Technical Analysis of the IRGFW

Understanding The Iranian Great Firewall

Report 1

December 2024

IRGFW
irgfw.report

伊朗大防火墙技术分析

理解伊朗大防火墙

报告 1

2024年12月

IRGFW
irgfw.report

IRGFW
irgfw.report

Glossary

Word	Meaning	Word	Meaning	Word	Meaning	Word	Meaning
ASIC	Application-Specific Integrated Circuit	ECH	Encrypted Client Hello	ISP	Internet Service Provider	TCP	Transmission Control Protocol
ASN	Autonomous System Number	ESNI	Encrypted Server Name Indication	L2TP	Layer 2 Tunneling Protocol	TIC	Telecommunication Infrastructure Company
CDN	Content Delivery Network	GFW	Great Firewall	MCI	Mobile Communication Company of Iran	TLS	Transport Layer Security
CIDR	Classless Inter-Domain Routing	GRPC	gRPC Remote Procedure Call	MTN	Irancell	TOR	The Onion Router
CPU	Central Processing Unit	HTTP	Hypertext Transfer Protocol	NIN	National Information Network	UDP	User Datagram Protocol
DDOS	Distributed Denial-of-Service	HTTPS	Hypertext Transfer Protocol Secure	OBFS	Obfuscation	UL	Upload
DL	Download	HU	HttpUpgrade *(=WebSocket)*	P2P	Peer-to-Peer	UTLS	Universal Transport Layer Security
DNS	Domain Name System	ICMP	Internet Control Message Protocol	PPTP	Point-to-Point Tunneling Protocol	VPN	Virtual Private Network
DOH	DNS-over-HTTPS	IKEV2	Internet Key Exchange version 2	QUIC	Quick UDP Internet Connections	VPS	Virtual Private Server
DOQ	DNS-over-QUIC	IPM	Institute for Research in Fundamental Sciences	SNI	Server Name Indication	WS	WebSocket
DOT	DNS-over-TLS	IPSEC	Internet Protocol Security	SSH	Secure Shell		
DOU	DNS-over-UDP	IPV4	Internet Protocol Version 4	SSTP	Secure Socket Tunneling Protocol		
DPI	Deep Packet Inspection	IPV6	Internet Protocol Version 6	SYN	Synchronize		
DTLS	Datagram Transport Layer Security	IRGFW	Iranian Great Firewall	TCI	Telecommunication Company of Iran		

(Table 1 – Glossary)

术语表

Word	含义	Word	含义	Word	含义	Word	含义
ASIC	应用专用型集成电路	ECH	加密客户端问候	ISP	互联网服务提供商	TCP	传输控制协议
ASN	自治系统编号	ESNI	加密服务器名称指示	L2TP	第二层隧道协议	TIC	电信基础设施公司
CDN	内容分发网络	GFW	防火长城	MCI	移动通信伊朗公司	TLS	传输层安全
CIDR	无类别域间路由	GRPC	gRPC 远程过程调用 Call	MTN	Irancell	TOR	洋葱路由
CPU	中央处理器	HTTP	超文本传输协议	NIN	国家信息网络	UDP	用户数据报协议
DDOS	分布式拒绝服务攻击	超文本传输安全协议	超文本传输协议安全	OBFS	混淆	UL	上传
DL	下载	HU	HttpUpgrade (=WebSocket)	P2P	点对点	UTLS	通用传输层安全
DNS	域名系统	ICMP	Internet控制消息协议	PPTP	点对点隧道协议	VPN	虚拟专用网络
DOH	基于HTTPS的DNS	IKEV2	互联网密钥交换第二版	QUIC	快速UDP互联网连接连接	VPS	虚拟专用服务器
DOQ	基于QUIC的DNS	IPM	基础科学研究所基础科学研究所	SNI	服务器名称指示	WS	WebSocket
DOT	基于TLS的DNS	IPSec	互联网协议安全	SSH	安全外壳协议		
DOU	基于UDP的DNS	IPV4	互联网协议第4版	SSTP	安全套接字隧道协议		
DPI	深度包检测	IPV6	互联网协议第六版	SYN	同步		
DTLS	数据报传输层安全	IRGFW	伊朗大防火墙	TCI	伊朗电信公司		

（表 1 – 术语表）

Brief History of the Iranian Great Firewall (IRGFW)

Before the tragic death of Mahsa Amini[1][2], the Islamic Regime of Iran's internet filtering system was relatively unsophisticated, the primary methods used were DNS and SNI blocking, which blocked non-TLS and TLS connections to foreign IP addresses. Deep packet inspection *(DPI)* and active probing technologies were minimal and largely invisible, indicating a less comprehensive approach to controlling internet traffic.

The situation drastically changed following Mahsa Amini's death and the subsequent nationwide protests. The Telecommunication Infrastructure Company *(TIC)* and other entities significantly upgraded the nation's internet censorship infrastructure. This involved acquiring and importing advanced firewall and DPI hardware, marking a shift towards a more rigorous and sophisticated internet control system.[3]

Iran has been increasingly inspired by the Chinese internet censorship model, often called the "Great Firewall of China."[4] Despite official claims that Iran is not directly following China's example, there are undeniable parallels in the methods and strategies employed.[5] Iran has developed its national internet infrastructure, aiming to increase domestic Internet traffic to 70% of the total internet traffic in the country, similar to China's promotion of local internet services to reduce reliance on global platforms.[6][7][8]

In addition to hardware upgrades, Iran has imposed stricter regulations on internet platforms, requiring them to comply with local laws or face censorship. This project, called “Sianat” aims to create a controlled internet environment that minimizes the influence of foreign platforms and increases the government's control over digital content and communication.[9]

伊朗大防火墙简史(IRGFW)

在玛莎・阿米尼的悲剧发生之前[1][2],伊朗伊斯兰政权的互联网过滤系统相对简陋，主要使用的方法是域名系统封锁和服务器名称指示封锁，用以阻止通往国外IP地址的非TLS和TLS连接。深度包检测*(DPI)*和主动探测技术极少使用且基本不可见，这表明其对互联网流量的控制采取了一种不那么全面的方式。

玛莎・阿米尼死亡及随后的全国性抗议发生后，情况发生了急剧变化。电信基础设施公司*(TIC)*及其他实体大幅升级了该国的互联网审查基础设施。这包括采购和引进先进的防火墙和深度包检测硬件，标志着其向更严格、更复杂的互联网控制系统转变。[3]

伊朗越来越受到中国互联网审查模式（常被称为“中国防火墙”）的启发。[4]尽管官方声称伊朗并非直接效仿中国的做法，但两者所采用的方法和策略存在不可否认的相似之处。[5]伊朗已发展其国家互联网基础设施，目标是使国内互联网流量达到该国互联网总流量的70%，这类似于中国推广本地互联网服务以减少对全球平台的依赖。[6][7][8]

除了硬件升级，伊朗还对互联网平台实施了更严格的监管，要求它们遵守当地法律，否则将面临审查。这个名为“Sianat”的项目旨在创建一个受控的互联网环境，以最大限度地减少外国平台的影响，并增加政府对数字内容和通信的控制。[9]

IRGFW: Digital Boundaries

The IRGFW also features extensive use of DPI to inspect and filter internet traffic at a granular level. This technology allows the government to block specific websites, monitor internet usage, and prevent access to certain content. The primary consumer ISPs in Iran, such as the **Mobile Communication Company of Iran** *(MCI)*, **IranCell** *(MTN)*, and the **Telecommunication Company of Iran** *(TCI)*, connect upstream to the **TIC** *(AS49666)*, which houses the primary firewall. This centralization ensures that blocking and filtering measures are uniformly enforced across all ISPs. [10][11]

The nationwide implementation of these advanced technologies and strict regulatory policies has significantly enhanced Iran's ability to monitor and control internet usage. This transformation reflects a broader trend towards increasing digital authoritarianism, leveraging state-of-the-art technologies to control information and suppress dissent.

The Iranian Great Firewall (IRGFW) is a complex and repressive internet censorship and surveillance apparatus. By employing advanced network filtering and traffic inspection techniques, it enforces pervasive restrictions on online communication and information access. Despite its oppressive nature, it is considered among the world's more formidable national censorship systems, integrating multiple technological and policy layers to strictly regulate and monitor internet traffic within Iran. This report provides a detailed technical analysis of the IRGFW's infrastructure and operational mechanisms to better understand how it achieves its high level of control.

At its core, the IRGFW operates through a coordinated effort involving major Internet Service Providers (ISPs) and the Telecommunication Infrastructure Company *(TIC)*, which serves as the primary upstream provider. Through deep packet inspection *(DPI)*, IP blocking, and other advanced network management techniques, the IRGFW enforces stringent controls over data flow. Understanding the architecture and operation of this system is crucial for comprehending the extent and efficiency of internet censorship in Iran.

First, we need some basic understanding of how and where the IranGFW *(IRGFW)* works. And for sure, we can say it is a unique set of firewalls.

IRGFW：数字边界

IRGFW还广泛采用DPI技术，以精细地检查和过滤互联网流量。这项技术使政府能够屏蔽特定网站、监控互联网使用情况并阻止访问某些内容。伊朗的主要消费者互联网服务提供商，例如**伊朗移动通信公司***(MCI)*、**IranCell***(MTN)* 和**伊朗电信公司***(TCI)*，都向上连接到**TIC***(AS49666)*，其中部署着主要的防火墙。这种集中化确保了所有互联网服务提供商都统一执行屏蔽和过滤措施。[10][11]

这些先进技术和严格监管政策的全国性实施，极大地增强了伊朗监控和管理互联网使用的能力。这一转变反映了一个更广泛的趋势，即日益增长的数字威权主义正利用尖端技术来控制信息和压制异议。

伊朗大防火墙（IRGFW）是一个复杂且具有压制性的互联网审查与监控体系。它通过采用先进的网络过滤和流量检测技术，对在线通信和信息访问实施普遍限制。尽管其性质具有压迫性，它仍被认为是全球更强大的国家审查系统之一，融合了多层技术和政策，以严格规范和监控伊朗境内的互联网流量。本报告对IRGFW的基础设施和运行机制进行了详细的技术分析，以更好地理解其如何实现如此高水平的控制。

IRGFW的核心运作依赖于主要互联网服务提供商（ISP）和作为主要上游提供商的电信基础设施公司（TIC）之间的协同努力。通过深度包检测（DPI）、IP封锁和其他先进的网络管理技术，IRGFW对数据流实施严格的控制。理解该系统的架构和运行方式，对于把握伊朗互联网审查的范围和效率至关重要。

首先，我们需要对伊朗防火墙 *(IRGFW)* 的工作原理和部署位置有一些基本的了解。可以肯定地说，它是一套独特的防火墙系统。

Iran AS Cones and Firewalls

Major consumer Iranian ISPs are:

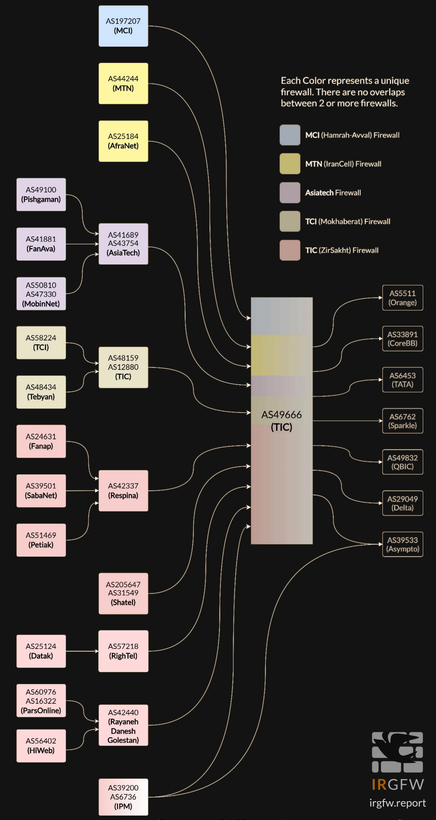
1. **MCI** *(AS197207)* [12]
2. **MTN** *(AS44244)* [13]
3. **TCI** *(AS48159 + AS12880)* [14][15]

All internet operators will go upstream to the **Telecommunication Infrastructure Company** *(TIC)* – **AS49666**, which houses the primary firewall and gateway.[15]

Each of these ISPs has different types of firewalls, and all other operators will follow one of these firewalls. For instance, when an IP address gets blocked on ASIATECH, it is blocked on PISHGAMAN, FANAVA, and MOBINNET. This is true for the DPI and firewall itself as well. The most advanced firewall belongs to the MCI operator *(the biggest mobile operator in Iran)*.

However, these firewalls *(especially MCI)* are sometimes turned off due to countrywide events. The TIC *(AS49666)* primary firewall will be used when the ISP firewall is turned off. For instance, when the MCI firewall is turned on, the TIC firewall will be overridden; thus, when an IP address gets blocked on MCI, it won't be blocked on TIC to some extent. After some time *(based on a "time-pattern" that we discuss in this report)*, the blacklist database will be synced to TIC *(AS49666)*, and then it's blocked on all ISPs.

The IPM *(AS6736)* Internet provides access to free Internet without the restrictions of the primary *(AS49666)* firewall. This organization is one of the oldest and was originally established for elite individuals, government officials, and verified researchers. Additionally, it has a strict bandwidth limit, typically capping at 10/100 Mbps.[16][17]



(Diagram 1 – IR AS Firewalls)
[Link]

伊朗自治系统锥与防火墙

主要的伊朗消费者互联网服务提供商有：

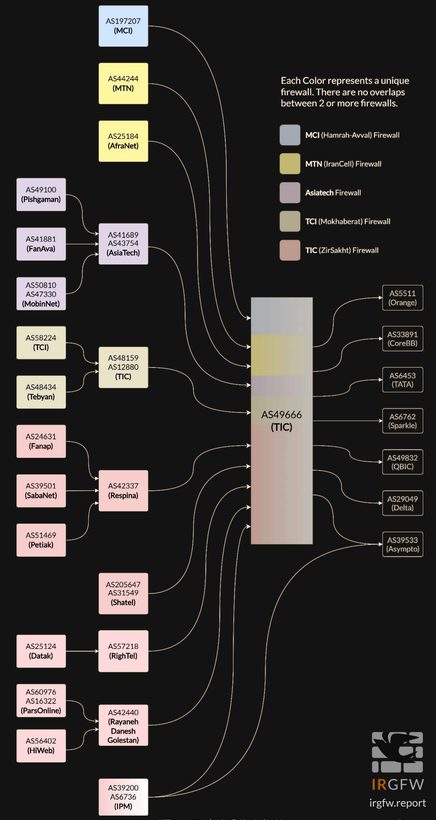
1. **MCI** (自治系统AS197207) [12]
2. **MTN** (AS44244) [13]
3. **TCI** (自治系统*4B159 +* AS12880) [14][15]

所有互联网运营商都将上行至**电信基础设施公司** *(TIC)* –**AS49666**，这里部署着主要的防火墙和网关。[15]

这些互联网服务提供商各自拥有不同类型的防火墙，而所有其他运营商都将遵循其中一种防火墙。例如，当一个IP地址在ASIATECH上被封锁时，它在PISHGAMAN、FANAVA和MOBINNET上也会被封锁。深度包检测和防火墙本身也是如此。最先进的防火墙属于MCI运营商（伊朗最大的移动运营商）。

然而，这些防火墙(尤其是MCI) 有时会因全国性事件而被关闭。当互联网服务提供商的防火墙关闭时，将使用TIC*(AS49666)* 的主防火墙。例如，当MCI防火墙开启时，TIC防火墙将被覆盖；因此，当一个IP地址在MCI上被阻止时，它在某种程度上不会在TIC上被阻止。一段时间后(基于我们将在本报告中讨论的"时间模式")， 黑名单数据库将同步至TIC*(AS49666)*， 随后该IP地址将在所有互联网服务提供商处被阻止。

IPM *(AS6736)* 互联网提供不受主*(AS49666)* 防火墙限制的免费互联网接入。该组织是最古老的机构之一，最初是为精英人士、政府官员和经过验证的研究人员设立的。此外，它有严格的带宽限制，通常上限为10/100 Mbps。[16][17]



(图1 – IR 自治系统防火墙)
[链接]

DNS Situation

DNS 状况

DNS Situation

DNS requests are subject to DPI, which often results in frequent poisoning and disruptions of DNS queries. Requests to well-known DNS providers are consistently graylisted, regardless of the encryption method used—whether DNS over UDP *(DoU)*, DNS over TLS *(DoT)*, DNS over HTTPS *(DoH)*, or DNS over QUIC *(DoQ)*. This issue is so widespread that, in many cases, it is necessary to rely on the DNS servers provided by the ISP for domestic (local) connections, particularly for traffic routed through the IRGFW.[25]

However, users can mitigate these disruptions by setting up their own DNS servers using encrypted protocols *(DoT, DoH, or DoQ)*. This approach enables users to bypass DNS poisoning, but it introduces two key challenges:

- **Graylisting of Destination IPs**: If the destination IP address is graylisted, the TLS handshake process may fail, preventing the establishment of DoT and DoH connections.
- **Using DoQ**: If UDP traffic is allowed to the destination IP, DoQ can be used to overcome the restrictions associated with the TLS handshake, ensuring secure DNS resolution despite DPI interference.

These challenges underscore the need for advanced DNS management techniques that address both the use of encrypted DNS protocols and the complexities introduced by DPI and graylisting.

DNS 状况tion

DNS查询会受到深度包检测的审查，这常常导致DNS查询频繁被投毒和干扰。无论使用何种加密方式——无论是基于UDP的DNS *(DoU)*、基于TLS的DNS *(DoT)*、基于HTTPS的DNS *(DoH)*、还是基于QUIC的DNS *(DoQ)*，对知名DNS服务商的请求都会被持续列入灰名单。这个问题非常普遍，以至于在许多情况下，对于国内（本地）连接，尤其是流经IRGFW的流量，必须依赖互联网服务提供商提供的DNS服务器。[25]

然而，用户可以通过使用加密协议(DoT、DoH或DoQ) 搭建自己的DNS服务器来减轻这些干扰。这种方法能帮助用户绕过DNS投毒，但会带来两个主要挑战：

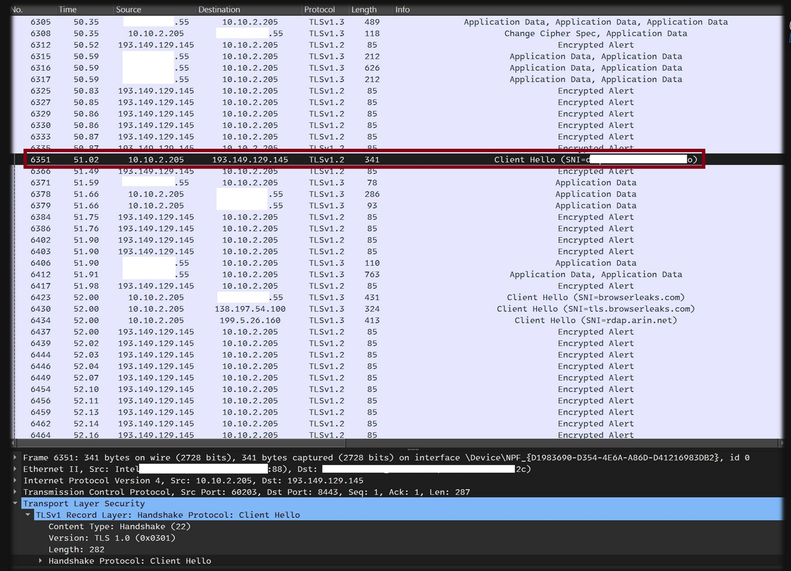
- **目标IP地址被列入灰名单**：如果目标IP地址被列入灰名单，TLS握手过程可能失败，从而无法建立DoT和DoH连接。
- **使用DoQ**：如果允许UDP流量到达目标IP地址，则可以使用DoQ来突破限制。与TLS握手相关联，确保在DPI干扰下仍能进行安全的DNS解析。

这些挑战凸显了采用先进域名系统管理技术的必要性，这些技术需同时应对加密的DNS协议以及由DPI和灰名单引入的复杂性。

DNS Situation

DNS-over-HTTPS *(DoH)*

In this scenario, we configure a DoH server with a whitelisted IP address and an SNI domain. The server listens for DoH requests on both port 443 and port 8443, with Nginx acting as the web server for general HTTP/HTTPS traffic. Both ports are accessible from a standard web browser, allowing the associated website to load normally. However, when using a popular DNS client *(e.g., YogaDNS)*, DoH requests are blocked. Specifically, ClientHello messages are successfully transmitted to the server, but no corresponding ServerHello messages are returned, causing the DoH connection to time out. This behaviour suggests a filtering mechanism affecting the DoH handshake process, preventing the successful resolution of DNS queries over HTTPS.



No.	Time	Source	Destination	Protocol	Length	Info
6305	50.35	.55	10.10.2.205	TLSv1.3	489	Application Data, Application Data, Application Data
6308	50.35	10.10.2.205	.55	TLSv1.3	118	Change Cipher Spec, Application Data
6312	50.52	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6315	50.59	.55	10.10.2.205	TLSv1.3	212	Application Data, Application Data
6316	50.59	.55	10.10.2.205	TLSv1.3	626	Application Data, Application Data
6317	50.59	.55	10.10.2.205	TLSv1.3	212	Application Data, Application Data
6325	50.83	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6327	50.85	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6329	50.86	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6330	50.86	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6333	50.87	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6335	50.87	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6351	51.02	10.10.2.205	193.149.129.145	TLSv1.2	341	Client Hello (SNI=[redacted]o)
6366	51.40	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6371	51.59	.55	10.10.2.205	TLSv1.3	78	Application Data
6378	51.66	10.10.2.205	.55	TLSv1.3	286	Application Data
6379	51.66	10.10.2.205	.55	TLSv1.3	93	Application Data
6384	51.75	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6386	51.76	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6402	51.90	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6403	51.90	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6406	51.90	.55	10.10.2.205	TLSv1.3	110	Application Data
6412	51.91	.55	10.10.2.205	TLSv1.3	763	Application Data, Application Data
6417	51.98	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6423	52.00	10.10.2.205	.55	TLSv1.3	431	Client Hello (SNI=browserleaks.com)
6430	52.00	10.10.2.205	138.197.54.100	TLSv1.3	324	Client Hello (SNI=tls.browserleaks.com)
6434	52.00	10.10.2.205	199.5.26.160	TLSv1.3	413	Client Hello (SNI=rdap.arin.net)
6437	52.00	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6439	52.02	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6444	52.03	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6446	52.04	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6449	52.07	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6454	52.10	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6456	52.11	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6459	52.13	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6462	52.14	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6464	52.16	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert

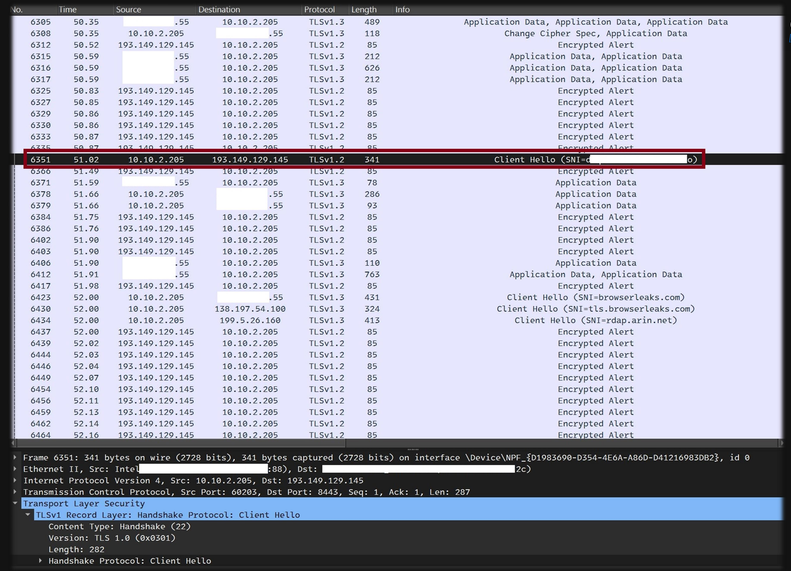
▸ Frame 6351: 341 bytes on wire (2728 bits), 341 bytes captured (2728 bits) on interface \Device\NPF_{D1983690-D354-4E6A-A86D-D41216983DB2}, id 0
▸ Ethernet II, Src: Intel[redacted]:88), Dst: [redacted]2c)
▸ Internet Protocol Version 4, Src: 10.10.2.205, Dst: 193.149.129.145
▸ Transmission Control Protocol, Src Port: 60203, Dst Port: 8443, Seq: 1, Ack: 1, Len: 287
▾ Transport Layer Security
  ▾ TLSv1 Record Layer: Handshake Protocol: Client Hello
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 282
    ▸ Handshake Protocol: Client Hello

(Image 1 – DoH)
[Link]

DNS 状况

基于HTTPS的DNS (DOH)

在此场景中，我们配置了一个带有白名单IP地址和服务器名称指示域名的DOH服务器。该服务器监听443端口和8443端口上的DOH请求，由Nginx充当通用HTTP/超文本传输安全协议流量的网页服务器。两个端口均可通过标准网页浏览器访问，使关联网站能正常加载。然而，当使用流行的DNS客户端（例如YogaDNS）时，DOH请求会被阻止。具体而言，ClientHello消息能成功发送至服务器，但未返回相应的ServerHello消息，导致DOH连接超时。此行为表明存在一种影响DOH握手过程的过滤机制，阻碍了通过超文本传输安全协议成功解析DNS查询。



No.	Time	Source	Destination	Protocol	Length	Info
6305	50.35	.55	10.10.2.205	TLSv1.3	489	Application Data, Application Data, Application Data
6308	50.35	10.10.2.205	.55	TLSv1.3	118	Change Cipher Spec, Application Data
6312	50.52	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6315	50.59	.55	10.10.2.205	TLSv1.3	212	Application Data, Application Data
6316	50.59	.55	10.10.2.205	TLSv1.3	626	Application Data, Application Data
6317	50.59	.55	10.10.2.205	TLSv1.3	212	Application Data, Application Data
6325	50.83	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6327	50.85	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6329	50.86	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6330	50.86	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6333	50.87	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6335	50.87	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6351	51.02	10.10.2.205	193.149.129.145	TLSv1.2	341	Client Hello (SNI=[redacted]o)
6366	51.40	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6371	51.59	.55	10.10.2.205	TLSv1.3	78	Application Data
6378	51.66	10.10.2.205	.55	TLSv1.3	286	Application Data
6379	51.66	10.10.2.205	.55	TLSv1.3	93	Application Data
6384	51.75	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6386	51.76	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6402	51.90	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6403	51.90	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6406	51.90	.55	10.10.2.205	TLSv1.3	110	Application Data
6412	51.91	.55	10.10.2.205	TLSv1.3	763	Application Data, Application Data
6417	51.98	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6423	52.00	10.10.2.205	.55	TLSv1.3	431	Client Hello (SNI=browserleaks.com)
6430	52.00	10.10.2.205	138.197.54.100	TLSv1.3	324	Client Hello (SNI=tls.browserleaks.com)
6434	52.00	10.10.2.205	199.5.26.160	TLSv1.3	413	Client Hello (SNI=rdap.arin.net)
6437	52.00	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6439	52.02	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6444	52.03	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6446	52.04	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6449	52.07	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6454	52.10	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6456	52.11	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6459	52.13	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6462	52.14	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert
6464	52.16	193.149.129.145	10.10.2.205	TLSv1.2	85	Encrypted Alert

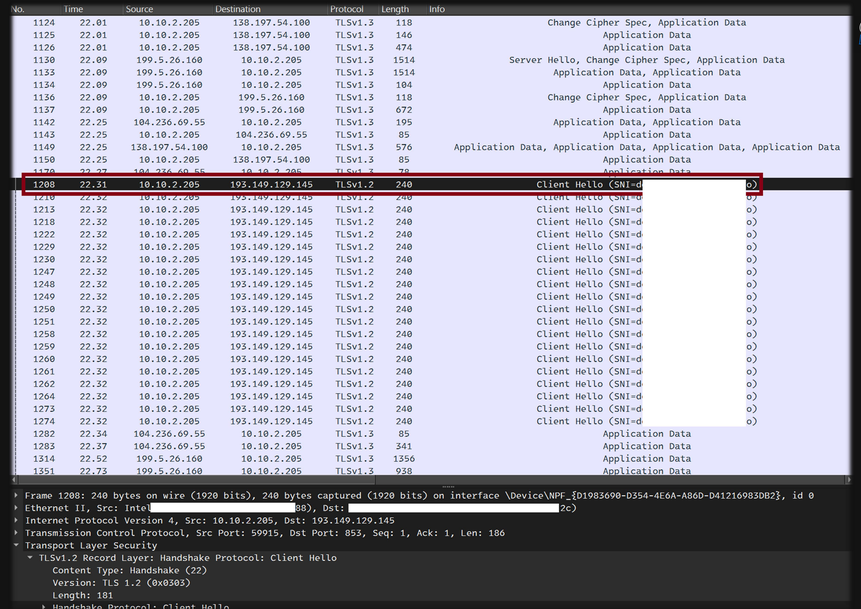
▸ Frame 6351: 341 bytes on wire (2728 bits), 341 bytes captured (2728 bits) on interface \Device\NPF_{D1983690-D354-4E6A-A86D-D41216983DB2}, id 0
▸ Ethernet II, Src: Intel[redacted]:88), Dst: [redacted]2c)
▸ Internet Protocol Version 4, Src: 10.10.2.205, Dst: 193.149.129.145
▸ Transmission Control Protocol, Src Port: 60203, Dst Port: 8443, Seq: 1, Ack: 1, Len: 287
▾ Transport Layer Security
  ▾ TLSv1 Record Layer: Handshake Protocol: Client Hello
      Content Type: Handshake (22)
      Version: TLS 1.0 (0x0301)
      Length: 282
    ▸ Handshake Protocol: Client Hello

（图 1 – DOH）
[链接]

DNS Situation

DNS-over-TLS *(DoT)*

In this scenario, we configure a DoT server to listen on port 853, (which also supports DoQ). A DoT request is sent to the server; however, similar to the previous DoH scenario, the TLS handshake fails. While ClientHello messages are transmitted successfully to the server, no corresponding ServerHello responses are received, resulting in a timeout. This indicates that the TLS negotiation is being blocked or interrupted, preventing the establishment of a secure connection for DNS resolution over TLS.



No.	Time	Source	Destination	Protocol	Length	Info
1124	22.01	10.10.2.205	138.197.54.100	TLSv1.3	118	Change Cipher Spec, Application Data
1125	22.01	10.10.2.205	138.197.54.100	TLSv1.3	146	Application Data
1126	22.01	10.10.2.205	138.197.54.100	TLSv1.3	474	Application Data
1130	22.09	199.5.26.160	10.10.2.205	TLSv1.3	1514	Server Hello, Change Cipher Spec, Application Data
1133	22.09	199.5.26.160	10.10.2.205	TLSv1.3	1514	Application Data, Application Data
1134	22.09	199.5.26.160	10.10.2.205	TLSv1.3	104	Application Data
1136	22.09	10.10.2.205	199.5.26.160	TLSv1.3	118	Change Cipher Spec, Application Data
1137	22.09	10.10.2.205	199.5.26.160	TLSv1.3	672	Application Data
1142	22.25	104.236.69.55	10.10.2.205	TLSv1.3	195	Application Data, Application Data
1143	22.25	10.10.2.205	104.236.69.55	TLSv1.3	85	Application Data
1149	22.25	138.197.54.100	10.10.2.205	TLSv1.3	576	Application Data, Application Data, Application Data, Application Data
1150	22.25	10.10.2.205	138.197.54.100	TLSv1.3	85	Application Data
1208	22.31	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1213	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1218	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1222	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1229	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1230	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1247	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1248	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1249	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1250	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1251	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1258	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1259	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1260	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1261	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1262	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1264	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1273	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1274	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1282	22.34	104.236.69.55	10.10.2.205	TLSv1.3	85	Application Data
1283	22.37	104.236.69.55	10.10.2.205	TLSv1.3	341	Application Data
1314	22.52	199.5.26.160	10.10.2.205	TLSv1.3	1356	Application Data
1351	22.73	199.5.26.160	10.10.2.205	TLSv1.3	938	Application Data

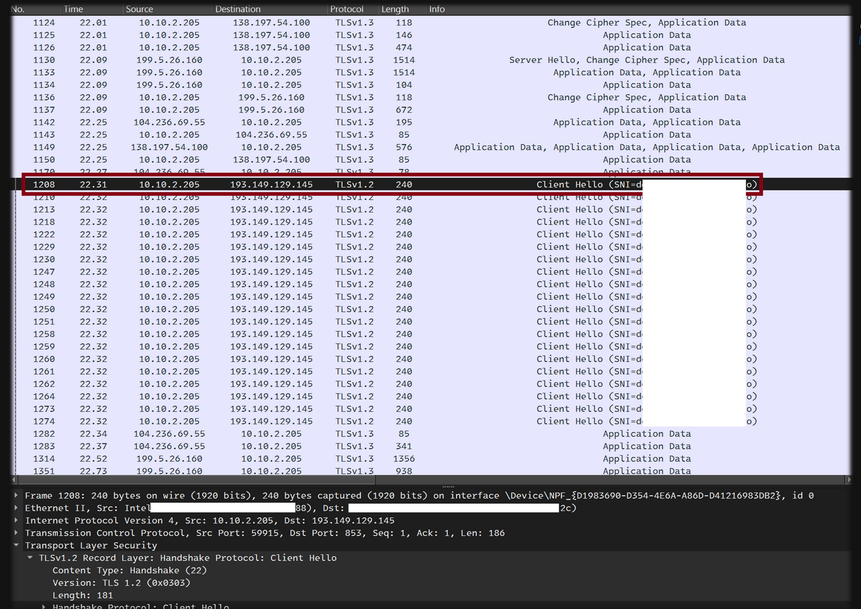
▸ Frame 1208: 240 bytes on wire (1920 bits), 240 bytes captured (1920 bits) on interface \Device\NPF_{D1983690-D354-4E6A-A86D-D41216983DB2}, id 0
▸ Ethernet II, Src: Intel[redacted]88), Dst: [redacted]2c)
▸ Internet Protocol Version 4, Src: 10.10.2.205, Dst: 193.149.129.145
▸ Transmission Control Protocol, Src Port: 59915, Dst Port: 853, Seq: 1, Ack: 1, Len: 186
▾ Transport Layer Security
  ▾ TLSv1.2 Record Layer: Handshake Protocol: Client Hello
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 181
    ▸ Handshake Protocol: Client Hello

(Image 2 – DoT)
[Link]

DNS 状况

基于TLS的DNS *(DoT)*

在此场景中，我们配置了一个DoT服务器监听853端口（该端口也支持DNS-over-QUIC）。一个DoT请求被发送至服务器；然而，与之前的DOH场景类似，TLS握手失败。虽然ClientHello消息成功传输至服务器，但未收到相应的ServerHello响应，导致超时。这表明TLS协商正被阻断或中断，从而阻止了通过TLS建立用于DNS解析的安全连接。



No.	Time	Source	Destination	Protocol	Length	Info
1124	22.01	10.10.2.205	138.197.54.100	TLSv1.3	118	Change Cipher Spec, Application Data
1125	22.01	10.10.2.205	138.197.54.100	TLSv1.3	146	Application Data
1126	22.01	10.10.2.205	138.197.54.100	TLSv1.3	474	Application Data
1130	22.09	199.5.26.160	10.10.2.205	TLSv1.3	1514	Server Hello, Change Cipher Spec, Application Data
1133	22.09	199.5.26.160	10.10.2.205	TLSv1.3	1514	Application Data, Application Data
1134	22.09	199.5.26.160	10.10.2.205	TLSv1.3	104	Application Data
1136	22.09	10.10.2.205	199.5.26.160	TLSv1.3	118	Change Cipher Spec, Application Data
1137	22.09	10.10.2.205	199.5.26.160	TLSv1.3	672	Application Data
1142	22.25	104.236.69.55	10.10.2.205	TLSv1.3	195	Application Data, Application Data
1143	22.25	10.10.2.205	104.236.69.55	TLSv1.3	85	Application Data
1149	22.25	138.197.54.100	10.10.2.205	TLSv1.3	576	Application Data, Application Data, Application Data, Application Data
1150	22.25	10.10.2.205	138.197.54.100	TLSv1.3	85	Application Data
1208	22.31	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1213	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1218	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1222	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1229	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1230	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1247	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1248	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1249	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1250	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1251	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1258	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1259	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1260	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1261	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1262	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1264	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1273	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1274	22.32	10.10.2.205	193.149.129.145	TLSv1.2	240	Client Hello (SNI=d[redacted]o)
1282	22.34	104.236.69.55	10.10.2.205	TLSv1.3	85	Application Data
1283	22.37	104.236.69.55	10.10.2.205	TLSv1.3	341	Application Data
1314	22.52	199.5.26.160	10.10.2.205	TLSv1.3	1356	Application Data
1351	22.73	199.5.26.160	10.10.2.205	TLSv1.3	938	Application Data

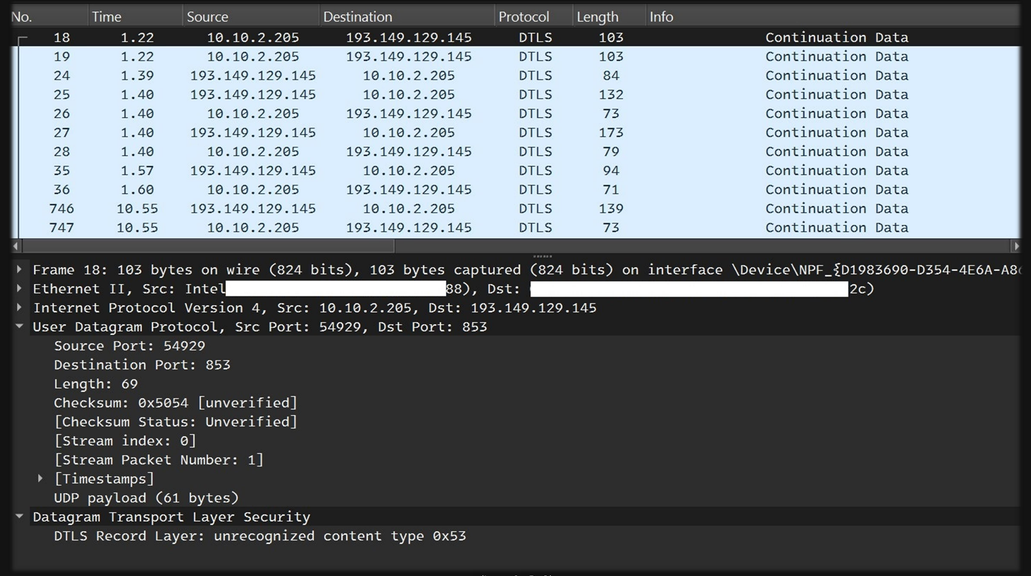
▸ Frame 1208: 240 bytes on wire (1920 bits), 240 bytes captured (1920 bits) on interface \Device\NPF_{D1983690-D354-4E6A-A86D-D41216983DB2}, id 0
▸ Ethernet II, Src: Intel[redacted]88), Dst: [redacted]2c)
▸ Internet Protocol Version 4, Src: 10.10.2.205, Dst: 193.149.129.145
▸ Transmission Control Protocol, Src Port: 59915, Dst Port: 853, Seq: 1, Ack: 1, Len: 186
▾ Transport Layer Security
  ▾ TLSv1.2 Record Layer: Handshake Protocol: Client Hello
      Content Type: Handshake (22)
      Version: TLS 1.2 (0x0303)
      Length: 181
    ▸ Handshake Protocol: Client Hello

(图 2 – DoT)
[链接]

DNS Situation

DNS-over-QUIC *(DoQ)*

In this scenario, DoQ requests are transmitted using the Datagram Transport Layer Security *(DTLS)* protocol, which operates over UDP but provides encryption similar to TLS. Since the DoQ requests are encapsulated in DTLS, they bypass the traditional filtering mechanisms of the IRGFW— as the firewall does not yet recognize this DTLS fingerprint from this specific client. Consequently, the connection is established successfully without disruption, allowing DNS queries to be resolved over QUIC without interference.

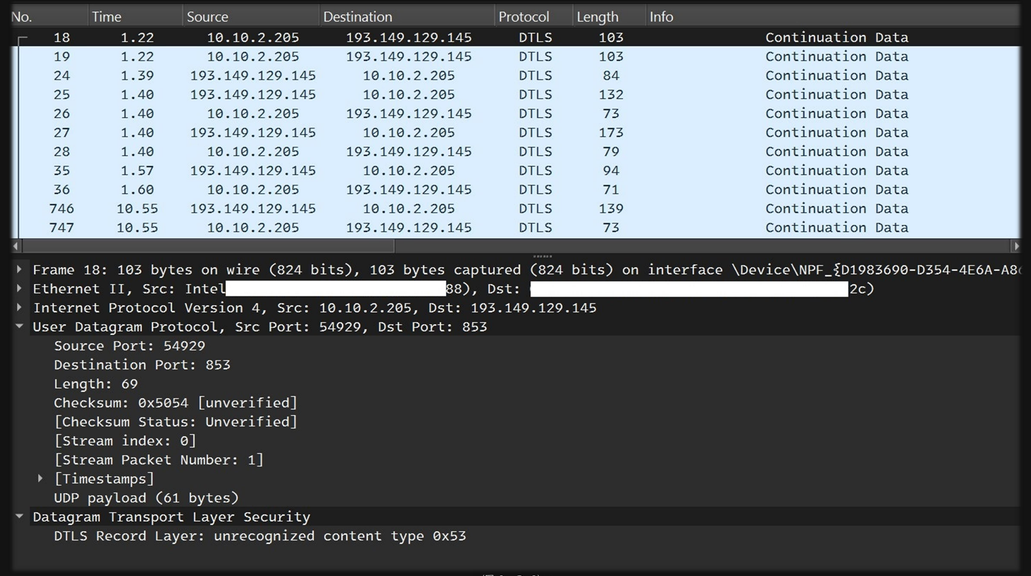


(Image 3 – DoQ)
[Link]

DNS 状况

基于QUIC的DNS *(DoQ)*

在此场景中，DNS-over-QUIC请求使用数据报传输层安全*(DTLS)* 协议进行传输，该协议基于UDP运行，但提供与TLS类似的加密。由于DNS-over-QUIC请求被封装在DTLS中，它们绕过了IRGFW的传统过滤机制——因为防火墙尚未识别出来自此特定客户端的此DTLS指纹。因此，连接得以成功建立且未受干扰，使得DNS查询能够通过QUIC顺畅解析。



(图 3 – DoQ)
[链接]

UDP Situation

UDP 状况

UDP Situation

First, we should separate the regular UDP and the Unidentified UDP *(or Unknown UDP)*. Regular UDP or UDP generally has fingerprints and identification, like Skype, Zoom and Facetime video calls. Also, a normal Wireguard handshake is based on known UDP; thus, it's easily identifiable and fingerprinted.[18][19]

On the other hand, Unknown UDP refers to UDP handshake or traffic that cannot be immediately recognized or matched to a known application or traffic by network monitoring and security tools. This type of traffic is often characterized by its need for more identifiable signatures, making it challenging to determine its purpose or source.

Unknown-UDP Characteristics:

- **Obfuscation:** Used to hide true traffic nature, common in VPN services like obfuscated WireGuard.
- **Proprietary Protocols:** Custom applications using unique communication methods.
- **Encryption:** Encrypted traffic does not match known patterns, often seen in P2P applications and security tools.

UDP 情况

首先，我们需要区分常规 UDP 和未知UDP (或 Unknown UDP)。 常规 UDP 或 UDP 通常具有指纹特征和识别标识，例如 Skype、Zoom 和 FaceTime 视频通话。此外，正常的 WireGuard 握手也基于已知的 UDP；因此，它很容易被识别和标记指纹。[18][19]

另一方面，未知UDP指的是UDP握手或流量，这些流量无法被网络监控和安全工具立即识别或匹配到已知的应用程序或流量。这类流量的特点通常是缺乏更多可识别的特征签名，因此难以确定其用途或来源。

未知UDP特征：

- **混淆：** 用于隐藏真实流量性质，常见于VPN服务中，如混淆WireGuard。
- **专有协议：** 使用独特通信方法的自定义应用程序。
- **加密：** 加密流量与已知模式不匹配，常见于点对点应用程序和安全工具中。

UDP Situation

In I.R. Iran, WireGuard handshakes to foreign IPs are likely blocked through the IRGFW by silently dropping UDP packets when it detects what appears to be a standard WireGuard handshake.[20] In some cases, rate limiting may also be applied to degrade the performance of such connections. This blocking mechanism can potentially be bypassed by adding noise or simulating other handshakes *(or any other known bytes)* before the actual WireGuard handshake. The firewall seems to rely on identifying specific byte patterns in the handshake rather than employing complex regex or deep inspection techniques, which may allow for obfuscation to evade detection.

The firewall appears to buffer or DPI up to 17 KB of UDP *(and TCP as well)* traffic connection per IP:Port combination, analyzing this data to detect WireGuard-specific fingerprints. Beyond this buffer, further traffic is not inspected. The blocking mechanism seems to target high UDP ports (above 1024), while widely used ports like 443 generally remain unaffected. This focus on high ports might make typical WireGuard configurations more susceptible to inspection.

Although UDP is stateless, firewalls often maintain a temporary "connection-like" state for UDP traffic. For example, they associate packets with an IP:Port pair and treat it as a pseudo-session for a limited time *(typically five seconds)*. This state allows the firewall to monitor multiple packets in a flow and identify patterns, such as a WireGuard handshake.

One possible approach to mitigate detection could involve implementing variable-interval port hopping[21], where the port changes at randomized time intervals. This might reduce the likelihood of fingerprinting by introducing unpredictability into traffic patterns. Additionally, altering handshake patterns dynamically and obfuscating payloads may further complicate the firewall's ability to effectively identify and block WireGuard traffic.

Over time, the firewall appears to adapt by recognizing and blocking specific handshake patterns, particularly in cases where repeated traffic is observed between specific IP ranges or data centers. This behaviour suggests the possibility that the firewall can learn and respond to repeated patterns. These observations underline the need for continued experimentation with obfuscation techniques and randomized traffic behaviour to maintain reliable connectivity and potentially outpace the firewall's evolving detection capabilities.

UDP 现状 tion

在伊朗，WireGuard 握手发往国外IP地址的行为很可能通过IRGFW进行阻断，其方式是在检测到看似标准的WireGuard 握手时，静默丢弃UDP数据包。[20] 在某些情况下，还可能应用速率限制以降低此类连接的性能。这种阻断机制有可能通过在实际的WireGuard 握手之前添加噪声或模拟其他握手（或任何其他已知字节）来绕过。防火墙似乎依赖于识别握手过程中的特定字节模式，而非采用复杂的正则表达式或深度检测技术，这或许为通过混淆来规避检测提供了可能。

防火墙似乎会对每个IP:端口组合缓冲或进行深度包检测，最多处理17 KB的UDP（TCP亦然） 流量连接，并分析这些数据以检测WireGuard特有指纹。超出此缓冲区的后续流量则不再受检。其阻断机制似乎针对高位UDP端口（1024以上），而443等广泛使用的端口通常不受影响。这种对高位端口的关注可能使典型的WireGuard配置更容易受到检测。

尽管UDP是无状态的，但防火墙通常会对UDP流量维持一个临时的“类连接”状态。例如，它们将数据包与IP:端口对关联，并在有限时间内（通常为五秒） 将其视为一个伪会话。这种状态使得防火墙能够监控流中的多个数据包并识别模式，例如WireGuard握手。

一种可能的缓解检测方法是实施变间隔端口跳变[21]，，即在随机时间间隔内变更端口。这可能通过引入流量模式的不可预测性来降低指纹识别的可能性。此外，动态改变握手模式和进行载荷混淆可能会进一步干扰防火墙有效识别和阻断WireGuard流量的能力。

随着时间的推移，防火墙似乎能够通过识别并阻断特定的握手模式来适应，尤其是在观察到特定IP段或数据中心之间存在重复流量的情况下。这种行为表明，防火墙可能具备学习和应对重复模式的能力。这些观察结果强调了需要持续试验混淆技术和随机化流量行为，以维持可靠的连接，并可能超越防火墙不断演进的检测能力。

UDP Situation

In this scenario, we observe that a standard WireGuard handshake is consistently being blocked by the firewall at the packet level, occurring at intervals of every 5 seconds. Notably, this 5-second interval is not part of a KeepAlive mechanism, as it's explicitly disabled in this configuration. Despite the firewall's targeted blocking of the WireGuard handshake, there are no ICMP error messages observed, and the destination IP address continues to respond to ping requests without issue. This behaviour indicates that while the firewall is specifically filtering out WireGuard handshake packets, it does not interfere with general network traffic such as ICMP, ensuring basic connectivity checks remain operational.

No.	Time	Source	Destination	Protocol	Length	Info	Dest Port
7657	9.33	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xAB8758DE	54571
8523	14.33	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xF55CBEA3	54571
9241	19.33	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xD2F7CD1F	54571
12590	24.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xB11AB00C	54571
13709	29.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xB40F9646	54571
17883	34.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xE3F156D5	54571
21228	39.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0x352C1132	54571
22784	44.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0x410EF8D6	54571
26031	49.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xD8D20074	54571
28588	54.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xBF418730	54571
30795	59.35	62.	45.13	WireGuard	190	Handshake Initiation, sender=0x1A46B998	54571
32225	64.35	62.	45.13	WireGuard	190	Handshake Initiation, sender=0x35928FC8	54571
35004	69.35	62.	45.13	WireGuard	190	Handshake Initiation, sender=0x9E962A9A	54571

(Image 4 – Normal Wireguard)
[Link]

UDP 现状

在此场景中，我们观察到，防火墙在数据包层面持续阻断标准的WireGuard握手，其发生间隔为每5秒一次。值得注意的是，这个5秒间隔并非KeepAlive机制的一部分，因为在此配置中已明确禁用了该机制。尽管防火墙有针对性地阻断了WireGuard握手，但并未观察到ICMP错误消息，且目标IP地址对ping请求的响应依然正常。这种行为表明，防火墙虽然专门过滤WireGuard握手数据包，但并未干扰如ICMP等常规网络流量，从而确保了基本的连通性检查仍可正常运行。

No.	Time	Source	Destination	Protocol	Length	Info	Dest Port
7657	9.33	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xAB8758DE	54571
8523	14.33	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xF55CBEA3	54571
9241	19.33	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xD2F7CD1F	54571
12590	24.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xB11AB00C	54571
13709	29.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xB40F9646	54571
17883	34.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xE3F156D5	54571
21228	39.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0x352C1132	54571
22784	44.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0x410EF8D6	54571
26031	49.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xD8D20074	54571
28588	54.34	62.	45.13	WireGuard	190	Handshake Initiation, sender=0xBF418730	54571
30795	59.35	62.	45.13	WireGuard	190	Handshake Initiation, sender=0x1A46B998	54571
32225	64.35	62.	45.13	WireGuard	190	Handshake Initiation, sender=0x35928FC8	54571
35004	69.35	62.	45.13	WireGuard	190	Handshake Initiation, sender=0x9E962A9A	54571

(图 4 – 正常的 WireGuard)
[链接]

UDP Situation

In this case study, we observe the behaviour of a firewall that actively blocks both the WireGuard handshake and ICMP communication to the destination IP.

Initially, we send minimal "junk" and "noise" packets to simulate traffic before attempting the WireGuard handshake. These packets accumulate in the firewall buffer up to packet number 9472 without triggering any response. Following this, we initiate a QUIC handshake, which successfully bypasses the firewall buffer by sending similar pre-handshake noise and junk packets. This behaviour demonstrates the firewall's ability to handle and scrutinize WireGuard traffic differently compared to QUIC.

The critical observation here is that when the firewall blocks the WireGuard handshake, it also prevents ICMP *(ping)* communication to the destination IP address. This simultaneous blocking indicates a stringent firewall policy that disrupts both VPN handshakes and ICMP traffic.

Importantly, the ICMP connectivity check here is independent of the WireGuard protocol's usual KeepAlive mechanism. Instead, a separate program is employed to test the ICMP connection *(although in Wireshark it's written "Port unreachable", but it results in timeouts in normal ping command)*, isolating the behaviour of the firewall towards diagnostic traffic alongside VPN handshakes.

No.	Time	Source	Destination	Protocol	Length	Info	Dest Port
4163	3.33	45.	6	WireGuard	190	Handshake Initiation, sender=0xF787EFB4	58040
4164	3.33	6	45	ICMP	218	Destination unreachable (Port unreachable)	
6152	8.44		6	WireGuard	190	Handshake Initiation, sender=0xD1BB7F12	58040
6153	8.44	6	45	ICMP	218	Destination unreachable (Port unreachable)	
6875	13.68		6	WireGuard	190	Handshake Initiation, sender=0xD373E425	58040
6876	13.68	6	45	ICMP	218	Destination unreachable (Port unreachable)	
7630	18.90		6	WireGuard	190	Handshake Initiation, sender=0x585838EB	58040
7631	18.90	6	45	ICMP	218	Destination unreachable (Port unreachable)	
8653	24.22		6	WireGuard	190	Handshake Initiation, sender=0xF8CBB029	58040
8654	24.22	6	45	ICMP	218	Destination unreachable (Port unreachable)	
9472	27.40	6	45	QUIC	65	Handshake, DCID=c55c844ce8700531[Malformed Packet]	35197
9474	27.41	6	45	QUIC	65	Handshake, DCID=c55c844ce8700531[Malformed Packet]	35197
9476	27.41	6	45	QUIC	67	Protected Payload (KP0)	35197
9478	27.42	6	45	QUIC	67	Protected Payload (KP0)	35197
9479	27.43	6	45	QUIC	69	Protected Payload (KP0)	35197
9481	27.44	6	45	QUIC	70	Protected Payload (KP0)	35197
9482	27.44	6	45	QUIC	65	Handshake, DCID=b6d42c6c7177df70[Malformed Packet]	35197
9484	27.44	6	45	QUIC	68	Protected Payload (KP0)	35197
9485	27.44	6	45	QUIC	65	Handshake, DCID=c55c844ce8700531[Malformed Packet]	35197
9486	27.44	6	45	QUIC	65	Handshake, DCID=c55c844ce8700531[Malformed Packet]	35197
9487	27.44	6	45	QUIC	69	Protected Payload (KP0)	35197

(Image 5 – Modified Wireguard)
[Link]

UDP 场景

在本案例研究中，我们观察到一个防火墙的行为，该防火墙会主动阻断 WireGuard 握手和对目标IP的 ICMP 通信。

最初，我们在尝试 WireGuard 握手之前，发送了最少的“垃圾”和“噪音”数据包来模拟流量。这些数据包在防火墙缓冲区中累积，直到数据包编号 9472，都没有触发任何响应。随后，我们发起了一个 QUIC 握手，它成功地绕过了

防火墙缓冲区，方法是发送类似的握手前噪音和垃圾数据包。这一行为表明防火墙能够以不同于 QUIC 的方式处理和

审查 WireGuard 流量。

这里的关键观察是，当防火墙阻断 WireGuard 握手时，它同时也阻止了到目标IP地址的 ICMP *(ping)* 通信。这种同时阻断的行为表明防火墙采用了严格的策略，会同时中断 VPN 握手和 ICMP 流量。

重要的是，这里的 ICMP 连通性检查独立于 WireGuard 协议通常的 KeepAlive 机制。相反，我们使用一个单独的程序来测试 ICMP 连接（虽然在 Wireshark 中它被记录为“端口不可达”，但在普通的 ping 命令中会导致超时），以此将防火墙对诊断流量的行为与 VPN 握手行为分离开来。

No.	Time	Source	Destination	Protocol	Length	Info	Dest Port
4163	3.33	45.	6	WireGuard	190	Handshake Initiation, sender=0xF787EFB4	58040
4164	3.33	6	45	ICMP	218	Destination unreachable (Port unreachable)	
6152	8.44		6	WireGuard	190	Handshake Initiation, sender=0xD1BB7F12	58040
6153	8.44	6	45	ICMP	218	Destination unreachable (Port unreachable)	
6875	13.68		6	WireGuard	190	Handshake Initiation, sender=0xD373E425	58040
6876	13.68	6	45	ICMP	218	Destination unreachable (Port unreachable)	
7630	18.90		6	WireGuard	190	Handshake Initiation, sender=0x585838EB	58040
7631	18.90	6	45	ICMP	218	Destination unreachable (Port unreachable)	
8653	24.22		6	WireGuard	190	Handshake Initiation, sender=0xF8CBB029	58040
8654	24.22	6	45	ICMP	218	Destination unreachable (Port unreachable)	
9472	27.40	6	45	QUIC	65	Handshake, DCID=c55c844ce8700531[Malformed Packet]	35197
9474	27.41	6	45	QUIC	65	Handshake, DCID=c55c844ce8700531[Malformed Packet]	35197
9476	27.41	6	45	QUIC	67	Protected Payload (KP0)	35197
9478	27.42	6	45	QUIC	67	Protected Payload (KP0)	35197
9479	27.43	6	45	QUIC	69	Protected Payload (KP0)	35197
9481	27.44	6	45	QUIC	70	Protected Payload (KP0)	35197
9482	27.44	6	45	QUIC	65	Handshake, DCID=b6d42c6c7177df70[Malformed Packet]	35197
9484	27.44	6	45	QUIC	68	Protected Payload (KP0)	35197
9485	27.44	6	45	QUIC	65	Handshake, DCID=c55c844ce8700531[Malformed Packet]	35197
9486	27.44	6	45	QUIC	65	Handshake, DCID=c55c844ce8700531[Malformed Packet]	35197
9487	27.44	6	45	QUIC	69	Protected Payload (KP0)	35197

(图5 – 修改后的WireGuard)
[链接]

QUIC Situation

In Iran, the deployment and utilization of QUIC and HTTP/3 protocols face significant challenges due to stringent government filtering policies. Although HTTP/3 has been partially adopted, its performance is severely throttled, leading to slower speeds than HTTP/2. QUIC handshake/traffic to many international data centers is often blocked, impacting performance inconsistently depending on the destination IP range.

Users attempting to circumvent these restrictions with tools that use QUIC as a tunnelling proxy but experience varying success, as the effectiveness of these tools heavily relies on the specific foreign IP addresses being accessed. Consequently, while these proxy tools can sometimes provide faster and more secure connections, their reliability is significantly based on Iran's pervasive and unpredictable filtering practices.[22]

In addition to these limitations, it has been observed that QUIC traffic to certain foreign IP ranges may be blocked selectively within the same data center, where some IPs remain accessible while others are entirely restricted.[23] This filtering appears to target QUIC handshakes, with specific byte patterns being flagged and blocked after repeated use. For example, frequent QUIC handshakes from Iranian IPs to a particular foreign IP can lead to a complete block on that connection. The filtering mechanism also demonstrates an ability to adapt and block high-frequency QUIC traffic originating from specific IPs after reaching a threshold of traffic volume or repeated patterns. Furthermore, QUIC traffic to Cloudflare has recently declined significantly, potentially indicating targeted restrictions against its widespread use. [24]

To address these challenges, tools relying on QUIC need to introduce dynamic handshake and traffic obfuscation mechanisms to evade identification by Iranian DPI systems. Adjusting handshake patterns or introducing randomness into QUIC traffic flows may help improve their effectiveness against these restrictions.

QUIC 现状tion

在伊朗，由于严格的政府过滤政策，QUIC 和 HTTP/3 协议的部署和使用面临重大挑战。尽管 HTTP/3 已被部分采用，但其性能受到严重限制，导致速度比 HTTP/2 更慢。通往许多国际数据中心的 QUIC 握手/流量经常被阻断，根据目标IP范围的不同，对性能的影响也不一致。

用户尝试使用将 QUIC 作为隧道代理的工具来规避这些限制，但成功率不一，因为这些工具的有效性在很大程度上取决于所访问的特定外国 IP地址。因此，虽然这些代理工具有时能提供更快、更安全的连接，但其可靠性在很大程度上取决于伊朗普遍存在且不可预测的过滤做法。[22]

除了这些限制之外，还观察到，在同一数据中心内，对某些外国 IP段的 QUIC 流量可能会被选择性屏蔽，其中一些 IP地址仍然可以访问，而另一些则完全被限制。[23]这种过滤似乎针对 QUIC 握手，特定的字节模式在重复使用后会被标记并阻止。例如，从伊朗 IP地址到特定外国 IP地址的频繁 QUIC 握手可能导致该连接被完全阻断。该过滤机制还表现出一种适应能力，在来自特定 IP地址的高频 QUIC 流量达到一定阈值或出现重复模式后，能够对其进行阻断。此外，最近发往 Cloudflare 的 QUIC 流量已显著下降，这可能表明针对其广泛使用采取了针对性限制。[24]

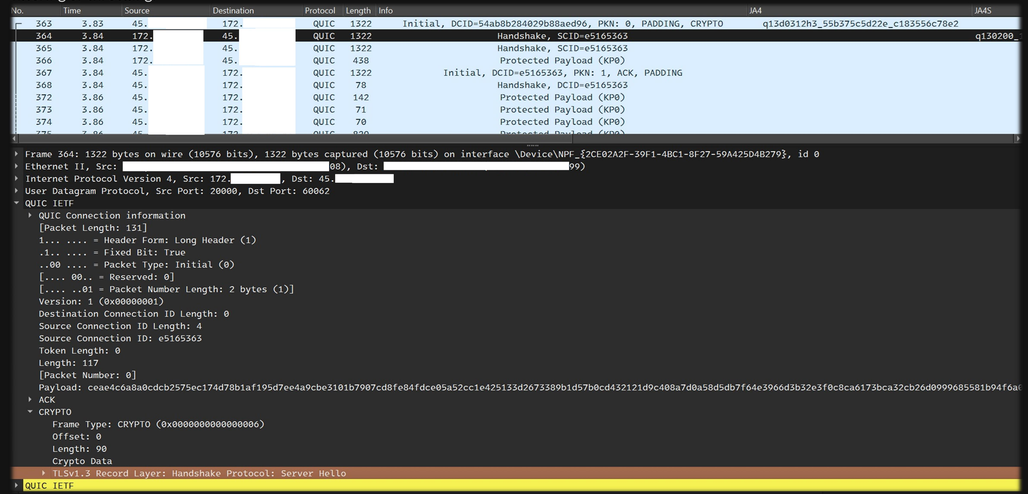
为应对这些挑战，依赖QUIC的工具需要引入动态握手和流量混淆机制，以规避伊朗深度包检测系统的识别。调整握手模式或在QUIC流量中引入随机性，可能有助于提升其应对这些限制措施的有效性。

QUIC Situation

In this scenario, we tested connectivity to a domain with a specific destination IP where UDP and QUIC traffic are unrestricted. The handshake process was observed in Wireshark, confirming the following sequence:

1. The ClientHello was sent from the client.
2. The ServerHello was received, completing the QUIC handshake.
3. Application-layer payloads were successfully exchanged without any interruptions.

The target server is running a Nginx with HTTP/3 *(QUIC)* support enabled by default. Both curl with HTTP/3 and Hysteria2 were used to validate connectivity and handshake consistency. The results confirm that this domain and IP are fully operational for QUIC traffic, with no evidence of filtering or throttling.



(Image 6 – QUIC Handshake OK)
[Link]

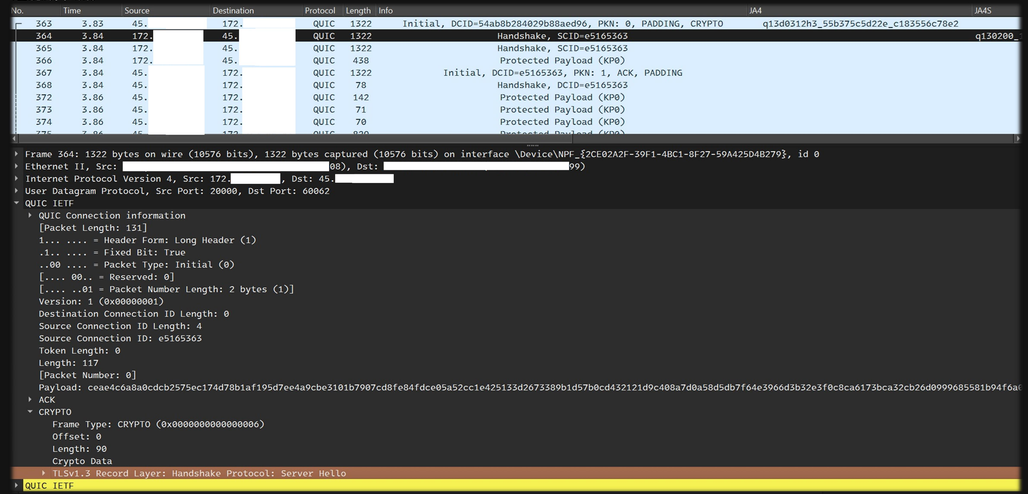
QUIC 状况

在此场景中，我们测试了连接到一个具有特定目标IP的域名，该域名处UDP和QUIC流量不受限制。我们在Wireshark中观察到了握手过程，并确认了以下序列：

1. ClientHello 已由客户端发送。
2. ServerHello 已接收，QUIC 握手完成。
3. 应用层载荷已成功交换，未发生任何中断。

目标服务器运行着默认启用了 HTTP/3 *(QUIC)* 支持的 Nginx。我们同时使用了支持 HTTP/3 的 curl 和 Hysteria2 来验证连通性与握手一致性。结果证实，此域名和IP地址完全支持 QUIC 流量，且没有迹象表明

过滤或节流。



（图 6 – QUIC 握手正常）
[Link]

QUIC Situation

In this specific scenario, the destination IP address can be connected with an obfuscated Wireguard, indicating UDP traffic is not blocked to this IP. Then we attempt to initiate a QUIC handshake with a whitelisted domain in Iran. Despite UDP traffic successfully reaching the destination IP, the QUIC handshake fails to complete.

When analyzing the traffic in Wireshark, we observe the client sending the ClientHello. However, all subsequent ClientHello packets are retransmissions, indicating that the client is not receiving a response from the server. No ServerHello is observed or received by the client, which confirms that the handshake is being disrupted after the initial client transmission.

This pattern highlights a filtering mechanism that allows UDP packets through but actively blocks the QUIC handshake process at a protocol-specific level. Such targeted behaviour underscores the sophistication of the filtering system and the need for advanced obfuscation techniques to bypass these restrictions. However, when testing with a non-blocked domain, the blockage consists.

No.	Time	Source	Destination	Protocol	Length	Info	JA4
9354	4.32	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 0, PADDING, CRYPTO	q13d0312h3_55b375c5d22e_c183556c78e
9381	4.52	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 1, PADDING, CRYPTO	
9382	4.52	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 2, PADDING, CRYPTO	
10095	4.93	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 3, PADDING, CRYPTO	
10096	4.93	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 4, PADDING, CRYPTO	
11007	5.73	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 5, PADDING, CRYPTO	
11008	5.73	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 6, PADDING, CRYPTO	
11544	7.33	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 7, PADDING, CRYPTO	
11545	7.33	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 8, PADDING, CRYPTO	

```
Frame 9354: 1322 bytes on wire (10576 bits), 1322 bytes captured (10576 bits) on interface ens192, id 0
Ethernet II, Src: VMware [redacted]f9), Dst: [redacted]8c)
Internet Protocol Version 4, Src: 6[redacted], Dst: 172.232.44.81
User Datagram Protocol, Src Port: 57951, Dst Port: 20000
QUIC IETF
  QUIC Connection information
  [Packet Length: 1280]
  1... .... = Header Form: Long Header (1)
  .1.. .... = Fixed Bit: True
  ..00 .... = Packet Type: Initial (0)
  [.... 00.. = Reserved: 0]
  [.... ..01 = Packet Number Length: 2 bytes (1)]
  Version: 1 (0x00000001)
  Destination Connection ID Length: 15
  Destination Connection ID: 62b0b3c512a1a4601e9b1b0a00575d
  Source Connection ID Length: 0
  Token Length: 0
  Length: 1255
  [Packet Number: 0]
  Payload [truncated]: 661240ed17a4ae52719087a84dc55ee0f23ebc7a8a2d1a8dbe64014caf5ce5b6bb78fc1950358039810 0bc952f3ddeb525da2a6c2058fb50083ffb2b22e4ae18632219b3079fefe78b740c8c395ceef
  PADDING Length: 958
  CRYPTO
    Frame Type: CRYPTO (0x0000000000000006)
    Offset: 0
    Length: 275
    Crypto Data
  TLSv1.3 Record Layer: Handshake Protocol: Client Hello
JA4 Fingerprint
```

(Image 7 - QUIC Handshake NotOK)
[Link]

QUIC 现状

在此特定场景中，目标IP地址可以通过混淆WireGuard进行连接，这表明UDP流量未被阻挡至此IP。随后，我们尝试与伊朗的一个白名单域名发起 QUIC 握手。尽管UDP流量成功抵达目标IP，但 QUIC 握手未能完成。

在 Wireshark 中分析流量时，我们观察到客户端发送了 ClientHello。然而，所有后续的 ClientHello 数据包都是重传，这表明客户端没有收到服务器的响应。客户端没有观测到或收到任何 ServerHello，这证实了握手在客户端初始传输后即被中断。

这种模式突显了一种过滤机制，它允许 UDP数据包 通过，但在协议特定层面主动阻断了 QUIC 握手 过程。这种有针对性的行为凸显了过滤系统的复杂性，以及需要采用高级 混淆技术 来绕过这些限制。然而，当使用未被屏蔽的域名进行测试时，阻断依然存在。

No.	Time	Source	Destination	Protocol	Length	Info	JA4
9354	4.32	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 0, PADDING, CRYPTO	q13d0312h3_55b375c5d22e_c183556c78e
9381	4.52	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 1, PADDING, CRYPTO	
9382	4.52	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 2, PADDING, CRYPTO	
10095	4.93	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 3, PADDING, CRYPTO	
10096	4.93	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 4, PADDING, CRYPTO	
11007	5.73	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 5, PADDING, CRYPTO	
11008	5.73	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 6, PADDING, CRYPTO	
11544	7.33	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 7, PADDING, CRYPTO	
11545	7.33	6	172.232.44.81	QUIC	1322	Initial, DCID=62b0b3c512a1a4601e9b1b0a00575d, PKN: 8, PADDING, CRYPTO	

```
Frame 9354: 1322 bytes on wire (10576 bits), 1322 bytes captured (10576 bits) on interface ens192, id 0
Ethernet II, Src: VMware [redacted]f9), Dst: [redacted]8c)
Internet Protocol Version 4, Src: 6[redacted], Dst: 172.232.44.81
User Datagram Protocol, Src Port: 57951, Dst Port: 20000
QUIC IETF
  QUIC Connection information
  [Packet Length: 1280]
  1... .... = Header Form: Long Header (1)
  .1.. .... = Fixed Bit: True
  ..00 .... = Packet Type: Initial (0)
  [.... 00.. = Reserved: 0]
  [.... ..01 = Packet Number Length: 2 bytes (1)]
  Version: 1 (0x00000001)
  Destination Connection ID Length: 15
  Destination Connection ID: 62b0b3c512a1a4601e9b1b0a00575d
  Source Connection ID Length: 0
  Token Length: 0
  Length: 1255
  [Packet Number: 0]
  Payload [truncated]: 661240ed17a4ae52719087a84dc55ee0f23ebc7a8a2d1a8dbe64014caf5ce5b6bb78fc1950358039810 0bc952f3ddeb525da2a6c2058fb50083ffb2b22e4ae18632219b3079fefe78b740c8c395ceef
  PADDING Length: 958
  CRYPTO
    Frame Type: CRYPTO (0x0000000000000006)
    Offset: 0
    Length: 275
    Crypto Data
  TLSv1.3 Record Layer: Handshake Protocol: Client Hello
JA4 Fingerprint
```

(图 7 – QUIC 握手 失败)
[Link]

IP Address Situation

IP地址状况Situation

IP Address Situation

IRGFW has three lists: **WhiteList**, **GrayList**, and **BlackList**. The history of an IP address is a significant factor in this matter.

- **White IP:** The IP should be from a relatively unknown data center; no one has used it for VPN/Proxy for the last three months. *(Or more!)* It should also be manually whitelisted on ISP databases. Thus, sometimes, an extremely unknown data center IP address could be blocked faster because it has not been whitelisted in the IRGFW database.

- **Gray IP:** The IRGFW designates specific IP addresses as "gray" when suspected of being used for VPN or proxy purposes but lacking sufficient evidence to warrant an immediate block. These IP addresses, often belonging to major data centers, are subject to periodic traffic analysis and data collection, likely contributing to limited upload speeds and high jitter. By default, the IRGFW categorizes an IP address as gray and continuously monitors it, gathering traffic samples. Based on the collected data and observed usage patterns over time, the IRGFW will decide whether to block the IP address permanently.

- **Black IP:** After analyzing sufficient data from Gray IPs, the IRGFW may escalate an IP address to Black IP status. This results in complete or partial blockage using different patterns:

    - **TIC and TCI:** These patterns block all types of traffic to the IP, including ICMP, SSH, TLS*(v1.0–v1.3)*, HTTP, and others.

    - **MCI:** When the firewall is active, it explicitly blocks the ServerHello phase of the TLS handshake, disrupting secure connections.

    - **MTN:** This pattern inconsistently blocks traffic, sometimes targeting SSH and TLS protocols and only TLS.

These strategies are part of the IRGFW's comprehensive approach to controlling and limiting VPNs and proxies within the country.

IPv6 Situation

IPv6 has not yet reached mainstream adoption across most operators. However, it is available for mobile users on networks like MCI and MTN, provided the user manually enables it. On these IPv6 addresses, DPI is typically disabled by default, making them less scrutinized. Nevertheless, the fundamental IRGFW rules—such as categorizing IP addresses into WhiteList, GrayList, and BlackList—still apply, though with less stringent enforcement compared to IPv4.

IP地址状况

IRGFW有三个名单：**白名单**、**灰名单**和**黑名单**。一个IP地址的历史记录是此事中的一个重要因素。

- **白IP：**该IP应来自一个相对不知名的数据中心；在过去三个月内，没有人将其用于VPN/代理。（或更久！）它还应已在ISP数据库中手动加入白名单。因此，有时一个极其不知名的数据中心IP地址可能会更快被封锁，因为它尚未在IRGFW数据库中被列入白名单。

- **灰IP：**当IRGFW怀疑特定IP地址被用于VPN或代理目的，但缺乏足够证据来立即封锁时，会将其指定为“灰色”。这些IP地址通常属于大型数据中心，会受到定期的流量分析和数据

    收集，这很可能导致上传速度受限和抖动较高。默认情况下，IRGFW会将一个IP地址归类为灰色，并持续
    监控它，收集流量样本。根据收集到的数据以及随时间观察到的使用模式，IRGFW将决定是否将其封锁。
    该IP地址永久性地。

- **黑IP：**在分析足够多的灰IP数据后，IRGFW可能会将某个IP地址升级为黑IP状态。这将导致使用不同模式进行完全或部分阻断：

    - **TIC与TCI：**这些模式会阻断发往该IP的所有类型流量，包括ICMP、SSH、TLS*(v1.0–v1.3)*、 HTTP以及其他协议。

    - **MCI：**当防火墙激活时，它会明确阻断TLS握手的ServerHello阶段，从而破坏安全连接。

    - **MTN：**此模式会不一致地阻断流量，有时针对SSH和TLS协议，有时仅针对TLS。

这些策略是IRGFW在国内控制和限制VPN与代理的综合性方法的一部分。

IPv6 现状

IPv6尚未在大多数运营商中达到主流采用。然而，对于MCI和MTN等网络的移动用户来说，只要用户手动启用，即可使用IPv6。在这些IPv6地址上，深度包检测通常默认处于禁用状态，因此受到的审查较少。尽管如此，基本的IRGFW规则——例如将IP地址分类为白名单、灰名单和黑名单——仍然适用，只是与IPv4相比，执行力度不那么严格。

Time Pattern

We have identified specific patterns related to block timings. The TIC primary firewall synchronizes daily at 6:00 AM and 12:00 PM (UTC+03:30). Consequently, when referring to a TIC firewall test, it implies that the TIC will block the servers exclusively during these synchronization periods. In contrast, the MCI firewall may block an IP address or domain at any time during the day, following its time-based patterns.

For clarity, "moderate" traffic is defined as symmetrical traffic of 100 Mbps on the server.

- **Time pattern 1:** 4h - 1d - 4d - 1w - 40d
- **Time pattern 2:** 1h - 4h - 2d - 2w - 40d

1. **Time Pattern 1:** Set up a proxy server with Xray-core like VLESS-TCP-Reality*(Vision) (Combination is unimportant)*. Flow some moderate traffic on it. If the IP address didn't block after 4 hours, it will likely work for 1 day *(The TIC firewall test)*. If the IP has not been blocked, it will likely work for 4 days *(Another TIC firewall test)*. And if it is not blocked yet, it will probably go for 1 week *(The MCI firewall test)*. If passed, it would likely work for 40 days, but after this period, there were so many random factors that we couldn't find any patterns.

2. **Time Pattern 2:** Set up a proxy server like the above. Flow some moderate traffic on it. If the IP address didn't block after 1 hour, it will likely work for 4 hours. If the IP has not been blocked, it will likely work for 2 days *(TIC firewall test)*. And if it is not blocked yet, it will probably go for 2 weeks *(The MCI firewall test)*. If passed, it would likely work for 40 days, but after this period, there were so many random factors that we couldn't find any patterns.

When an IP address is Graylisted, it will never go to Whitelist again! So, when IRGFW throttles the IP address, we can say the IP is gray, and when the IP is blocked, it is in BlackList. Most of the time, after 40 days, the IP will be unblocked again, but now the IP is gray and may have some limitations on DL/UL speed and high jitter in some cases. This pattern will occur for every foreign IP address range, primarily for famous data centers and hosting services that can be used for VPN/Proxy servers; or too infamous ASNs that are not in the default firewalls whitelists.

This "gray-listing" can be used for protocols as well. As we discussed, HTTP3/QUIC and UDP are Graylisted by default unless the client's fingerprint *(e.g. User-Agent in HTTP handshakes or UTLS in Client-Hello)* does not match any of the firewall databases and the destination IP address has not been graylisted yet.

时间模式

我们已经识别出与阻断时间相关的特定模式。TIC主防火墙每天在UTC+03:30的上午6:00和中午12:00进行同步。因此，当提及TIC防火墙测试时，意味着TIC将仅在上述同步时段内阻断服务器。相比之下，MCI防火墙则可能在其基于时间的模式驱动下，在一天中的任何时间阻断某个IP地址或域名。

为明确起见，“中等流量”定义为服务器上100 Mbps的对称流量。

- **时间模式1：** 4小时 - 1天 - 4天 - 1周 - 40天
- **时间模式2：** 1小时 - 4小时 - 2天 - 2周 - 40天

1. **时间模式1：** 使用Xray-core设置一个代理服务器，例如VLess-TCP-Reality(Vision)（具体组合不重要）。 在其上传输一些中等流量。如果IP地址在4小时后未被屏蔽，它很可能能工作1天（TIC防火墙测试）。如果IP仍未屏蔽，它很可能能工作4天（另一次TIC防火墙测试）。如果此时仍未屏蔽，它可能可以持续1周（MCI防火墙测试）。如果通过，它很可能能工作40天，但在此之后，随机因素过多，我们未能发现任何规律。

2. **时间模式2：** 如上所述设置一个代理服务器。 在其上产生一些中等流量。如果该IP地址在1小时后未被阻断，则很可能能工作4小时。如果该IP仍未受阻，则很可能能工作2天(TIC防火墙测试)。如果至此仍未受阻，则可能持续2周(MCI防火墙测试)。如果通过，则很可能能工作40天，但在此期限之后，存在太多随机因素，我们未能发现任何规律。

当一个IP地址被列入灰名单后，它将永远不会再进入白名单！因此，当IRGFW限制该IP地址时，我们可以说该IP是灰色的；而当该IP被阻断时，它便进入了黑名单。大多数情况下，40天后该IP会再次被解封，但此时IP已处于灰色状态，在某些情况下可能会对下载/上传速度有所限制并出现高抖动。这种模式将出现在每一个境外IP地址段，主要针对那些可用于VPN/代理服务器的知名数据中心和托管服务；或者那些因不在默认防火墙白名单中而声名狼藉的ASN。

这种“灰名单”机制同样适用于协议。正如我们讨论过的，HTTP3/QUIC和UDP协议默认处于灰名单中，除非客户端的指纹(例如HTTP握手时的User-Agent或Client-Hello中的UTLS)与防火墙数据库中的任何记录都不匹配，且目标IP地址尚未被列入灰名单。

Active-Probes

活跃探测

Active-Probes

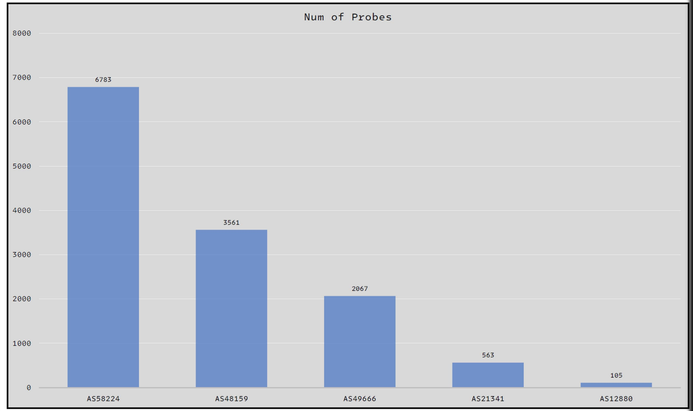
IRGFW “had” an active-probing system back in September 2023, and we extracted some of the used IP addresses.[26] Some of our test servers were even impacted by various DDoS attacks, which maxed out the server CPU usage.[27] These IP addresses were handled in these tests on the server using Xray-core.

But from early January 2024, IRGFW no longer uses Active-Probes. There are no signs of probes on any servers, and we guess they upgraded the IRGFW to be more precise and optimized on the passive side, as we'll discuss in this report.

In the image below we recorded most of the IP CIDRs that we detected as Probers. Our test method is inspired by gfw.report team.[28]

In the following pages, we cumulated all of our Active-probe tests into **three types**. Most tests were done with Xray-core and others with various cores and methods in Iran.

Iran's GFW Active Probing Test			
AS Number	Org/Name	Num of Probes	IP Ranges
AS58224	*Iran Telecommunication Company PJS (TCI)*	**6783**	80.191.0.0/16 *(80.191.69.0/24)* *(80.191.64.0/24)* 78.38.0.0/16 78.39.0.0/16 217.218.0.0/16 *(217.218.80.0/16)* 2.187.0.0/16
AS48159	*Telecommunication Infrastructure Company (TIC-AS)*	**3561**	2.189.42.0/24 2.184.0.0/16
AS49666	*Telecommunication Infrastructure Company (TIC-GW-AS)*	**2067**	2.188.28.0/24
AS21341	*Soroush Rasanheh Company Ltd (SINET-AS)*	**563**	62.220.121.0/24
AS12880	*Information Technology Company (ITC) (DCI-AS)*	**105**	2.188.170.0/24



(Image 8 – APO)
[Link]

主动探测Probes

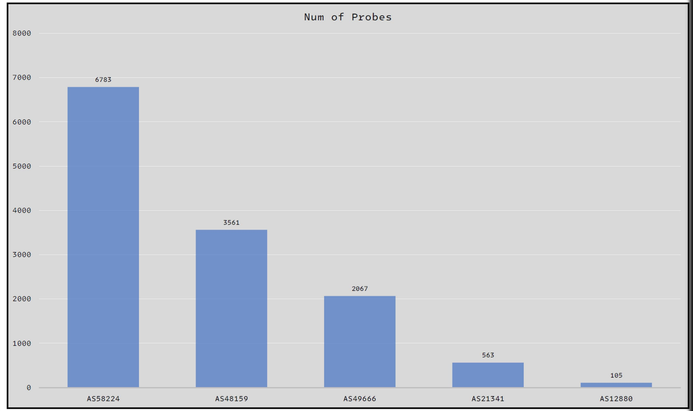
IRGFW在2023年9月时“曾”拥有一个主动探测系统，我们提取了部分使用过的IP地址。[26] 我们的一些测试服务器甚至受到了各种DDoS攻击的影响，这些攻击使服务器的中央处理器使用率达到峰值。[27] 在服务器的这些测试中，这些IP地址是通过Xray-core进行处理的。

但从2024年1月初开始，IRGFW不再使用主动探测。在所有服务器上均未发现探测迹象，我们猜测IRGFW已升级，在被动侧变得更加精确和优化，正如我们将在本报告中讨论的那样。

在下图中，我们记录了检测到的大部分作为探测者的IP CIDR块。我们的测试方法灵感来源于gfw.report团队。[28]

在接下来的几页中，我们将所有的主动探测测试结果汇总为**三种类型**。大部分测试使用Xray-core完成，其余则在伊朗使用多种核心和方法进行。

Iran's GFW Active Probing Test			
AS Number	Org/Name	Num of Probes	IP Ranges
AS58224	*Iran Telecommunication Company PJS (TCI)*	**6783**	80.191.0.0/16 *(80.191.69.0/24)* *(80.191.64.0/24)* 78.38.0.0/16 78.39.0.0/16 217.218.0.0/16 *(217.218.80.0/16)* 2.187.0.0/16
AS48159	*Telecommunication Infrastructure Company (TIC-AS)*	**3561**	2.189.42.0/24 2.184.0.0/16
AS49666	*Telecommunication Infrastructure Company (TIC-GW-AS)*	**2067**	2.188.28.0/24
AS21341	*Soroush Rasanheh Company Ltd (SINET-AS)*	**563**	62.220.121.0/24
AS12880	*Information Technology Company (ITC) (DCI-AS)*	**105**	2.188.170.0/24



（图8 – APO）
[链接]

Active-Probes

Test Type 1

Here, The server, utilizing VLess-TCP-Reality protocol *(Port 2053)*, operated for 24 hours, transferring ~2TB of data before being blocked. Legitimate SYN requests grew steadily, peaking at 332 in hour 19. However, probing activity—*likely from the IRGFW*—began at hour 5, with a sharp increase during hour 16 *(64 probe SYNs alongside 301 legitimate SYNs)*. This suggests deliberate targeting as part of censorship enforcement mechanisms.

Key Observations:

- The Iranian firewall's probes escalated alongside traffic, indicating active surveillance and filtering efforts targeting circumvention tools.
- Despite blocking IR domains and IPs, the server was overwhelmed due to insufficient fallback mechanisms *(e.g., Nginx fallback)* and the absence of adaptive defensive strategies.
- Traffic and probe spikes during hours 16–19 reflect a coordinated probing strategy, likely aiming to detect and disrupt encrypted communication methods.

Number of SYN Received Across Time		
Relative Time (Hours)	*Legitimate Clients*	*Probers*
0	0	0
1	17	0
2	13	0
3	25	0
4	34	0
5	37	12
6	46	6
7	46	8
8	47	8
9	40	10
10	138	18
11	178	36
12	186	31
13	140	23
14	195	55
15	119	15
16	301	64
17	287	81
18	281	35
19	332	41
20	13	4
21	12	0
22	8	0
23	3	0
24	1	0
25	0	0

XRAY CONFIG
VLess-TCP-Reality
Port: 2053
Flow: None
SNI: ftp.debian.org:443

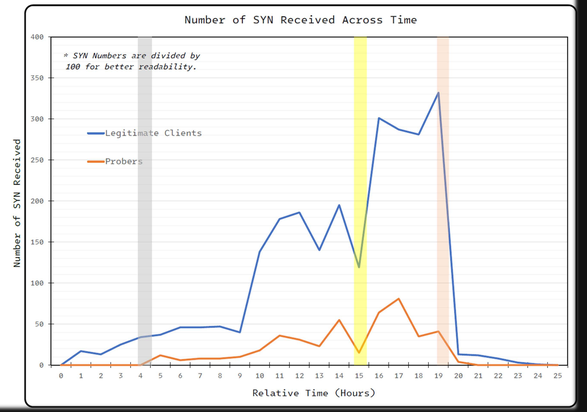
~300Gb Traffic Transferred. Probe Started.

Server Status	BLOCKED
Runtime	24 Hours
Total Traffic	2 TB

Nginx Fallback	No
IR Domains & IPs	Blocked on Server

Traffic Increased
Probes Increased

Server Blocked. ~1950Gb Traffic Transferred.



(Image 9 – AP1)
[Link]

主动探测

测试类型 1

此处，服务器采用VLess-TCP-Reality协议(端口2053)运行了24小时，传输了~2TB数据后被阻断。合法的SYN请求稳步增长，在第19小时达到332个的峰值。然而，探测活动——很可能来自IRGFW——于第5小时开始，并在第16小时急剧增加(探测SYN包64个，合法SYN包301个)。这表明了作为审查执行机制的一部分，存在蓄意的针对性行为。

关键观察：

- 伊朗防火墙的探测活动随流量增加而升级，表明其针对规避工具正在进行主动监控和过滤。
- 尽管封锁了IR域名和IP地址，但由于回退机制不足（例如，Nginx回退）且缺乏自适应防御策略，服务器仍不堪重负。
- 在第16至19小时期间流量和探测活动激增，反映了一种协调的探测策略，其目的可能是为了检测并干扰加密通信方法。

Number of SYN Received Across Time		
Relative Time (Hours)	*Legitimate Clients*	*Probers*
0	0	0
1	17	0
2	13	0
3	25	0
4	34	0
5	37	12
6	46	6
7	46	8
8	47	8
9	40	10
10	138	18
11	178	36
12	186	31
13	140	23
14	195	55
15	119	15
16	301	64
17	287	81
18	281	35
19	332	41
20	13	4
21	12	0
22	8	0
23	3	0
24	1	0
25	0	0

XRAY CONFIG
VLess-TCP-Reality
Port: 2053
Flow: None
SNI: ftp.debian.org:443

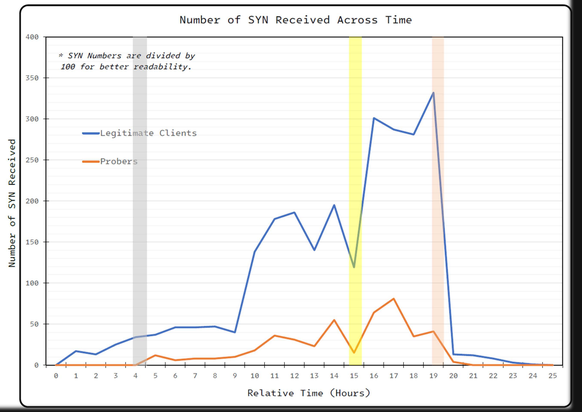
~300Gb Traffic Transferred. Probe Started.

Server Status	BLOCKED
Runtime	24 Hours
Total Traffic	2 TB

Nginx Fallback	No
IR Domains & IPs	Blocked on Server

Traffic Increased
Probes Increased

Server Blocked. ~1950Gb Traffic Transferred.



(图9 – AP1)
[链接]

Active-Probes

Test Type 2

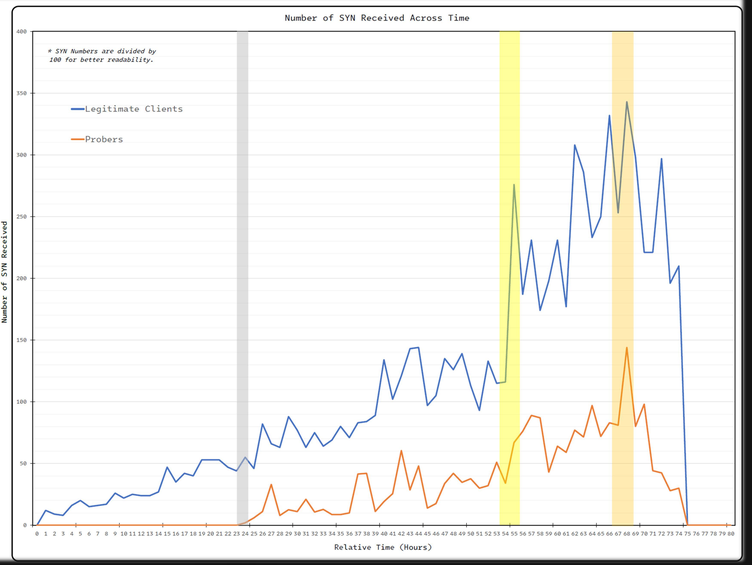
The server, running VLess-TCP-Reality on Port 8080, operated for 75 hours, transferring ~8TB of data without being blocked. Legitimate traffic steadily increased, peaking at 343 SYNs by hour 68. Probes, likely from the Iranian firewall, began after transferring 1.2TB of data *(hour 24)* and spiked during hours 55 and 68, reflecting active targeting by censorship mechanisms.

Key Observations:

- Probes escalated alongside legitimate traffic, peaking at 343 SYNs *(hour 68)*, indicating persistent attempts to disrupt encrypted bypass mechanisms.
- Despite sustained probing and increased traffic, the server remained operational, demonstrating resilience against active filtering efforts.

Number of SYN Received Across Time

Relative Time (Hours)	Legitimate Clients	Probers	XRAY CONFIG
0	0	0	VLess-TCP-Reality
1	12	0	Port: 8080
2	9	0	Flow: xtls-rprx-vision
3	8	0	SNI: oakfaserver .8080
4	16	0	
5	20	0	
6	15	0	
7	16	0	Server Status: NOT BLOCKED
8	17	0	Runtime: 75 Hours
9	26	0	Total Traffic: 8 TB
10	22	0	
11	25	0	
12	24	0	
13	24	0	Nginx Fallback: Yes
14	27	0	IR Domains & IPs: Blocked on Server
15	47	0	
16	35	0	
17	42	0	
18	40	0	
19	53	0	
20	53	0	
21	53	0	
22	47	0	
23	44	0	
24	55	2	1.2 TB Traffic Transferred: Probes Started.
25	46	6	
26	82	11	
27	66	33	
28	63	8	
29	88	13	
30	77	11	
51	85	30	Redundent Data Hidden.
52	133	32	
53	115	51	
54	116	34	
55	276	67	Major Traffic Increased: Probes Increased.
56	187	76	
57	231	89	
58	174	87	
59	198	43	
60	231	64	
61	177	59	
62	308	77	
63	286	72	
64	233	97	
65	250	72	
66	332	83	
67	253	81	
68	343	144	Major Traffic Increased: Probes Increased.
69	298	80	
70	221	98	
71	221	44	
72	297	42	
73	196	28	
74	210	30	
75	0	0	Server Shutdown.
76	0	0	
77	0	0	
78	0	0	
79	0	0	
80	0	0	



(Image 10 - AP2)
[Link]

活跃探测Probes

测试类型2

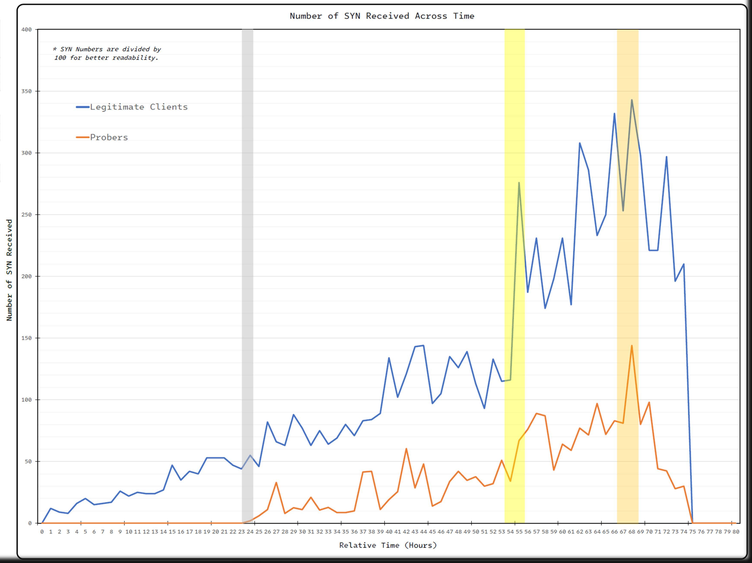
该服务器在8080端口运行VLess-TCP-Reality，持续运行了75小时，传输了 ~8TB数据而未被阻断。合法流量稳步增长，在第68小时达到峰值343个SYN包。很可能来自伊朗防火墙的探测，在传输了1.2TB数据后开始(第24小时)，并在第55和68小时激增，这反映了审查机制正对其进行主动锁定。

关键观察：

- 探测活动随着合法流量一同升级，在第68小时达到峰值343个SYN包(第68小时)，这表明其持续试图干扰加密绕过机制。
- 尽管面临持续的探测和流量增长，服务器仍保持运行，展现了对主动过滤措施的抵御能力。

Number of SYN Received Across Time

Relative Time (Hours)	Legitimate Clients	Probers	XRAY CONFIG
0	0	0	VLess-TCP-Reality
1	12	0	Port: 8080
2	9	0	Flow: xtls-rprx-vision
3	8	0	SNI: oakfaserver .8080
4	16	0	
5	20	0	
6	15	0	
7	16	0	Server Status: NOT BLOCKED
8	17	0	Runtime: 75 Hours
9	26	0	Total Traffic: 8 TB
10	22	0	
11	25	0	
12	24	0	
13	24	0	Nginx Fallback: Yes
14	27	0	IR Domains & IPs: Blocked on Server
15	47	0	
16	35	0	
17	42	0	
18	40	0	
19	53	0	
20	53	0	
21	53	0	
22	47	0	
23	44	0	
24	55	2	1.2 TB Traffic Transferred: Probes Started.
25	46	6	
26	82	11	
27	66	33	
28	63	8	
29	88	13	
30	77	11	
51	85	30	Redundent Data Hidden.
52	133	32	
53	115	51	
54	116	34	
55	276	67	Major Traffic Increased: Probes Increased.
56	187	76	
57	231	89	
58	174	87	
59	198	43	
60	231	64	
61	177	59	
62	308	77	
63	286	72	
64	233	97	
65	250	72	
66	332	83	
67	253	81	
68	343	144	Major Traffic Increased: Probes Increased.
69	298	80	
70	221	98	
71	221	44	
72	297	42	
73	196	28	
74	210	30	
75	0	0	Server Shutdown.
76	0	0	
77	0	0	
78	0	0	
79	0	0	
80	0	0	



(图10 - *AP2*)[链接]

Active-Probes

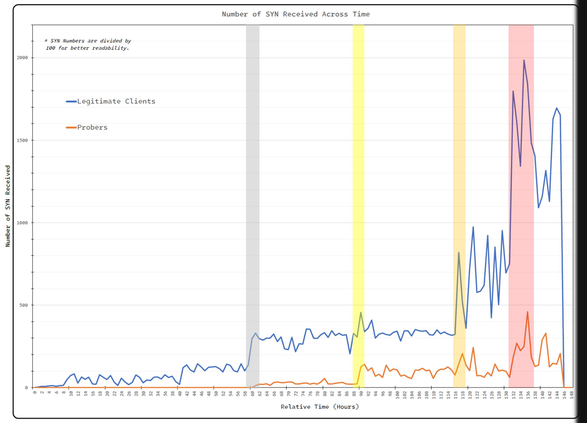
Test Type 3

The server, running VLess-TCP-Reality on Port 443, operated for 145 hours, transferring ~21TB of data without being blocked. Legitimate SYN requests steadily increased throughout the runtime, peaking at 1986 during hour 135. Probing activity, likely originating from the Iranian firewall, began after 4.2TB of data was transferred *(hour 61)* and intensified during three major spikes: hours 90, 117, and 132.

Key Observations:

- Probes began at hour 61 and grew significantly during major traffic surges. Probes peaked alongside legitimate traffic at 1986 SYNs during hour 135.
- Each major increase in legitimate traffic triggered a corresponding spike in probes, indicating systematic filtering efforts targeting high-traffic periods.
- Despite heavy traffic and persistent probing, the server remained operational, demonstrating robustness against the censorship mechanisms deployed.

Number of SYN Received Across Time			
Relative Time (Hours)	Legitimate Clients	Probers	XRAY CONFIG
0	0	0	VLess-TCP-Reality
1	3	0	Port: 443
2	7	0	Flow: xtls-rprx-vision
3	7	0	SNI: steal.personal-domain.with.com (Id
4	10	0	
5	12	0	
6	8	0	
7	12	0	Server Status: NOT BLOCKED
8	13	0	Runtime: 145 Hours
9	49	0	Total Traffic: 21 TB
10	73	0	
11	83	0	
12	27	0	
13	64	0	Nginx Fallback: Yes
14	50	0	IR Domains & IPs: Blocked on Server
15	63	0	
16	22	0	
17	20	0	
18	28	0	
19	145	0	
20	101	0	
59	104	0	
60	297	0	
61	330	12	4.2 TB Traffic Transferred; Probes Started.
62	297	20	
63	288	19	
64	300	23	
65	300	13	
66	325	30	
67	294	15	
86	325	22	
87	205	21	
88	329	21	
89	306	22	
90	856	141	Major Traffic Increased; Probes Increased.
91	359	103	
92	409	120	
94	300	69	
95	324	81	
96	333	61	
112	337	111	
114	225	125	
115	317	107	
116	322	76	
117	820	163	Major Traffic Increased; Probes Increased.
118	520	206	
119	359	136	
120	720	103	
121	975	244	
122	577	72	
123	587	71	
124	480	62	
125	923	92	
126	423	71	
127	853	142	
128	502	100	
129	953	106	
130	695	99	
131	751	63	
132	1798	180	Major Traffic Increased; Probes Increased.
133	1613	269	
134	1342	224	
135	1986	248	
136	1840	460	
137	1483	185	
138	1406	128	
139	1090	136	
140	1317	329	
141	1317	329	
142	1128	125	
143	1628	148	
144	1696	141	
145	1653	207	
146	0	0	Server Shutdown.
147	0	0	
148	0	0	



(Image 11 – AP3)
[Link]

主动探测

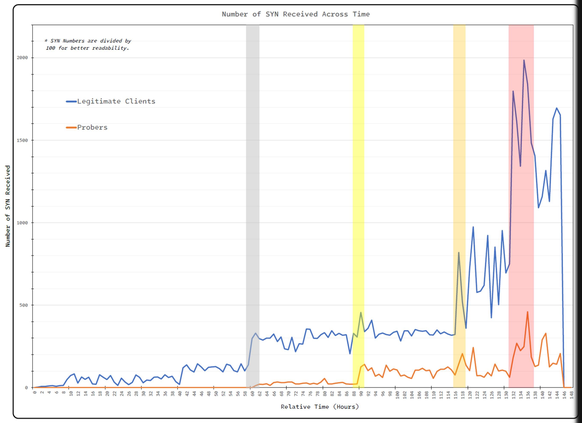
测试类型 3

该服务器在443端口上运行VLess-TCP-Reality，持续运行了145小时，传输了 ~21TB数据而未被阻断。合法的同步请求在运行期间稳步增长，并在第135小时达到峰值1986次。探测活动（可能源自伊朗防火墙）在传输了4.2TB数据后开始（第61小时），并在三个主要高峰时段加剧：第90、117和132小时。

关键观察结果：

- 探测始于第61小时，并在主要流量激增期间显著增长。探测与合法流量在第135小时同时达到峰值，为1986个SYN包。
- 每次合法流量的显著增加都会引发探测的相应激增，这表明存在针对高流量时期的系统性过滤措施。
- 尽管流量巨大且持续探测，服务器仍保持正常运行，这表明其对部署的审查机制具有强大的抵御能力。

Number of SYN Received Across Time			
Relative Time (Hours)	Legitimate Clients	Probers	XRAY CONFIG
0	0	0	VLess-TCP-Reality
1	3	0	Port: 443
2	7	0	Flow: xtls-rprx-vision
3	7	0	SNI: steal.personal-domain.with.com (Id
4	10	0	
5	12	0	
6	8	0	
7	12	0	Server Status: NOT BLOCKED
8	13	0	Runtime: 145 Hours
9	49	0	Total Traffic: 21 TB
10	73	0	
11	83	0	
12	27	0	
13	64	0	Nginx Fallback: Yes
14	50	0	IR Domains & IPs: Blocked on Server
15	63	0	
16	22	0	
17	20	0	
18	28	0	
19	145	0	
20	101	0	
59	104	0	
60	297	0	
61	330	12	4.2 TB Traffic Transferred; Probes Started.
62	297	20	
63	288	19	
64	300	23	
65	300	13	
66	325	30	
67	294	15	
86	325	22	
87	205	21	
88	329	21	
89	306	22	
90	856	141	Major Traffic Increased; Probes Increased.
91	359	103	
92	409	120	
94	300	69	
95	324	81	
96	333	61	
112	337	111	
114	225	125	
115	317	107	
116	322	76	
117	820	163	Major Traffic Increased; Probes Increased.
118	520	206	
119	359	136	
120	720	103	
121	975	244	
122	577	72	
123	587	71	
124	480	62	
125	923	92	
126	423	71	
127	853	142	
128	502	100	
129	953	106	
130	695	99	
131	751	63	
132	1798	180	Major Traffic Increased; Probes Increased.
133	1613	269	
134	1342	224	
135	1986	248	
136	1840	460	
137	1483	185	
138	1406	128	
139	1090	136	
140	1317	329	
141	1317	329	
142	1128	125	
143	1628	148	
144	1696	141	
145	1653	207	
146	0	0	Server Shutdown.
147	0	0	
148	0	0	



(图 11 – AP3)
[链接]

Active-Probes

Summary

- **Test Methodology**

  All Active Probe tests have been consolidated into three primary categories. The majority of these tests were conducted using Xray-core, supplemented by additional testing with various other cores and methods across Iran. The findings were consistent across both TLS and non-TLS protocols, indicating that the specific protocol used had minimal influence on the probing behaviour of the IRGFW. Notably, approximately 90% of our servers running TLS proxies and VPN tunnels were subject to probing by the IRGFW.

  To manage these probes effectively, the Nginx webserver can be employed. It is important to note that probers should not be blocked outright; instead, they should be configured to receive neutral HTTP status codes *(e.g., 2XX, 3XX, 404, etc.)*.

- **Probing Ratio**

  The average ratio of probers to legitimate SYN requests ranged from **0.2** to **0.3**. This meant that for every legitimate user, there were approximately **20%** to **30%** as many probes on average, indicating a relatively high level of active probing compared to legitimate traffic.

- **Test Period & Relevance**

  Please note that these results reflect tests conducted up until September 2023 and are provided to showcase the active probing capabilities of the IRGFW at that time. As of December 2024, these findings are no longer applicable, as the IRGFW has since ceased using any active probing mechanisms.

主动探测

摘要

- **测试方法**

  所有主动探测测试已整合为三大主要类别。其中绝大多数测试使用Xray-core进行，并在伊朗各地辅以其他多种核心和方法进行补充测试。研究结果在TLS和非TLS协议中均保持一致，这表明所使用的具体协议对IRGFW的探测行为影响甚微。值得注意的是，我们运行TLS代理和VPN隧道的服务器中，约有90%受到了IRGFW的探测。

  为了有效管理这些探测者，可以采用Nginx网页服务器。需要注意的是，不应完全阻止探测者；相反，应将其配置为接收中性的HTTP状态码（例如，2XX、3XX、404等）。

- **探测比例**

  探测者与合法SYN请求的平均比例在**0.2**到**0.3**之间。这意味着，平均而言，每有一位合法用户，就有大约**20%**到**30%**的探测者，这表明与合法流量相比，主动探测的水平相对较高。

- **测试周期与相关性**

  请注意，这些结果反映的是截至2023年9月进行的测试，旨在展示IRGFW当时的主动探测能力。截至2024年12月，这些发现已不再适用，因为IRGFW此后已停止使用任何主动探测机制。

The DPI
(Deep Packet Inspection)

深度包检测
(深度包检测)

The DPI

TLS Situation

The IRGFW consistently performs deep inspection and fingerprinting of TLS ClientHello handshakes to identify potential VPN or proxy traffic based on distinctive patterns, regardless of the TLS version used. While tools like uTLS can be employed to obscure some of these fingerprints, they do not fully eliminate detection, as uTLS itself has vulnerabilities that can still be identified by sophisticated DPI techniques.

We developed a series of tools to measure and analyze these behaviours. For instance, we set up a Nginx server hosting a standard website on a public *(whitelisted)* IP address. The site was accessible without issues across all major Iranian ISPs using Chrome and Firefox. However, when a DNS query *(using DoT or DoH)* was initiated from a popular DNS client on Windows, the TLS handshake failed to complete, resulting in a timeout.

When we tested with uTLS *(both official and fragmentation modes)*, the handshake was completed, indicating that the IRGFW had fingerprinted the DNS client. This issue also affects major VPN clients: despite having a whitelisted server IP and SNI domain, the TLS handshake times out.[32] However, when using a less common or non-standard client with different fingerprinting characteristics, the handshake succeeds, and the VPN tunnel is established without issues.

深度包检测

TLS现状

IRGFW持续对TLS ClientHello握手进行深度检测和指纹识别，以基于独特模式识别潜在的VPN或代理流量，无论所使用的TLS版本如何。虽然可以使用uTLS等工具来掩盖部分指纹，但它们并不能完全消除检测，因为uTLS本身也存在漏洞，仍可能被复杂的深度包检测技术识别。

我们开发了一系列工具来测量和分析这些行为。例如，我们在一个公共（白名单） IP地址上搭建了一个托管标准网站的Nginx服务器。使用Chrome和Firefox浏览器，该网站在所有主要的伊朗ISP上均可正常访问。然而，当从Windows上一个流行的DNS客户端发起DNS查询（使用DoT或DoH）时，TLS握手无法完成，导致超时。

当我们使用UTLS（包括官方模式和分片模式）进行测试时，TLS握手顺利完成，这表明IRGFW已对DNS客户端进行了指纹识别。此问题同样影响主流的VPN客户端：尽管服务器IP和SNI域名在白名单中，TLS握手仍会超时。[32] 然而，当使用指纹特征不同、较不常见或非标准的客户端时，握手成功，VPN隧道也能顺利建立。

The DPI

IRGFW DPI consists of two central systems:

1. **The Active Part:** Check each international connection's first 1–17 kilobytes. This system looks for predefined signatures in the first packets of each stream, such as 0x16 0x3, which indicates a possible TLS type. It then looks for the SNI extension in this packet, which starts with 0x1 and includes the packet length. After identifying the SNI, it determines whether it is in the blocking hashtable. If the packet is not of the TLS type, the system looks for other signatures, such as SSH or HTTP. Regarding HTTP, the system looks for the Host header.

   Previously, the system was case-sensitive and sensitive to extra spaces, but it has been updated to eliminate all spaces. This active signature checking appears to be performed on specialized ASICs due to their high processing load, but even with powerful processors, delays and increased ping occur. People return home in the afternoon and activate their VPNs, causing the blocking system to become congested. It's worth noting that the operators in the active part differ, each having their own set of bugs, indicating that the system isn't wholly consistent.

2. **The Passive Part:** Before the recent update *(late Dec 2023 / early Jan 2024)*, the DPI system was fully active and could be deceived without causing any issues. However, after the update, MCI randomly samples some of each person's connections, passively capturing patterns of circumvention. These patterns include TLS in TLS, authentications, and standard VPN packet headers. For example, when using VLess *(V2ray/Xray)*, VLess sends a small authentication packet to each connection before sending the mainstream, ensuring the client is legitimate. Furthermore, when establishing a new VPN connection with another TLS connection, the passive blocking system searches for repeating patterns in small packets containing TLS or V2ray/Xray patterns. If the IP addresses and domains are discovered, they are flagged and reported to the blocking system every 4 hours *(time-pattern)*, where they are either throttled or blocked entirely.

深度包检测

IRGFW 深度包检测由两个核心系统组成：

1. **主动检测部分：** 检查每条国际连接的前1–17千字节。该系统会在每个流的前几个数据包中寻找预定义的签名，例如表示可能是TLS类型的0x16 0x3。接着，它在该数据包中寻找以0x1开头并包含数据包长度的SNI扩展。识别出SNI后，系统判断其是否位于阻断哈希表中。如果数据包不是TLS类型，系统会寻找其他签名，例如SSH或HTTP。对于HTTP，系统会查找Host头。

   此前，该系统对大小写和多余空格敏感，但现已更新以消除所有空格。这种主动特征检查似乎是在专用集成电路上进行的，因为其处理负载很高，但即便使用强大的处理器，也会出现延迟和ping值增加的情况。人们在下午回家并激活他们的VPN，导致阻断系统变得拥堵。值得注意的是，主动部分的运营商各不相同，各自都有其特定的漏洞，这表明该系统并非完全一致。

2. **被动部分：** 在最近的更新(2023年12月底/2024年1月初)之前，深度包检测系统是完全主动的，可以被欺骗而不会引起任何问题。然而，更新之后，MCI会随机抽样每个人的部分连接，被动地捕获规避模式。这些模式包括TLS in TLS、认证数据包和标准的VPN数据包头。例如，当使用VLess(*V2ray/Xray*)时，VLess会在发送主流数据之前向每个连接发送一个小的认证数据包，以确保客户端的合法性。此外，当与另一个TLS连接建立新的VPN连接时，

   被动封锁系统会搜索包含TLS或V2ray/Xray模式的小型数据包中的重复模式。如果发现IP地址和域名，它们会被标记并每隔4小时(时间模式)向封锁系统报告，随后被限速或完全封锁。

The DPI

Possible Solution

To mitigate the risk of server blocking, the goal is to disrupt the patterns that enable detection. One way to do this is by modifying traffic patterns that are easily identifiable by servers. Injecting randomized packets at the start of each stream can help obscure the traffic's intent, making it harder for detection algorithms to classify it. Additionally, multiplexing multiple streams into fewer connections reduces the visibility of individual traffic flows, further decreasing the chance of detection.

For authentication traffic, injecting randomized packets and fragmenting them with varying padding and sizes can prevent the server from recognizing predictable patterns. By making the authentication process less uniform, you reduce the likelihood of it being flagged.

Blocking effectiveness relies on the inability to modify protocols or propagate changes to users easily. If users can adjust traffic patterns dynamically and apply these changes broadly, it undermines the server's ability to block based on fixed patterns. The ability to modify protocols *(such as through encryption, traffic obfuscation, or fragmentation)* helps maintain anonymity and reduce the risk of detection, making blocking attempts less effective.

This strategy hinges on continuous adaptation to avoid predictable behaviour that could be used for blocking or filtering.

深度包检测

可能的解决方案

为了降低服务器封锁的风险，目标是破坏那些使检测成为可能的模式。一种方法是修改容易被服务器识别的流量模式。在每个流开始时注入随机化数据包，有助于掩盖流量的意图，使检测算法更难对其进行分类。此外，将多个流多路复用到更少的连接中，可以减少单个流量流的可见性，从而进一步降低被检测到的机会。

对于认证流量，注入随机化数据包并使用可变填充和不同大小进行分片，可以防止服务器识别出可预测的模式。通过使认证过程不那么统一，可以降低其被标记的风险。

阻断的有效性依赖于无法轻易修改协议或向用户广泛传播变更。如果用户可以动态调整流量模式并广泛应用这些更改，就会削弱服务器基于固定模式进行阻断的能力。修改协议的能力（例如通过加密、流量混淆或分片）有助于保持匿名性并降低被检测的风险，从而使阻断尝试效果不佳。

此策略的关键在于持续适应，以避免可能被用于阻断或过滤的可预测行为。

Protocols Overview

协议概览

IRGFW
irgfw.report
33

Protocols Overview

These tests are conducted intensively with MahsaServer.com *(whenever possible)*; other tests were conducted anonymously in the real world and with Iranian users via the top five ISPs. The number of tests varies from 4 to 20 servers and tests for each protocol or method. The results are averaged, and the median of the results of all protocol tests. Also, all tests are conducted directly on a foreign server, and no middle or tunnelled servers are involved.

- **Socks5, SSTP, PPTP, IKEv2/IPsec:** Blocked by their fingerprints to all foreign IP addresses. **(Blacklist)**
- **L2TP:** Blocked. Many government officials use this protocol, but their Iranian IP addresses or IMEIs have been whitelisted. **(Blacklist)**
- **OpenVPN:** Completely blocked by its fingerprint in all major ISPs. **(Blacklist)**
    - **OpenVPN + Cloak:** Partially functional. Cloak was recently detected by IRGFW [29], resulting in minimal UL/DL speeds with high jitter. **(Graylist)**
- **Wireguard:** Completely blocked by all major ISPs but can function without limitations on some ISPs with a clean IP address and minimal traffic. Higher traffic leads to quick blocking.
    - **Obfuscated Wireguard:** As discussed in the UDP situation section, it can be used by modifying the handshake, but it's vulnerable to fingerprinting.
- **Shadowsocks** *(old and new encryptions and methods)*: Mostly blocked, occasionally graylisted. Some modifications allow connectivity but with high packet loss and jitter. **(Graylist)**
    - **ShadowSocks + Cloak:** Partially functional. Detected by IRGFW with minimal UL/DL speeds and high jitter **(Graylist).**
- **MTProto:** Mostly graylisted. When functional, it follows a strict time-pattern, leading to IP blockage within four days, but it can be extended to 2 weeks or more.
- **SoftEther:** Similar to Wireguard. Blacklisted by fingerprint and follows a strict time-pattern. **(Blacklist)**
- **SSH:** Partially functional on some ISPs and Gray-listed on others. Often follows a loose time-pattern. **(Graylist)**
    - **SSH-over-TLS:** Partially functional and often follows a loose time-pattern. **(Graylist)**

协议概述 Overview

这些测试主要与MahsaServer.com(在可能的情况下)密集进行； 其他测试是在现实世界中匿名进行的，并通过前五大互联网服务提供商与伊朗用户进行。测试数量从4到20台服务器不等，涵盖每种协议或方法的测试。结果取平均值，并以所有协议测试结果的中位数为准。此外，所有测试均直接在境外服务器上进行，不涉及任何中间或隧道服务器。

- **Socks5、SSTP、PPTP、IKEv2/IPsec：** 因其指纹特征对所有境外IP地址被屏蔽。**(黑名单)**
- **L2TP：** 被屏蔽。许多政府官员使用此协议，但其伊朗IP地址或IMEI已被加入白名单。**(黑名单)**
- **OpenVPN：** 在所有主要互联网服务提供商中，因其指纹特征被完全屏蔽。**(黑名单)**
    - **OpenVPN + Cloak：** 部分可用。Cloak最近被IRGFW检测到，[29], 导致上传/下载速度极低且抖动很高。**(灰名单)**
- **WireGuard：** 被所有主要互联网服务提供商完全封锁，但在某些拥有干净IP地址且流量极低的互联网服务提供商上可无限制使用。流量较高会导致快速封锁。
    - **混淆WireGuard：** 如UDP情况部分所述，可通过修改握手过程来使用，但它容易受到指纹识别攻击。
- **Shadowsocks** （新旧加密方式及方法）：大部分被封锁，偶尔进入灰名单。某些修改版本允许连接，但伴随高丢包率和抖动。**（灰名单）**
    - **ShadowSocks + Cloak：** 部分可用。会被IRGFW检测到，上传/下载速度极低且抖动高**（灰名单）**。
- **MTProto：** 多数情况下处于灰名单。可用时遵循严格的时间模式，导致IP在四天内被封锁，但可延长至两周或更久。
- **SoftEther：** 与WireGuard类似。通过指纹识别被列入黑名单，并遵循严格的时间模式。**（黑名单）**
- **SSH：** 在部分互联网服务提供商处功能受限，在其他服务商处则被列入灰名单。通常遵循宽松的时间模式。**（灰名单）**
    - **SSH-over-TLS：** 部分功能可用，通常遵循宽松的时间模式。**（灰名单）**

Protocols Overview

- **V2Ray/XRay/SingBox** *(v5.22.0/v24.12.18/v1.10.5)*:
    - **VMess-(TCP/WS/HU/GRPC)-NonTLS:** Works with a clean IP *(MCI and TCI firewalls only)* but is usually blocked within four days and up to two weeks in some cases.
    - **(VLess/VMess)-(TCP/WS/HU/GRPC/H2)-TLS:** Works with a clean IP but is often blocked within two weeks *(time-pattern)*.
    - **REALITY/ShadowTLSv3:** Mostly blocked within four days *(sometimes within 24 hours)* unless used with a whitelisted SNI but usually blocked within two weeks, even with a whitelisted SNI. This behaviour strongly suggests that the IRGFW employs a reverse DNS mapping system to identify and block these types of protocols and destination IP addresses.
    - **Trojan:** Similar to V2Ray/Xray with TLS. Graylisted and follows a time-pattern.
- **Hysteria2:** Requires a QUIC-enabled destination IP *(Page 8 - UDP section)*.
    - **Hysteria2 + Obfs** *(Salamander)*: QUIC may be completely disabled to some IPs, but Salamander Obfs can sometimes bypass this restriction if UDP works appropriately.
- **TUIC/JUICITY:** Similar to plain Hysteria2. Gray-listed with limited UL/DL bandwidth and high jitter.
- **Obfs4** *(for any protocols like OpenVPN/ShadowSocks/Tor)*: Mostly blocked but can work on some ISPs. Gray-listed and has exceptionally high jitter and UL limitations.
- **TOR** *(with every bridge combination)*: Mostly blocked. And rarely gray-listed with a limited speed.

协议概述

- **V2Ray/XRay/SingBox**
(v5.22.0/v24.12.18/v1.10.5):
    - **VMess-(传输控制协议/WebSocket/HttpUpgrade/gRPC)-非TLS:** 适用于干净的IP地址(仅限MCI和TCI防火墙），但通常在四天内被封锁，某些情况下最多可达两周。
    - **(VLess/VMess)-(传输控制协议/WebSocket/HttpUpgrade/gRPC/H2)-TLS:** 适用于干净的IP地址，但通常在两周内被封锁(时间模式)。
    - **REALITY/ShadowTLSv3:** 大多在四天内被封锁(有时在24小时内），除非使用白名单中的服务器名称指示，但即使使用白名单中的服务器名称指示，通常也会在两周内被封锁。这种行为强烈表明IRGFW采用了反向DNS映射系统来识别和封锁这些类型的协议以及目标IP地址。
    - **Trojan:** 与带TLS的V2ray/Xray类似。被列入灰名单并遵循时间模式。
- **Hysteria2:** 需要目标IP支持QUIC(第8页 - UDP部分)。
    - **Hysteria2 + Obfs** *(Salamander)*：某些IP地址可能会完全禁用QUIC，但如果UDP工作正常，Salamander Obfs有时可以绕过此限制。
- **TUIC/JUICITY：** 与普通Hysteria2类似。被灰名单限制，上传/下载带宽有限且抖动很高。
- **Obfs4** (适用于OpenVPN/Shadowsocks/Tor等任何协议)： 大部分被封锁，但在某些互联网服务提供商上可能可用。被灰名单限制，抖动极高且上传受限。
- **Tor** (含所有网桥组合): 多数情况下被封锁。极少情况下被列入灰名单并限制速度。

Protocols Overview

- **CDN (Content Delivery Network):**

  Certain Content Delivery Networks *(CDNs)*, such as Cloudflare, are compatible with specific protocols that enhance security and privacy. A common configuration is VLess+(WS/gRPC)+TLS, which works effectively to conceal a Virtual Private Server *(VPS)* IP address by routing traffic through a CDN. This setup takes advantage of the CDN to obfuscate the source server's IP, making it harder for adversaries like the IRGFW to directly target the VPS.

  However, the SNI/Host field in the protocol configuration often serves as a vulnerability. When this field is located, the IRGFW can block it, effectively neutralizing the traffic. To mitigate this, fragmentation techniques are employed. Fragmentation involves splitting the SNI/Host domain into smaller components to prevent the firewall from reading or interpreting it properly. This method aims to outsmart the filtering mechanisms.[30]

  Despite these efforts, there are limitations. The IRGFW may escalate its countermeasures by blocking all connections to certain CDNs that are unable to interpret fragmented SNI/Host data. Furthermore, as of November 2024, Cloudflare appears to have implemented stricter security measures aimed at filtering out “bot-like” traffic. Unfortunately, traffic generated by tools such as V2ray/Xray is classified as bot traffic under these guidelines, leading to connection interruptions or outright blocking.

- **ECH/ESNI:**

  ECH, formerly known as ESNI, serves a similar purpose as fragmentation: preventing firewalls from reading the SNI domain. By encrypting the handshake process, ECH ensures that the SNI remains hidden from middleboxes and censorship mechanisms. This encryption disrupts the IRGFW's ability to inspect the unencrypted handshake, effectively thwarting many censorship attempts.

  Historically, ECH and its predecessor ESNI faced outright blocking in countries with stringent censorship policies, such as Iran and China. However, in recent years, Iran has allowed the use of ECH, providing a potential avenue for bypassing restrictions. This is in contrast to China, where ECH and ESNI continue to be actively blocked by the Great Firewall *(GFW)*.[31]

  While ECH offers robust protection by encrypting the SNI, it remains vulnerable to infrastructure-level blocks. As noted in the CDN section, if the underlying network infrastructure *(e.g., IRGFW or Cloudflare)* decides to block certain types of encrypted traffic, ECH configurations can become ineffective. This vulnerability highlights the ongoing arms race between censorship circumvention techniques and the countermeasures deployed by oppressive regimes.

协议概述 Overview

- **内容分发网络：**

  某些内容分发网络 (CDN), 例如 Cloudflare，兼容特定的能增强安全与隐私的协议。一种常见的配置是 VLess+(WebSocket/gRPC)+TLS，它通过将流量路由经由一个内容分发网络，能有效隐藏虚拟专用服务器 *(VPS)* 的 IP地址。这种设置利用了内容分发网络来混淆源服务器的 IP，使得像 IRGFW 这样的对手更难直接攻击虚拟专用服务器。

  然而，协议配置中的SNI/主机字段常常成为弱点。一旦该字段被定位，IRGFW便可将其阻断，从而有效遏制流量。为缓解此问题，人们采用了分片技术。分片是指将SNI/主机域名拆分为更小的组成部分，以防止防火墙正确读取或解析。此方法旨在绕过过滤机制。[30]

  尽管付出了这些努力，仍存在局限性。IRGFW可能会升级其反制措施，阻断所有连接至某些无法解析分片SNI/主机数据的CDN的流量。此外，截至2024年11月，Cloudflare似乎已实施更严格的安全措施，旨在过滤“类似机器人”的流量。不幸的是，根据这些准则，由V2ray/Xray等工具产生的流量被归类为机器人流量，从而导致连接中断或直接被阻断。

- **ECH/ESNI：**

  ECH（前身为ESNI）与分片具有相似的目的：防止防火墙读取SNI域名。通过对握手过程进行加密，ECH确保SNI对中间盒和审查机制保持隐藏。这种加密破坏了IRGFW检查未加密握手的能力，从而有效挫败了许多审查企图。

  历史上，ECH及其前身ESNI在实施严格审查政策的国家，如伊朗和中国，曾面临彻底封锁。然而，近年来，伊朗已允许使用ECH，这为绕过限制提供了一条潜在途径。这与中国的做法形成对比，在中国，ECH和ESNI仍被防火长城(GFW)主动封锁。[31]

  尽管ECH通过加密SNI提供了强大的保护，但它仍然容易受到基础设施层面的封锁。正如CDN部分所述，如果底层网络基础设施（例如IRGFW或Cloudflare）决定阻止某些类型的加密流量，ECH配置可能会失效。这一漏洞凸显了审查规避技术与压迫政权部署的反制措施之间持续不断的军备竞赛。

November 2024 Update

2024年11月更新

Update

IRGFW
irgfw.report
37

Update on the IRGFW

As of December 2024 *(and at the time of writing this report)*, the IRGFW has significantly scaled back its DPI functions. This reduction has led to the deactivation or minimal enforcement of previously rigorous blocking rules, time-based restriction patterns, and active probing protocols that formed the core of IRGFW's stringent internet control.

Currently, the primary ISP firewalls remain operational; however, they function with reduced thresholds, allowing only basic filtering without the in-depth traffic inspection and monitoring that DPI typically provides. Consequently, many protocols, such as VPNs, encrypted connections, and various UDP-based services that would normally face high rates of throttling, blocking, or graylisting, are experiencing fewer restrictions and lower instances of disruption. The current state reflects a temporary easing of censorship measures, as IRGFW's normally advanced DPI capabilities *(like detecting and fingerprinting traffic patterns, active packet sampling, and blocking via synchronized blacklists)* are not being actively applied.

This reduced control intensity may allow for increased data flow and somewhat more open access to previously restricted internet services. However, this shift may be reversible depending on future policy decisions and technological adjustments. While this shift may be temporary, it represents a notable pause in IRGFW's otherwise pervasive control measures, allowing for a brief window of increased connectivity and reduced censorship across Iran's internet landscape.

关于IRGFW的最新情况

截至2024年12月（以及撰写本报告时），IRGFW已大幅缩减其深度包检测功能。这一削减导致先前严格的封锁规则、基于时间的限制模式以及构成IRGFW严格互联网管控核心的主动探测协议被停用或仅得到最低限度的执行。

目前，主要的ISP防火墙仍在运行；然而，它们以降低的阈值运行，仅允许进行基本过滤，而缺乏深度包检测通常提供的深度流量检查与监控。因此，许多协议，如VPN、加密连接以及各种基于UDP的服务，这些通常面临高比例限速、封锁或灰名单处理的，现在遇到的限制更少，中断情况也更少。当前状态反映了审查措施的暂时放松，因为IRGFW通常具备的高级深度包检测能力（如检测和指纹识别流量模式、主动数据包采样以及通过同步黑名单进行封锁）并未被积极应用。

这种控制强度的降低可能允许数据流量的增加，以及对先前受限互联网服务的访问变得相对开放。然而，这一转变可能是可逆的，取决于未来的政策决策和技术调整。尽管这种转变可能是暂时的，但它代表了IRGFW原本无处不在的控制措施出现了一个显著的暂停期，为伊朗互联网环境提供了一个连接性增强、审查减少的短暂窗口。

Last Words

Censorship and circumvention engage in a dynamic and relentless battle. Circumvention methods are continuously developed, deployed, and refined, only to be identified, disrupted, and neutralized by increasingly sophisticated filtering systems. In response, new strategies emerge, temporarily restoring access and perpetuating this endless cycle of adaptation and counter-adaptation.

It's crucial to recognize that the current reduction in filtering intensity by the Islamic Republic of Iran is not a permanent shift or a sign of leniency. Instead, it is a calculated pause, likely designed to provide time for the IRGFW and its associated systems to train and evolve. These systems are being fine-tuned to better detect and counteract new circumvention methods, preparing for a stricter and more effective resurgence. Such measures will enable tighter control during politically or socially critical periods when managing the flow of information is essential for maintaining authority.

In this environment, relying on a single method of circumvention is not just ineffective—it's dangerous. A sustainable approach demands a diverse toolkit of techniques, used in parallel. Employing multiple methods simultaneously—ranging from different protocols and encrypted channels to traffic obfuscation and fragmentation—greatly reduces the risk of complete disruption. Redundancy ensures that if one method is compromised, others remain functional, maintaining connectivity and access.

Ultimately, adaptability and strategic diversification are essential to counter increasingly advanced censorship mechanisms. Success in this battle requires constant innovation, proactive thinking, and the deployment of a wide range of tools to stay ahead of oppressive systems that continue to evolve. The fight for digital freedom is not a static challenge; it demands resilience, creativity, and a readiness to meet each new restriction with stronger, more agile solutions.

最后的话

审查与规避展开了一场动态且无休止的较量。规避方法被不断开发、部署和完善，却又被日益复杂的过滤系统识别、干扰和压制。作为回应，新的策略应运而生，暂时恢复访问，并使得这种适应与反适应的无尽循环持续下去。

至关重要的是要认识到，伊朗伊斯兰共和国目前审查强度的降低并非永久性转变或宽容迹象。相反，这是一次有计划的暂停，其目的很可能是为IRGFW及其相关系统提供训练和演化的时间。这些系统正在被微调，以更好地检测和对抗新的规避方法，为更严格、更有效的卷土重来做准备。此类措施将使当局在政治或社会关键时期能够实施更严密的控制，因为管理信息流对于维持权威至关重要。

在这种环境下，仅依赖单一的规避方法不仅是无效的，而且是危险的。可持续的策略需要一个多样化的技术工具包，并行使用。同时采用多种方法——从不同的协议和加密通道，到流量混淆和分片——可以极大降低被完全阻断的风险。冗余性确保即使一种方法被攻破，其他方法仍能保持功能，维持连接和访问。

最终，适应性和战略多样化对于对抗日益先进的审查机制至关重要。要在这场斗争中取得成功，需要持续创新、前瞻性思维，并部署广泛的工具，以领先于不断演化的压制性系统。争取数字自由的斗争不是一场静态的挑战；它需要韧性、创造力，并准备好以更强大、更灵活的解决方案应对每一个新的限制。

References

1. https://www.amnesty.org/en/latest/news/2023/09/what-happened-to-mahsa-zhina-amini/
2. https://www.ohchr.org/en/press-releases/2023/09/iran-one-year-anniversary-jina-mahsa-aminis-death-custody-heightened
3. https://github.com/net4people/bbs/issues/125
4. https://gfw.report
5. https://github.com/net4people/bbs/issues/129
6. https://apps.dtic.mil/sti/citations/AD1107324
7. https://dig.watch/updates/iran-to-implement-national-information-network-to-keep-people-off-the-internet
8. https://www.science.org/content/article/iran-s-researchers-increasingly-isolated-government-prepares-wall-internet
9. https://www.en-hrana.org/tag/tarh-e-sianat/
10. https://bgp.tools/rankings/IR?sort=cone
11. https://internetabad.factnameh.com/fa
12. https://mci.ir/introduction
13. https://irancell.ir/p/4471/
14. https://www.tci.ir/%D8%AA%D8%A7%D8%B1%DB%8C%D8%AE%DA%86%D9%87/
15. https://www.tic.ir/fa/introduce
16. http://www.iranet.ir/
17. https://www.ipm.ir/
18. https://www.paloaltonetworks.com/blog/2014/06/udp-malware-hiding-place-of-choice/
19. https://knowledgebase.paloaltonetworks.com/KCSArticleDetail?id=kA10g000000Clc6CAC
20. https://github.com/net4people/bbs/issues/140
21. https://ieeexplore.ieee.org/document/1404672
22. https://etchamber.ir/internet-report-2
23. https://github.com/net4people/bbs/issues/113
24. https://irgfw.report/blog/post1/
25. https://github.com/net4people/bbs/issues/224#issuecomment-1462268182
26. https://t.me/irgfw/6
27. https://github.com/XTLS/Xray-core/issues/2778
28. https://gfw.report/blog/gfw_shadowsocks/
29. https://github.com/net4people/bbs/issues/327
30. https://github.com/GFW-knocker/gfw_resist_tls_proxy
31. https://github.com/net4people/bbs/issues/43
32. https://github.com/net4people/bbs/issues/153

参考

1. https://www.amnesty.org/en/latest/news/2023/09/what-happened-to-mahsa-zhina-amini/
2. https://www.ohchr.org/zh/press-releases/2023/09/iran-one-year-anniversary-jina-mahsa-aminis-death-custody-heightened
3. https://github.com/net4people/bbs/issues/125
4. https://gfw.report
5. https://github.com/net4people/bbs/issues/129
6. https://apps.dtic.mil/sti/citations/AD1107324
7. https://dig.watch/updates/伊朗-to-implement-national-information-network-to-keep-people-off-the-internet
8. https://www.science.org/content/article/iran-s-researchers-increasingly-isolated-government-prepares-wall-internet
9. https://www.en-hrana.org/tag/tarh-e-sianat/
10. https://bgp.tools/rankings/IR?sort=cone
11. 超文本传输安全协议://internetabad.factnameh.com/fa
12. https://mci.ir/introduction
13. https://irancell.ir/p/4471/
14. 超文本传输安全协议://www.tci.ir/%D8%AA%D8%A7%D8%B1%DB%8C%D8%AE%DA%86%D9%87/
15. 超文本传输安全协议://www.tic.ir/fa/introduce
16. http://www.iranet.ir/
17. https://www.ipm.ir/
18. 超文本传输安全协议://www.paloaltonetworks.com/blog/2014/06/UDP-恶意软件-隐藏位置的选择/
19. https://knowledgebase.paloaltonetworks.com/KCSArticleDetail?id=kA10g000000Clc6CAC
20. https://github.com/net4people/bbs/issues/140
21. https://ieeexplore.ieee.org/document/1404672
22. https://etchamber.ir/internet-report-2
23. https://github.com/net4people/bbs/issues/113
24. https://irgfw.report/blog/post1/
25. https://github.com/net4people/bbs/issues/224#issuecomment-1462268182
26. https://t.me/irgfw/6
27. https://github.com/XTLS/Xray-core/issues/2778
28. https://gfw.report/blog/gfw_shadowsocks/
29. https://github.com/net4people/bbs/issues/327
30. https://github.com/GFW-knocker/gfw_resist_tls_proxy
31. https://github.com/net4people/bbs/issues/43
32. https://github.com/net4people/bbs/issues/153

Technical Analysis of the IRGFW

Understanding The Iranian Great Firewall

Report 1

December 2024

IRGFW
irgfw.report

IRGFW技术分析

理解伊朗大防火墙

报告 1

2024年12月

IRGFW
irgfw.report