VSRC 安全周报（2021-08-17）

0x00 本周漏洞综述

本周需要关注漏洞共 5 个： Pulse Connect Secure 8 月多个安全漏洞；Microsoft 8 月多个安全漏洞；Palo Alto Networks PAN-OS 命令注入漏洞 (CVE-2021-3050)；Apache OFBiz 任意文件上传漏洞 (CVE-2021-37608)；Node.js 远程代码执行漏洞 (CVE-2021-22931)。

本周安全态势共 2 个：硬件 RNG 中的漏洞影响数十亿物联网设备；CVE-2021-28455：针对 Microsoft IIS 和 SQL Server 的新攻击面。

根据以上综述，本周安全威胁为中。

0x01 重要安全漏洞列表

1. Pulse Connect Secure 8 月多个安全漏洞

漏洞概述

2021 年 8 月 2 日，Ivanti 发布了 Pulse Connect Secure 系统软件版本 9.1R12，修复了 Pulse Connect Secure VPN 设备中的多个安全漏洞，成功利用这些漏洞的攻击者可以实现 RCE、XSS 攻击、命令注入或任意文件删除。目前，这些漏洞暂未发现在野利用。

漏洞详情

本次公开的 6 个漏洞都可以被远程利用，其中，CVE-2021-22937 和 CVE-2021-22935 最为严重。这些漏洞的详情如下：

Pulse Connect Secure 远程代码执行漏洞（CVE-2021-22937）

经过身份验证的攻击者可以利用此漏洞在 web 界面上传恶意文件来实现文件写入或执行代码。该漏洞的 CVSSv3 评分为 9.1。

Pulse Connect Secure 任意文件删除漏洞（CVE-2021-22933）

经过身份验证的攻击者可以通过恶意制作的 Web 请求实现任意文件删除。该漏洞的 CVSSv3 评分为 7.6。

Pulse Connect Secure 缓冲区溢出漏洞（CVE-2021-22934）

经过身份验证的攻击者可以通过恶意制作的 Web 请求造成 Pulse Connect Secure 设备缓冲区溢出。该漏洞的 CVSSv3 评分为 8.0。

Pulse Connect Secure 命令注入漏洞（CVE-2021-22935）

经过身份验证的攻击者可以通过未处理的 web 参数执行命令注入。该漏洞的 CVSSv3 评分为 9.1。

Pulse Connect Secure XSS 漏洞（CVE-2021-22936）

攻击者可以通过未处理的 web 参数对经过身份验证的管理员进行跨站脚本攻击。该漏洞的 CVSSv3 评分为 8.2。

Pulse Connect Secure 命令注入漏洞（CVE-2021-22938）

经过身份验证的攻击者可以通过管理员 Web 控制台中未处理的 Web 参数执行命令

注入。该漏洞的 CVSSv3 评分为 7.9。

影响范围

Pulse Connect Secure < 9.1R12

安全建议

目前这些漏洞已经修复。建议受影响的客户及时升级更新至 PCS 9.1R12 版本（已于 2021 年 8 月 2 日发布）。

下载链接：

https://www.ivanti.com/products/connect-secure-vpn?psredirect

参考链接：

https://kb.pulsesecure.net/articles/Pulse_Security_Advisories/SA44858

https://us-cert.cisa.gov/ncas/current-activity/2021/08/06/ivanti-releases-security-update-pulse-connect-secure

https://securityaffairs.co/wordpress/120880/security/pulse-connect-secure-vpn-flaw-2.html?

2. Microsoft 8 月多个安全漏洞

漏洞概述

2021 年 8 月 10 日，Microsoft 发布了 8 月份的安全更新，本次发布的安全更新共计修复了 51 个安全漏洞，其中有 7 个漏洞评级为严重，37 个漏洞评级为高危，其中包括 3 个 0 day 漏洞。

漏洞详情

本次发布的安全更新涉及 Azure、Microsoft Edge、Microsoft Office、Windows MSHTML Platform、Windows Print Spooler Components、Windows Services for NFS ONCRPC XDR Driver、Windows TCP/IP 和 Windows Update 等多个产品和组件。其中，Microsoft Edge (Chromium-based)中的 7 个漏洞评级为未知。在其它 44 个漏洞中，13 个为远程代码执行漏洞，8 个为信息泄露漏洞，2 个为拒绝服务漏洞，4 个为欺骗漏洞。

Microsoft 本次发布的安全更新共计修复了 3 个 0 day 漏洞，其中有 1 个在野外被积极利用。2 个公开披露但未被利用的 0 day 漏洞为：

- CVE-2021-36936 ：Windows Print Spooler 远程代码执行漏洞，其 CVSSv3 评分为 8.8，该漏洞的攻击复杂度和所需权限较低，且无需用户交互，Microsoft 可利用性评估为"更有可能被利用"。
- CVE-2021-36942 ： Windows LSA 欺骗漏洞（与 PetitPotam NTLM 中继攻击有关），其 CVSSv3 评分为 7.5，该漏洞的攻击复杂度低，无需特殊权限和用户交互即可利用，Microsoft 可利用性评估为"更有可能被利用"。Microsoft 表示，该漏洞影响了所有服务器，建议客户在应用安全更新方面应优先考虑域控制器。

1 个已被积极利用的漏洞为：

- CVE-2021-36948 ：Windows Update Medic Service 权限提升漏洞，其 CVSSv3 评分为 7.8，该漏洞的攻击复杂度和所需权限较低，且无需用户交互即可本地利用。目前此漏洞暂未公开披露，但微软安全响应中心（MSRC)和微软威胁情报中心(MSTIC)已经检测到该漏洞被利用。

7 个评级为严重的漏洞为：

- CVE-2021-34530：Windows 图形组件远程代码执行漏洞
- CVE-2021-34480：Scripting Engine 内存损坏漏洞
- CVE-2021-34535：Remote Desktop Client 远程代码执行漏洞
- CVE-2021-34534：Windows MSHTML 平台远程代码执行漏洞
- CVE-2021-36936：Windows Print Spooler 远程代码执行漏洞
- CVE-2021-26432：Windows Services for NFS ONCRPC XDR Driver 远程代码执行漏洞
- CVE-2021-26424：Windows TCP/IP 远程代码执行漏洞

除此之外，Microsoft 还修复了 Windows Print Spooler 远程代码执行漏洞（CVE-2021-34527，也称为 PrintNightmare，以及之前披露的 CVE-2021-34481） 和 PetitPotam NTLM 中继攻击，该攻击使用 MS-EFSRPC API 强迫设备与攻击者控制下的远程中继服务器进行协商，低权限的攻击者可以利用这种攻击来接管域控制器，从而控制整个 Windows 域。

安全建议

目前 Microsoft 已发布相关安全更新，鉴于漏洞的严重性，且部分漏洞正在被积极利用，建议用户尽快修复。

（一） Windows update 更新

自动更新：

Microsoft Update 默认启用，当系统检测到可用更新时，将会自动下载更新并在下一次启动时安装。

手动更新：

1、点击“开始菜单”或按 Windows 快捷键，点击进入“设置”

2、选择“更新和安全”，进入“Windows 更新”（Windows 8、Windows 8.1、Windows Server 2012 以及 Windows Server 2012 R2 可通过控制面板进入“Windows 更新”，具体步骤为“控制面板”->“系统和安全”->“Windows 更新”）

3、选择“检查更新”，等待系统将自动检查并下载可用更新。

4、重启计算机，安装更新系统重新启动后，可通过进入“Windows 更新”->“查看更新历史记录”查看是否成功安装了更新。对于没有成功安装的更新，可以点击该更新名称进入微软官方更新描述链接，点击最新的 SSU 名称并在新链接中点击“Microsoft 更新目录”，然后在新链接中选择适用于目标系统的补丁进行下载并安装。

（二） 手动安装更新

Microsoft 官方下载相应补丁进行更新。

下载链接：

https://msrc.microsoft.com/update-guide/vulnerability

参考链接：

https://msrc.microsoft.com/update-guide/vulnerability

https://msrc.microsoft.com/update-guide/en-US/vulnerability/CVE-2021-36948

https://www.bleepingcomputer.com/news/microsoft/microsoft-august-2021-patch-tuesday-fixes-3-zero-days-44-flaws/

3. Palo Alto Networks PAN-OS 命令注入漏洞（CVE-2021-3050)

漏洞概况

CVE ID	CVE-2021-3050	时 间	2021-08-11
类 型	命令注入	等 级	高危
远程利用	是	影响范围	
攻击复杂度	低	可用性	高
用户交互	无	所需权限	低
PoC/EXP	已公开	在野利用	

漏洞详情

PAN-OS 是 Palo Alto Networks 为其防火墙设备开发的操作系统。

2021 年 8 月 11 日，Palo Alto Networks 发布安全公告，修复了 PAN-OS 中的一个命令注入漏洞（CVE-2021-3050），该漏洞的 CVSSv3 评分为 8.8。

该漏洞存在于 PAN-OS Web 界面中，经过身份验证的远程攻击者能够执行任意系统命令并提升权限，但要利用此漏洞，攻击者需要访问 PAN-OS Web 界面进行身份验证。

Palo Alto Networks 表示暂未发现该漏洞被利用，但此漏洞的 EXP 已公开。

安全建议

目前此漏洞已经修复。鉴于此漏洞为外部发现，且漏洞利用公开可用，建议受影响用户参考下表及时升级更新：

版本	受影响版本	修复版本
PAN-OS 10.1	>= 10.1.0	>= 10.1.2
PAN-OS 10.0	>= 10.0.0	>= 10.0.8
PAN-OS 9.1	>= 9.1.4	>= 9.1.11
PAN-OS 9.0	>= 9.0.10	>= 9.0.15
PAN-OS 8.1	None	8.1.*

注：Prisma Access 防火墙和运行 PAN OS 8.1 版本的防火墙不受此漏洞的影响。

下载链接：

https://www.paloaltonetworks.cn/

参考链接：

https://security.paloaltonetworks.com/CVE-2021-3050

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-3050

https://nvd.nist.gov/vuln/detail/CVE-2021-3050

4. Apache OFBiz 任意文件上传漏洞（CVE-2021-37608）

漏洞概况

CVE ID	CVE-2021-37608	时　间	2021-08-11
类　型	文件上传	等　级	高危
远程利用	是	影响范围	
攻击复杂度		可用性	高
用户交互	无	所需权限	
PoC/EXP	未公开	在野利用	否

漏洞详情

Apache OFBiz 是一款企业流程自动化软件，可以帮助用户实现企业内业务的自动化，它为用户提供了如 ERP 企业资源规划、CRM 客户关系管理等多种管理功能。

2021 年 8 月 11 日，Apache 发布安全公告，公开了 OFBiz 中的一个任意文件上传漏洞（CVE-2021-37608）。由于 Apache OFBiz 存在校验错误，恶意攻击者可以利用此漏洞上传任意文件，并远程执行恶意代码。

影响范围

Apache OFBiz < 17.12.08

安全建议

目前此漏洞已经修复。建议受影响用户及时升级更新到 17.12.08 或更高版本。

下载链接：

http://ofbiz.apache.org/download.html#vulnerabilities

补丁链接：

https://issues.apache.org/jira/browse/OFBIZ-12297

参考链接：

http://mail-archives.apache.org/mod_mbox/www-announce/202108.mbox/%3C40716d3e-150d-10d6-ee27-aca4ae0480fb@apache.org%3E

https://issues.apache.org/jira/browse/OFBIZ-12297

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-37608

5. Node.js 远程代码执行漏洞（CVE-2021-22931）

漏洞概况

CVE ID	CVE-2021-22931	时 间	2021-08-11
类 型	RCE	等 级	高危
远程利用	是	影响范围	
攻击复杂度		可用性	
用户交互		所需权限	
PoC/EXP		在野利用	

Node.js 是一个基于 Chrome V8 引擎的 JavaScript 运行环境，它使用高效、轻量级的事件驱动、非阻塞 I/O 模型。Node.js 中的包管理器 npm，是全球主流的开源库生态系统。

2021 年 8 月 11 日，Node.js 发布了 v16.x、v14.x 和 v12.x 发行版的安全更新，修复了 Node.js 中的一个远程代码执行漏洞（CVE-2021-22931，高危），详情如下：

由于 Node.js DNS 库中的域名服务器返回的主机名缺少输入验证，这可能导致输出错误的主机名（可能导致域名劫持）和使用该库的应用程序中存在注入漏洞，远程攻击者可利用此漏洞执行 XSS 攻击、使应用程序崩溃（拒绝服务）或远程执行恶意代码。

此外，Node.js 本次发布的安全更新还修复了 rejectUnauthorized 参数的不完整验证问题（CVE-2021-22939，低危）；以及一个 Use-after-free 漏洞（CVE-2021-22940，高危），该漏洞是 CVE-2021-22930 的修复不完整导致的，攻击者可以利用内存损坏来改变进程行为。

影响范围

Node.js 12.x < 12.22.5 (LTS)

Node.js 14.x < 14.17.5 (LTS)

Node.js 16.x < 16.6.2 (Current)

安全建议

目前此漏洞已经修复。建议受影响用户及时升级更新到以下版本：

Node.js v12.22.5 (LTS)

Node.js v14.17.5 (LTS)

Node.js v16.6.2 (Current)

下载链接：

https://nodejs.org/en/download/

参考链接：

https://nodejs.org/en/blog/vulnerability/aug-2021-security-releases/

https://nodejs.org/en/blog/release/v12.22.5/

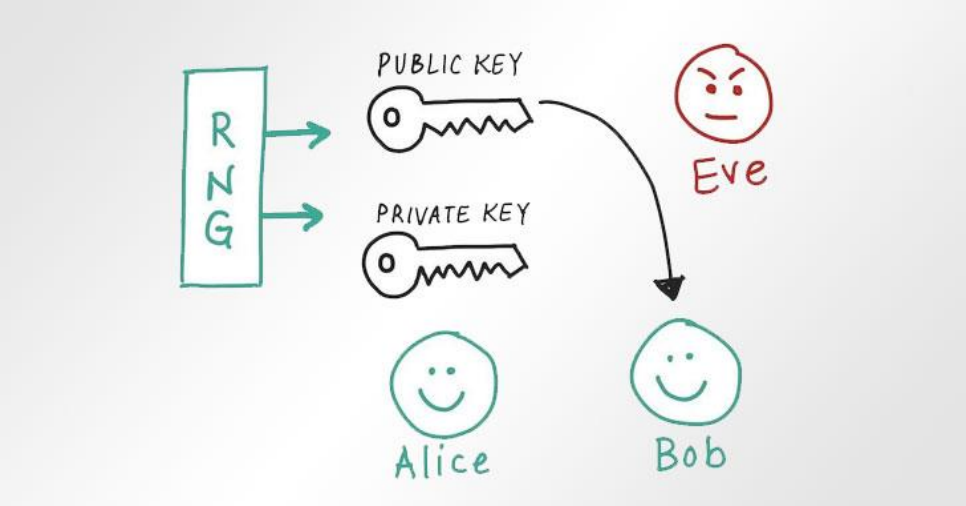
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2021-22931

0x02 本周安全态势

1. 硬件 RNG 中的漏洞影响数十亿物联网设备

风险概述

2021 年 08 月 05 日，研究人员公开披露了全球 350 亿台物联网（IoT）设备中使用的随机数生成器（RNG）中的一个漏洞，该漏洞将导致无法正确生成随机数，从而破坏 IoT 设备的安全性并使其面临被攻击的风险。



攻击详情

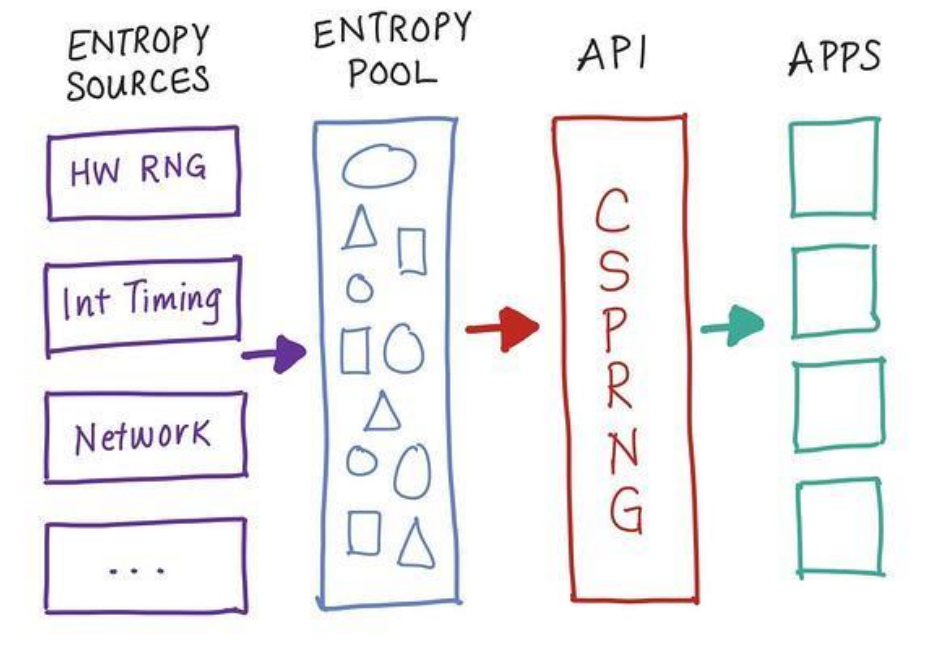
随机数生成（RNG）是一个关键的过程，它支持多种加密应用程序，包括密钥生成、随机数等，它是密码学、访问控制、身份验证等技术的基础。在传统的操作系统中，它来自一个加密安全的伪随机数生成器（CSPRNG），该生成器使用从高质量种子源获得的熵（entropy）。

当涉及到物联网设备时，这是由一个 system-on-a-chip（SoC）提供的，其中包含一个专用的硬件 RNG 外设，称为真随机数发生器（TRNG），用于从物理过程或现象中捕获

随机性。研究人员表示，该外设目前被调用的方式是不正确的，其缺乏对错误代码响应的全面检查，导致产生的随机数非随机且可预测、部分 entropy 及未初始化的内存，甚至包含全 0 的加密密钥。

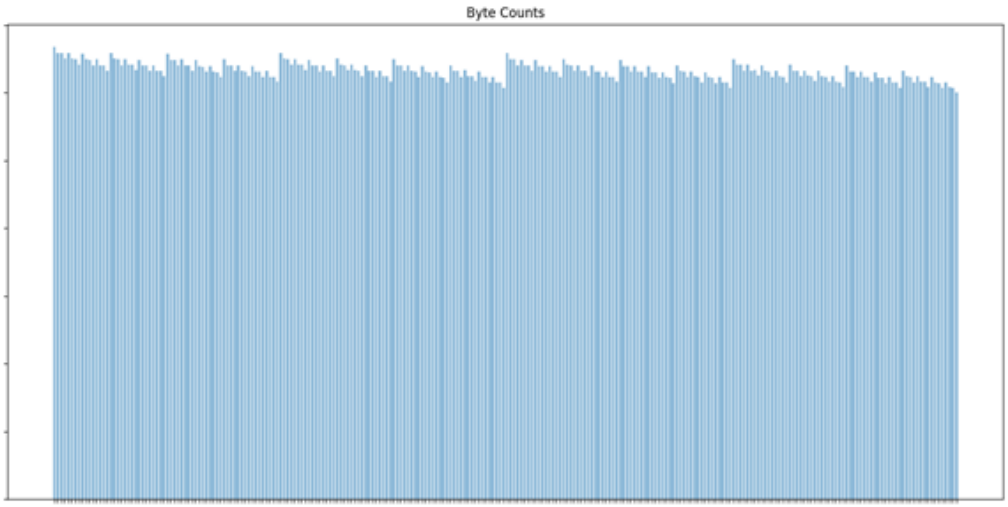
研究人员指出，RNG 外围设备的 HAL 功能可能因各种原因而失败，但迄今为止最常见（和可利用）的是设备的 entropy 已用尽，且其每秒产生的随机位数有限，如果试图在 RNG HAL 函数没有任何随机数可分配的时候调用它，它就会失败并返回一个错误代码。因此，如果设备试图过快地获取较多随机数，将导致调用失败。

该问题是物联网领域独有的，因为它们通常缺乏带有随机性 API 的操作系统（例如，类似 Unix 系统中的/dev/random 或 Windows 中的 BCryptGenRandom）。研究人员强调实施 CSPRNG 子系统，以消除 entropy 源中的任何单点故障，并表示针对此类问题，必须在物联网设备中设计出一个重要而复杂的功能。



研究人员证明，物联网硬件 RNG 外围设备的原始 entropy 质量差异很大。在联发科

7697、北欧半导体 nrf52840 和 STM32-L432KC 测试中，大多数设备未能通过统计分析测试。研究人员表示，这些结果通常取决于单个设备本身，即使是相同品牌和型号的设备也会因为小的制造缺陷产生不同的测试结果。因此，不应孤立地信任从硬件 RNG 中获取的原始 entropy。CSPRNG 子系统提供的密钥拉伸和 entropy 池从根本上是避免执行 IoT RNG 所必需的。



MediaTek 7697 SoC 上每个字节 0 到 255 的频率直方图

风险等级

严重。

- 该漏洞影响了整个物联网行业（全球大约 350 亿台 IoT 设备）。
- 物联网需要使用 CSPRNG 子系统。
- RNG 代码存在安全风险，应该改为使用由较低抽象层提供的 CSPRNG 子系统。
- 切勿直接从 RNG 硬件使用 entropy。

影响范围

全球范围内物联网设备中使用的硬件 RNG，具体受影响厂商和型号暂未发布。

安全建议

研究人员表示，虽然这些问题可以通过软件更新来修复，但理想的解决方案是物联网设备制造商和开发商使用 CSPRNG API，该 API 从一组不同的 entropy 源中播种，并确保代码不会忽略错误条件，或无法阻止当没有更多的 entropy 可用时调用 RNG。

缓存措施

1.针对设备所有者：

及时更新设备，该问题可以通过软件解决，但可能需要一些时间。在此期间，注意不要过于信任物联网小工具。对于需要互联网连接的家庭设备，将它们放在一个只能对外联系的专用网段中。这将有助于遏制任何漏洞，使其不会蔓延到网络的其它部分。

2.针对物联网设备开发商：

选择包括 CSPRNG API 的物联网设备，这些设备的种子来自各种 entropy 源，包括硬件 RNG 的。如果没有 CSPRNG 且没有其它选择，请仔细审查环境中所依赖的库及代码，以确保代码不会从未初始化的内存中读取、忽略硬件 RNG 外设寄存器或错误的代码条件，或者在没有更多 entropy 可用时无法阻塞。仔细考虑对实时情况的影响，在这种情况下，阻塞并不是一个可行的选择。

3.针对设备制造商/物联网操作系统：

在 SDK 中取消或禁用任何直接使用 RNG HAL 的功能。可以使用 CSPRNG API，该

API 使用具有适当硬件 RNG 处理的强大和多样化的 entropy 源进行处理。Linux 内核对/dev/urandom 的实现可以作为一个很好的参考。

参考链接：

https://labs.bishopfox.com/tech-blog/youre-doing-iot-rng

https://thehackernews.com/2021/08/a-critical-random-number-generator-flaw.html

2. CVE-2021-28455：针对 Microsoft IIS 和 SQL Server 的新攻击面

执行摘要

近日，Unit 42 在 2021 年亚洲黑帽大会上分享了对微软互联网信息服务（IIS）和 SQL 服务器的新攻击面的信息。在我们的演讲中，我们介绍了一种以前未公开的技术，即在 SQL 注入或 ad hoc（临时攻击）下对 IIS 和 SQL Server 的远程数据库执行 SQL 查询。我们还讨论了我们在三个月内发现的大约 100 个 Jet 漏洞中挑选的三个典型案例。在本文中，我们介绍了该技术的细节，该技术允许攻击者远程攻击 IIS 和 SQL Server，通过使用微软 Jet 数据库引擎漏洞获得 SYSTEM 权限。

作为响应，微软发布了一个复杂的补丁来缓解这个攻击面。但是，该补丁在默认情况下是关闭的，大多数 Jet 漏洞仍然没有得到修补。我们强烈建议客户主动开启缓解措施，在注

册表中禁用远程表访问，并对这类攻击保持谨慎。除此之外，访问连接引擎（ACE）的攻击面的缓解措施仍然不完善，我们正在与微软合作，为 MS Jet 和 ACE 发布完整的补丁。

攻击面

这个新的攻击面与微软 Jet 数据库引擎中支持的远程数据库访问有关，包括 MS Jet Red（Jet Red 数据库引擎）和 ACE（Access Connectivity Engine）。这是一个实用的功能，但也会带来潜在的安全问题。当被滥用时，该功能允许远程攻击者在其控制的服务器上的数据库文件（完全控制）执行 SQL 查询。一旦合法的数据库文件被替换成错误格式的数据库文件，在上面执行 SQL 查询就会破坏微软 Jet/ACE 的代码前提条件和假设，导致许多 Jet 组件中的漏洞。

这些 Jet 漏洞对安全边界的整体影响和破坏取决于执行 SQL 查询的位置。典型的攻击场景是 SQL 注入和 ad hoc。在这两种情况下，攻击者可以对 IIS 和 SQL Server 中格式错误的数据库执行任何 SQL 查询，由此产生的 Jet 漏洞将影响 IIS 和 SQL Server。具体来

说，用户可以在MS Jet中的表前面添加数据库路径，并在ACE中使用OPENDATASOURCE、OPENROWSET 或 addlinkedserver，从而在表上执行 SQL 查询时分配远程数据库，如图 1 所示。

```
Access
Select * from [ExternalDatabase][table]
```

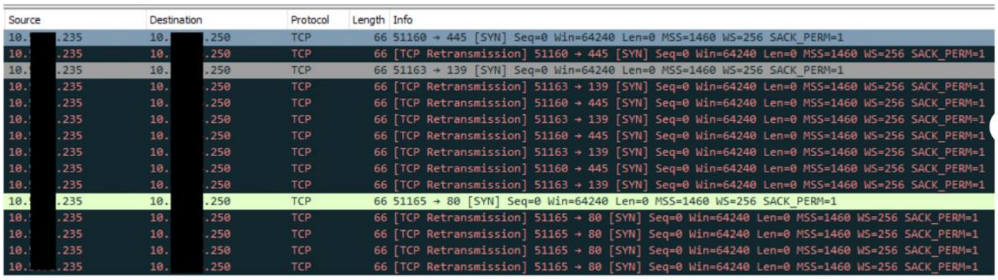
```
SQL Server
SELECT * FROM opendatasource('provider', 'data source=ExternalDatabase ')...[table]

SELECT * FROM OPENROWSET(‘provider', 'Database=ExternalDatabase', 'SELECT * FROM [table]')

EXEC sp_addlinkedserver
@server = ‘ServerName',
@srvproduct = ‘ServerProduct',
@provider = ‘provider',
@datasrc = ‘ExternalDatabase',
@provstr = ‘ProviderString';
```

图 1. Access 和 SQL Server 中的远程数据库访问 SQL。

在 MS Jet 和 ACE 内部，调用 CreateFile 来打开 IIS 和 SQL Server 中的远程数据库文件。鉴于远程数据库的输入路径是一个 UNC 路径，服务器信息块（SMB）和基于 web 的分布式授权和版本管理（WebDAV）都将被用来打开远程数据库，如图 2 所示。

The hidden feature for CreateFile(UNC) in IIS and SQL Server

• CreateFile(UNC) in IIS and SQL Server uses SMB and **WEBDAV**



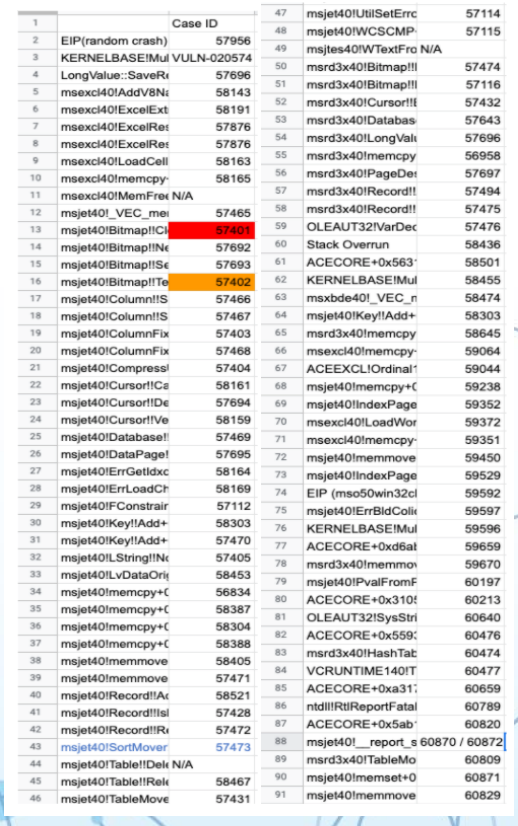
Source	Destination	Protocol	Length	Info
10.[redacted].235	10.[redacted].250	TCP	66	51160 → 445 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	[TCP Retransmission] 51160 → 445 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	51163 → 139 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	[TCP Retransmission] 51163 → 139 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	[TCP Retransmission] 51160 → 445 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	[TCP Retransmission] 51163 → 139 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	[TCP Retransmission] 51160 → 445 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	[TCP Retransmission] 51163 → 139 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	[TCP Retransmission] 51160 → 445 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	[TCP Retransmission] 51163 → 139 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	51165 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	[TCP Retransmission] 51165 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	[TCP Retransmission] 51165 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	[TCP Retransmission] 51165 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
10.[redacted].235	10.[redacted].250	TCP	66	[TCP Retransmission] 51165 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1

图 2. IIS 和 SQL Server 中 CreateFile(UNC)的隐藏功能。

SQL 注入和 ad hoc 只是两种可能的攻击场景，同样，IIS 和 SQL Server 也只是两个潜在的攻击对象。任何在 Windows 上支持 MS Jet 和 ACE 的组件都可能受到攻击，只要该组件允许用户在 MS Jet 和 ACE 的可控数据库上执行任何查询。

IIS 和 SQL Server 中的漏洞

远程访问数据库使攻击者能够用一个格式错误的数据库替换合法的数据库。根据我们的研究，替换数据库问题是发现 MS Jet 和 ACE 漏洞的关键之一。假设 MS Jet 和 ACE 的代码开发和测试可能没有考虑到数据库格式错误的情况，因此我们有了改变 SQL 查询和数据库文件以进行测试的想法。通过这种模糊测试策略，我们在 MS Jet 和 ACE 中发现了大约 100 个漏洞，如图 3 所示。其中大部分可用于在 SQL 注入和 ad hoc 下攻击 IIS 和 SQL Server。



	Case ID
EIP(random crash)	57956
KERNELBASE!Mul	VULN-020574
LongValue::SaveR	57696
msexcl40!AddV8N	58143
msexcl40!ExcelExt	58191
msexcl40!ExcelRe	57876
msexcl40!ExcelRe	57876
msexcl40!LoadCell	58163
msexcl40!memcpy	58165
msexcl40!MemFree	N/A
msjet40!_VEC_me	57465
msjet40!Bitmap!!Cl	57401
msjet40!Bitmap!!Ne	57692
msjet40!Bitmap!!Se	57693
msjet40!Bitmap!!Te	57402
msjet40!Column!!S	57466
msjet40!Column!!S	57467
msjet40!ColumnFix	57403
msjet40!ColumnFix	57468
msjet40!Compressl	57404
msjet40!Cursor!!Ca	58161
msjet40!Cursor!!De	57694
msjet40!Cursor!!Ve	58159
msjet40!Database!!	57469
msjet40!DataPage!	57695
msjet40!ErrGetIdxc	58164
msjet40!ErrLoadCh	58169
msjet40!FConstrair	57112
msjet40!Key!!Add+	58303
msjet40!Key!!Add+	57470
msjet40!LString!!Ne	57405
msjet40!LvDataOri	58453
msjet40!memcpy+(	56834
msjet40!memcpy+(	58387
msjet40!memcpy+(	58304
msjet40!memcpy+(	58388
msjet40!memmove	58405
msjet40!memmove	57471
msjet40!Record!!Ac	58521
msjet40!Record!!Isl	57428
msjet40!Record!!Re	57472
msjet40!SortMover	57473
msjet40!Table!!Del	N/A
msjet40!Table!!Rel	58467
msjet40!TableMove	57431

msjet40!UtilSetErrc	57114
msjet40!WCSCMP	57115
msjtes40!WTextFro	N/A
msrd3x40!Bitmap!!I	57474
msrd3x40!Bitmap!!I	57116
msrd3x40!Cursor!!E	57432
msrd3x40!Databas	57643
msrd3x40!LongVal	57696
msrd3x40!memcpy	56958
msrd3x40!PageDe	57697
msrd3x40!Record!!	57494
msrd3x40!Record!!	57475
OLEAUT32!VarDec	57476
Stack Overrun	58436
ACECORE+0x563	58501
KERNELBASE!Mul	58455
msxbde40!_VEC_r	58474
msjet40!Key!!Add+	58303
msrd3x40!memcpy	58645
msexcl40!memcpy	59064
ACEEXCL!Ordinal1	59044
msjet40!memcpy+(	59238
msjet40!IndexPage	59352
msexcl40!LoadWor	59372
msexcl40!memcpy	59351
msjet40!memmove	59450
msjet40!IndexPage	59529
EIP (mso50win32cl	59592
msjet40!ErrBldColl	59597
KERNELBASE!Mul	59596
ACECORE+0xd6a	59659
msrd3x40!memmo	59670
msjet40!PvalFromF	60197
ACECORE+0x310	60213
OLEAUT32!SysStri	60640
ACECORE+0x559	60476
msrd3x40!HashTab	60474
VCRUNTIME140!T	60477
ACECORE+0xa31	60659
ntdll!RtlReportFatal	60789
ACECORE+0x5ab	60820
msjet40!__report_s	60870 / 60872
msrd3x40!TableMo	60809
msjet40!memset+0	60871
msjet40!memmove	60829

图 3. 约 100 个 MS Jet 漏洞。

在我们的演讲中，我们证明了数据库文件中仅仅一个字节的变化就可以导致 MS Jet 漏洞，如图 4 所示。

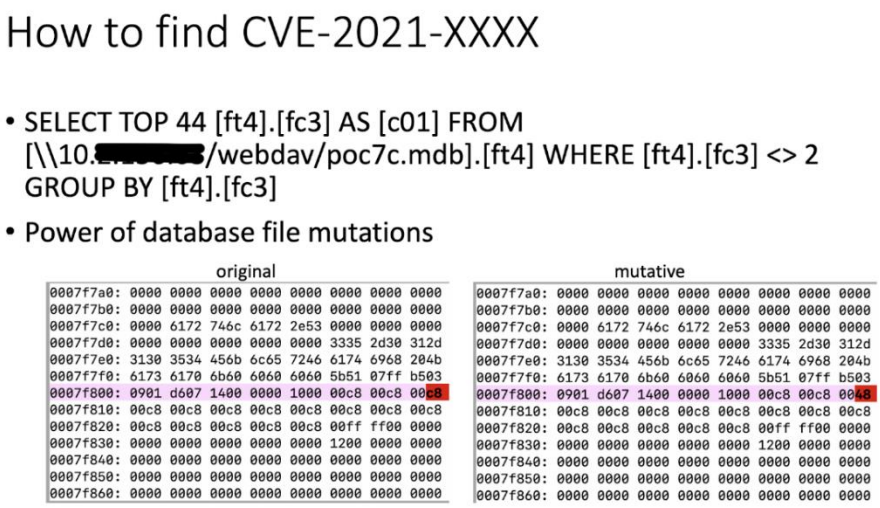


图 4. 数据库中单个字节的更改。

微软补丁

随着 2021 年 5 月补丁日发布的 Windows 更新，微软将 CVE-2021-28455 分配给我们的发现，并修复了我们报告的新攻击面。该补丁为用户引入了一个选项，以禁用 MS Jet 组件和 ACE 组件中的远程数据库访问，但它没有修复每个 JET 漏洞，而是在使用 MS Jet 的多个应用程序（如 IIS 和 Access）中缓解了我们报告中披露的整个攻击面。

正如我们在图 5 中看到的，在 rgtib 结构中的偏移量 904h 有一个新字段（由 ebx 寄存器表示），用于远程访问数据库。在_ltibAllocate 函数中，它被默认设置为 1，这意味着它被默认启用。

```
Address          Function                        Instruction
.text:10003C88                                   dd 7010904h, 0E030B05h, 0D06000Ah
.text:10003E04                                   dd 0D08000Fh, 7090C03h, 1060A05h, 70B000Dh, 0A010904h
.text:10025D30   ErrReadRegistry                 add    eax, 904h
.text:10025DED   ErrReadRegistry                 add    eax, 904h
.text:10037C4D   _ItibAllocate@0                 mov    dword ptr [ebx+eax+904h], 1
.text:10080490   _ErrIsamCopyRecords@32  var_904 = byte ptr -904h
.text:1009BF22   _ErrGetOutputDatabaseId@12      cmp    [ecx+eax+904h], edi
.text:100C27F1   _ErrQEMCompileQuery@52          cmp    dword ptr [ecx+eax+904h], 0
```

图 5. 为 rgtib 结构中的新字段 AllowQueryRemoteTables 设置的默认值。

然后在 ErrReadRegistry 函数中调用 UtilRegQueryValue2 函数，以获得 HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Jet\4.0\engines 注册表项下的注册表键 - AllowQueryRemoteTables 的值。之后，它将该值存储到图 6 所示的 rgtib 结构的 AllowQueryRemoteTables 字段中。

```
int __stdcall ErrReadRegistry(int a1, int a2)
{
  HKEY v2; // edi
  HKEY v4; // eax
  HKEY v5; // edi
  struct HKEY__ phkResult; // [esp+B8h] [ebp-20h] BYREF
  HKEY hKey; // [esp+BCh] [ebp-1Ch] BYREF
  char Data[20]; // [esp+C0h] [ebp-18h] BYREF

  if ( !a2 )
    AssignDefaults();
  if ( !UtilRegOpenKey(HKEY_LOCAL_MACHINE, "SOFTWARE\\Microsoft\\Jet\\4.0\\Engines", &hKey) )
  {
    v2 = hKey;
    if ( hKey )
    {
      phkResult.unused = 20;
      if ( !UtilRegQueryValueW(hKey, 0, L"JetDisabled", (LPBYTE)Data, &phkResult) && !ascii_wcsicmp(L"YES", Data) )
        return -5500;
      if ( !a2 )
        ReadRegTree(v2);
      phkResult.unused = 518;
      UtilRegQueryValueW(v2, 0, L"SystemDB", (LPBYTE)rgtib + 2312 * a1, &phkResult);
      phkResult.unused = 4;
      UtilRegQueryValue2(v2, 0, "CompactByPKey", (LPBYTE)rgtib + 2312 * a1 + 2284, &phkResult);
      phkResult.unused = 4;
      UtilRegQueryValue2(v2, 0, "PrevFormatCompactWithUNICODECompression", (LPBYTE)rgtib + 2312 * a1 + 2304, &phkResult);
      phkResult.unused = 4;
      UtilRegQueryValue2(v2, 0, "AllowQueryRemoteTables", (LPBYTE)rgtib + 2312 * a1 + 2308, &phkResult);// here
      UtilRegCloseKey(v2);
    }
  }
  v4 = (HKEY)OpenShadowTree(a1);
  v5 = v4;
  if ( v4 )
  {
    if ( !a2 )
      ReadRegTree(v4);
    phkResult.unused = 518;
    UtilRegQueryValueW(v5, 0, L"SystemDB", (LPBYTE)rgtib + 2312 * a1, &phkResult);
    phkResult.unused = 4;
    UtilRegQueryValue2(v5, 0, "CompactByPKey", (LPBYTE)rgtib + 2312 * a1 + 2284, &phkResult);
    phkResult.unused = 4;
    UtilRegQueryValue2(v5, 0, "PrevFormatCompactWithUNICODECompression", (LPBYTE)rgtib + 2312 * a1 + 2304, &phkResult);
    phkResult.unused = 4;
    UtilRegQueryValue2(v5, 0, "AllowQueryRemoteTables", (LPBYTE)rgtib + 2312 * a1 + 2308, &phkResult);
    UtilRegCloseKey(v5);
00025138 ErrReadRegistry:29 (10025D38)
```

图 6. 从 UtilRegQueryValue2 函数的注册表中获取 AllowQueryRemoteTables 字段。

之后，两个函数（_ErrGetOutputDatabaseId 和_ErrQEMCompileQuery）检查 rgtib 结构（由 ecx 寄存器表示）中的 AllowQueryRemoteTables 字段，如图 7 所示。

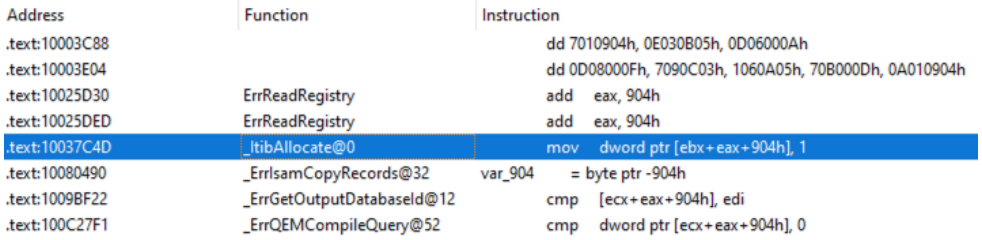


图 7. 检查 rgtib 结构中的 AllowQueryRemoteTables 字段。

从图 8 中我们可以看到，如果 AllowQueryRemoteTables 字段被设置为 0，那么 _ErrGetOutputDatabaseId 函数将返回一个错误，并且不会调用 ErrTryOpenDatabase 函数来打开数据库文件，无论该数据库文件是远程还是本地。这有效地缓解了远程数据库访问的攻击面。

```
  if ( !*((_DWORD *)rgtib + 578 * v3 + 577) )   // if AllowQueryRemoteTables = 0, return
    return 0xFFFFFC17;
  v7 = (void *)*((_DWORD *)lpAddress + 43);
  v12 = v5;
  result = ErrReconcileDbConn(*((_DWORD *)lpAddress + 10), (int)&v12, v7, v15, 261);
  if ( result >= 0 )
  {
    if ( *v4 || (v8 = v12) == 0 )
    {
LABEL_16:
      ErrDispGetDatabaseInfo(*((_DWORD *)lpAddress + 10), *((_DWORD *)lpAddress + 12), &v14, 4, 20);
      result = ErrGetIsamType(*((_DWORD *)lpAddress + 11), v7, &v13);
      if ( result >= 0 )
      {
        if ( v13 == 0xFFFF
          || v13 == 65534
          || (result = ErrTryOpenDatabase(
                         *((_DWORD *)lpAddress + 10),
                         v12,
                         (STRSAFE_LPCWSTR)v7,
                         a3,
                         (v14 != 0 ? 0x100 : 0) | 2,
                         1),
              result < 0) )
        {
          result = ErrTryOpenDatabase(
                     *((_DWORD *)lpAddress + 10),
                     v12,
                     (STRSAFE_LPCWSTR)v7,
                     a3,
                     v14 != 0 ? 0x100 : 0,
                     1);
0009B322 _ErrGetOutputDatabaseId@12:29 (1009BF22)
```

图 8. _ErrGetOutputDatabaseId 函数中的 AllowQueryRemoteTables 字段检查。

然而，这个功能在默认情况下是不开启的。要禁用远程数据库访问，用户需要在相应的注册表中添加一个名为 AllowQueryRemoteTables 的注册表，如微软文档（KB5002984:

配置 Jet Red Database Engine 和 Access Connectivity Engine 以阻止对远程数据库的访问）所述，并将 dword 值设为 0。

CVE-2021-28455 的影响范围

目前此漏洞已在微软 5 月补丁日中发布的安全更新中修复。根据微软的安全公告，该漏洞的影响范围如下所示：

Windows Server 2012 R2 (Server Core installation)

Windows Server 2012 R2

Windows Server 2012 (Server Core installation)

Windows Server 2012

Windows Server 2008 R2 for x64-based Systems Service Pack 1 (Server Core installation)

Windows Server 2008 R2 for x64-based Systems Service Pack 1

Windows Server 2008 for x64-based Systems Service Pack 2 (Server Core installation)

Windows Server 2008 for x64-based Systems Service Pack 2

Windows Server 2008 for 32-bit Systems Service Pack 2 (Server Core installation)

Windows Server 2008 for 32-bit Systems Service Pack 2

Windows RT 8.1

Windows 8.1 for x64-based systems

Windows 8.1 for 32-bit systems

Windows 7 for x64-based Systems Service Pack 1

Windows 7 for 32-bit Systems Service Pack 1

Windows Server 2016 (Server Core installation)

Windows Server 2016

Windows 10 Version 1607 for x64-based Systems

Windows 10 Version 1607 for 32-bit Systems

Windows 10 for x64-based Systems

Windows 10 for 32-bit Systems

Windows Server, version 20H2 (Server Core Installation)

Windows 10 Version 20H2 for ARM64-based Systems

Windows 10 Version 20H2 for 32-bit Systems

Windows 10 Version 20H2 for x64-based Systems

Windows Server, version 2004 (Server Core installation)

Windows 10 Version 2004 for x64-based Systems

Windows 10 Version 2004 for ARM64-based Systems

Windows 10 Version 2004 for 32-bit Systems

Windows Server, version 1909 (Server Core installation)

Windows 10 Version 1909 for ARM64-based Systems

Windows 10 Version 1909 for x64-based Systems

Windows 10 Version 1909 for 32-bit Systems

Windows Server 2019 (Server Core installation)

Windows Server 2019

Windows 10 Version 1809 for ARM64-based Systems

Windows 10 Version 1809 for x64-based Systems

Windows 10 Version 1809 for 32-bit Systems

Windows 10 Version 1803 for ARM64-based Systems

Windows 10 Version 1803 for x64-based Systems

Windows 10 Version 1803 for 32-bit Systems

Microsoft Office 2013 Service Pack 1 (64-bit editions)

Microsoft Office 2013 Service Pack 1 (32-bit editions)

Microsoft Office 2013 RT Service Pack 1

Microsoft Access 2016 (64-bit edition)

Microsoft Access 2016 (32-bit edition)

Microsoft Office 2016 (64-bit edition)

Microsoft Office 2016 (32-bit edition)

Microsoft 365 Apps for Enterprise for 64-bit Systems

Microsoft 365 Apps for Enterprise for 32-bit Systems

Microsoft Office 2019 for 64-bit editions

Microsoft Office 2019 for 32-bit editions

Microsoft Access 2013 Service Pack 1 (64-bit editions)

Microsoft Access 2013 Service Pack 1 (32-bit editions)

结论

IIS 和 SQL Server 是微软生态系统中的基本组件，它们已被广泛部署在许多生产系统和服务中。微软的 Jet 数据库引擎，包括 MS Jet 和 ACE，已经有 20 多年的历史了，绝大部分的 Jet 模块都被发现是容易被利用的，因为漏洞缓解措施有限。远程数据库访问功能将 Jet 的漏洞与 IIS 和 SQL Server 组件连接起来，从而将它们的安全性降到与 Jet 数据库引擎相同的水平。攻击者可能会利用这种缺陷攻击 IIS 和 SQL Server，并通过 SQL 注入远程获得 SYSTEM 权限。

建议客户遵循微软的指导，禁用远程数据库访问，以缓解这种严重的攻击面，并有助于防止攻击者利用 Jet 漏洞来破坏 IIS 和 SQL Server。

通用安全建议

- 及时修复漏洞，定期更新软件、程序和应用程序，确保应用程序是最新的，以保护系统免受漏洞利用。
- 加强系统和网络的访问控制，修改防火墙策略，关闭非必要的应用端口或服务，减少将危险服务（如 SSH、RDP 或数据库等）暴露到公网，以减少攻击面。
- 预防 0day 漏洞和恶意软件，安全产品实时更新最新规则或相关防护指标。
- 加强系统用户和权限管理，启用多因素认证机制和最小权限原则，用户和软件权限应保持在最低限度。
- 启用强密码策略并设置为定期修改。
- 使用最新、全面的威胁情报信息，监控网络和安全事件，以快速响应攻击。

原文链接：

https://unit42.paloaltonetworks.com/iis-and-sql-server/