作者是Konstantin Taranov,16 February, 2022，arXiv

这篇文章主要是写NeVerMore发现了RDMA协议中的新漏洞，揭示了对启用RDMA的应用程序和NVMe-oF协议的几个攻击向量，表明当前的NVMe-oF协议的安全机制并不能解决使用RDMA带来的安全漏洞。使用四种针对RDMA协议的攻击和七种针对NVMe-oF协议的攻击，并在NVMe-oF的两个最流行的实现上进行了验证，使用两种威胁模型，分别为TLU和TRA，这两个设计缺陷的组合使数据包注入到任何本地RDMA端点而无需任何管理特权，在本地无特权攻击者（TLU）的情况下会受到数据包注入攻击。以及管理权限的攻击者如何绕过NVMe o F协议和操作系统的现有安全机制，将数据写入远程NVMe设备。提出的解决措施是使用在相应的NVMe of消息中发送的附加MAC来验证此数据。对于NVMe oF写入，通过RDMA写入数据计算的MAC应该是NVMe请求的一部分。对于NVMe oF读取，通过RDMA读取数据计算的MAC应在NVMe响应中。

**Abstract**

本文介绍了InfiniBand架构(一种流行的RDMA标准)和nvme -over- fabric(一种杰出的工业分类存储协议)的安全性分析，nvme -over- fabric利用RDMA协议实现对远程固态设备的低延迟和高带宽访问。

我们将**展示无特权用户如何将数据包注入到本地网络控制器上创建的任何RDMA连接中，绕过操作系统及其内核的安全机制，以及如何使用注入获得对NVMe-oF设备的未经授权的块访问。**

**Introduction**

资源分解正在成为数据中心设计中的一个重要工具，它将现有的**单块服务器拆分为多个统一的单资源池，通过高速互连进行通信**。

这种方法提高了硬件资源的利用率和部署灵活性，因为计算节点和存储节点都可以使用不同类型的服务器硬件，并且可以独立进行尺寸划分。

尽管有这些优点，但分解开辟了新的攻击载体，因为它通常在低延迟、高带宽但不可信的网络上实现。

NVMe over fabric (NVMe- of)协议是一种领先的存储分解协议

**NVMe- of结合了两种最新的高性能技术:NVM Express (NVMe)[16]和远程直接内存访问(RDMA)。**

**NVMe- of采用RDMA连接发送NVMe请求，这些请求通常通过PCIe发送到本地固态驱动器(SSD)，网络结构具有几微秒的超低延迟。**

尽管**RDMA网络能够实现低延迟和高带宽**，但它们已被证明存在安全弱点。

主要原因是缺乏安全通道和向远程方公开内存访问。

尽管存在这些风险，但部署NVMe-oF的安全影响和危险在很大程度上仍未得到研究。

我们展示了任何试图利用RDMA的系统都会打开一个攻击面，允许本地用户绕过操作系统及其内核的安全机制。

重要的是，我们展示了任何无特权用户都可以将包注入到本地网络控制器上创建的RDMA连接中，即使它们是在内核空间中创建的。

因此，**任何从内核空间使用RDMA的系统都会打开一个攻击面，允许本地用户操作启用了RDMA的内核模块，例如NVMe-oF块设备。**

此外，我们还展示了对手如何通过RDMA连接管理器、RDMA资源共享和RDMA拥塞机制中的漏洞破坏、阻止和减慢RDMA连接，从而进行拒绝服务攻击。

鉴于已经发现的RDMA漏洞，我们随后分析了NVMe-oF协议的安全机制，以及它在存储性能开发工具包(SPDK)和Linux内核中的实现

我们的分析NeVerMore涵盖了最近提出的对NVMe-oF的安全扩展，其中包括带内认证和使用IPsec的安全通道。展示了与RDMA网络使用相关的NVMe-oF协议的设计和实现中的多个漏洞和缺陷。

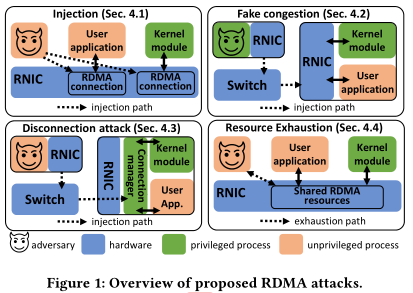
特别地，我们展示了非特权用户如何获得对NVMe-oF设备的块访问权限，以及如何操作已存储和加载的数据。

因此，尽管存储行业在为NVMe-oF实现这些安全机制上花费了大量精力，但用于RDMA网络的InfiniBand架构的底层实现和设计中的漏洞可能会削弱这些努力。

**InfiniBand （IB）** 是一种用于高性能计算的计算机网络通信标准，具有非常高的吞吐量和非常低的延迟。它用于计算机之间和计算机内部的数据互连。InfiniBand 还用作服务器和存储系统之间的直接互连或交换互连，以及存储系统之间的互连。

因此，我们认为应该强调使用RDMA所继承的安全问题，以进一步影响NVMe-oF安全机制的发展。

总之，我们提出了四类针对RDMA协议的攻击，它们支持针对NVMe- of协议的七种不同的攻击向量，允许无特权的对手操纵远程NVMe设备以及任何NVMe- of客户机的内存状态。



最后，我们讨论了针对每个发现的漏洞的潜在缓解技术，包括针对NVMe-oF请求的应用程序级源和数据身份验证的建议

* 1. **Overview of implemented attack classe**

非特权用户可以将数据包注入到本地网络控制器上创建的任何RDMA连接中，包括由特权内核模块创建的连接

因此，任何使用RDMA的内核应用程序都会打开一个攻击面，允许攻击者从用户空间通过向它们的连接注入RDMA请求来操纵内核级应用程序。

对于NVMe- of，可以绕过操作系统和文件系统的安全机制，在块级别直接操作NVMe磁盘，而无需管理特权。

**Fake congestion.**

特权用户可以伪造RDMA协议的拥塞通知包，迫使远程网络控制器减慢速度。

因此，攻击者可以破坏任何可达的支持rdma的应用程序的正常操作。

对于NVMe- of，攻击者可以显著降低对远程NVMe磁盘的访问性能。

**Disconnection attack.**

一个没有特权的用户可以伪造RDMA连接管理器的数据包，允许它断开网络中的任何RDMA连接，包括由本地和远程内核模块创建的连接。

因此，任何用户都可以破坏任何支持rdma的应用程序的正常操作。

对于NVMe- of，攻击者可以暂时断开网络连接的NVMe磁盘，阻止操作系统访问这些磁盘。

**Resource exhaustion.**

无特权用户可以阻塞本地RDMA资源，阻止本地应用程序打开RDMA连接。任何本地用户都可以破坏任何支持rdma的本地应用程序的正常操作。

对于NVMe- of，攻击者可以通过之前的攻击将网络连接的NVMe磁盘断开，然后禁止其重新连接，从而使操作系统在较长时间内无法访问存储。

**2 BACKGROUND ON NVME OVER RDMA**

NVMe-oF协议使用ib架构中指定的RDMA网络协议，包括本机ib、RoCEv1和RoCEv2协议。

不管底层的RDMA协议是什么，开发人员都可以通过RDMA谓词用户空间库来使用RDMA通信。

每个可靠的RDMA连接由两个称为队列对的端点处理程序组成，它们允许应用程序发出RDMA通信请求。

用户通过rdma的队列对处理程序直接向支持rdma的网络控制器(RNIC)提交异步通信请求，绕过操作系统并减少CPU开销。

RNIC使用集成的DMA模块执行所有数据访问，该模块可以直接写入和读取本地内存。

一旦RNIC完成了通信请求的执行，它将生成一个完成事件，该事件被写入用户空间完成队列，指示请求的完成。

**NVMe**

NVMe协议是一种存储协议，旨在利用快速PCIe接口将请求直接发送到高性能存储媒体。

**NVMe协议类似于RDMA协议，其中通信端点是主机CPU和存储设备**。

应用程序通过直接向使用专用DMA硬件访问内存的存储设备发送异步工作请求来使用NVMe。NVMe提供微秒级的访问延迟和每秒数百万的I/O操作，显著优于传统存储协议，如SAS和SATA。

**NVMe-oF。**

NVMe- of协议定义了一种支持通过网络进行NVMe块存储的通用架构。

作为一种网络协议，NVME-oF利用RDMA将数据移动转移到支持RDMA的网卡上，因此减少了处理远程I/O请求所涉及的处理开销。

NVMe- of协议将NVMe命令和响应封装到fabric-neutral中，并将其传递到RDMA传输。

fabric-neutral表示从NVMe- of客户端传输到具有本地附加NVMe设备(即通过PCIe)的远程NVMe- of目标的NVMe单元。

**RDMA connection manager.**

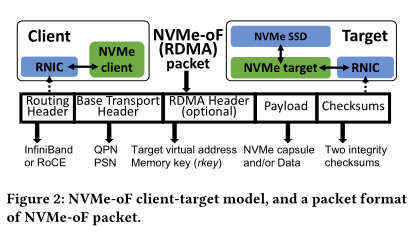
NVMe- of客户端和远程NVMe- of目标端需要建立RDMA连接，以确保NVMe胶囊的可靠交付。

使用RDMA连接管理器、一个用户空间库和一个内核模块建立连接，其目的是建立和管理类似于TCP套接字的RDMA连接。

否则，RDMA应用程序将不得不在TCP连接上实现连接参数的交换，并使用RDMA谓词库手动创建RDMA端点，这很容易出错，并使RDMA连接的维护变得复杂。

**End-to-end overview.**

图2说明了NVMe-oF的一个部署示例。



存储节点(目标)运行一个NVMe- of目标应用程序，该应用程序有权通过NVMe协议访问本地NVMe设备，并侦听来自其他集群节点的RDMA连接。计算节点(客户机)运行一个使用RDMA连接管理器连接到目标的NVMe-oF客户机应用程序。客户机可以是本地挂载远程磁盘的内核模块，也可以是允许连接到远程磁盘的特权应用程序。

目前，NVMe-oF目标可以使用IP过滤器或带内认证协议，提供双向挑战-响应认证，以防止未经授权的连接.

每个RDMA包由**一个路由报头(包含网口标识符)、一个传输报头(包含与端点相关的信息，如连接标识符(QPN)和包序列号(PSN))、包有效载荷和两个完整性校验和组成**。

RNIC处理RDMA报文的顺序如下。

首先，它检查校验和以及目标连接端点是否存在，RNIC检查打开连接的本地表是否包含路由信息和目标连接标识符(QPN)的连接。RNIC不仅匹配QPN，还匹配路由信息(例如，RoCEv2的发送方的IP地址)。如果存在连接，RNIC将报文序列号(PSN)与本地报文计数器进行比较。如果它们匹配，RNIC将包含胶囊的有效载荷复制到目标应用程序在预发布的RDMA接收请求中指定的内存位置。然后应用程序处理胶囊并将响应发送给客户机。

**NVMe-oF协议中的数据通信是通过RDMA写和读请求进行的，该请求还包含一个RDMA头，其中包含一个内存键(rkey)和一个目标虚拟地址。**

为了限制未经授权的内存访问，启用rdmaa的应用程序需要显式地为RDMA访问注册内存，以生成它的内存密匙，该密匙应该包含在发起者的请求中。

为了处理单侧RDMA包，RNIC通过检查目标连接标识符(QPN)访问指定内存的权限来验证RDMA头，给定所提供的内存键。

在这些检查之后，RNIC使用它的DMA引擎访问请求的虚拟地址。

1. **Threat Model**

我们考虑两种威胁模型，分别称为TLU和TRA在这两个模型中，我们都考虑连接位于不同机器上的两个端点的受害连接。连接可以是任何基于InfiniBand架构的RDMA连接，例如RoCEv1、RoCEv2和InfiniBand。在NVMe-oF存储分解的情况下，在NVMe-oF客户端和NVMe-oF目标端之间建立连接。此外，针对RDMA连接管理器的攻击假定连接是使用RDMA连接管理器建立的。对于所有其他攻击，可以使用RDMA连接管理器或通过RDMA谓词库使用本机接口建立连接。

注意，NVMe-oF总是使用RDMA连接管理器来建立连接。

**威胁模型本地用户(TLU)。**

我们考虑位于受害者连接的一个端点上的对手(即，它与NVMe-oF目标或客户机共存)。攻击者是一个无特权的用户，并被假定使用合法的方法获得了对计算机的访问权限。我们假设攻击者与NVMe-oF实体共享相同的物理RNIC，并且两者都可以使用它进行通信。我们假设攻击者和NVMe-oF实体没有通过RNIC虚拟化分离。

TLU模型在使用RDMA和NVMe-oF加速工作负载的私有集群中非常普遍。

NVMe-oF实体之间的链路可以使用基于RoCE[23]的IPsec来保护，RoCE[23]是一种将RoCEv2报文封装到IPsec报文中的协议，并由本地管理员配置相应的IPsec策略。

NVMe-oF实体和攻击者都不能访问用于IPsec安全策略的加密密钥，但可以使用安全链接与远程实体通信。

因此，在没有使用IPsec保护机器之间的链路的情况下，所有在威胁模型TLU假设下实现的攻击也可以执行。

**威胁模型TRA (Remote Administrator)。**

我们假设攻击者位于与NVMe-oF连接端点不同的机器上。攻击者在其机器上拥有管理特权，这使其能够伪造并将消息注入网络。这些特权允许攻击者改变网络接口的配置(例如，它的IP地址)。该模型是在ReDMArk中为包注入提出的，称为T2模型。我们假设在NVMe-oF实体之间的链路上没有启用通过RoCE的IPsec。这允许攻击者向受害连接的端点发送伪造数据包。因此，通过在两个受害端点使用的路径上启用IPsec通道，可以减轻在该模型中实现的所有攻击。

**Adversary constraints.**

我们假设NVMe-oF目标只接受来自良性NVMe-oF客户端的连接。对手不能直接与NVMe-oF目标建立连接。现有的NVMe目标应用程序可以为此使用IP过滤器。目前正在开发的NVMe-oF的安全扩展假定可以防止使用挑战握手身份验证协议的所有未经授权的连接。由于无法窃听现有的连接，对手受到了进一步的限制。

例如，攻击者不需要从现有连接中嗅探RDMA连接参数(例如，连接标识符或包序列号)，而是需要猜测这些参数，以便成功模拟NVMe-oF端点之一。

使用ReDMArk中发现的攻击可以方便地猜测这些参数。

**4 NVME-OF中RDMA协议的安全性分析**

在对NVMeoF协议及其实现的安全性分析中，发现了所有针对RDMA的攻击，并成功地进行了验证。对NVMe-oF协议的安全性造成威胁。

例如，允许将数据包注入到RDMA连接的攻击会使注入到NVMe-oF连接中。

尽管如此，所有的攻击都可以成功地应用于其他支持rdma的系统和协议。

针对支持RoCE和InfiniBand协议的各种网络控制器，分析了发现的RDMA漏洞和攻击。

**4.1报文注入**

4.1报文注入首先，我们分析了允许报文注入到基于ib协议的攻击，包括RoCE和原生ib。与现有的注入工具相比，我们的包注入攻击在没有管理特权的情况下是可行的，假设攻击者位于与受害者相同的机器上(参见表1)。

在传统TCP套接字的情况下，操作系统通过不向无特权用户公开“原始”套接字并隔离属于不同进程的套接字，来防止包注入到本地连接。由于rnic不提供特权的“原始”端点，因此被认为不可能伪造原生InfiniBand包。我们证明，任何用户都可以将数据包注入到本地RNIC上创建的任何连接中，而不管它是从用户空间还是内核空间创建的，从而绕过操作系统及其内核的安全机制。

在创建连接时缺乏健全检查。

**Lack of sanity checks during connection creation.**

通过直接使用RDMA动词库(我们进一步将其表示为本机RDMA连接建立)，用户可以在不依赖RDMA连接管理器的情况下创建RDMA端点。

这种方法不传递任何消息，因为它期望应用程序通过其他通信方式(例如，使用带外TCP连接)接收连接参数。

为了建立一个RDMA连接，每个主机手动创建一个RDMA端点，其中包含关于目标主机的信息，例如它的连接标识符、路由信息以及传入和传出包计数器的状态。

不幸的是，在使用本机RDMA连接建立RDMA端点的创建过程中，没有一个被测试的rnic硬件提供程序执行基本的完整性检查。这允许创建和使用以相同的远程RDMA端点(即具有相同的目标QPN)为目标的多个RDMA端点，但具有不同的本地标识符。

因此，有两个不同的过程。在一个主机上可以通信到相同的目的RDMA端点。

这允许攻击者创建受害者现有连接的几乎相同的副本(除了源QPN)，并使用它与相同的RDMA端点通信。由于使用本机RDMA接口创建RDMA端点不涉及任何网络通信，因此对这种模拟尝试的检测是有限的。

RDMA报文中不包含源QPN。

基于infiniband的协议在包中不包含源连接标识符(QPN)，而只在基本传输头中包含目标标识符这种设计选择基于RDMA连接是点对点通道这一事实。

因此，当RDMA连接建立时，将源QPN通信给接收端，然后将源QPN存储在接收端端点的连接表中。尽管这种设计选择本身看起来不像一个漏洞，但它可以与前面的设计缺陷结合，允许对手模仿具有不同源QPN的现有连接，在没有管理特权的情况下向现有连接注入数据包。

RDMA数据包中没有包含源QPN这一事实，使得从远程RDMA端点的角度来看，使用伪造连接的对手发送的数据包与常规RDMA数据包难以区分 因此，这两个设计缺陷的组合使数据包注入到任何本地RDMA端点而无需任何管理特权

实现---我们使用本机InfiniBand接口实现了包注入攻击，该接口使用RDMA谓词库手动构建连接端点。为此，攻击者需要知道受害者端点的信息，如网络端口地址、其连接标识符(QPN)和其当前包序列号(PSN)。

•目标网口地址被认为是公共信息(例如，它是RoCEv2的目标RNIC的IP地址)。

•连接标识符是由RNIC生成的24位数字。然而，在所有测试的rnic中，连接标识符的生成器都显示出了缺陷(见表2)。连接标识符是按顺序分配的，设备在重新启动后使用静态初始化值，这使得攻击者可以猜测一个有效的连接标识符。

假定攻击者与目标端点位于同一位置，并共享相同的RNIC，它可以创建自己的RDMA端点，并根据分配给其连接的标识符获取先前建立的连接的信息。类似地，如果攻击者可以合法地连接到目标计算机上的应用程序，它可以查询与该应用程序的连接的远程连接标识符。否则，攻击者可能试图在预定的重新启动后开始攻击，并注入标识符大于静态启动种子的包。我们测试了表2中列出的5个模型的31个不同rnic，起始种子在0x10到0x600的范围内。

•由于起始数据包序列号是由RDMA连接管理器随机生成的，并且序列号错误的数据包注入不会影响受害者的连接，我们建议枚举其所有状态(24位)。

**TLU模型下的注入。**

**Injection under the TLU model.**

利用讨论的漏洞，攻击者可以模拟主机上的一个端点，并对其连接执行注入。

为了验证这一点，我们实现了一个不需要管理权限且可以由任何用户运行的工具。

我们已经在InfiniBand和RoCE网络上测试了该工具，并成功地注入了RDMA请求。

有趣的是，我们最多可以从攻击者的连接中注入128个包，因为它从来没有收到来自目标端点的确认，因此，不能取得进一步的进展。

**4.3攻击RDMA连接管理器在本节中，我们展示了无特权攻击者如何欺骗RDMA端点，使其认为其连接已断开。**

**为此，攻击者利用RDMA连接管理器中的漏洞，允许攻击者伪造断开连接请求，迫使用户接收伪造的断开连接事件。**

NVMe oF及其实现NVMe o F协议允许通过支持RDMA的网络在远程固态存储设备或系统上执行NVMe命令。为此，它通过将NVMe命令和响应封装到RDMA消息的有效负载中，在当前主机和目标主机之间传输它们。NVMe协议区分了控制平面和数据平面胶囊。前者使用RDMA发送进行传输，而后者使用RDMA读写操作等单边内存访问进行传输。

NVMe oF读取。要从远程NVMe设备读取数据，客户端发送一个NVMe oF读取请求，其中包含目标应在何处写入数据的信息。目标将从磁盘读取数据，然后使用RDMA写入请求将其写入客户端的缓冲区。由于RDMA写入是一个“静默”的单边操作，目标通过RDMA发送发送响应以指示请求的完成。

5.2.2 NVMe of响应的欺骗。对手可以欺骗目标向客户端发出的响应。接收到客户端的响应意味着受影响的通信缓冲区已被使用，现在可以取消注册或重新用于新请求。因此，向NVMe of客户端注入NVMe o F响应可能会导致Linux内核实现的内存过早失效，以及SPDK的内存过早突变。

我们测试了此攻击是否会影响写入请求的NVMe。对于TLU模型，攻击工具与NVMe of目标在同一台机器上执行。攻击如图9所示，对手在RDMA读取请求到达目标后立即发送响应。由于客户机较早收到响应，它可能会在目标完全读取数据之前改变数据，从而导致将损坏的数据存储到磁盘中。因此，我们的NVMe oF响应注入允许在没有管理权限的情况下损坏远程磁盘上的数据。

5.2.3使用RDMA写入的内存损坏。攻击者可以注入RDMA写入请求，以更改NVMe客户端和NVMe目标的RDMA可访问内存。理论上，NVMeoF目标应该能够抵御这种攻击，因为它可能只为本地访问注册内存，因为客户端从不发出RDMA读写（见图7）。尽管如此，SPDK目标为远程访问注册内存，使其易受此攻击。Linux NVMe oF目标模块不存在此问题。

5.3 **NVMe oF漏洞的缓解NVMe工作组建议利用IPsec和使用DH-HMAC-CHAP协议的端点身份验证来缓解InfiniBand体系结构当前的安全缺陷。**

然而，即使部署了这些机制，也并不是所有建议的攻击都得到了缓解（汇总见表3）。问题是，所提出的技术不提供RDMA安全传输，因此在本地无特权攻击者（TLU）的情况下会受到数据包注入攻击。因此，我们建议采取以下额外的缓解机制。

NVMe oF漏洞的缓解NVMe工作组建议利用IPsec和使用DH-HMAC-CHAP协议的端点身份验证来缓解InfiniBand体系结构当前的安全缺陷。

NVMe oF消息验证。NVMe oF协议可以利用应用层安全性来验证NVMeoF消息，包括单边RDMA请求，即使RDMA的源验证无法在应用层完全实现。原因是RNIC在不涉及CPU的情况下执行单边RDMA请求，因此在执行这些数据包之前无法验证其真实性。例如，由于远程客户端可以使用RNIC的DMA引擎访问内存而不需要CPU的参与，因此不可能在应用程序层保护RDMA读取。

为了在不同的RDMA请求之间实现这种保护，NVMe oF应用程序需要在完成单边RDMA请求后使内存注册无效，以防止远程对手的RDMA写入注入导致内存突变。内存无效后，数据接收方应使用来自NVMeoF请求或响应的MAC验证数据的完整性。否则，攻击者可以在检查数据后，但在将其提交到本地NVMe设备之前修改数据。请注意，SPDK的当前实现选择性能而不是安全性，并且根本不会使内存注册无效。总的来说，NVMe oF规范应该为每个NVMe o F消息引入两个MAC，以独立验证数据和NVMe O F消息。如果它们使用通过消息和数据串联计算的一个MAC，则可能会导致部分执行伪造的NVMe of写入请求，因为NVMe o F目标需要发出RDMA读取以获取数据。

如何使用RDMA协议中的漏洞绕过NVMe of的安全机制。为了对NVMe oF实现进行攻击，我们设计了几种攻击工具，可用于攻击其他支持RDMA的应用程序。值得注意的是，我们展示了如何在没有管理权限的情况下将RDMA数据包欺骗到受害者连接中，即使启用了RoCE上的IPsec。关于NVMe oF协议，我们展示了没有管理权限的攻击者如何绕过NVMe o F协议和操作系统的现有安全机制，将数据写入远程NVMe设备。此外，我们还展示了如何伪造NVMe数据响应，从而迫使受害者客户端观察NVMe设备的伪造状态，而不实际修改NVMe的状态。我们预计，我们的工作将推动对高性能互连和利用它们的系统的安全研究，从而实现更安全的高性能网络和系统。