫生系统 Memorial Health System，该系统在俄亥俄州和西弗吉尼亚州经营着许多医院、诊所和医疗保健站点，受攻击影响，相关医院被迫使用纸质图表，外科手术和使用放射学仪器的检查项被迫停止。医院在联邦调查局及其保险公司的帮助下与攻击者谈判，最终支付了赎金以获取解密密钥，医院系统在周末得以恢复上线。

2022 年 3 月 19 日，加利福尼亚州的 Partnership HealthPlan 遭到了 Hive 的勒索攻击，该组织也是一家非营利性的社区医疗保健组织，通过和政府签订合同，为 14 个县的 60 多万人提供卫生保健服务。Hive 勒索组织声称窃取了大量的个人信息和超过 400GB 的服务器数据，而该组织最终也支付了赎金，勒索信息在泄密网站被移除。

2022 年 5 月 31 日，哥斯达黎加公共卫生机构被 Hive 勒索组织攻击，除了核心的勒索动作，被攻击的打印机还打印出数十页充满杂乱无意义字符的页面，而在此次攻击前不久，Conti 勒索组织已经攻击了哥斯达黎加政府系统导致总统宣布进入国家紧急状态。据研究员分析，Conti 至少从 2021 年 11 月开始就在和 Hive 合作，Hive 经常使用 Conti 提供的初始攻击访问和渗透测试服务。

2 技术分析

2.1 入侵和数据窃取

初始入侵的方式取决于目标网络的情况，已知的方式有以下几种：

- ❶ 包含恶意附件的钓鱼邮件
- ❷ RDP、VPN、其它远程网络连接协议的初始凭证
- ❸ 漏洞

Hive 勒索组织使用过的部分漏洞如下：

- :: CVE-2020-12812 Fortinet_FortiOS_SSL_VPN 授权错误漏洞
- :: CVE-2021-31207 Microsoft_Exchange_MailboxExportRequest 任意文件写入漏洞
- :: CVE-2021-34473 Microsoft_Exchange_Server 远程代码执行漏洞
- :: CVE-2021-34523 Microsoft_Exchange_Server_Proxyshell 远程代码执行漏洞

初始入侵成功之后，攻击者可能会下载运行 Cobalt Strike 后门、创建新账户用于持久化，使用 Mimikatz 获取更多凭证。需要窃取数据时，**Hive 勒索组织会通过以下文件共享服务中的某个服务来上传文件：**

- hxxps://anonfiles.com
- hxxps://mega.nz
- hxxps://send.exploit.in
- hxxps://ufile.io
- hxxps://www.sendspace.com
- hxxps://privatlab.net
- hxxps://privatlab.com

|||||

2021 年 6 月：病毒样本首次出现，运行在 Windows 系统中，使用 Golang 编译，未经过特殊处理

2021 年 11 月：增加 Linux/FreeBSD、ESXi 版本（ESXi 版本使用 C/C++ 实现，可能是考虑系统兼容性）

2021 年 12 月：出现了经过代码混淆的样例，使用控制流平坦化技术

2022 年 3 月：编程语言由 Golang 转为 Rust（执行更高效，更难被分析），更改了加密算法

下面我们选择最初的 Golang 样本和使用 Rust 编写的样本进行分析。

 Golang 样本

sha256: 77a398c870ad4904d06d455c9249e7864ac92dda877e288e5718b3c8d9fc6618

UPX 脱壳后，可以发现样本是使用 Golang 编写的，如图 2 所示：



程序执行时，可选参数如图 3:



参数解释整理如下:

-kill	要杀死的进程，大小写不敏感，默认为"msspub mssdesktop"
-no-clean	不清理磁盘空间，默认清理（创建随机内容的文件再删除达到覆盖磁盘数据目的）
-skip	跳过不加密的文件，默认为"\.lnk"
-stop	要停止的服务，大小写不敏感，默认为"bmr sql oracle postgres redis vss backup ssdp"
-t	加密线程数量，默认为 10

输入的参数一部分被拼接为正则字符串，用于后续匹配：

00050921	jne 77a3_unupxed.505963	
00050923	mov eax,dword ptr ss:[esp+44]	[edi+28]:&"(?i:bmr sql oracle postgres redis vss backup sstp)"
00050927	mov eax,dword ptr ss:[esp+4c]	[edi+2C]:&"(?i:mspub msdesktop)"
0005092A	mov eax,dword ptr ss:[esp+48]	[edi+30]:&"(?i:^C:\\\\Windows\\\.\\\.\\1nk\\ \\\\\$\\ \\\\Windows\\ \\\\ \\\\ADMIN\\ \\\\ \\\\IPC\\ \\\\\$)"
00050931	mov eax,dword ptr ss:[esp+48]	
00050935	mov dword ptr ds:[edi+30],eax	
00050938	mov eax,dword ptr ss:[esp+50]	[edi+34]:&"(.+\\. (.+\\.+)?\\.hive\$"
0005093C	mov dword ptr ds:[edi+34],eax	
0005093F	mov dword ptr ss:[esp+54],edi	
00050946	mov eax,dword ptr ss:[esp+58]	
0005094A	mov dword ptr ss:[esp+E0],eax	
00050951	mov eax,dword ptr ss:[esp+40]	
00050955	mov dword ptr ss:[esp+E4],eax	
0005095C	add esp,88	
00050962	ret	

图4 样本正则字符串拼接 ▲

导出 key 之后，根据输入的参数，程序将杀死指定进程、停止指定服务、删除 ShadowCopy 文件、加密文件、清除加密 key、覆盖磁盘数据、删除自身等，如图 5 所示：

```

google_com_encryptor__App_ExportKey(a1);
result = v2;
if ( !v2 )
{
    v4 = v3;
    google_com_encryptor__App_KillProcesses(a1);
    google_com_encryptor__App_StopServices(a1);
    google_com_encryptor__App_RemoveItself();
    google_com_encryptor__App_RemoveShadowCopies(a1);
    google_com_encryptor__App_Notify(a1);
    google_com_encryptor__App_ScanFiles(a1);
    google_com_encryptor__App_EncryptFiles(a1, 0);
    google_com_encryptor__App_EraseKey(a1);
    if ( !*( _BYTE *) (a1 + 36) )
        google_com_encryptor__App_CleanSpace(a1);
    v5[0] = (int)&unk_51B2E0;
    v5[1] = (int)&off_5679AC; // "Finished"
    log_Println((int)v5, 1, 1);
    result = v4;
}

```

图5 样本功能函数调用 ▲

将之前随机生成的密钥流加密，保存为 C:\L7HSI01UNIselRJHL8BQMw.key.hive:

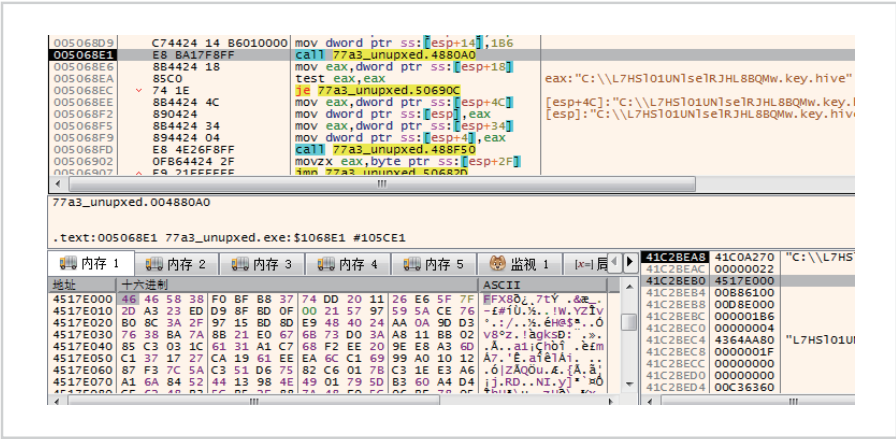


图6 样本保存密钥流

文件名开头的字符串是文件内容 md5 值经过 base64 编码得到的，最后 2 个符号做了调整:

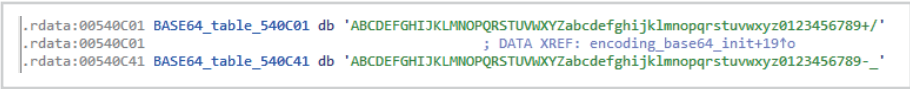


图7 样本中调整的base64字符

释放执行 bat 脚本删除勒索程序，因为勒索程序刚执行时相关的进程还存在，所以文件一开始不会删除成功，这里设置了循环，如果勒索程序还存在，就休眠 1 秒然后再次尝试删除直至成功，最后脚本会自删除:



图8 删除勒索程序的脚本

删除 ShadowCopy 是通过调用 vssadmin 实现的，最后脚本同样会自删除，如图 9:

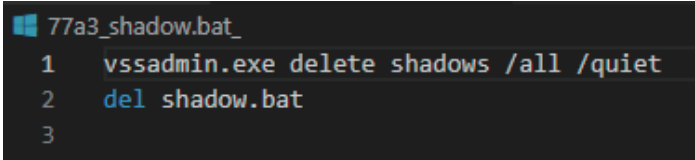


图9 删除ShadowCopy的脚本

加密文件的算法为密钥流异或，加密方式并不是全文件加密，而是以间隔的方式加密文件的部分内容，以提高加密效率：

```
while ( 1 )
{
    v20 = v17 >> 31;
    if ( (HIDWORD(v61) != v20 || v49 <= (unsigned int)v61) && SHIDWORD(v61) >= v20 )
        break;
    if ( HIDWORD(v61) )
        runtime_panicExtendIndex(v35, v38, v40);
    if ( (unsigned int)v61 >= 0x1000 )
        runtime_panicIndex();
    v21 = ( __int64)(v61 + __PAIR64__(v19, v18)) % 0x100000;
    if ( HIDWORD(v21) )
        runtime_panicExtendIndex(v35, v38, v40);
    if ( v63 - v64 <= (unsigned int)v21 )
        runtime_panicIndex();
    v22 = ( __int64)(v61 + __PAIR64__(v19, v18)) % 1024;
    if ( HIDWORD(v22) )
        runtime_panicExtendIndex(v35, v38, v40);
    if ( (unsigned int)v22 >= 0x400 )
        runtime_panicIndex();
    v65[v61] ^= (*_BYTE *) (v22 + v66) ^ (*_BYTE *) (v21 + v67);
    v17 = v49;
    v61 = (unsigned int)v61 + 1164;
    v19 = HIDWORD(v59);
    v18 = v59;
}
```

图10 使用密钥流加密文件内容

加密后的文件会被添加 ".hive" 后缀，添加的字符串和密钥流相关，在桌面和各文件夹下会生成名为 HOW_TO_DECRYPT.txt 的勒索信，见图 11：

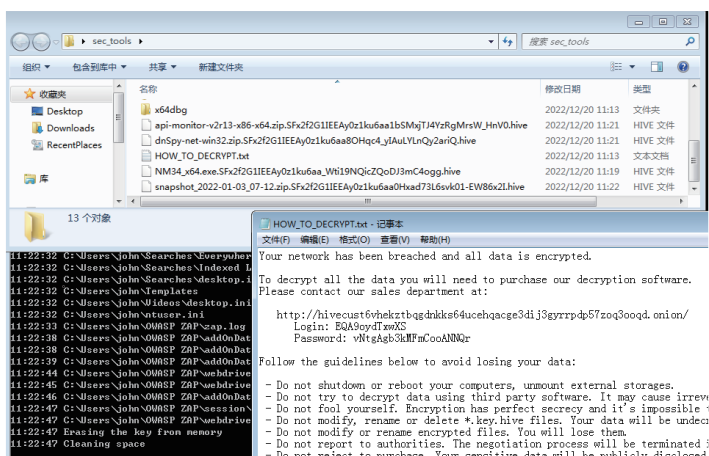


图11 加密结果和勒索信

通过写大量随机数据到文件再删除文件来覆盖硬盘数据，防止从硬盘恢复文件：

```
os_OpenFile(v6, v10, 578, 438, v10);
result = v13;
if ( !v13 )
{
    v17 = v11;
    v21[2] = (int)&os_Remove_ptr;
    v22 = v18;
    v23 = v16;
    v21[1] = (int)os_File_Close_ptr;
    v21[0] = v11;
    runtime_makeslice(&unk_51B4A0, 0xA00000, 0xA00000, v7);
    v19 = v8;
    math_rand_Rand_Read(dword_61E4D4, v8, 0xA00000, 0xA00000, v11, 0);
    do
    {
        os_File_Write(v17, v19, 0xA00000, 0xA00000, v12, v14, v15);
        if ( v14 )
            break;
        os_File_Sync(v17, v3, v4);
    }
    while ( !v3 );
    os_File_Close(v21[0], v3, v4);
    result = os_Remove(v22, v23, v5, v9);
}
return result;
```

图12 随机内容覆盖硬盘数据

由于网站已经更新，旧版的加密软件自动生成的用户名密码无法登录，如图 13 所示：

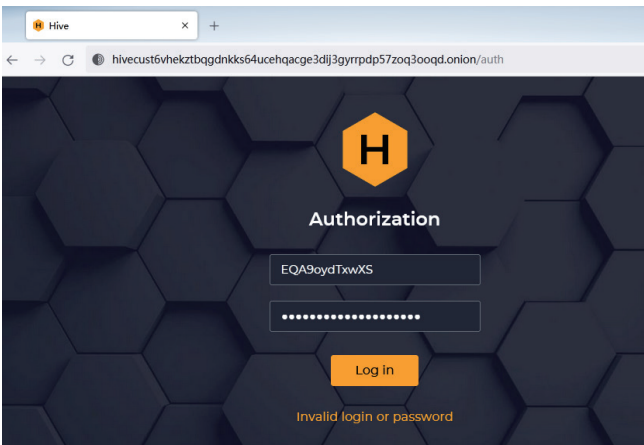


图13 Hive提供的联系网站登录页面

 Rust 样本

原始文件名	windows_x64_encrypt.exe
MD5	44aba241dd3f0d156c6ed82a0ab3a9e1
文件大小	403 KB (413,184 字节)
文件类型	PE32+, exe
编译信息	GNU linker ld (GNU Binutils)(2.34)[Console64,console]

sha256: 065208b037a2691eb75a14f97bdbd9914122655d42f6249d2cca419a1e4ba6f1

为了防止其他获取到样本的人也能登录交流网站查看信息，该版本程序在启动时必须指定用户名和密码，测试时可以设置任意值，如图 14：

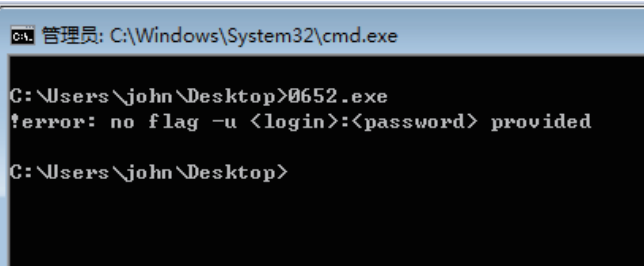


图14 Hive Rust样本需要指定用户名密码运行

样本中使用的字符串都被单独异或，使用时再异或拼接恢复。

.text:00000000040B314	movzx	eax, word ptr [rax+0B4CEh]
.text:00000000040B318	xor	eax, 5B25h
.text:00000000040B320	xor	rcx, 19C46170h
.text:00000000040B327	mov	dword ptr [rsp+648h+Src], ecx
.text:00000000040B32E	mov	word ptr [rsp+648h+Src+4], ax
.text:00000000040B336	cmp	r8, 6
.text:00000000040B33A	jnz	short loc_40B364

图15 异或恢复字符串

部分参数整理解释如下表：

-u	指定用户名密码，格式为"user:password"，必选参数
-f/-force/--force	提升权限强制执行
-no-local/--no-local	不加密本地文件
-no-mounted/--no-mounted	不加密网络分享加载的文件
-no-discovery/--no-discovery	不尝试发现网络共享文件
-min-size/--min-size	最小的加密文件大小，若大小不足则不加密

需要调整执行权限，使后续操作有足够权限：

```
v209 = GetCurrentProcess();
if ( OpenProcessToken(v209, 0x28u, (PHANDLE)TokenHandle) )
{
    *(_OWORD *)Luid = xmmword_458090;
    v729[0] = &unk_459795;
    v729[1] = &unk_4597A5;
    LOWORD(v729[2]) = 0;
    LODWORD(v729[3]) = 1;
    sub_40856D(lpMem, v729);
    v210 = lpMem[0];
    LookupPrivilegeValueW(0i64, (LPCWSTR)lpMem[0], (PLUID)&Luid[4]);
    v211 = 8i64;
    if ( 2 * (__int64)lpMem[1] )
        HeapFree(hHeap, 0, v210);
    AdjustTokenPrivileges(*(HANDLE *)TokenHandle, 0, (PTOKEN_PRIVILEGES)Luid, 0, 0i64, 0i64);
    CloseHandle(*(HANDLE *)TokenHandle);
}
```

图16 调整执行权限

调用 vssadmin.exe 等程序删除备份文件，修改配置，防止恢复文件：

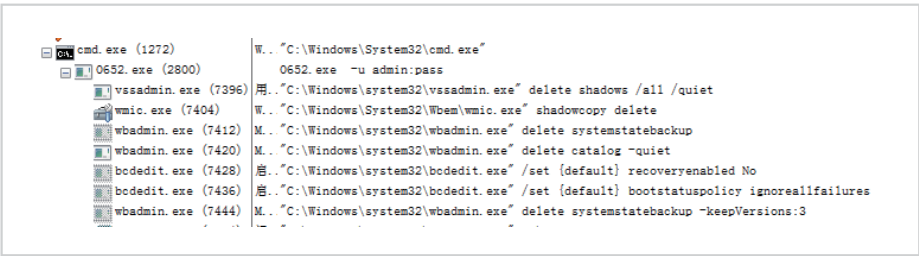


图17 删除备份文件

在系统盘下创建 2 个 ".key" 文件并写入加密后的密钥流，密钥流的生成较 Golang 版本复杂很多，文件内容经过自定义 hash 算法处理后，前 6 字节被 base64 编码作为文件名，如图 18 所示：

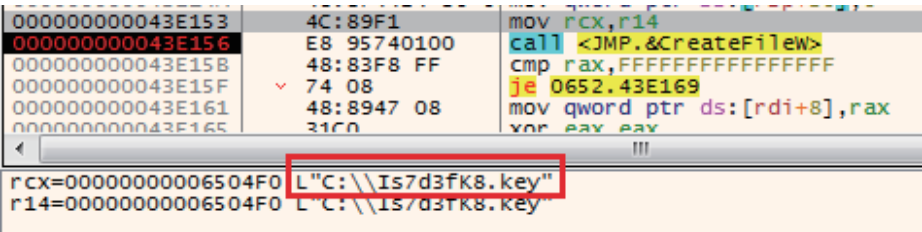


图18 创建密钥流文件

解密后缀字符串并进行比较，
后缀可以是 sql/txt/rtf/xml/pdf/xls/
xlsx/doc/docx 等。

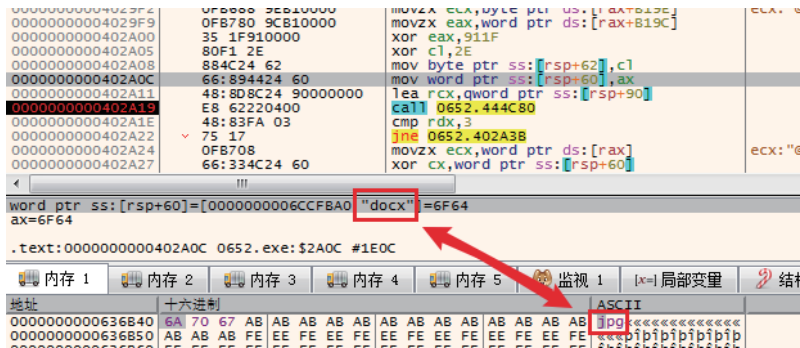


图19 对比文件后缀判断是否需要加密 ▲

加密文件算法为密钥流异或，加密的方法和 Golang 版本一致，间隔加密部分内容以提高加密效率。

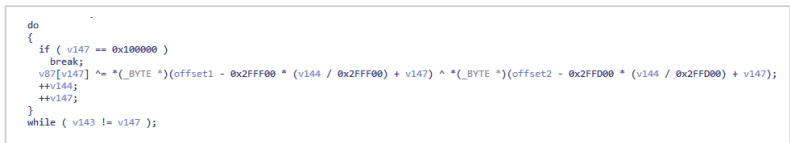


图20 使用密钥流加密文件内容 ▲

加密后的文件会被添加后缀，该后缀和使用的密钥流相关，在桌面和各文件夹下会生成名为 HOW_TO_DECRYPT.txt 的勒索信，见图 21:

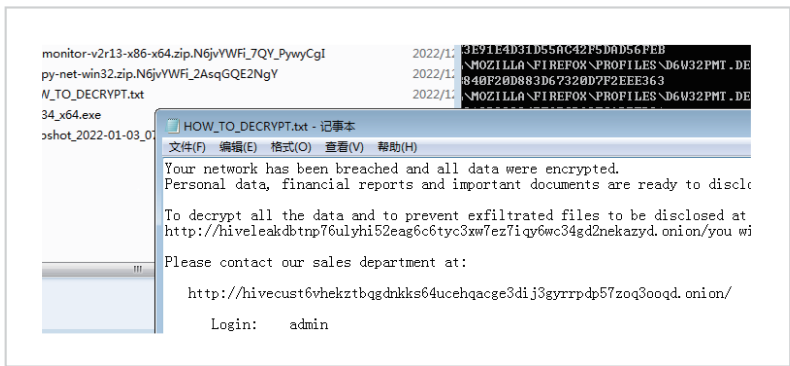


图21 加密结果和勒索信 ▲

3 总结

从技术上看，Hive 勒索组织从出现开始就选择了较容易实现跨平台的 Golang 来编写勒索程序，随后通过一系列方式加强了分析难度，如字符串加密、执行流平坦化等。在观察到 BlackCat 勒索组织首次使用 Rust 实现了勒索程序后，Hive 也能够很快跟进，短时间内也转到了 Rust，使得勒索程序执行更快而且更难被分析。

从策略上看，Hive 并不在攻击目标上给自己设限，不仅如此，还能够积极和其他强大的勒索组织合作以发起更多的攻击，这也是其发展如此迅速的原因。

作为防守一方，一旦被勒索组织入侵成功，公司不仅会遭受经济损失，声誉也会受到影响，更严重的是核心的数据和技术可能被泄露，危及公司生存，种种这些对公司的打击可能是致命的。通过以上分析，我们可以了解 Hive 勒索组织的特点，针对性地采取防御措施保护公司资产。

LockBit 勒索组织分析

文 / 鲍旭丹

1 组织概述

1.1 背景

LockBit 勒索软件最早于 2019 年 9 月以 ABCD 勒索软件出现在大众视野中，后更名为 LockBit。为扩大规模，LockBit 开始以 RaaS 的模式运营，并成立了专门用于泄露受害者数据的站点 DLS(Data Leak Site)，通过该站点管理和处理勒索业务，如赎金协商，如图 1 所示。

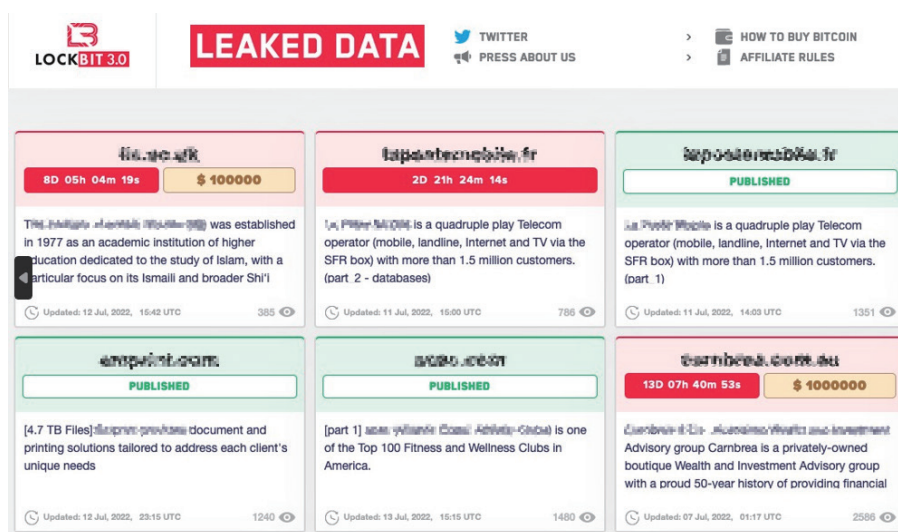


图1 LockBit勒索数据泄露站点（图源 Sentinel Lab）

2021 年 7 月，LockBit 高调发布 LockBit 2.0 版本，声称该版本对比其他主流的勒索软件具有最快的加密速度，并且能够自动化分发加密以及实时泄露数据，目的是吸引更多的附属机构加入。发布之后，便在世界各地发起数次针

对性勒索攻击事件。2022 年 7 月，LockBit 又一次升级为 3.0 版本，在此版本中，LockBit 还率先引入“漏洞赏金计划”，该计划旨在邀请安全研究人员提交漏洞报告，根据贡献程度换取 1000 到 100 万美元不等的奖金。LockBit 一直致力于更快速、更隐蔽、更先进的勒索攻击，并不断构思新的方法以超越同类竞品，近两年来已经成为最活跃的勒索组织之一。

1.2 著名事件

- 2021 年 8 月，全球财富 500 强的咨询领域巨头埃森哲公司遭受勒索软件攻击，泄密数据高达 6TB；
- 2021 年 12 月，一家为荷兰警方和紧急服务处理敏感文件的荷兰技术公司 Abiom 遭到 LockBit 勒索软件袭击。该公司拒绝支付赎金，导致包括身份证和发票在内的 39,000 份文件在网上泄露；
- 2022 年 7 月，LockBit 窃取了意大利税务局 78GB 的数据，其中包括公司文件、扫描件、财务报告和合同；
- 2022 年 10 月，巴西利亚银行遭到 LockBit 勒索软件的攻击，该团伙要求赎金 50 BTC，赎金相当于 520 万巴西雷亚尔；
- 2022 年 11 月，法国国防和技术公司 Thales 遭到 LockBit 针对性勒索攻击，随后，LockBit 泄露其 9.5GB 的存档文件，包含公司技术文件、客户文件、会计文件等。

2 技术分析

2.1 初始入侵

在 LockBit 的 RaaS 运营模式中，运营机构通过附属机构分发 LockBit 勒索软件，并在地下市场通过广告招募 IAB，或者与其他恶意组织合作（如 Maze），招募公司内部人员来获得 RDP 或 VPN 凭据，用于实现初始入侵。除了通过代理，LockBit 的其他入侵方式还包括：

- ❶ 钓鱼邮件
- ❷ 弱口令爆破
- ❸ 漏洞利用：Fortinet VPN (CVE-2018-13379)、ProxyShell (CVE-2021-34473)、SonicWall SRA (CVE-2021-20028)
- ❹ 通过第三方凭据或者公开凭据获得对服务器的访问权限

一旦成功入侵到受害者机器，LockBit 便开始使用多种反分析、反调试技术来阻碍静态和动态分析以达到最终加密勒索的目的。为了对 LockBit 的加密与勒索过程进行详细描述，本文选取了较为典型的 LockBit 2.0 勒索载荷，具体分析过程如下。

2.2 样本分析

原始文件名	LockBit_2ED873D43FE5DD38
MD5	63dcf75ad743b292e4a6cd067ffc2c18
文件大小	959.50 KB
文件类型	Win32 EXE
时间戳	2021-07-26

反调试

如图 2 所示,在样本入口点处,LockBit 会判断 PEB 中的 NtGlobalFlag 成员值是否为 0x70,如果是,则下面三个标志已被设置,表示进程正在被调试,LockBit 将自身无限期挂起。

0x10 FLG_HEAP_ENABLE_TAIL_CHECK
0x20 FLG_HEAP_ENABLE_FREE_CHECK
0x40 FLG_HEAP_VALIDATE_PARAMETERS

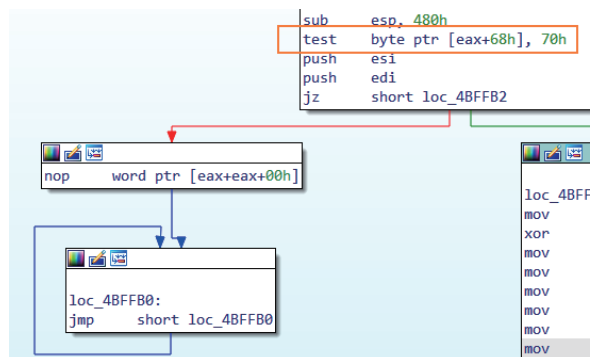


图2 反调试技术

动态加载 API

LockBit 导入表中仅仅加载了包括 Kernel32.dll、Advapi32.dll 在内的少量 DLL 和 API。实际上,LockBit 无法仅用这些导入的 API 完成强大的恶意功能。为躲避分析,LockBit 通过 LoadLibraryA 动态加载了其他的 DLL 模块。以 gdiplus.dll 为例,如图 3 所示,LockBit 将硬编码的“GDIPLUS.DLL”字符串先转为小写,通过 PEB 解析出 LoadLibraryA 再加载 gdiplus.dll 到内存中。

被加载的动态链接库有: gdiplus.dll、ws2_32.dll、shell32.dll、advapi32.dll、user32.dll、ole32.dll、netapi32.dll、gpedit.dll、oleaut32.dll、shlwapi.dll、msvcrt.dll、activeds.dll、gdiplus.dll、mpr.dll、bcrypt.dll、crypt32.dll、iphlpapi.dll、wtsapi32.dll、win2u.dll、Comdlg32.dll、cryptbase.dll、combase.dll、winspool.drv。

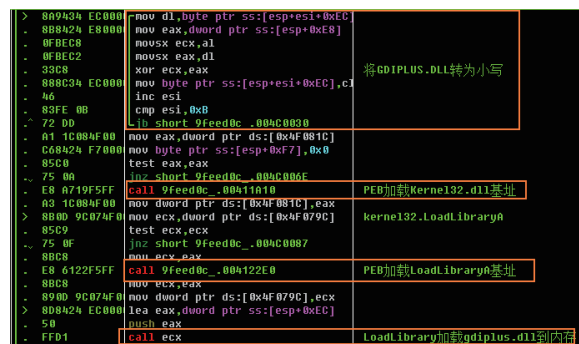


图3 动态加载API

字符串混淆

LockBit 对一些 DLL 字符串进行了异或、加减法等混淆,如图 4 展示了 ws2_32.dll 在被动态解析前的混淆字符串为“{w6c762hpp”。

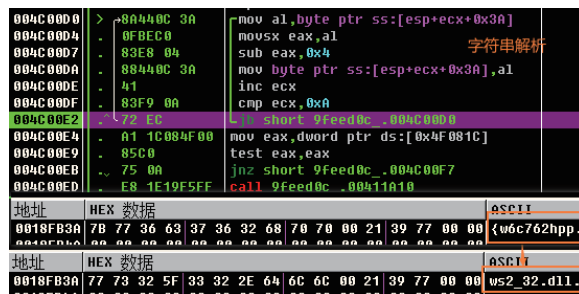


图4 堆栈字符串混淆

检查系统语言

在加载所有 DLL 到内存后, LockBit 会判断当前系统语言是否在自己的白名单列表内, 如果命中白名单, 则退出当前勒索进程, 如图 5 所示:



图5 检查系统语言是否在白名单中

拒绝访问当前勒索进程

LockBit 解析调用 RtlAllocateAndInitializeSid 为系统 Everyone 组分配并初始化一个权限为 SECURITY_WORLD_SID_AUTHORITY 的 SID。

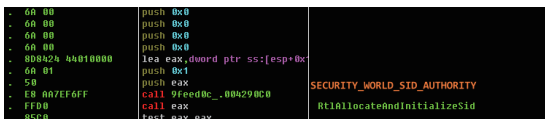


图6 为Everyone用户组分配SID

随后, 调用 RtlAddAccessDeniedAce 给当前 Everyone 组添加一个新的拒绝访问的 ACL, 再将新的 ACL 指向当前进程, 这样, Everyone 组的任何用户将无法访问当前勒索进程, 图 7 所示:

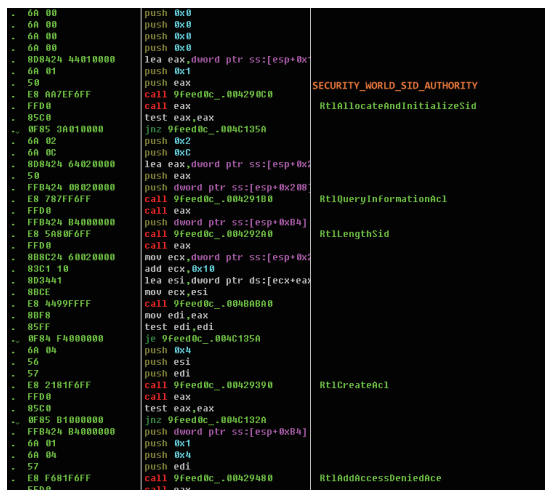


图7 为Everyone组创建拒绝访问的ACL

关闭系统错误警告

如图 8 所示, 为了防止系统出现错误提示或者弹窗, 使受害者得知具体问题, LockBit 将进程的错误模式设置为 0x7, 对应选项如下:

0x1 SEM_FAILCRITICALERRORS (系统不显示关键错误处理程序弹框)

0x2 SEM_NOALIGNMENTFAULTEXCEPT(系统不显示 Windows 错误报告弹框)

0x4 SEM_NOGPFAULTERRORBOX(系统自动修复对齐错误)

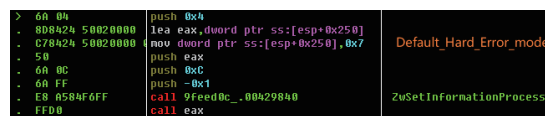


图8 关闭系统错误警告

进程提权

LockBit 调用 RtlAdjustPrivilege 实现进程提权，并启动 SE_TAKE_OWNERSHIP_PRIVILEGE，在后续加密过程中获取文件对象的所有权。

```
808424 3F020000 lea eax,dword ptr ss:[esp+0x23F]
50          push eax
6A 00       push 0x0
6A 01       push 0x1
6A 09       push 0x9 → SE_TAKE_OWNERSHIP_PRIVILEGE
E8 8085F6FF call 9feed0c_.00429930 RtlAdjustPrivilege
FFD0       call eax
```

图9 获取文件所有权

禁用进程和服务

LockBit 从硬编码的数据中解密出进程列表和服务列表清单，并禁用这些进程和服务，如图 10 所示：

```
68 608A4F00 push 9feed0c_.004F8A60 进程列表
68 3C8A4F00 push 9feed0c_.004F8A3C
BA A5040000 mov edx,0x4A5
B9 607A4E00 mov ecx,9feed0c_.004E7A60
E8 D0000000 call 9feed0c_.004B9B50 decrypt_config
83C4 40     add esp,0x40
BA 0F030000 mov edx,0x30F
B9 00AC4E00 mov ecx,9feed0c_.004EAC00
68 348A4F00 push 9feed0c_.004F8A34 服务列表
68 4C8A4F00 push 9feed0c_.004F8A4C
E8 B4000000 call 9feed0c_.004B9B50 decrypt_config
```

图10 禁用进程和服务

进程清单：

wxServer,wxServerView,sqlmangr,RAGui,supervise,Culture,Defwatch,winword,QBW32,QBDBMgr,qbupdate,axlbridge,httpd,fdlauncher,MsdTsrvr,java,360se,360doctor,wdswwfsafe,fdhost,GDscan,ZhuDongFangYu,QBDBMgrN,mysqld,AutodeskDesktopApp,acwebbrowser,Creative Cloud,Adobe Desktop Service,CoreSync,Adobe CEF,Helper,node,AdobelPCBroker,sync-taskbar,sync-worker,InputPersonalization,AdobeCollabSync,BrCtrlCntr,BrCcUxSys,SimplyConnectionManager,Simply.SystemTrayIcon,fbguard,fbserver,ONENOTEM,wsa_service,koaly-exp-engine-service,TeamViewer_Service,TeamViewer,tv_w32,tv_x64,TitanV,Ssms,notepad,RdrCEF,sam,oracle,ocssd,dbsnmp,synctime,agntsvc,sqlplussvc,xfssvccon,mydesktopservice,ocautoupds,encsvc,tbirdconfig,mydesktopqos,ocomm,dbeng50,sqlbcoreservice,excel,infopath,msaccess,mspub,onenote,outlook,powerpnt,steam,thebat,thunderbird,visio,wordpad,bedbh,vxmon,benetns,bengien,pvlsvr,beserver,raw_agent_svc,vsnapvss,CagService,DellSystemDetect,EnterpriseClient,ProcessHacker,Procexp64,Procexp,GlassWire,GWCtrlSrv,WireShark,dumpcap,j0gnjko1,Autoruns,Autoruns64,Autoruns64a,Autorunsc,Autorunsc64,Autorunsc64a,Sysmon,Sysmon64,procexp64a,procmon,procmon64,procmon64a,ADEplorer,ADEplorer64,ADEplorer64a,tcpview,tcpview64,tcpview64a,avz,tdsskiller,RaccineElevatedCfg,RaccineSettings,Raccine_x86,Raccine,Sqlservr,RTVscan,sqlbrowser,tomcat6,QBIDPService,notepad++,SystemExplorer,SystemExplorerService,SystemExplorerService64,Totalcmd,Totalcmd64,VeeamDeploymentSvc

服务清单：

wrapper、DefWatch、ccEvtMgr、ccSetMgr,SavRoam,Sqlservr,sqlagent,sqladhlp,Culserver,RTVscan,sqlbro

wser,SQLADHLP,QBIDPService,Intuit.QuickBooks.FCS,QBCFMonitorService, msmdsrv,tomcat6,zhudongfangy
u,vmware-usbarbitator64,vmware-converter,dbsrv12,dbeng8,MSSQL\$MICROSOFT##WID,MSSQL\$VEEAMSQL
L2012,SQLAgent\$VEEAMSQL2012,SQLBrowser,SQLWriter,FishbowlMySQL,MSSQL\$MICROSOFT##WID,My
SQL57,MSSQL\$KAV_CS_ADMIN_KIT,MSSQLServerADHelper100,SQLAgent\$KAV_CS_ADMIN_KIT,msftesql-
Exchange,MSSQL\$MICROSOFT##SSEE,MSSQL\$SBSMONITORING,MSSQL\$SHAREPOINT,MSSQLFDLaun
cher\$SBSMONITORING,MSSQLFDLauncher\$SHAREPOINT,SQLAgent\$SBSMONITORING,SQLAgent\$SHAR
EPOINT,QBFCService,QBVSS,YooBackup,YoolIT,vss,sql,svc\$,MSSQL,MSSQL\$,memtas,mepocs,sophos,veea
m,backup,bedbg,PDVFSService,BackupExecVSSProvider,BackupExecAgentAccelerator,BackupExecAgentBro
wser,BackupExecDiveciMediaService,BackupExecJobEngine,BackupExecManagementService,BackupExecRP
CService,MVArmor,MVarmor64,stc_raw_agent,VSNAPVSS,VeeamTransportSvc,VeeamDeploymentService,V
eeamNFSSvc,AcronisAgent,ARSM,AcrSch2Svc,CASAD2DWebSvc,CAARCUpdateSvc,WSBExchange,MSExc
hange,MSExchange\$

创建互斥量

为避免同时运行多个勒索软件实例，LockBit 将硬编码的字符填充到互斥量对象中 “: \BaseNamedObjects\{3FE573D4-3FE5-DD38-399C-886767BD8875}”，如图 11 所示，如果搜索到相同的互斥量，表明当前已有勒索进程在运行，则退出当前进程。

50
57
53
8085 9CFFFFFF
50
8085 68FCFFFF
50
FF95 94FFFFFF
83C4 48
8085 68FCFFFF
50
8085 70FFFFFF

push eax
push edi
push ebx
lea eax,[local.89]
push eax
lea eax,[local.230]
push eax
call [local.91]
add esp,0x48
lea eax,[local.230]
push eax
lea eax,[local.100]

0018F760, (Unicode "\BaseNamedObjects\{3FE573D4-3FE5-DD38-399C-886767BD8875}")
0038

HEX 数据	Unicode
5C 00 42 00 61 00 73 00 65 00 4E 00 61 00 6D 00	\BaseNam
65 00 64 00 4F 00 62 00 6A 00 65 00 63 00 74 00	edObject
73 00 5C 00 7B 00 33 00 46 00 45 00 35 00 37 00	s\{3FE5
33 00 44 00 34 00 2D 00 33 00 46 00 45 00 35 00	3D4-3FE5
2D 00 44 00 44 00 33 00 38 00 2D 00 33 00 39 00	-DD38-39
39 00 43 00 2D 00 38 00 38 00 36 00 37 00 36 00	9C-88676
37 00 42 00 44 00 38 00 38 00 37 00 35 00 7D 00	7BD8875}

图11 创建互斥量对象

持久化

LockBit 通过注册表将自己添加到 windows 启动项实现持久性，以防止勒索进程被系统中途关闭。

51
8D4D 08
51
6A 00
68 1F000200
6A 00
6A 00
6A 00
8D4D 92
51
68 01000080
FFD0
33C9
85C8
0F85 D3130000

push ecx
lea ecx,[local.10]
push ecx
push 0x0
push 0x2001F
push 0x0
push 0x0
push 0x0
lea ecx,dword ptr ss:[ebp-0x6E]
push ecx
push 0x80000001
call eax
xor ecx,ecx
test ecx,ecx
jnz 9feed0c_0049CE8E

advapi32.RegCreateKeyExA
advapi32.RegCreateKeyExA
{ASCII "SOFTWARE\Microsoft\Windows\CurrentVersion\Run"}

图12 创建注册表

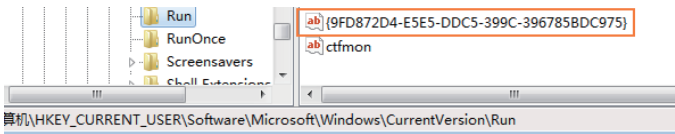


图13 注册启动项

添加扩展名图标

LockBit 在注册表中将扩展名 .LockBit 的默认图标设置为 “Random_Str{5}” .ico。



图14 在注册表中添加.LockBit扩展名图标

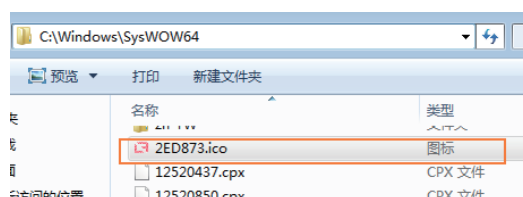


图15 LockBit扩展名图标样式

删除卷影副本

LockBit 首先解析控制命令字符串：

- /c vssadmin Delete Shadows /All /Quiet
- /c bcdedit /set {default} recoveryenabled No
- /c bcdedit /set {default} bootstatuspolicy ignoreallfailures
- /c wmic SHADOWCOPY /nointeractive
- /c wevtutil cl security
- /c wevtutil cl system
- /c wevtutil cl application

为了防止加密文件的恢复，LockBit 创建进程启动 cmd.exe 执行上述命令，功能包括：删除卷影副本、禁用 Windows 启动修复，删除 windows 备份，如图 16 所示。

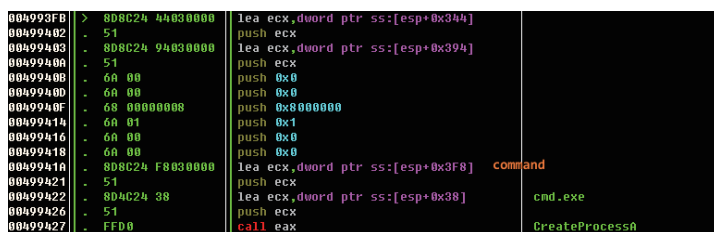


图16 执行命令，删除卷影副本

多线程加密文件

在进行加密前，LockBit 会针对当前受害者生成一个 32 字节的私钥和对应的 32 字节的公钥，公钥和私钥是调用加密库 Libsodium 中 crypto_box_keypair 的函数生成的，如图 17 所示。



图17 创建用于加密的公钥和私钥

接着，LockBit 创建由随机字符串构成的注册表项，将私钥和公钥写入其中，分别命名为 Private 和 Public。完成后，LockBit 便从内存中擦除私钥。



名称	类型	数据
(默认)	REG_SZ	(数值未设置)
Private	REG_BINARY	e1 6f 2e df 5b d4 70 f0 fd ae b4 a8 33 a5 58 8b 24 b3 34 6b fc 3...
Public	REG_BINARY	18 7a 3e 08 07 bc 79 ca 83 a2 3f 80 e3 e8 c1 f5 d2 fb 2d 41 ef 9...

图18 创建注册表写入公钥和私钥

设置加密密钥后，LockBit 创建多线程进行加密，其中，子线程数量 = 处理器内核数量。同时，它调用 NtCreateIoCompletion 创建 I/O 完成端口辅助加密，如图 19。LockBit 3.0 中子线程数量 = 处理器内核总数 * 2 + 1，加密速度比 LockBit2.0 更快。

```
NtCreateIoCompletion = sub_415400(&stru_4E2520, 2031619, 0, v35);
if ( NtCreateIoCompletion(v41, v43, v45, v47) >= 0 )
{
    dword_4F889C = AllocVirtualMem((4 * dword_4F8898));
    if ( dword_4F889C )
    {
        v37 = 0;
        if ( !dword_4F8898 )
            return 1;
        while ( 1 )
        {
            *(dword_4F889C + 4 * v37) = CreateThread(v49, v50, v51, v52, v53[0], v53[1]);
            child_thread_handle = *(dword_4F889C + 4 * v37);
            if ( child_thread_handle == -1 )
                break;
        }
    }
}
```

图19 创建多线程进行加密

加密主要使用 AES-CBC 模式，如图 20 所示：

```
{
    case 1:
        AES_key = v85 + 16;
        if ( byte_4F8C84 )
        {
            AES_KeyExpansion(v100, AES_key);
            AES_CBC_Encrypt(v15 + 3, v15[10], v15[10]);
            Memory_Fill(v31, 0xFF, 4);
        }
        else
        {
            AES_KeyExpansion_2(v99, AES_key);
            AES_CBC_Encrypt_2(v15 + 3, v15[10], v15[10]);
            Memory_Fill(v99, 0xFF, 0x118);
        }
}
```

图20 使用AES-CBC加密数据

此外，LockBit 还会调用 FindFirstFileExW 和 FindNextFileW 来遍历本地磁盘进行文件加密，遍历过程主要分为三步：

❶ 判断当前文件是否位于当前目录 (./)、上一层目录 (../)，如果不是则进行第 2 步判断；

```
result = FindFirstFileExW(v260, 0, v263, 0, 0, 0);
v9 = result;
v505 = result;
if ( result != -1 )
{
    v285[0] = 's\0%';
    v285[1] = '%\0\\';
    v285[2] = 's';
    while ( v264 == '.' )
    {
        if ( v265 == '.' )
        {
            if ( v266 )
                break;
        }
        else if ( v265 )
        {
            break;
        }
    }
}
```

图21 第1步 文件目录判断

② 判断当前文件是否是文件夹，如果是文件夹且不在文件夹白名单中，则递归调用，如果不是文件夹，则进行第 3 步判断；

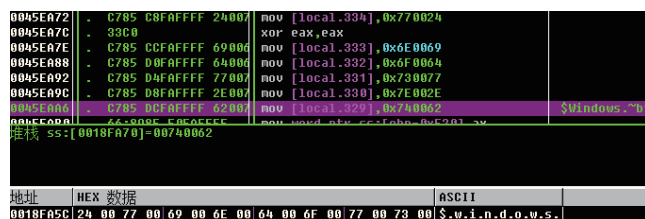


图22 第2步 文件夹判断

文件夹白名单:

\$Windows.~bt, intel, msocache, \$recycle. bin, \$windows.~ws, tor browser, boot, windows nt, msbuild, microsoft, all users, system volume information, perflog, google, application data, windows, windows.old, appdata, mozilla, microsoft.net, microsoft shared, internet explorer, common files, opera, windows journal, windows defender, windowsapp, windowspowershell, usoshared, windows security, windows photo viewer

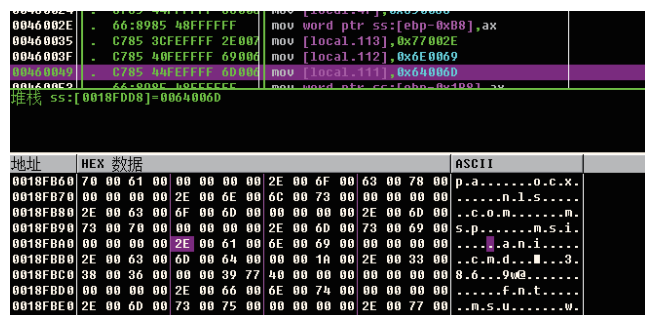


图23 第3步 文件扩展名判断

③ 判断当前文件的扩展名是否在白名单中，如果不是，则加密当前文件；同时，如果文件的属性是 readonly, LockBit 会将其属性设置为 normal, 然后再加密；

文件扩展名白名单:

.386, .cmd, .ani, .adv, .msi, .msp, .com, .nls, .ocx, .mpa, .cpl, .mod, .hta, .prf, .rtp, .rpd, .bin, .hlp, .shs, .drv, .wpx, .bat, .rom, .msc, .spl, .msu, .ics, .key, .exe, .dll, .lnk, .ico, .hlp, .sys, .drv, .cur, .idx, .ini, .reg, .mp3, .mp4, .apk, .ttf, .otf, .fon, .fnt, .dmp, .tmp, .pif, .wav, .wma, .dmg, .iso, .app, .ipa, .xex, .wad, .msu, .icns, .lock, .LockBit, .theme, .diagcfg, .diagcab, .diagpkg, .msstyles, .gadget, .woff, .part, .sfcache, .winmd

勒索信

加密完成后，在每个文件路径下，都会生成以 .LockBit 为后缀名的加密文件以及勒索信 Restore-My-Files.txt, 勒索信中指出，如果想恢复被加密的文件，受害者需要访问指定网站与他们取得联系。

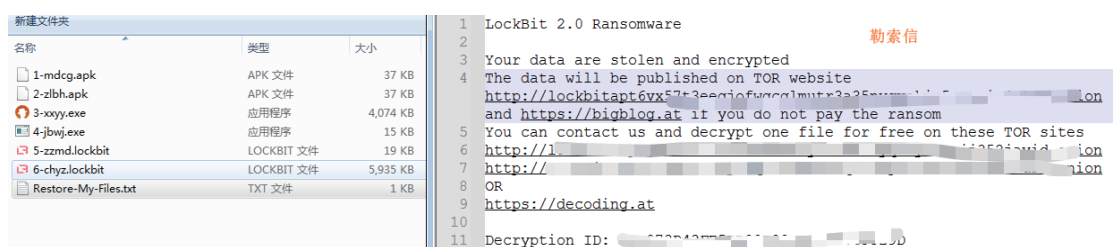


图24 LockBit勒索信

设置壁纸

为了更直观地提醒受害者，LockBit 调用 SystemParametersInfoW 函数将事先存放在系统 %Temp% 路径下的 .bmp 图片设置为壁纸。

```
SystemParametersInfoW = (v230 + (*(v274 + 28) + 4 * (*(v274 + 36) + 2 * v272 + v230) + v230));
LABEL_357:
dword_4F8E5C = SystemParametersInfoW;
LABEL_358:
SystemParametersInfoW(20, 0, v271, 3); // (SPI_SETDESKWALLPAPER,0,wall_path, 3)
LABEL_359:
Nt_FreeVirtualMemory(v271);
return 1;
```

图25 设置桌面壁纸为勒索信 ▲

其中，LockBit 3.0 勒索壁纸如图 26 所示。



图26 LockBit 3.0壁纸 ▲

LockBit 2.0 勒索壁纸如图 27 所示。



图27 LockBit 2.0壁纸 ▲

删除自身

加密完成后，LockBit 构建自我删除命令，并启动 cmd.exe 执行。

```
}
++v54;
v56 = (deleteself_command + 4);
v208 = v54;
deleteself_command += 4;
if ( v54 == *(v55 + 24) )
goto LABEL_86;
}
v203[13] = 67;
v203[14] = 0x29;
v204 = 0;
for ( j = 0; j < 0xF; ++j )
v203[j] ^= v202; // cmd.exe
Shell_dll = dword_4F8BD4;
// /C ping 127.0.0.7 -n 3 > Nul & fsutil file setZeroData
// offset=0 length=524288 "%s" & Del /f /q "%s"
ShellExecuteW = Shell_dll + *(v209[7] + 4 * *(v209[9] + 2 * v208 + Shell_dll) + Shell_dll);
return (ShellExecuteW)(v64);
```

图28 构建命令删除自身 ▲

3 ATT&CK

根据 LockBit 的惯用攻击手法，绘制 LockBit 的 ATT&CK，如图 29。

持久化(19)	提权(13)	防御规避(39)		发现(29)		凭据访问(17)	影响(13)
操纵账户	滥用提升控制权限机制	滥用提升控制权限机制	网络边界桥接	发现帐户	发现软件	利用中间人攻击 (MITM)	删除帐户访问权限
利用BITS服务	操纵访问令牌	操纵访问令牌	混淆文件或信息	发现应用程序窗口	发现系统信息	暴力破解	数据销毁
引导或登录自动启动执行	引导或登录自动执行	利用BITS服务	操作系统预启动	发现浏览器书签	发现系统位置	从存储密码的位置获取凭据	加密数据以产生影响
引导或登录初始化脚本	引导或登录初始化脚本	规避调试器	进程注入	发现云基础设施	发现系统网络配置	利用凭证访问漏洞	操纵数据
添加浏览器扩展插件	创建或修改系统进程	混淆/解密文件或信息	反射代码加载	云服务仪表盘	发现系统网络连接	强制认证	污损破坏
篡改客户端软件	域策略修改	直接访问卷	注册恶意域控制器	发现云服务	发现系统所有用户	伪造Web凭证	磁盘擦除
创建账户	容器逃逸	域策略修改	使用Rootkit	发现云存储对象	发现系统服务	输入捕捉	端点侧拒绝服务
创建或修改系统进程	事件触发执行	执行范围保护	损坏信任控制	调试器规避	发现系统时间	修改身份验证流程	固件损坏
事件触发执行	利用漏洞提权	利用防御规避	系统程序代理执行	发现域信任	虚拟化/沙箱逃逸	多因素身份验证 (MFA) 拦截	禁止系统恢复
利用外部远程服务	执行流程劫持	文件和目录权限修改	系统脚本代理执行	发现文件和目录		多因素身份验证 (MFA) 请求轰炸	网络侧拒绝服务
执行流程劫持	进程注入	隐藏行为	模板注入	发现组策略		网络嗅探	资源劫持
植入容器映像	计划任务/工作	执行流程劫持	使用流量信令	发现网络服务		操作系统凭据转储	服务停止
修改身份验证流程	利用有效账户	削弱防御机制	利用受信任的开发工具执行	发现网络共享		窃取应用程序访问令牌	系统关闭/重启
启动Office应用程序		删除主机中的信标	未使用/不支持的云区域	网络嗅探		窃取或伪造身份验证证书	
在操作系统前启动		间接执行命令	使用备用身份验证材料	发现密码策略		窃取或伪造 Kerberos 凭证	
利用计划任务/工作		仿冒	利用有效账户	发现外围设备		窃取Web会话 Cookie	
利用服务器软件组件		修改身份验证过程	虚拟化沙箱逃逸	发现权限组		不安全的凭证	
使用流量信令		修改云计算基础设施	削弱加密	流程发现			
利用有效账户		修改注册表	利用XSL文件执行脚本	查询注册表			
		修改系统映像		发现远程系统			

4 总结

LockBit 勒索软件攻击手法复杂多变，版本迭代更新快，尤其是通过精心设计的多线程架构，使得攻击过程十分隐蔽，加密过程也非常迅速，给受害者的数据安全、经济和声誉都造成了严重损失。当面临着复杂的勒索软件，要深知事前防御效果远胜于事后补救，H3C 攻防实验室将密切关注 LockBit 勒索组织的最新活动，及时更新有关勒索软件的最新动态，持续输出高质量威胁情报。

5 防护建议

针对勒索攻击的特点，给出几点有效建议：

- 定期进行重要文件备份；
- 及时开启系统软件、补丁的自动更新；
- 开启具有防病毒功能的安全软件；
- 及时升级最新特征库。

REvil 勒索组织分析

文 / 汪林

1 组织概述

REvil，又称 Sodinokibi，是一个勒索软件网络犯罪组织，于 2019 年 4 月首次出现，因擅长实施“大型狩猎”活动和多重勒索攻击而受到关注。REvil 被认为是另一个同样臭名昭著的勒索组织 GandCrab 的继承者，后者于 2019 年 6 月宣布关闭运营，此后 REvil 就开始非常活跃，并且两者使用的勒索软件在代码结构和行为方面存在大量相似之处。REvil 曾于 2021 年下半年一度因执法机构的压力被迫关闭运营，但在 2022 年 3 月，研究人员发现了 REvil 勒索软件卷土重来的迹象，至今仍然活跃。

REvil 是最早采用 RaaS 运营模式的勒索组织之一，通常以全球范围内的大中型企业作为勒索攻击目标，但会避开一些特定国家和地区；值得一提的是，在最新的 REvil 勒索软件版本中删除了地区白名单豁免功能。勒索赎金的数量基于目标的年收入，REvil 组织从中收取 20%-30% 的份额，其余部分则为其雇佣的附属机构所得，赎金使用比特币、门罗币等加密货币进行支付。该组织曾声称通过勒索攻击在一年内获得 1 亿美元的赎金收入。

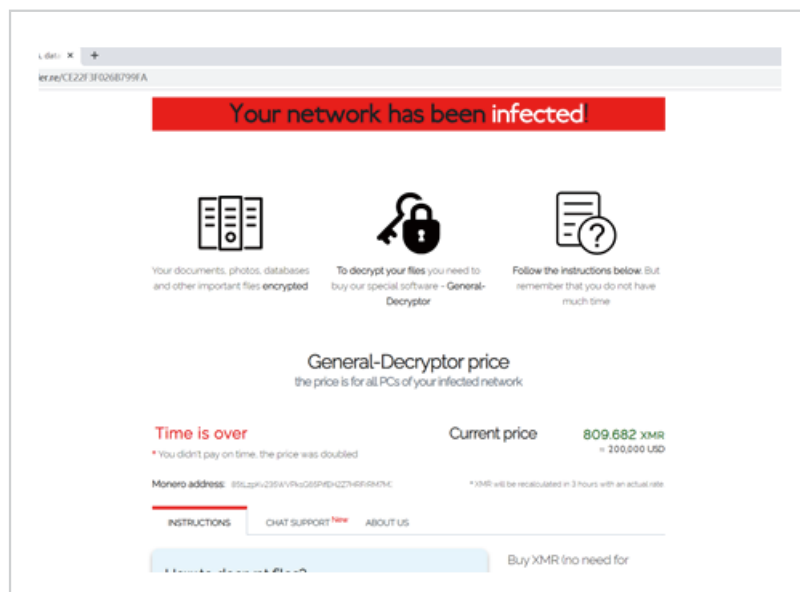


图1 REvil组织提供给受害者的付款页面

1.1 勒索策略的改进

为了提高勒索成功率,REvil不断改进勒索策略。2020年1月,REvil开始采用双重勒索策略,建立了一个名为“The Happy Blog”的暗网数据泄露站点,要求附属机构在加密操作前窃取数据,对于那些拒绝支付赎金、依靠备份恢复数据的受害者,REvil会把窃取到的数据暴露在该站点上并进行拍卖。2021年3月,REvil为其附属机构提供了一项新服务,通过对受害者使用 DDoS 攻击,向新闻媒体和受害者的商业伙伴透露相关消息,以向受害者施加压力。此外,REvil也是首个进行“间接勒索”的组织,2021年4月,REvil在受害者拒绝支付赎金后,转而向受害者的客户索要赎金。事件经过为,REvil攻击了全球第二大笔记本电脑设计制造商广达(Quanta),窃取了该公司重要合作伙伴苹果公司的产品设计蓝图,索要5000万美元赎金,但遭到拒绝;之后,REvil试图勒索苹果公司,要求其支付赎金来“回购”被窃取的产品蓝图。

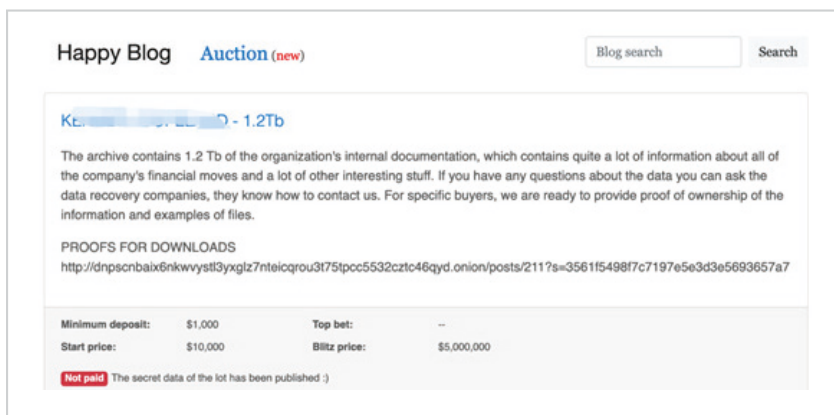


图2 REvil在暗网“Happy Blog”上拍卖拒不支付赎金的受害者数据(图源网络) ▲

1.2 “大型狩猎”活动达到顶峰

REvil擅长进行“大型狩猎”活动,从出现至今,已有不少大型企业和机构沦为受害者,索要的赎金越来越多,勒索攻击的恶劣影响程度不断攀升,在2021年上半年达到顶峰:

- 2021年1月,泛亚大型零售连锁运营商 Dairy Farm Group 受到 REvil 勒索软件攻击,被索要 3000 万美元赎金。
- 2021年3月,REvil勒索攻击了计算机设计制造巨头宏碁(Acer),窃取了该公司大量数据,并索要5000万美元赎金。
- 2021年5月,全球最大肉类食品加工厂商 JBS 遭受了 REvil 勒索软件攻击,被迫将所有美国牛肉工厂暂时关闭,并中断了家禽和猪肉工厂的运营,导致美国境内短期的肉食产品供应短缺。该公司被索要赎金 2250 万美元,经谈判最终支付了 1100 万美元赎金。
- 2021年7月初,REvil通过 Kaseya VSA 平台的 0-day 漏洞发动了供应链攻击,导致下游 1000 多家客户网络遭受勒索软件攻击,其中 Kaseya 被 REvil 索要 7000 万美元赎金,后下调为 5000 万美元。

1.3 卷土重来

REvil在2021年上半年成功实施了多次影响恶劣的勒索攻击事件后,遭到了国际执法机构的围堵。其部分服务器遭到入侵和控制,数据泄露站点也被劫持,该组织被迫停止了勒索软件的运营。从2021年10月开始,REvil的

部分核心成员和附属成员陆续被指控和逮捕，在 2022 年 1 月，俄罗斯政府宣布逮捕了 REvil 勒索软件团伙的 14 名成员。

然而，在销声匿迹几个月后，2022 年 4 月，REvil 使用过的 Tor 基础设施再次开始运行，但新的网站与旧的网站并不相同。此外，研究人员还发现了从 REvil 源码编译而来的新版勒索软件。因此研究人员推测，REvil 组织并没有完全被毁灭，其剩余的核心开发人员之一重启了新的勒索软件运营。

2 技术分析

2.1 初始入侵手段

REvil 组织雇佣大量附属组织来分发 REvil 勒索软件，这些附属组织对受害者网络进行初始入侵的方式主要有以下几种：

- 软件漏洞利用
- 钓鱼邮件攻击
- RDP 弱口令暴力破解
- 供应链攻击
- 通过僵尸网络分发

REvil 曾利用 Weblogic 反序列化漏洞（CVE-2019-2725）、Confluence 漏洞（CVE-2019-3396）、Pulse Secure VPN 漏洞（CVE-2019-11510）、Microsoft Exchange 服务器漏洞（CVE-2021-26855）等。在 JBS 攻击事件中，研究人员发现 REvil 勒索软件与 Qbot 僵尸网络可能进行了合作。

一旦入侵成功，攻击者即可在目标内网进行横向渗透，寻找高价值的服务器和主机，先窃取大量敏感数据，然后部署 REvil 勒索软件进行文件加密。

2.2 勒索软件分析

行为特征

REvil 勒索软件执行完成后，会在受害者主机上留有如下行为特征：

- 替换桌面背景壁纸，提示受害者其主机上的文件已被加密。

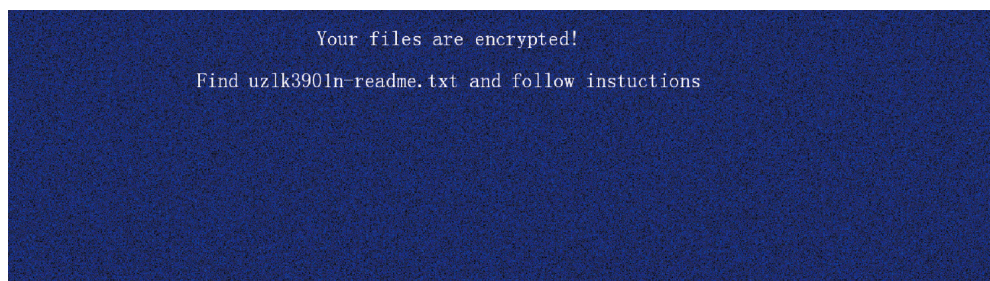


图3 替换后的桌面背景

- 加密后的文件名为原文件名后面加上新的扩展名，新扩展名由 5-10 个随机生成的字符（数字和小写字母）组成

名称	修改日期	类型	大小
AAct_x64.exe	2019/2/16 23:21	应用程序	1,499 KB
de4dot.zip.uzlk3901n	2023/3/16 11:24	UZLK3901N 文件	1,387 KB
dnSpy-4.1.8.zip.uzlk3901n	2023/3/16 11:24	UZLK3901N 文件	23,098 KB
MicrosoftRootCertificateAuthority2011.cer.uzlk3901n	2023/3/16 11:24	UZLK3901N 文件	2 KB
ndp48-devpac-enu.exe	2020/12/28 22:57	应用程序	140,259 KB
uzlk3901n-readme.txt	2023/3/16 11:24	文本文档	8 KB
windows6.1-kb2533623-x64.msu	2020/12/28 1:11	Microsoft Updat...	2,261 KB

图4 加密后的文件

- 在每个文件夹下都会创建一个勒索信文件，提示受害者去完成赎金支付操作，勒索信文件的名称格式为 {EXT}-readme.txt，{EXT} 为加密后的文件扩展名。

```

uzlk3901n-readme.txt
1  ----- Welcome. Again. -----
2
3  >> Whats Happen?
4  Your files are encrypted, and currently unavailable. You can check it: all files on your system has extension uzlk3901n
5  By the way, everything is possible to recover (restore), but you need to follow our instructions. Otherwise, you CANT return your data (NEVER).
6
7  >> What guarantees?
8
9  Its just a business. We absolutely do not care about you and your deals, except getting benefits. If we do not do our work and liabilities - nobody will not cooperate with us. Its not
10 to check the ability of returning files. You should go to our website. There you can decrypt one file for free. That is our guarantee.
11 If you will not cooperate with our service - for us, its does not matter. But you will lose your time and data, cause just we have the private key. In practice - time is much more val
12
13 >> Sensitive Data
14 Sensitive data on your network was DOWNLOADED.
15 If you DON'T WANT your sensitive data to be PUBLISHED in our blog - you have to act quickly.
16 !!! You should check our blog, using Tor Browser, your data could already be published !!!
17 http://blogxx7563u3arv76old7cykqkqyowvz44adk3vplrvd5d.onion
18
19 Data includes:
20 - Employees personal data, CVs, SS, SSN.
21 - Complete network map including credentials for local and remote services.
22 - Private financial information including: clients data, bills, budgets, annual reports, bank statements.
23 - Manufacturing documents including: datagrams, schemas, Drawings in solidworks format
24 - And more...
25
26 >> How to get access to the website?
27
28 Using a TOR browser:
29 1) Download and install TOR browser from this site: https://torproject.org/
30 2) Open our website: http://landseaf2hoy12jy0wsktypid6ucabmh7x3x3jypharvkatpad.onion
31 3) When you open our website, put the following data in the input form:
32
33 Key:
34
35 a4db3ab70d835108f8c3b3c7b71134d0d0a8f78d5555b0e9gkznuobvnd
36 v7v9lN0ymeeqcrw0a0a55Y5pVq13ba0B88x1c0gpg84x0M6g4v1A9hVgo
37 49tCHpVp0gs3ruq7lo1d8dew/b0fzmkrvA8J7m86C8f8m0rvz20t8g97f8590gk

```

图5 勒索信文件

功能演变

REvil 的开发人员定期更新 REvil 勒索软件，为了提高使用的灵活性，REvil 勒索软件采用可配置项以及可选的命令行参数来控制勒索软件的运行，下面通过比较配置项和命令行参数的变化来讨论其功能演变。

1.07 版本 (2019.11, ea4cae3d6d8150215a4d90593a4c30f2)

使用的配置项有 pk、pid、sub、dbg、wht、prc、svc、dmn、net、nobody、nname、img、fast、arn、exp。

使用的命令行参数有 -nolan、-nolocal、-path。

2.01 版本 (2020.03, bfb8e910f9480d64e8ff6ecf4b10ef4b)

使用的配置项有 pk、pid、sub、dbg、wht、prc、svc、dmn、net、nobody、nname、img、et、spsize、arn。相较于 1.07 版本，et 项替代了原 fast 项的功能，并结合 spsize 项提供了快速加密和完全加密之外

的第三种加密类型，称之为混合加密类型，即对于一个大文件，可以取多个 1MB 数据块进行加密，每个块之间的数据不进行加密，这些块之间间隔的数据大小由 spsize 项指定。删除使用 exp 配置项，不再利用漏洞 CVE-2018-8453 进行权限提升。

② 使用的命令行参数有 -nolan、-nolocal、-path、-fast、-full。相较于 1.07 版本，增加了两个参数：-fast，-full，同样是对 1.07 版本的 fast 配置项的代替。若指定命令行参数 -fast 或 -full，则会覆盖 et 配置项功能。

2.05 版本 (2021.03, 2075566e7855679d66705741dabe82b4)

① 使用的配置项有 pk、pid、sub、dbg、wht、prc、svc、dmn、net、nobody、nname、img、et、spsize、arn、rdmcut、exp。相较于 2.01 版本，恢复了 exp 配置项的使用，但功能进行了变更，被用来指示软件是否进行权限提升。增加了新的 rdmcut 配置项。

② 使用的命令行参数有 -nolan、-nolocal、-path、-fast、-full、-silent、-smode。增加了 -silent 和 -smode。-silent 用来指示是否跳过原有的进程、服务关闭和卷副本删除操作，-smode 用来重启进入 windows 安全模式运行。

新的 1.0 版本 (2022.03, db2401798c8b41b0d5728e5b6bbb94cf)

① 配置信息里存在的配置项有 pk、pid、sub、dbg、wht、prc、accs、svc、net、nbody、nname、exp、img、et、spsize、arn、rdmcut。相较于 2021 年的 2.05 版本，去掉了 dmn 项，增加了 accs 项，该项保存用于访问目标共享网络资源的登录凭证。此外，dbg、net、exp、arn 相关功能被删除，pid、spsize、rdmcut 未被使用。sub 的值与之前格式不同，此前版本为整数，该版本为 Guid。

② 增加 t 参数，该参数值用于计算正确的 rc4 密钥长度，错误的值会导致程序无法运行。



图6 使用参数“t”计算rc4密钥长度

③ 删除了地区白名单检查操作。

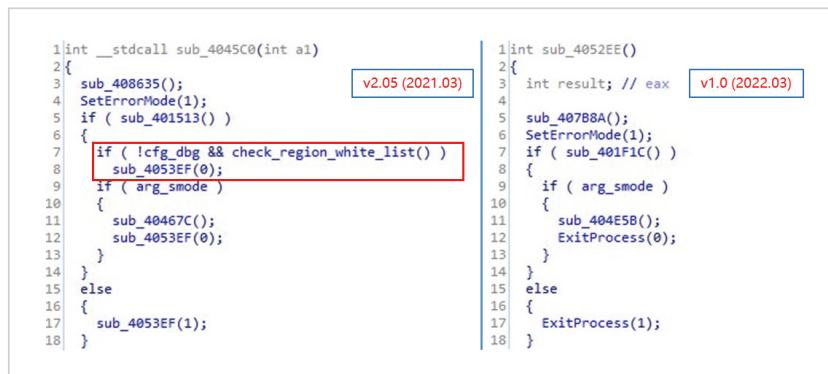


图7 删除地区白名单检查操作

表 1. 配置信息中各配置项功能描述汇总

配置项	描述
pk	Base64 编码的攻击者公钥
pid	附属组织的 ID
sub	攻击活动标识符
dbg	指示是否在 Debug 模式下运行，Debug 模式不检查键盘布局 and 系统语言。
et	指示文件加密类型，-0：完全加密；-1：快速加密，只加密文件起始处的 1MB 数据；-2：混合加密，每隔 spsize 项指定的数据量加密 1MB 数据块。
spsize	给出加密类型 et 为 2 时要跳过的 MB 数。
wipe	指示是否应擦除 wfld 元素中指定的文件夹。（未使用）
wfld	要擦除的文件夹列表。（未使用）
wht	包含三个 REvil 不会加密的白名单列表：-fld：文件夹白名单；-fls：文件白名单；-ext：扩展名白名单。
prc	需要关闭的进程名列表。
net	指示是否回传相关信息。
dmn	C2 域名列表，域名之间由“；”分隔。
svc	需要关闭的服务名列表。
nbody	Base64 编码的勒索信模板。
nname	勒索信文件名模板。
exp	指示是否进行权限提升。
img	显示在桌面背景中的文本模板，以 Base64 编码。
arn	指示是否设置注册表实现持久化。
rdmcnt	将勒索信息文件保存的最大文件夹数。如果为零，将勒索信息文件保存到所有文件夹下。
accs	一个登录凭证（用户名和密码）列表，用于访问目标受保护的共享网络资源。

表 2. REvil 勒索软件的命令行参数功能

参数	描述
-nolan	不加密共享网络存储的文件。
-nolocal	不加密本地存储的文件。
-path	指定加密的目录，提供此参数时，桌面背景图像保持不变。
-full	加密目标文件中的所有数据。
-fast	只加密每个文件的前 1 MB 数据，与 -full 参数互斥。
-silent	跳过关闭进程和服务、及删除卷影副本操作。
-smode	在 windows 安全模式下重启运行。

详细分析

本文对新的 1.0（2022.03）版本样本进行详细分析。

原始文件名	
MD5	db2401798c8b41b0d5728e5b6bbb94cf
文件大小	85 KB (87040 字节)
文件类型	PE32+, exe
编译信息	Microsoft Linker(14.0, Visual Studio 2015 14.0*)[GUI32]

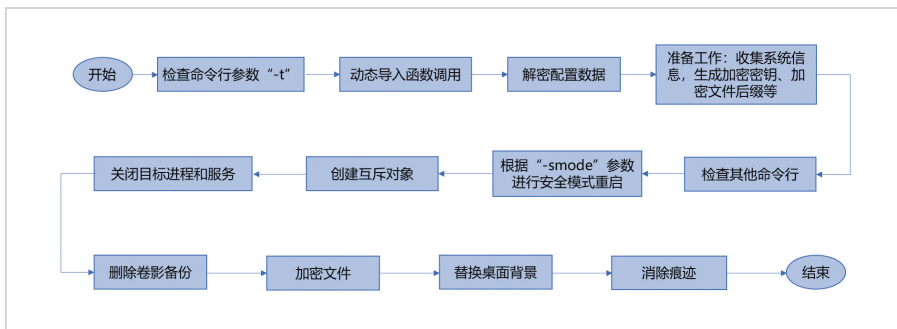


图8 样本执行流程

Revil 勒索软件使用的字符串都是通过 rc4 算法加密保存，在使用时解密，这样可以对一些敏感字符串进行保护，Rc4 密钥和密文前后相邻保存在一起。程序运行首先解析命令行中的“-t”参数，若没有则不能执行，并且值也是特定的，用于在解密字符串时获得 rc4 密钥的长度，该样本所使用的“-t”参数值为“798690758”。

Revil 勒索软件的函数导入表为空，在运行时动态导入所需函数调用，这样一定程度上可以躲避杀毒软件的检测；其将混淆后的函数导入表保存为一个 DWORD（4 字节）数组，每一个 DWORD 值代表一个函数，动态导入的过程如下：

- ❶ 被混淆的 DWORD 值通过异或和移位进行转换，新的 DWORD 值被拆分为两部分，前 11 bits 指定 DLL 名，后 21 bits 指定此 DLL 中导出的函数；
- ❷ DLL 通过 LoadLibrary（）函数加载；
- ❸ 使用自定义算法对该 DLL 中导出的每个函数名进行哈希，将哈希值与 21 bits 值进行比较，如果它们匹配，则用正确函数地址替换混淆的 DWORD 值，完成动态导入操作。

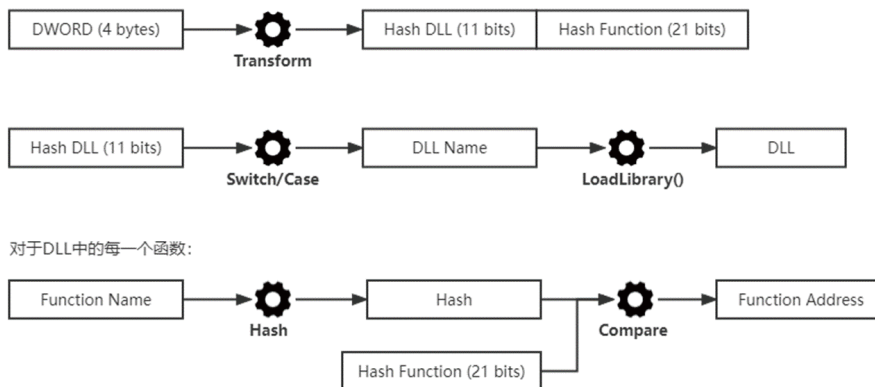


图9 动态解析函数调用过程

接下来，Revil 将解密配置数据，配置数据同样是用 rc4 算法加密。在解密之前先用 crc32 校验数据是否被篡改，明文数据为 JSON 格式，然后解析各个字段值。

rc4密钥	Crc32校验值	密文长度	配置数据密文
-------	----------	------	--------

图10 配置信息密文保存格式

配置信息明文 JSON 格式如下：

```

2  "pk": "XnG+yLIqCkX9s+D2nHAFt7kFgmKXtqvmC2w8ePx2I=",
3  "pid": "b78ad33f-32e9-4756-8c7b-83d46f8a0e19",
4  "sub": "bb9ced79-3e79-4649-8fcc-3d4295fab08",
5  "dbg": false,
6  "wht": {
7    "fld": ["program files","intel","msocache","tor browser","system volume information","$windows-~ws","program f
8    "fls": ["boot.ini","autorun.inf","ntuser.ini","ntldr","desktop.ini","bootfont.bin","ntuser.dat","thumbs.db","b
9    "ext": ["cab","ico","ani","rom","diagpkg","diagcfg","dll","msu","nomedia","rtp","psi","key","diagcab","exe","o
10  },
11  "pnc": ["encsvc","infopath","ocssd","xfssvcon","firefox","visio","outlook","ocomm","oracle","tbirdconfig","isqlpl
12  "acss": ["STRATFORD\\dkadmin5Stratford1","STRATFORD\\wilis43lpd3sk","STRATFORD\\tcadmin\\Martingpcpal","STRATFORD\\
13  "svc": ["mepocs","svc$","sql","veeam","sophos","memtas","vss","backup" ],
14  "net": false,
15  "nbody": "LQALACBAPQA9AD8A1ABXAGUABjAG8ABQ8IAC4A1ABBAGcAYQBpAG4ALgAgAD8APQA9ACBALQATAABACgANAAPgA+ACAAVvBoAGEA
16  "nname": "(EXT)-readme.txt",
17  "exp": false,
18  "img": "WQBvAHUJAcgAGAGYaaQBsAGUAcuAGAGEAcgB1ACAAZQBuAGMACgB5AHAAAdAB1AGQAIQANAAADQAKAEYAaQBuAGQAIAB7AEUAwABUAH0ALQ
19  "et": 2,
20  "spsize": 0,
21  "arn": false,
22  "rdmcnt": 0

```

图11 配置信息明文JSON格式

接下来，Revil 会做一些加密前的准备工作：

- 使用 ECDH 算法生成加密会话密钥对，会话私钥用配置里“pk”项保存的攻击者公钥进行 ECDH 和 AES 混合加密后保存，会话公钥和攻击者公钥直接保存。

```

108 generate_key_pair(session_private_key, (int*)&session_public_key);
109 size1 = 0;
110 size2 = 0x20;
111 size3 = 0x20;
112 encrypted = ecdh_aes_encrypt(session_private_key, (int*)&attack_public_key_config, 0x20, &size1);
113 zero(session_private_key, 0x20u);
114 if ( !encrypted )
115     return 0;
116 copy_((int*)&secret, encrypted, size1);
117 if ( !reg_set_value((int)v10, 0x80000002, (int)v22, 3, (int*)&attack_public_key_config, size3) )
118     reg_set_value((int)v10, 0x80000001, (int)v22, 3, (int*)&attack_public_key_config, size3);
119 if ( !reg_set_value((int)v10, 0x80000002, (int)v20, 3, (int*)&session_public_key, size2) )
120     reg_set_value((int)v10, 0x80000001, (int)v20, 3, (int*)&session_public_key, size2);
121 if ( !reg_set_value((int)v10, 0x80000002, (int)v12, 3, (int*)&secret, size1) )
122     reg_set_value((int)v10, 0x80000001, (int)v12, 3, (int*)&secret, size1);

```

图12 密钥生成

- 利用系统驱动的卷序列号和 cpuid 获取的信息为受害者生成一个标识 id (UID)，该 id 会保存到勒索信中，并返回给攻击者。

```

15 uid = heap_alloc(34);
16 computer_name = (WORD *)get_computer_name(); // computer name
17 v2 = (int)computer_name;
18 if ( computer_name )
19     v3 = (unsigned __int16)computer_name[sub_4059C0(computer_name)] ^ (unsigned __int16)*computer_name;
20 else
21     v3 = 1337;
22 if ( !uid )
23     return 0;
24 memset((char *)cpu_info, 0, sizeof(cpu_info));
25 get_cpu_info(cpu_info); // cpu info
26 windows_vol_sn = get_windows_volume_serial_number(); // 系统盘序列号
27 windows_vol_sn_crc = crc32((unsigned __int8 *)&windows_vol_sn, v3, 4);
28 decrypt_rc4(argv_t ^ 0x2F980DCB, (int)&unk_4150C3, (int)&unk_4150C3 + (argv_t ^ 0x2F980DCB), 16, format_string); // L"%08X%08X"
29 v10 = 0;
30 windows_vol_sn = windows_vol_sn;
31 cpu_info_crc = crc32((unsigned __int8 *)cpu_info, windows_vol_sn_crc, strlen((const char *)cpu_info));
32 wprintf(uid, format_string, cpu_info_crc, windows_vol_sn);

```

图13 UID生成

- 生成新的加密文件扩展名，一般 5-10 个随机字符。

- 获取用户、系统和勒索软件相关信息，保存到一个 json 格式中，经过加密后保存到勒索信中。

```
38 decrypt_rc4(argv_t ^ 0x2F9B0DC0, (int)&unk_414A31, (int)&unk_414A31 + (argv_t ^ 0x2F9B0DC0), 320, format_string);//
39
40 // {"ver": "%d", "pid": "%s", "sub": "%s", "pk": "%s",
41 // "uid": "%s", "sk": "%s", "unm": "%s", "net": "%s",
42 // "grp": "%s", "lng": "%s", "bro": false, "os": "%s",
43 // "bit": "%d", "dsk": "%s", "ext": "%s"}
44
45 v9 = 0;
46 wprintf(
47     info,
48     0x20000,
49     format_string,
50     0x100,
51     cfg_sub,
52     cfg_sub,
53     cfg_pk,
54     uid,
55     secret_base64,
56     user_name,
57     computer_name,
58     domain,
59     system_lang,
60     os_name,
61     cpu_arch,
62     disk_info,
63     file_extension + 2);
64 v6 = strlen(info);
65 *this = 0;
66 v7 = 2 * v6;
67 v3 = v7 ? ecdh_aes_encrypt(info, (int)&attack_public_key_binary_2, v7, this) : 0;
```

图14 KEY生成

上述信息也会被保存到注册表中。接下来，Revil 检查其他命令行参数，如图 15 所示。

```
183 decrypt_rc4(argv_t ^ 0x2F9B0DC0, (int)&unk_4148A9, (int)&unk_4148A9 + (argv_t ^ 0x2F9B0DC0), 12, v33);// -nolan
184 v34 = 0;
185 decrypt_rc4(v11 ^ 0x2F9B0DCC, (int)&unk_414C31, (int)&unk_414C31 + (v11 ^ 0x2F9B0DCC), 16, v39);// -nolocal
186 v40 = 0;
187 decrypt_rc4(v11 ^ 0x2F9B0DCB, (int)&unk_4145B5, (int)&unk_4145B5 + (v11 ^ 0x2F9B0DCB), 10, v25);// -path
188 v26 = 0;
189 decrypt_rc4(v12, (int)&unk_413E4B, (int)&unk_413E4B + v12, 14, v37);// -silent
190 v38 = 0;
191 decrypt_rc4(v11 ^ 0x2F9B0DCB, (int)&unk_414173, (int)&unk_414173 + (v11 ^ 0x2F9B0DCB), 12, v35);// -smode
192 v36 = 0;
193 arg_nolan = sub_401E0F(v33) == 0;
194 arg_nolocal = sub_401E0F(v39) == 0;
195 arg_path = sub_401D81(v25);
196 arg_silent = sub_401E0F(v37) == 0;
197 arg_smode = sub_401E0F(v35);
198 decrypt_rc4(argv_t ^ 0x2F9B0DCE, (int)&unk_413DAA, (int)&unk_413DAA + (argv_t ^ 0x2F9B0DCE), 10, v25);// -fast
199 v26 = 0;
200 arg_fast = sub_401E0F(v25);
201 decrypt_rc4(argv_t ^ 0x2F9B0DC0, (int)&unk_413DF2, (int)&unk_413DF2 + (argv_t ^ 0x2F9B0DC0), 10, v25);// -full
202 v26 = 0;
203 arg_full = sub_401E0F(v25);
```

图15 检查命令行参数

若使用了“-smode”参数，REvil 勒索软件将重启计算机，以带网络连接的 Windows 安全模式运行，这样可以避免 AV 检测和防护；首先修改登录密码，然后设置用户以管理员权限自动登录，接着设置自身下次启动自运行，并设置下次启动时使用 bootcfg 或 bcdedit 进入 Windows 安全模式，最后运行命令重启计算机。

```
82 decrypt_rc4(argv_t ^ 0x2F9B0DC0, (int)&unk_41408F, (int)&unk_41408F + (argv_t ^ 0x2F9B0DC0), 0x62, v21);// SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnce
83 v12 = 0;
84 decrypt_rc4(v10 ^ 0x2F9B0DD0, (int)&unk_41403C, (int)&unk_41403C + (v10 ^ 0x2F9B0DD0), 0x4E, v35);// "Aqftm6
85 v36 = 0;
86 decrypt_rc4(v10 ^ 0x2F9B0DD0, (int)&unk_414424, (int)&unk_414424 + (v10 ^ 0x2F9B0DD0), 14, v39);// "SrTtTZ
87 v40 = 0;
88
89 if (v30 && (unsigned __int64)sub_405F51() > 0x501u)
90     WoudServerToUserRedirection(0);
91 v12 = argv_t ^ 0x2F9B0DD0;
92 if (v35 >= 0x0000)
93 {
94     decrypt_rc4(v12, (int)&unk_4140CB, (int)&unk_4140CB + v12, 0x27, v33);// bcdedit /set (current) safeboot network
95     v34 = 0;
96     Wintex(v33, 5);
97     decrypt_rc4(argv_t ^ 0x2F9B0DC9, (int)&unk_414950, (int)&unk_414950 + (argv_t ^ 0x2F9B0DC9), 0x4E, v27);// "bcdedit /deletevalue (current) safeboot"
98     v36 = 0;
99     v17 = 2 * sub_4059C0(v27) + 2;
100     v14 = ((int (__cdecl *) (char *, int, _JORD *, int))sub_404E08)(v39, v15, v27, v17);
101 }
102 else
103 {
104     decrypt_rc4(v12, (int)&unk_4140FA, (int)&unk_4140FA + v12, 0x27, v33);// bootcfg /raw /a /safeboot:network /id 1
105     v34 = 0;
106     Wintex(v33, 5);
107     decrypt_rc4(argv_t ^ 0x2F9B0DC9, (int)&unk_414508, (int)&unk_414508 + (argv_t ^ 0x2F9B0DC9), 0x3C, v39);// bootcfg /raw /fastdetect /id 1
108     v36 = 0;
109     v14 = 2 * sub_4059C0(v27) + 2;
110     v14 = ((int (__cdecl *) (char *, int, _JORD *, int))sub_404E08)(v39, v15, v28, v14);
111 }
112 if (v14)
113     return 0;
114 if (v30 && (unsigned __int64)sub_405F51() > 0x501u)
115     WoudServerToUserRedirection(0);
116 if (sub_40800E((void *)19))
117     ShutdownSystem(1); // 关机
```

图16 安全模式重启

创建互斥对象，避免重复运行：

```
8 decrypt_rc4(argv_t ^ 0x2F9B0DC9, (int)&unk_415154, (int)&unk_415154 + (argv_t ^ 0x2F9B0DC9), 86, v3);// Global\03C88601-E0B0-EC70-7CF4-2624752A0Bf9
9 v4 = 0;
10 v0 = 0;
11 mutex_obj = CreateMutex(0, 0, v3);
12 if (mutex_obj && GetLastError(v1) == 183)
13     v0 = 1;
14 return v0;
```

图17 创建互斥量

默认情况下, REvil 会根据配置项 “prc” 和 “svc” 给定的列表来关闭目标服务和进程, 并删除卷影副本, 但 “-silent” 参数可以使 REvil 跳过这个操作, 此外还会清空回收站里的文件。

```
14 sub_40860E(20); // SeDebugPrivilege
15 if ( arg_silent )
16 {
17     v1 = CreateThread(0, 0, sub_4044B8, 0, 0, 0); // 创建COM对象wbem来关闭进程和服务
18     if ( v1 )
19         CloseHandle(v1);
20     sub_403A26(); // 遍历所有服务, 关闭目标服务
21     sub_4063FA(v2); // 遍历所有进程, 关闭目标进程
22     v3 = CreateThread(0, 0, sub_404775, 0, 0, 0); // 创建COM对象wbem来关闭进程和服务
23     if ( v3 )
24         CloseHandle(v3);
25 }
26 sub_40860E(9); // SeTakeOwnershipPrivilege
```

图18 关闭进程、服务, 删除卷影副本 ▲

上述操作完成后, 开始加密文件。REvil 勒索软件利用 IOCP 模型创建多线程, 使用 ECDH 和 Salsa20 共同实现加密用户文件, 对配置信息 “wht” 字段中指定的白名单进行豁免。Revil 将文件加密过程分为四步:

- ❶ 从文件中读取 1 MB 数据块 (如果文件小于 1 MB, 则读取整个文件);
- ❷ 数据块被加密并写回文件。根据加密类型 (快速加密、完全加密) 来确定下一步是 3) 还是 1);
- ❸ 将加密相关的元数据添加到文件的末尾;
- ❹ 修改文件扩展名。

```
46 switch ( *(_DWORD *) (v4 + 344) ) // step flag
47 {
48     case 1:
49         file_encrypt_step1(v4); // 读取文件内容
50         break;
51     case 2:
52         next_step = 1;
53         if ( cfg_et == 1 )
54             next_step = 3;
55         file_encrypt_step2(next_step); // 加密文件内容
56         break;
57     case 3:
58         file_encrypt_step3(); // 加密后数据写回原文件
59         break;
60     case 4:
61         file_encrypt_step4(a1, v4); // 修改文件扩展名
62         break;
63 }
```

图19 文件加密的4个步骤 ▲

替换桌面背景, 显示 “img” 配置项保存的文本。最后消除痕迹: 清理内存空间、删除样本文件。

加密逻辑

1 密钥生成

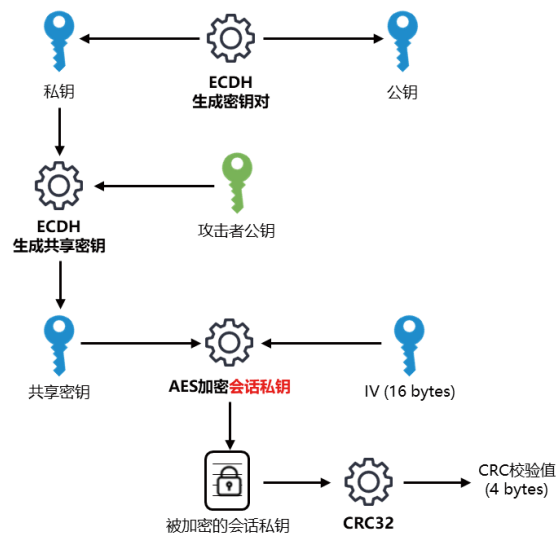
勒索软件使用复杂的密钥系统来保证文件加解密的可靠性, 以及确保无法被除自身之外的其他人解密。

● 生成会话密钥对: Revil 勒索软件首先使用椭圆曲线 Diffie-Hellman (ECDH) 算法生成一个会话密钥对, 使用的曲线算法为 Curve25519。



图20 会话密钥生成

● 加密会话私钥: 攻击者公钥保存在“pk”配置项中, 攻击者公钥和 ECDH 新生成的私钥生成一个共享密钥, 此共享密钥用于 AES 加密会话私钥。



$$\text{密文(88 bytes)} = \text{被加密的会话私钥(32 bytes)} + \text{公钥(32 bytes)} + \text{IV(16 bytes)} + \text{CRC校验值(4 bytes)}$$

图21 加密会话私钥

● 保存必要的密钥数据: 最后得到的密文数据长度为 88 字节, 和会话公钥及攻击者公钥分别保存到三个注册表表项中, 其他数据和密钥从内存中释放, 因为它们不会用于后续的文件加密。

2 文件加密

对于每个文件, 将生成一个新的密钥对。新的文件私钥和会话公钥通过 ECDH 算法生成文件加密密钥, 该密钥用于 Salsa20 算法来加密文件内容。因此, 每个文件都是使用不同的密钥进行加密, 没有攻击者的私钥无法进行解密。

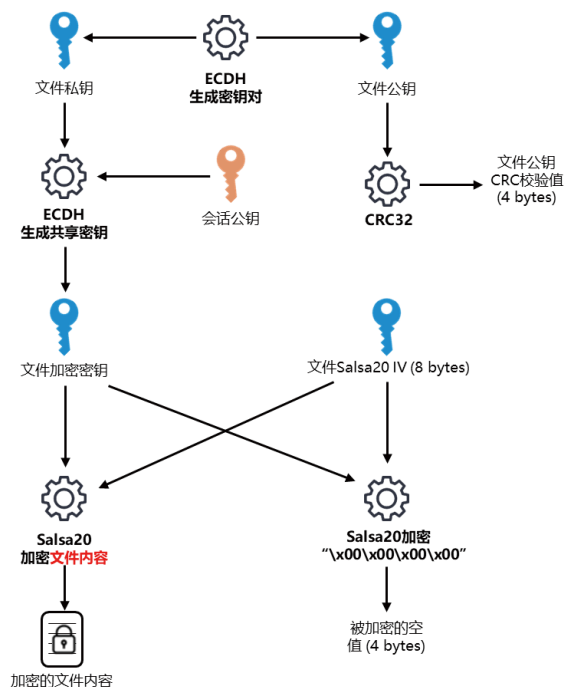


图22 文件加密过程

元数据被添加到加密文件的末尾，元数据包含解密该文件所需的除了攻击者私钥外的各种信息，使用这些信息配合攻击者私钥即可解密该文件。

被加密的文件内容
用配置信息里公钥加密的密文 (88 bytes)
用硬编码在样本里的公钥加密的密文 (88 bytes)
该文件的ECDH公钥(32 bytes)
该文件的Salsa20 IV值 (8 bytes)
该文件ECDH公钥的 CRC32 校验值 (8 bytes)
加密类型 et (4 bytes)
间隔大小 spsize (4 bytes)
加密的空值 (4 bytes)

图23 添加元数据到文件末尾 ▲

3

总结

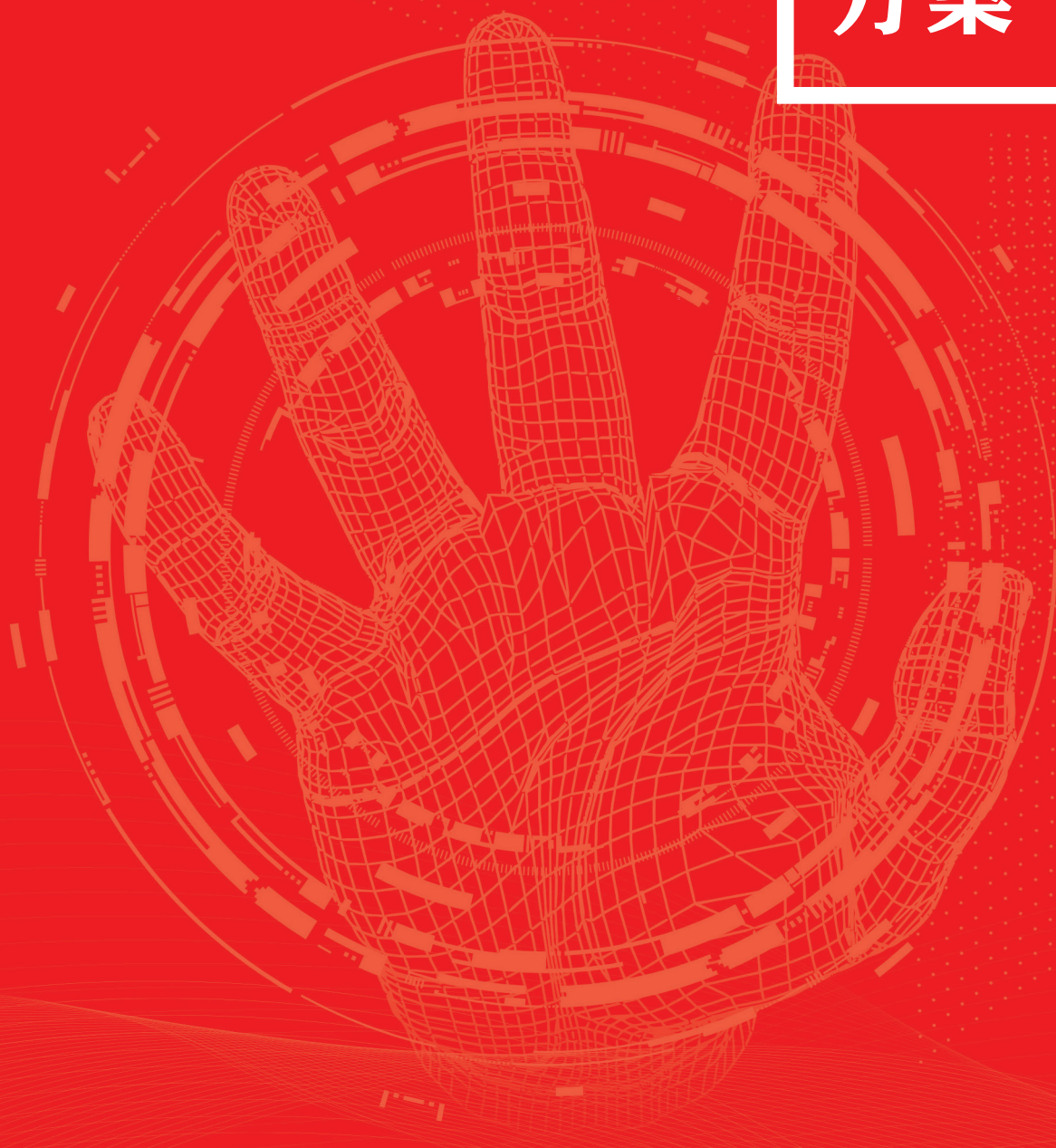
Revil 勒索组织自 2019 年初露锋芒后，就因其成熟的 RaaS 运营模式、多样化的勒索策略以及功能强大的勒索软件，成为最流行的勒索软件网络犯罪组织之一。由于其攻击行为愈发猖獗，终是遭受到了国际执法机构的追捕打击，一度销声匿迹，停止运营。然而，面对勒索软件攻击的暴利诱惑，残存的 Revil 勒索组织人员沉寂半年之后卷土重来。本文也详细介绍了 Revil 勒索软件所使用的各种技术手段，以帮助读者对该款勒索软件有更深层的认识。



Chapter Three

第三章

应用实践
方案



云—网—端—管融合， 构筑勒索防护新防线



文 / 张宇

2023 年伊始，勒索攻击事件就频繁登上网络安全版面的头条。2023 年 1 月，黑客组织宣布首次完成了对工控系统设备的勒索加密；2 月，勒索软件攻击者大规模主动利用一个 VMWare ESXi 漏洞攻击并破坏了欧美数千台服务器；同月，以色列最好的大学以色列理工学院被勒索团队攻击；4 月，台企微星遭到勒索攻击，被索要巨额赎金。勒索攻击已成为目前政府、企业、医院、学校等组织所受到的威胁最大的攻击方式之一，仿佛已成为企业头上的“达摩克斯之剑”。

1 攻防视角下的勒索攻击防护思路

勒索攻击的目的是获取关键信息数据的访问和控制权限，并进行数据加密和勒索，其入侵的攻击思路主要有以下三种：

- 🛡️ 利用应用漏洞、SQL 注入等手段入侵企业对外应用服务器，并以此为跳板再入侵数据中心关键服务器。
- 🛡️ 利用邮件钓鱼、社工钓鱼等方式投递钓鱼后门，直接获取办公 PC 权限，并以此为跳板入侵数据中心关键服务器。
- 🛡️ 获取分支机构终端和服务器权限，以此为跳板入侵数据中心。

通过对勒索攻击行为的分析，可以发现，勒索攻击具有目标明确、攻击加密、持续威胁的特点。在勒索病毒投递的过程中，往往采用加密流量进行传递，传统安全检测方式难以识别发现。此外，由于勒索攻击的目标明确，关键数据往往会遭到 24 小时不间断的攻击，这也要求了防御者需要具备持续监测和分析能力。

新华三针对勒索攻击的特点，站在攻防视角，防护手段覆盖在入侵、部署、勒索等各阶段，提升安全纵深防护效果，防止单点防护手段被绕过。

- 入侵阶段：发现攻击者攻击意图，禁止通过攻击进入；结合 APT 攻击防护产品，主动诱捕攻击者行为；结合威胁情报，协同攻击检测。
- 部署阶段：云网协同的 AI 病毒查杀，防范勒索样本投递和内网传播；分析勒索病毒传递途径，勒索内网传播态势感知展示；联动网内设备处置勒索，防止内网横向传播行为。
- 勒索阶段：一键终止勒索行为，禁止勒索病毒运行；勒索病毒攻击溯源，事件止损回溯；引入安全运营理念，提升安全防护管理效果。

2 新华三勒索防护安全解决方案

新华三勒索防护安全解决方案结合网络安全、终端安全、APT 攻击防护等理念，以勒索攻击为主要防御场景，“云-网-端-管”多层次多手段融合，实现勒索攻击可感、可见、可控、可管。

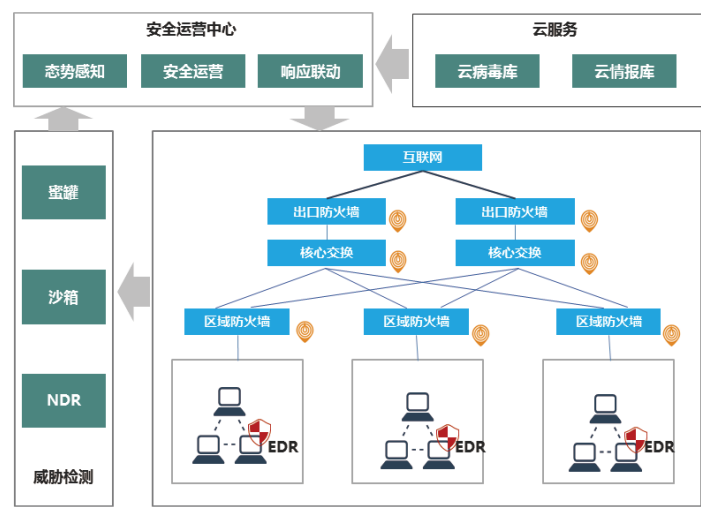


图1 新华三勒索防护安全解决方案

威胁感知以主动诱捕方案为核心。部署 APT 防护类产品，构造陷阱，混淆攻击目标，精确感知勒索攻击行为，将勒索攻击隔离到构造出的虚假的环境中，从而保护内部真实资产；同时，记录攻击行为，并获取黑客的网络身份和指纹信息，对其进行攻击取证和溯源。通过交换机、防火墙等网络设备和安全设备，与 APT 防护产品联动，充分利用网安设备部署的广度，最大化内部网络覆盖度，降低漏报概率，理论上可 100% 覆盖，解决传统 APT 防护最大的问题。

终端部署 EDR 产品，采用智能沙箱技术、机器学习技术、大数据关联分析技术，通过云端威胁情报、异常行为分析等方式，主动发现来自外部或内部的安全威胁，自动化完成发现、隔离、修复、补救、取证并协助防病毒软件进行策略部署等一系列工作，从而有效对终端进行全生命周期的安全防护，防止勒索风险的横向扩散。

分析管理以安全运营中心为核心，基于安全日志、用户行为、终端日志、业务数据、资产状态等多源数据，结合外部情报，通对外部攻击分析、系统合规自检等手段，实现“事态可评估”；通过对攻击趋势分析和终端行为检测，实现“趋势可预测”；通过对未知威胁的智能检测识别、资产的状态监控和多维度风险分析，实现“风险可感应”；通过对攻击溯源取证、云网端协同联动、工单流程闭环处理和设备策略自适应调整，实现“知行可管控”。全网勒索行为感知，溯源攻击行为，联动网络设备、安全设备，实现自动化响应处置。

③ 轻量化勒索防护防御方案

在中小企业、普教等场景下，用户没有太多预算，但勒索防护攻击又非常频繁，如何低成本高质量进行勒索防护，是此类场景下的核心。

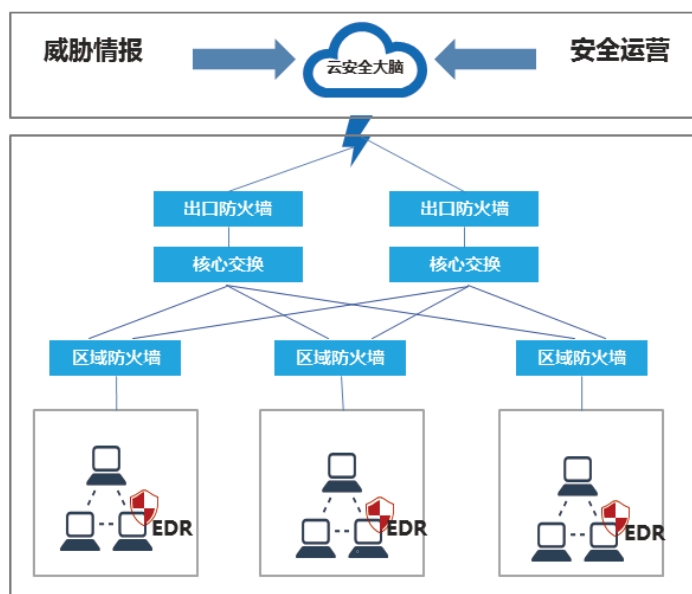


图2 勒索防护轻量化解决方案

新华三基于此场景，推出轻量化勒索防护防御方案。采用“本地产品+云服务”方式，本地轻量化部署，仅需要在出口和区域边界部署防火墙，端侧部署 EDR 产品，云端以安全大脑为核心，结合威胁情报，快速应对病毒变种，低成本实现专家级安全运营。

实现零配置开局，设备开局文件连接大脑平台，大脑自动化开局配置导入，分钟级完成开局上线流程。对设备状态实时监测，通过智能化巡检，规范化管理，提高运维管理效率。支持批量策略管理，支持批量优化勒索防护策略，支持设置定时任务进行安全策略批量配置，设备配置模板集中下发。

④ 勒索攻击防御没有终点

网络安全防御者和勒索攻击组织的对抗愈演愈烈，不断涌现的新组织和不断迭代的新病毒对网络安全造成了极大的威胁。悲观者说，“道高一尺，魔高一丈”，从博弈论的角度防御方往往处于劣势，网络失陷不可避免。但是，作为网络安全从业者，笔者始终坚信主动、智能、完善的安全防御体系能够有效对抗勒索组织。网络安全防御方应该站在攻防视角，充分了解勒索攻击方式，不断打磨安全解决方案，提升安全防护的有效性。网络防御并没有一个终点，应该不断演进，“道高一尺，魔高一丈，愈进愈阻，永无止息”。



Chapter Four

第四章

测论



/ 霍龙

1 勒索攻防过程分析

攻击链是指攻击者成功渗透网络并窃取数据所需的一系列阶段，每个阶段都展示了攻击者的一个特定目标路径。所以围绕攻击链来设计测试勒索病毒是一种有效的方法，因为它聚焦于实际的攻击是如何发生的。一般来说，勒索病毒的攻击链分为以下四大步骤：

- 1 边界突破：**攻击者首先通过侦察获取到目标的信息，然后通过 RDP 爆破、SSH 爆破、钓鱼邮件、扫描攻击、系统漏洞以及社会工程等手段来攻破目标的边界，搜集并获取未授权访问的设备信息。
- 2 利用漏洞：**攻击者一般会通过黑进一个开放的端口向内网传输恶意程序，建立命令控制管道，允许攻击者执行命令；或通过远控木马安装下载程序，从互联网下载额外的恶意程序，允许攻击者执行；或通过钓鱼邮件诱导用户点击安装，然后触发勒索病毒加密。
- 3 内网横向渗透：**由内网的某一台主机扩散至内网的其他主机，一般通过 RDP 爆破、SSH 爆破、端口扫描以及钓鱼邮件等。
- 4 加密勒索：**攻击者执行勒索病毒并向内网扩散，受害主机文件被加密并弹窗告知用户索要赎金。

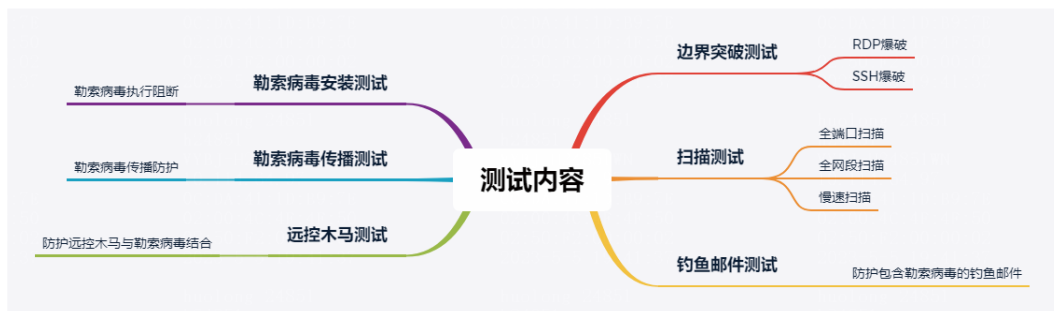


图1 新华三勒索病毒测试设计思维导图 ▲

环境搭建

为了测试勒索病毒攻击场景，我们需要模拟勒索病毒攻击的全过程。通过上文对勒索病毒攻击链的梳理来准备测试环境，如图 2 所示：

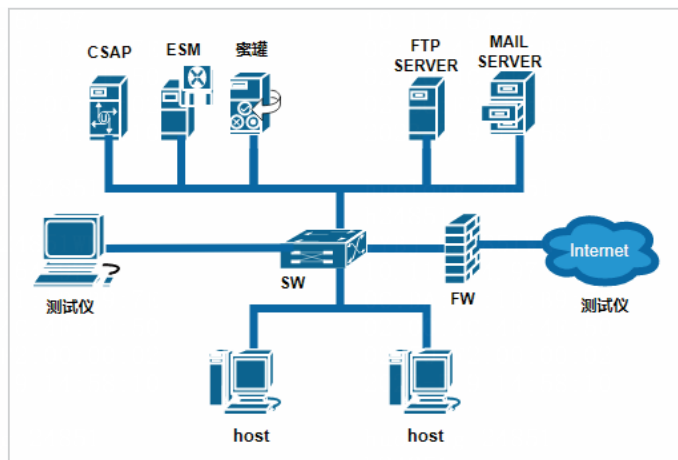


图2 新华三勒索病毒解决方案测试组网

CSAP（新华三态势感知平台）

态势感知平台通过采集全网原始流量数据，并结合云端的威胁情报，对海量安全数据进行挖掘和关联分析，对攻击、威胁、脆弱性、流量和行为等五大态势进行感知，生成全方位的安全全景视图，使我们能够快速准确地掌握当前网络地安全态势，并以此作为依据进行联动响应。

ESM（新华三终端安全管理系统）

终端安全管理系统的防病毒控制中心采取了病毒库检测、行为检测、云查杀等多项先进技术，通过技术组合，提供更为全面和有效的安全防护服务。此外，ESM 还提供了全网病毒查杀、全网远程设置、远程杀毒、集中式授权管理、全面监控邮件客户端、统一的管理界面、直接操纵客户端等多种复杂的网络管理功能，让以往繁重的日常管理工作变得轻松自如。

FW（新华三防火墙）

防火墙既可以阻止来自因特网的对受保护网络的未授权访问，也可以允许内部网络用户对因特网进行 Web 访问或收发邮件等。防火墙作为访问因特网的权限控制关口，既可以进行身份鉴别、信息加密、访问控制等，也可以进行病毒检测、异常行为检测隔离、拦截恶意邮件、恶意域名检测等。

蜜罐

蜜罐设备用于对攻击方进行欺骗，通过布置作为诱饵的主机或者网络服务等，诱使攻击方对诱饵实施攻击，从而捕获攻击行为并对其进行分析。通过了解攻击方所使用的工具与方法，推测攻击意图和动机，能够让我们清晰地了解所面对的安全威胁，并通过技术和管理手段来增强系统的安全防护能力。

勒索攻击尝试突破阶段 – 突破边界测试

暴力破解测试：攻击者主要采用穷举法，通过尝试大量可能的密码或密钥来破解用户账号或加密信息。

RDP 远程协议能够让用户通过本地电脑远程连接服务器，大部分的 Windows 服务器都有相应的客户端。SSH 是 Linux 系统主要的远程连接管理方式，现如今 SSH 暴力破解依然十分普遍。

- 对 root 账号进行 SSH 暴力破解测试；
- 对 administrator 账号进行 RDP 暴力破解测试；
- 使用中国姓名 top 1000 作为用户名进行 SSH/RDP 暴力破解测试；
- 使用 top 10000 password 字典进行 SSH/RDP 暴力破解测试。

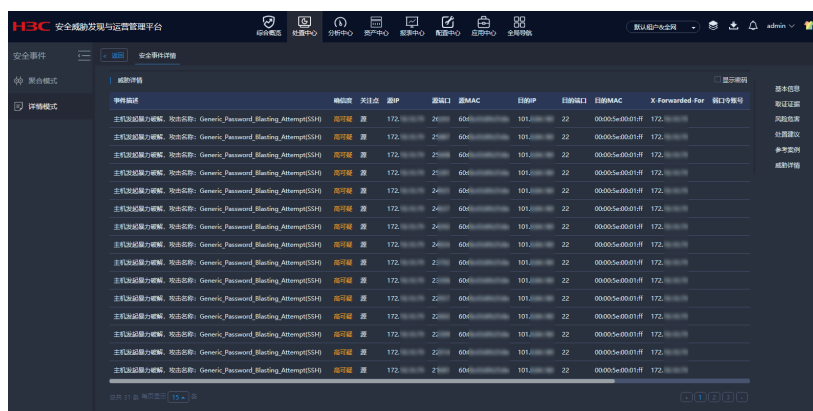


图3 新华三CSAP暴力破解检测效果

勒索攻击尝试入侵阶段 – 钓鱼邮件测试

攻击者伪装成同事、合作伙伴、朋友、家人等用户信任的人，通过发送电子邮件的方式，诱使用户回复邮件、点击嵌入邮件正文的恶意链接或者打开邮件附件以植入木马或者勒索病毒等，进而窃取用户敏感数据、账号密码等。

图 4 是模拟发送给 HR 的钓鱼邮件，其中附件压缩包文件包含勒索病毒程序：

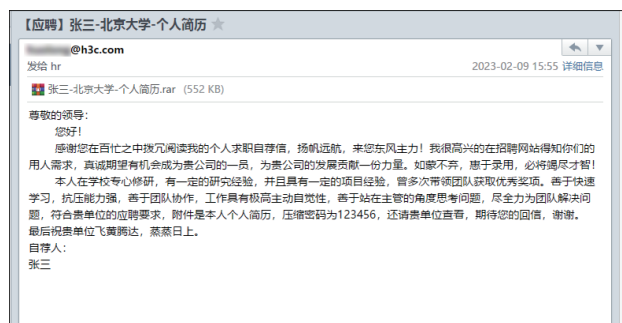


图4 钓鱼邮件示例

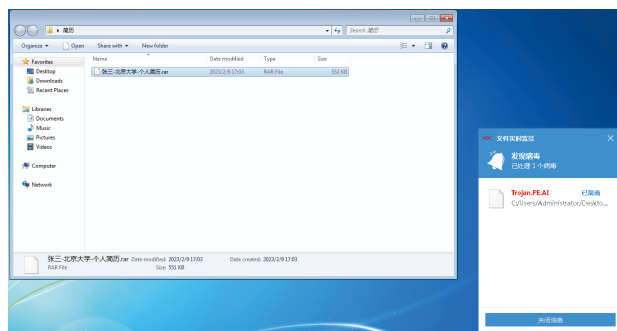


图5 新华三ESM拦截发起勒索攻击的病毒文件

勒索攻击尝试扫描阶段 - 扫描攻击测试

攻击者将网络请求发送到目标服务器或者目标网段，用于发现目标主机的工作状态，判断目标主机是否联网并处于开机状态；或者嗅探主机开放的端口，利用端口对应用程序中的漏洞进行攻击。

- 使用 NMAP 对内网进行全网扫描；
- 使用 NMAP 对内网主机的端口进行扫描；
- 使用 NMAP 对内网主机进行慢速扫描。

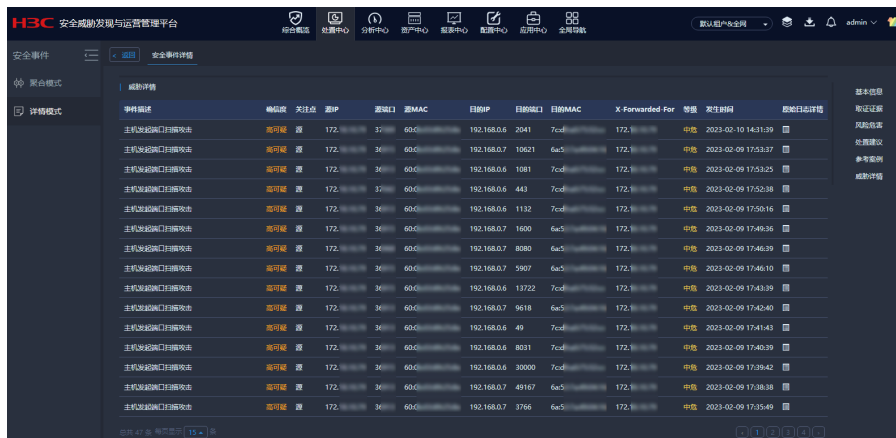


图6 新华三CSAP检测扫描攻击效果图 ▲

勒索攻击尝试植入阶段 - 远控木马测试

远程控制木马可以对目标计算机进行交互性访问，可以下发相应的指令触发恶意软件的功能，也能获取目标系统的各种数据，其交互是双向的。

利用工具 Cobalt Strike 进行远控木马正向连接测试;

- 执行远控木马反向连接测试；
- 通过远控木马植入勒索病毒。

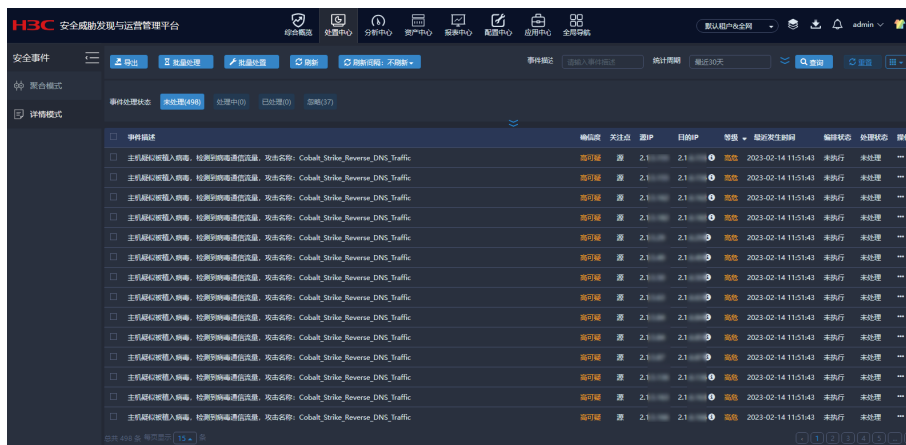


图7 新华三CSAP检测恶意木马连接 ▲

勒索攻击尝试传播阶段 – 勒索病毒传播测试

- 使用远程桌面入侵后投毒勒索传播；
- 通过网站挂马的方式进行勒索病毒传播；
- 利用已知漏洞进行勒索病毒的传播。
- 使用共享文件夹投毒勒索传播；
- 使用邮件进行勒索病毒传播；

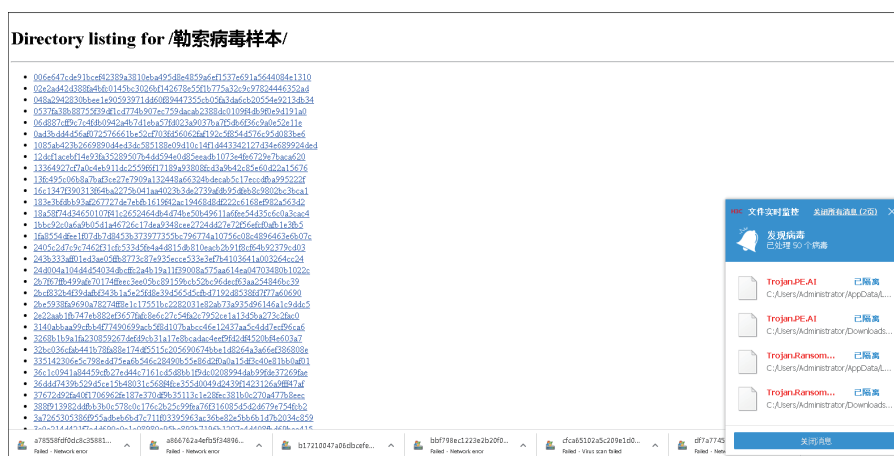


图8 新华三ESM拦截勒索病毒

勒索攻击尝试勒索阶段 – 勒索病毒程序运行测试

- 测试勒索病毒是否可以顺利运行；
- 测试勒索病毒在不同操作系统上的运行情况，如 Windows、Linux 等；
- 测试勒索病毒程序是否可以躲避杀毒软件的检测和拦截；
- 测试勒索病毒是否可以成功加密受害主机的文件。

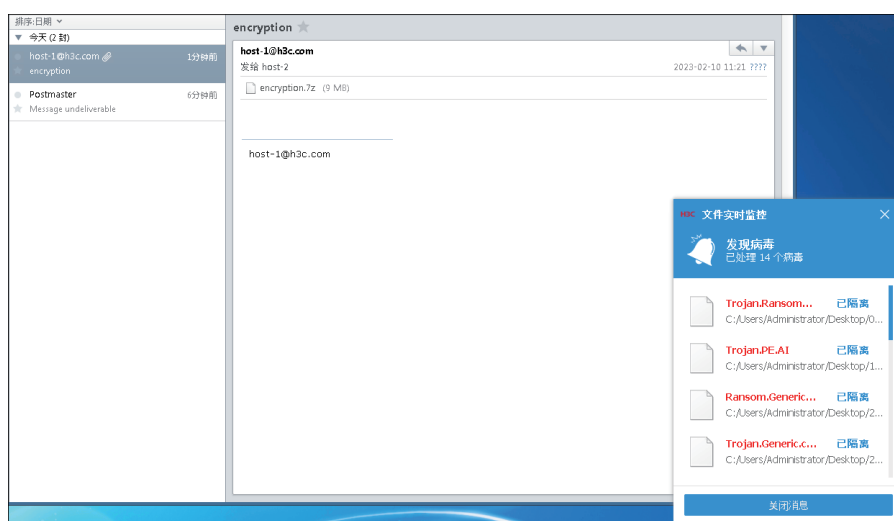


图9 新华三ESM阻止勒索病毒运行

3 新华三勒索防护解决方案效果

在测试环境中连接 100 台计算机，使用勒索病毒程序发起攻击，对比引入新华三勒索防护解决方案前后的测试结果，经过数次实验测试，使用新华三勒索防护解决方案后取得以下成果：

- ❶ 预防性防护：勒索病毒攻击计算机数量从 10 台降低到 1 台，攻击成功率下降了 90%；
- ❷ 防范性防护：终端安全防护系统检测出的勒索病毒数量从 85 个降低到 5 个，拦截率提高了 70%；
- ❸ 恢复性防护：在恢复备份数据时，耗时降低了约 40%，数据完整率提高了 30%；
- ❹ 告警通知：网络管理员在检测到勒索病毒时，可以立即收到告警邮件通知。

随着企业数字化进程的加速，信息安全威胁和风险也快速增加，网络攻击手段越来越多，尤其是近年来勒索攻击事件频发，其破坏力和影响力越来越大，勒索病毒已成为网络安全的最大威胁之一。面对未知的攻击，我们需要制定有效的安全策略，并进行具有前瞻性的安全防护，这样才能在现网中占据先手，百战不殆。



附录 1 缩略语清单

Abbreviations 缩略语	Full spelling 英文全名	Chinese explanation 中文解释
RaaS	Ransomware-as-a-Service	勒索软件即服务
MaaS	Malware-as-a-Service	恶意软件即服务
IAB	Initial Access Brokers	初始访问代理
RDP	Remote Desktop Protocol	远程桌面协议
VPN	Virtual Private Network	虚拟专用网络
DLL	Dynamic Link Library	动态链接库
SID	Security Identifiers	安全标识符
ACL	Access Control List	访问控制列表
ATT&CK	Adversarial Tactics, Techniques and Common Knowledge	攻击矩阵框架
MFA	Multi-Factor Authentication	多因素身份验证
APT	Advanced Persistent Threat	高级可持续威胁攻击
EDR	Endpoint Detection and Response	端点检测与响应的技术
SDN	Software Defined Network	软件定义网络
NFV	Network Function Virtualization	网络功能虚拟化
MSP	Managed Service Provider	管理服务提供商

附录 2 常用入侵方式介绍

凭据泄露

用户名和密码的组合是最常见的访问凭据之一。凭据的泄露可以让入侵者轻松获得内部人员的访问权限，有效地绕过了内部网络的边界检测。凭据泄露带来的风险程度因访问级别而有所不同。普通用户的访问凭据因其访问权限而受到限制，但当属于某个群组中的一项服务或应用程序遭受入侵或破坏时，与其共享凭据的所有应用程序都将暴露在威胁之中。另外，特权级别的访问凭据，不仅为企业内部管理人员所持有，企业的设备之间（包括服务器、网络设备、安全工具等）相互通信也正是依赖着这些高权限的访问凭据，一旦此类凭据被泄露，意味着在企业的整个安全体系架构中，入侵者可随意访问资源而不受限制。为了抵御风险，企业组织渐渐转向 MFA，即在登录请求和认证通过之间引入了一系列额外的用户身份确认步骤，如较为安全的生物识别（即生物数据），例如指纹或高级形式的面部识别，通常网络犯罪分子很难窃取或复制这类数据。

网络钓鱼

网络钓鱼是一种利用社会工程学的网络犯罪策略，常见的网络钓鱼包括钓鱼邮件和钓鱼链接两种类型。利用钓鱼邮件入侵目标网络是勒索攻击的一个重要手段。攻击者通过精心构造看似合理的邮件主题和内容，诱导企业内部人员运行携带恶意代码的文档或可执行程序，从而直接获取办公机器控制权。由于钓鱼邮件是针对具体清晰的目标，所以其内容可以高度定制化。钓鱼邮件的主题各式各样，通常包括福利发放、发票报销、重置账户等，对企业内部人员具有迷惑性和诱导性。此外，利用钓鱼链接也是勒索攻击者进行网络钓鱼的重要形式。勒索犯罪组织引导受害者访问恶意链接，在不知情的情况下下载并浏览携带恶意代码的文件、安装捆绑恶意软件的应用、模仿内部网站骗取工作人员内部访问凭据等。

凭据爆破

远程访问服务为员工异地访问内部资源提供了极大的便利，同时也为犯罪分子带来了可乘之机。使用弱口令爆破账户权限是一个技术门槛相对较低的攻击方式，但其有效性却十分显著。实际上大部分勒索攻击事件，其访问入口都与弱访问凭据有关。如果 VPN、RDP 和 SSH 等远程服务配置了诸如“123456”、“password”等较弱的访问密码，攻击者极有可能使用工具加载包含大量弱口令的字典，通过暴力破解方式获取敏感资源的访问权限。为了对抗弱密码带来的安全风险，很多系统对口令进行了字符和长度限制，一种较为常见的策略是：密码中必须包含字母、数字和特殊字符，且长度不能过低。但是，这种限制策略也并不代表密码就是绝对安全的，一些符合条件的“强密码”如“用户名 + 生日”的组合同样也面临着被爆破的风险。

漏洞利用

一些受害目标由于其网络安全意识淡薄，不注意对资产的漏洞进行及时修复，为勒索攻击者提供了一条高效获取

初始访问权限的攻击路径。不过，值得注意的是，即使是防护严密、注重安全的潜在目标，也可能存在长期被忽略的边缘资产，给组织带来极大的安全风险。特别的，每当重大漏洞被披露时，大型企业或机构由于网络资产数量庞大，难以在短时间内梳理出所有的漏洞资产并进行修复，这种情况带来的时间差，成为勒索攻击者的“黄金时间”。

供应链攻击

供应链攻击是一种新兴的高级攻击入侵方式。攻击者渗透进受害者的第三方软件和服务供应商的系统，在供应商提供给受害者的合法软件中植入恶意代码，受害者出于对供应商的信任不会对其软件进行严格检查，以至于攻击者的恶意代码可以成功运行。在供应链攻击中，攻击者都会在上游或中游供应商中介入，将其恶意活动传播给众多下游用户，与其他攻击方式相比，成功的供应链攻击往往规模更大，影响更深远。

僵尸网络分发

通过 MaaS 模式，僵尸网络为其他网络犯罪组织提供各种攻击服务来获利，如 DDoS、发送垃圾邮件、分发恶意软件、窃取敏感数据等。勒索软件攻击者通过向僵尸网络运营者购买目标受害者主机的访问权限，以此来向受害者分发勒索软件。





主办单位：新华三信息技术有限公司

顾 问：孙松儿 王其勇 梁力文

策 划：徐庆伟 袁江 孙秋男

主 编：王冉莹

本期编委：孙秋男 马文强 秦旭 鲍旭丹 汪林 张宇 霍龙
