【论文分享】QUICforge: 基于 QUIC 协议的客户端伪造请求攻击

引言

今天分享的论文是针对 QUIC（Quick UDP Internet Connection）协议发起客户端请求伪造攻击，该工作由柏林工业大学分布式基础设施安全实验室的 Yuri Gbur 和 Florian Tschorsch 共同完成。作者从**QUIC 协议的设计**出发，分析了 (1) 服务端初始请求伪造（Server Initial Request Forgery, SIRF），(2) 版本协商请求伪造 （Version Negotiation Request Forgery, VNRF) 以及 (3) 连接迁移请求伪造（Connection Migration Request Forgery, CMRF) 这三种典型的客户端请求伪造攻击模式。作者发现 VNRF 可用于模拟 DNS 等基于 UDP 的协议，SIRF 和 CMRF 可用于流量放大攻击。作者评估了 13 种开源的 QUIC 协议实现探索了请求伪造攻击的影响，结果表明，**所有实现都受 SIRF 和 VNRF 请求伪造攻击影响，QUIC 协议存在潜在的脆弱性。**

论文发表于国际网络安全四大顶级学术会议 NDSS 2023。

QUICforge: Client-side Request Forgery in QUIC

Yuri Gbur
Technische Universität Berlin
Berlin, Germany
yuri.gbur@posteo.de

Florian Tschorsch
Technische Universität Berlin
Berlin, Germany
florian.tschorsch@tu-berlin.de

全文共 3400 字，阅读时间约 10 分钟。

背景介绍

QUIC 协议

随着 QUIC 协议的标准化 [1] 以及 Apple, Cloudflare, Facebook 以及 Google 等大型互联网厂商对 QUIC 协议应用与推广，QUIC 协议变得越来越受欢迎。QUIC 协议建立在 **UDP 协议**之上，交互双方通过共享 CID（Connection ID）来维护应用层状态。它结合了 TCP 协议和 TLS 1.3 协议的功能，并且能够减少连接配置期间的 TTL（Time To Live），被作为新 HTTP/3 标准的核心协议。QUIC 协议导致网络协议栈发生的变化如图 1 所示 [2]：

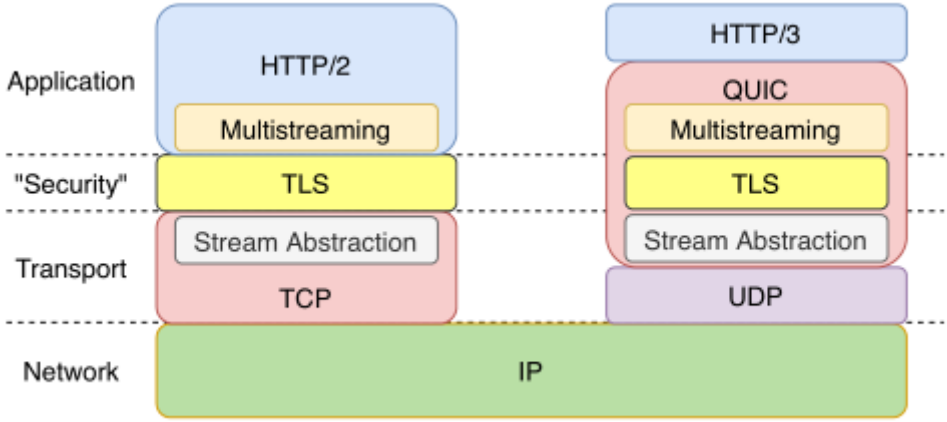


图1：HTTP/2 和HTTP/3协议栈对比

攻击方式

威胁模型

客户端请求伪造攻击是客户端（攻击者）触发受害方（服务器）向另一主机发送一个或多个“非预期”的网络请求，该攻击通常通过**滥用协议的特性**或者**滥用应用程序的功能**实现 [5-7]。通过请求伪造攻击，攻击者可以实现两个目标：

1. 获取服务器**更高的权限**，进而操纵目标主机
2. 利用服务器到目标主机**更高的带宽**发起其他攻击

作者假定攻击者可以修改 QUIC 数据包的内容并且目标主机至少有一个端口可以接受 QUIC 数据包，但不假设攻击者可以直接操作目标主机，攻击模型如图 2 所示：

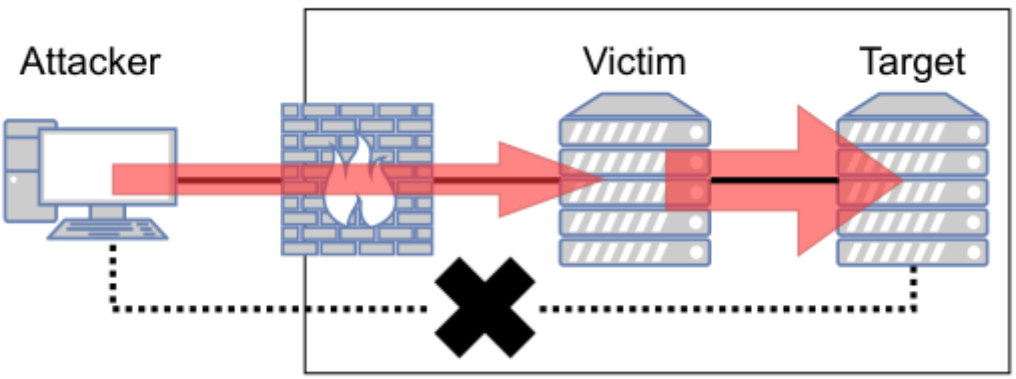


图 2：攻击模型

客户端伪造请求

客户端伪造请求主要包括服务端初始请求伪造（Server Initial Request Forgery，SIRF)，版本协商请求伪造（Version Negotiation Request Forgery，VNRF) 以及连接迁移请求伪造（Connection Migration Request Forgery，CMRF) 三种方法。

服务端初始请求伪造（SIRF）

服务端初始请求伪造（Server Initial Request Forgery，SIRF) 是 QUIC 协议中客户端最常使用的请求伪造技术。攻击者利用伪造的 QUIC 协议数据包（伪造其源 IP 和端口号），使用服务器支持的 QUIC 协议版本，向服务器发起 QUIC 握手，服务器利用攻击者伪造的 IP 进行握手回应 [2]，具体过程如图 3 所示。

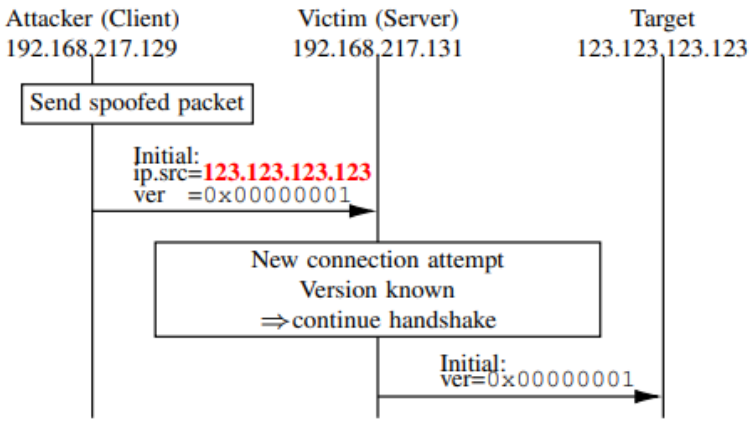


图 3：SIRF 交互过程

版本协商请求伪造（VNRF）

版本协商请求伪造（Version Negotiation Request Forgery，VNRF) 类似于 SIRF，通过利用 QUIC 协议握手的**另一个特性——如果客户端请求数据包中包含未知版本信息，服务器将使用版本协商数据包进行响应** [2]。攻击者可以通过伪造源 IP 和端口号并利用服务器不支持的 QUIC 版本信息来完成伪造攻击，具体过程如图 4 所示。

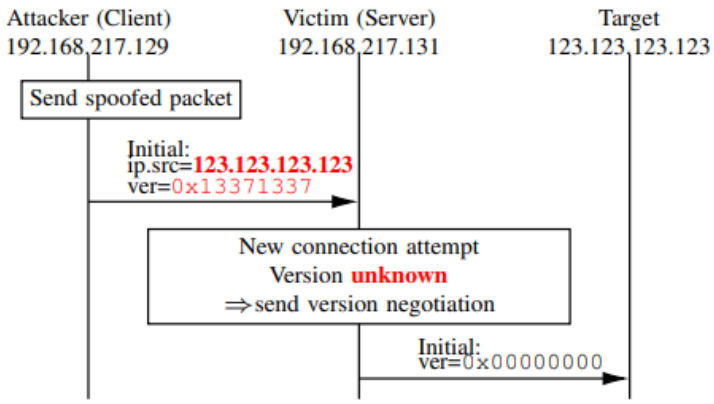


图 4：VNRF 交互过程

连接迁移请求伪造（CMRF）

QUIC 协议的连接迁移是指在网络地址或路径发生变化时，将连接从一个端点（Endpoint）转移到另一个端点，而不需要重新进行握手的过程，进而保持通信的连续性。CMRF 利用 QUIC 协议的连接迁移（Connection Migration）特性。攻击者需要启动连接并完成握手过程，然后向服务器端发送包含伪造 IP 的数据包，服务器端检测到客户端 IP 发生变化后，向攻击者伪造的 IP 地址发送包含 PATH_CHALLENGE 帧的 QUIC 数据包（验证迁移后的路径是否可达），具体过程如图 5 所示。

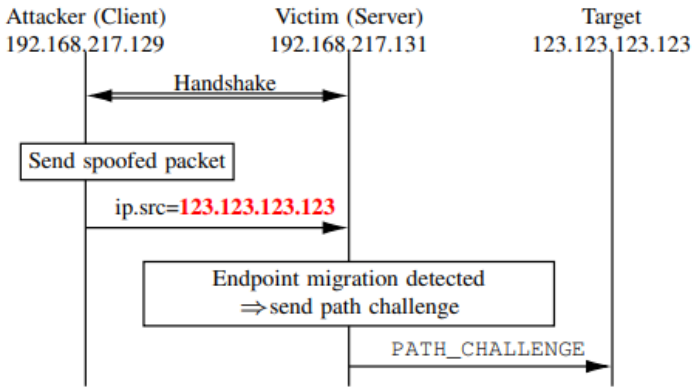


图 5：CMRF 交互过程

协议模拟（PROTOCOL IMPERSONATION）

QUIC 在技术上是一种应用层协议，攻击者可以通过伪造请求来攻击其他基于 UDP 的协议，这种攻击也被称为**跨协议请求伪造（CPRF）攻击**[3]，由于加密会导致密文难以控制，因此，作者主要对 QUIC 数据包中未加密的部分进行研究。由于 SIRF 和 CMRF 数据包中可以控制的部分有限，而 VNRF 数据包中有大量字节可以更改控制，故只有VNRF可进行协议模拟，版本协商数据包如图 6 所示，其中，前 8 个字节无法控制，其余自己均可被攻击者直接篡改设置。

```
Version Negotiation Packet {
        Header Form (1) = 1,
        Unused (7),
        Version (32) = 0,
        Destination Connection ID Length (8),
        Destination Connection ID (0..2040),
        Source Connection ID Length (8),
        Source Connection ID (0..2040),
        Supported Version (32) ...,
}
```

图 6：版本协商数据包 (Version Negotiation Packet)

模拟DNS数据包

作者手动构造一个 QUIC 协议数据包，如图 7 所示，通过 VNRF 将其发送到递归解析器（Recursive Resolver），进而触发对域名 tu-berlin.de 的解析，通过 Wireshark 抓包查看协议的交互过程，如图 8 所示。

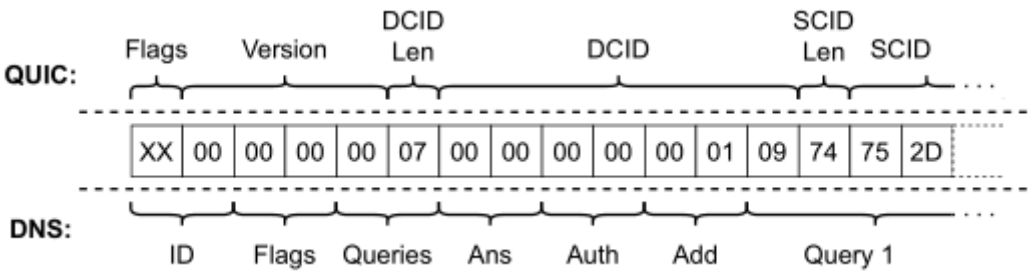


图 7：QUIC 协议伪造 DNS 协议

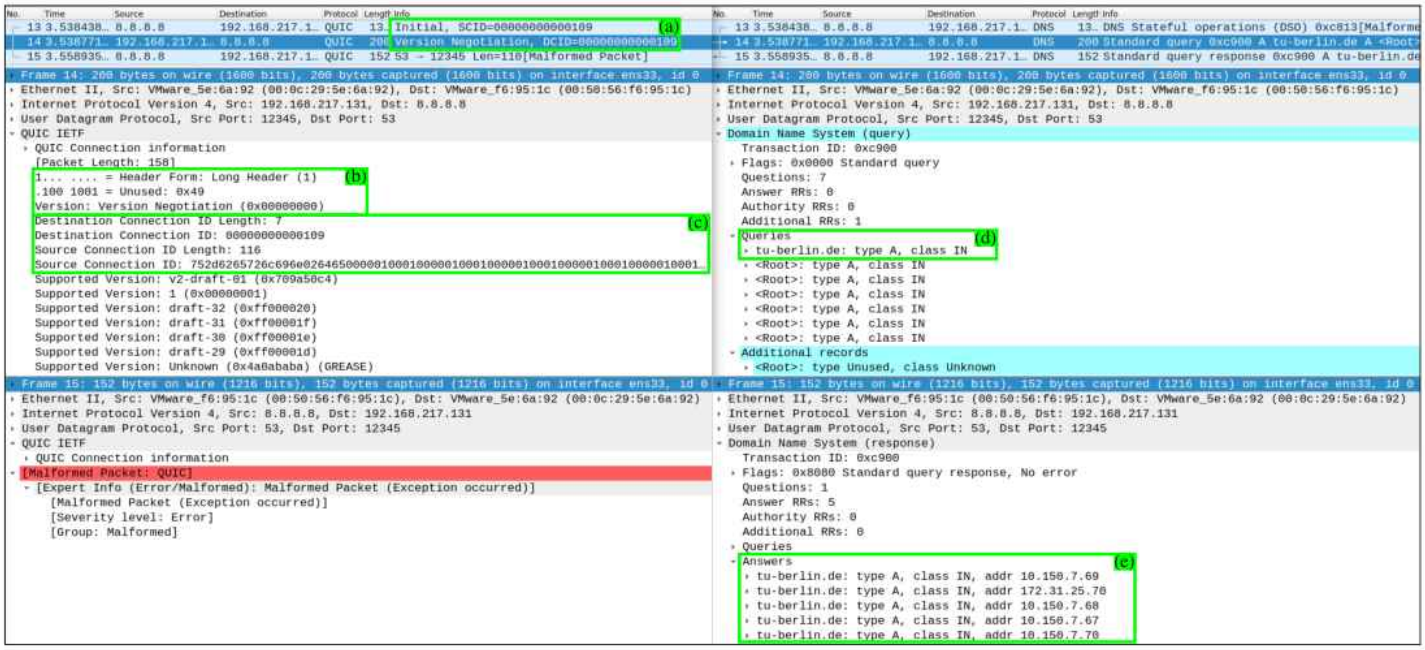


图 8：通过 Wireshark 抓包分析协议模拟过程（左边为构造的 QUIC 协议数据包，右边为返回的 DNS 协议数据包）

流量放大（TRAFFIC AMPLIFICATION）

流量放大是指攻击者控制服务器转发的数据量大于攻击者自己伪造的数据量，为了能够更直观的观察流量放大，作者使用路径放大因子（Path Amplification Factor，PAF）来衡量流量放大，如图 9 所示

$$\mathrm{PAF} = \frac{\#\text{ bytes from victim to target}}{\#\text{ bytes from attacker to victim with spoofed address}}$$

图 9：流量放大因子

QUIC 协议中可能导致放大攻击的因素有三个：

1. **PMTUD(Path Maximum Transmission Unit Discovery):** QUIC 协议规定，**网络中必须能够支持 1200 字节大小的数据包** [2]，故支持 QUIC 协议的服务器在传输第一个数据包时，会将数据包填充到 1200 字节（PADDING）以判断网络是否能够支持 QUIC 协议。
2. **TLS 参数：**在 QUIC 协议握手的过程中，服务端所需的 TLS 参数一般要比客户端需要的参数更大，故在密钥协商的过程中也会导致一定程度的流量放大 [2]
3. **可靠性：**为了确保可靠性，服务端可能会**一次发送多个重复数据包或者在一定时间未收到数据包之后进行超时重传** [8]

评估

作者评估了所有支持 QUIC 版本 1 开源的应用实现，由于互操作性以及应用本身存在的问题，作者最终选取了 13 个支持 QUIC 协议的应用作为评估对象 [4]，评估模型如图 10 所示，图中的控制流为实线，网络流量为虚线。

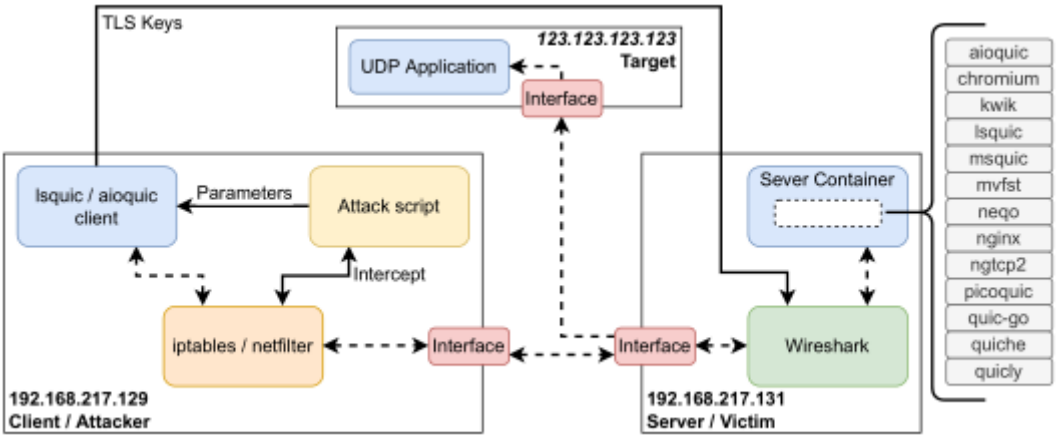


图 10：评估模型

协议的脆弱性评估

作者对 13 个 QUIC 实现进行脆弱性评估，评估的结果如表 1 所示。

Client	CMRF				SIRF			VNRF	
	Vuln.	Pad.	New CID	PAF>3	Vuln.	PAF>3	Ref. CID	Vuln.	CID>20
aioquic	✓	✗	✗	✗	✓	✓	✗	✓	✗
chromium	✓	✗	✗	✓	✓	✓	✗	✓	✓
kwik	✗	-	-	-	✓	✗	✗	✓	✓
lsquic	✓	✓	✓	✓	✓	✗	✗	✓	✗
msquic	✓	✗	✓	✗	✓	✓	✗	✓	✓
mvfst	✓	✓	✗	✓	✓	✓	✗	✓	✗
neqo	✓	✗	✓	✓	✓	✓	✗	✓	✓
nginx	✓	✗	✓	✓	✓	✗	✗	✓	✗
ngtcp2	✓	✗	✗	✗	✓	✗	✗	✓	✓
picoquic	✓	✗	✓	✓	✓	✗	✗	✓	✓
quic-go	✗	-	-	-	✓*	✗	✗	✓	✓
quiche	✗	-	-	-	✓*	✗	✗	✓	✓
quicly	✗	-	-	-	✓	✓	✗	✓	✓
Total	**9**	**2**	**5**	**6**	**11**(13)	**6**	**0**	**13**	**9**

* Sends retry packet instead of server initial packet

表 1：评估三种客户端请求伪造攻击

其中：

1. Vuln. 表示该实现是否易受该种请求伪造攻击
2. Pad. 表示服务器是否将包含 PATH_CHALLENGE 的第一个数据包填充到 1200 字节（PMTUD）
3. New CID 表示受害方（服务器转）在转发数据包时是否使用新的CID（Connection ID）
4. Ref.CID 表示服务端在转发数据包时是否直接将 DCID（Destination Connection ID）和 SCID（Source Connection ID ）调换位置
5. CID>20 表示服务器是否能够响应 CID 长度大于 20 字节的客户端初始数据包

结论：9 种软件实现受 CMRF 影响，所有软件实现均受 SIRF 和 VNRF 影响，客户端请求伪造攻击对于 QUIC 协议的影响是非常大的。

流量放大评估

由于 QUIC 协议中版本协商数据包较小，故 VNRF 不适用于流量放大攻击。作者对 CMRF 和 SIRF 所造成的流量放大进行研究，不同实现流量放大的结果如图 11 所示。

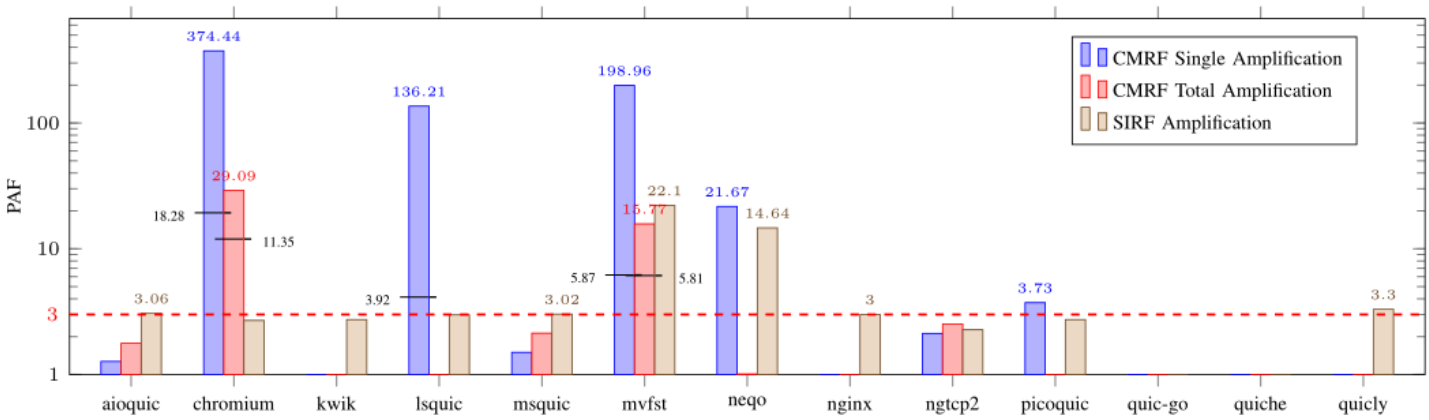
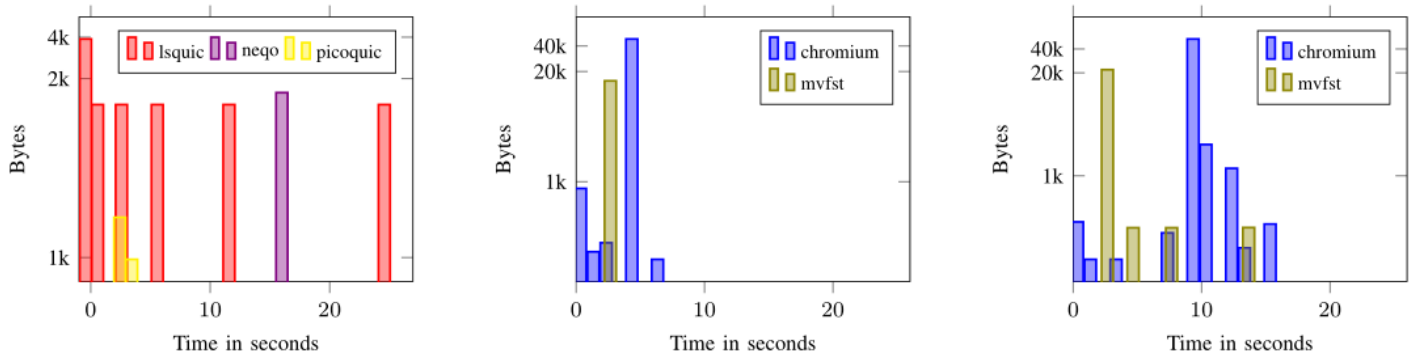


图 11：评估 CMRF 和 SIRF 对 QUIC 协议流量放大的影响

其中，红色虚线表示 QUIC 协议设计之时限制流量放大的最大倍数。可以看出，CMRF 可以使得 chromium 流量放大 374.44 倍、mvfst 流量放大 198.96 倍、lsquic 流量放大 136.21 倍。作者接下来分析了流量的产生随时间的变化，如图 12 所示。



(a) Single amplification for server with 10k bytes maximum peak.

(b) Single amplification for server with more than 10k bytes peak.

(c) Total amplification.

图 12：流量随时间的变化

结论：CMRF 可以使得部分应用产生严重的流量放大，但数据流量并非在一瞬间产生的，这个结果并不是攻击者想要看到的。

缓解措施

作者在和 QUIC 协议不同实现的开发团队沟通时得知，流量放大攻击产生的原因是在设计 QUIC 协议时**考虑了协议的向后兼容特性（Backwards Compatibility）**，作者针对协议模拟和流量放大提出以下几点缓解措施。

协议模拟

1. 服务器端应该始终重新选择新的 CID，并且 CID 长度不固定
2. 服务器端将 CID 值哈希处理，但这会导致 QUIC 协议性能下降（路由和负载均衡）
3. 在 QUIC 协议数据包的首部增加 32 bit 的掩码，再将数据包的其他部分和该掩码进行异或处理

但这些缓解措施都**不具有向后兼容性（Backwards Compatibility）**，如何解决协议更新迭代过程中向后兼容的特性是一个非常值得的问题。

流量放大

1. 更改可靠性机制，避免一次性发送多个重复数据包
2. 客户端将初始数据包填充到 1200 字节，避免服务器端进行 PMTUD 来探测网络是否能够支持 QUIC 协议

总结

随着 QUIC 协议的标准化以及各大厂商基于 QUIC 协议进行新的应用开发，QUIC 协议变得越来越受欢迎。作者从QUIC 协议的自身设计出发，分析了 SIRF，VNRF，CMRF 三种客户端伪造请求攻击模式，发现能够通过 VNRF 进行跨协议请求，通过 CMRF 和 SIRF 进行流量放大攻击。作者认为，在大规模采用 QUIC 协议之前，有必要对 QUIC 协议的安全性进行深入研究，如何解决 QUIC 协议当下存在的安全问题并保证协议的向后兼容性是一个亟待解决的问题。

文章链接：

https://www.ndss-symposium.org/ndss-paper/quicforge-client-side-request-forgery-in-quic/

参考文献

[1] M. Thomson, “V ersion-Independent Properties of QUIC,” RFC 8999, May 2021.

[2] M. Bishop, “Hypertext Transfer Protocol V ersion 3 (HTTP/3),” Internet Engineering Task Force, Internet-Draft draft-ietf-quic-http-34, Feb. 2021, work in Progress.

[3] T. Prynn, “Cross-protocol request forgery,” NCC Group Whitepaper, Oct. 2018.

[4] QUIC Working Group, “Implementations,” Aug. 2021. [Online]. Available: https://github.com/quicwg/base-drafts/wiki/Implementations

[5] J. Topf, “The HTML Form Protocol Attack,” Aug. 2001. [Online]. Available: https://www.jochentopf.com/hfpa/hfpa.pdf

[6] A. Barth, C. Jackson, and J. C. Mitchell, “Robust defenses for cross-site request forgery.” in Proceedings of the 15th ACM Conference on Computer and Communications Security, ser. CCS

’ 08. New Y ork, NY , USA: Association for Computing Machinery, 2008, p. 75–88.

[7] N. Jovanovic, E. Kirda, and C. Kruegel, “Preventing cross site request forgery attacks,” in 2006 Securecomm and Workshops, 2006, pp. 1–10.

[8] J. Iyengar and M. Thomson, “QUIC: A UDP-Based Multiplexed and Secure Transport,” RFC 9000, May 2021.

封面图