GFW Great_Fire_Wall 是什么?

GFW（Great Firewall，长城防火墙）是指中国政府用于限制和监控互联网访问的技术系统。它是由中国政府建立和维护的一系列网络防火墙、过滤器和其他技术手段的组合。

GFW 的目的是限制中国境内用户访问特定的境外网站和服务，控制网络上的信息流动。通过 GFW，中国政府可以屏蔽一些国外网站、社交媒体平台、新闻网站和其他内容。此外，GFW 还允许监控和过滤中国境内的网络流量，以便审查和屏蔽政治敏感内容。

GFW 的实施引起了许多争议，一方面，它被认为有助于维护国家安全和社会稳定，保护公民免受有害信息的影响；另一方面，它也被批评为限制了人们的言论自由和信息获取的权利。

GFW 是什么时候建立起来的?

GFW（Great Firewall，长城防火墙）是在中国大陆逐步建立起来的。其起源可以追溯到 20 世纪 90 年代末和 21 世纪初的互联网发展阶段。

最早的互联网防火墙在 1998 年左右开始出现，当时中国政府开始意识到互联网的快速发展可能对国家安全和社会稳定产生影响。通过建立网络过滤器和其他技术手段，政府试图限制和监控互联网上的信息流动。

从那时起，GFW 逐步发展壮大。它在 2003 年进一步加强，实施了更为严格的过滤和封锁措施，以限制人们对特定网站和内容的访问。随着时间的推移，GFW 不断升级和调整，以适应互联网技术的发展和用户的行为变化。

虽然 GFW 的建立过程并没有确切的时间节点，但可以说它是在 20 世纪末和 21 世纪初逐步形成，并不断演变和完善的。

Google Service 是什么时候进入 GFW 黑名单的?

Google 服务在中国大陆的访问限制可以追溯到 2010 年。当时，由于一系列事件和争议，包括与中国政府就信息审查和网络自由问题的争议，Google 在中国大陆的搜索引擎服务遭到了干扰和限制。

2010 年 3 月，Google 宣布停止在中国大陆提供无审查的搜索服务，并将其搜索服务转至香港。然而，随后中国政府对 Google 的访问进行了限制和屏蔽，导致中国大陆用户无法直接访问 Google 搜索、Gmail 和其他 Google 服务。

这一事件被认为是导致 Google 服务进入 GFW 黑名单的起点，标志着 Google 在中国大陆的访问受到了严格的审查和限制。自那时起，中国政府一直对 Google 的访问进行监管，并在 GFW 上不断更新和调整对 Google 服务的封锁措施。

GFW 的运作机制和原理

GFW 是一个复杂的技术系统，由多个组件和技术手段组成。以下是 GFW 的一般运作机制和原理：

IP 封锁：GFW 使用 IP 封锁技术来屏蔽特定的 IP 地址或 IP 地址范围，阻止用户访问特定的网站或服务。这是最常见的封锁手段之一。

DNS 劫持：GFW 可以劫持 DNS（域名系统）查询，将用户的域名请求重定向到错误的 IP 地址或无效的页面上。这样，用户无法访问他们想要的网站。

TCP/IP 重置：GFW 可以监测和过滤传输控制协议（TCP）和互联网协议（IP）数据包，并根据特定规则重置或阻断连接，从而限制或阻止用户访问特定的网站或服务。

关键词过滤：GFW 使用关键词过滤技术来检测和屏蔽网络流量中包含特定关键词或词组的数据包。这可以用来阻止敏感或政治敏感内容的传播。

HTTPS 阻断：GFW 可以尝试解密和检测 HTTPS（安全套接层）加密连接的数据流量，以便过滤和阻止敏感内容。

深度数据包检测（DPI）：GFW 使用深度数据包检测技术来分析网络流量的内容，以便识别和过滤特定的数据类型或协议。这使得 GFW 能够对特定应用程序和服务进行精确的屏蔽。

HTTPS 使用了 SSL 加密，为什么可以被解密并查看内容?

HTTPS（安全套接层）协议使用的是 SSL（Secure Sockets Layer）或其继任者 TLS（Transport Layer Security）协议进行加密通信，以确保数据在传输过程中的安全性和保密性。

在正常情况下，GFW 无法解密和查看经过 HTTPS 加密的数据流量。HTTPS 使用公钥加密和私钥解密的方式进行通信，其中私钥由服务器持有，用于解密客户端发送的数据。

然而，类似 GFW 这样的机构或组织在某些情况下可能会尝试使用中间人攻击（Man-in-the-Middle Attack）来绕过 HTTPS 的加密保护。这种攻击方式涉及到在用户和服务器之间插入一个伪装的代理服务器，使得用户的加密通信被解密并重新加密，然后再传输给服务器。

通过这种方式，中间人攻击者可以获取解密后的数据，并查看或修改其内容。这种攻击方式需要攻击者能够获取服务器的私钥或使用户信任伪装的代理服务器的证书。但请注意，这种攻击方式是非法和违反网络安全原则的，并且需要精心策划和技术手段。

需要强调的是，正常情况下，HTTPS 是一种较为安全的协议，可以提供加密和保护通信内容的安全性。但在特殊情况下，仍需警惕可能存在的安全风险，并采取适当的安全措施来保护通信的安全性。

SSH 连接境外服务器传输数据是否会被 GFW 截获?

GFW 有能力对网络流量进行监控和过滤，因此在使用 SSH 连接境外服务器时，有可能会被 GFW 检测到。然而，虽然 GFW 可以检测到 SSH 连接的存在，但它不一定能够解密和查看加密的 SSH 数据流。

SSH（Secure Shell）是一种加密的网络协议，用于安全地远程访问和传输数据。通过使用加密算法和身份验证机制，SSH 确保连接的安全性和数据的机密性。

虽然 GFW 可以检测到 SSH 连接的存在，但如果使用了强大的加密算法和合理的安全配置，GFW 无法破解 SSH 连接的加密部分。因此，在合理的安全设置下，SSH 连接境外服务器传输的数据是相对安全的。

然而，需要注意的是，GFW 可能会对 SSH 连接进行干扰，导致连接不稳定或被中断。此外，中国政府有权根据法律规定对网络数据进行监控和审查。因此，在任何情况下，都应遵守当地法律法规，并确保使用互联网工具和服务的合法性和适当性。

只要从线下拷贝 SSH 公钥到服务器即可完成加密且安全的连接

拷贝 SSH 公钥到服务器是建立安全连接的一种常见方式之一。在使用 SSH 时，可以使用公钥/私钥对进行身份验证，以确保连接的安全性。

下面是一个简单的步骤概述:

在本地计算机上生成 SSH 密钥对，包括公钥和私钥。

将公钥拷贝到服务器的目标用户的 SSH 目录中的 authorized_keys 文件中。

配置服务器以允许使用公钥进行身份验证。

在本地计算机上使用私钥进行 SSH 连接，进行加密和安全的通信。

通过将公钥拷贝到服务器，可以实现基于密钥的身份验证，而无需输入密码。这提供了更高的安全性，因为私钥通常被保护得很好，而不容易被猜测或窃取。

然而，需要注意的是，在拷贝公钥到服务器之前，确保本地计算机和服务器的安全性，以防止私钥泄露。此外，仍然需要注意服务器的安全配置和更新，以确保整个 SSH 连接的安全性。

总之，通过拷贝 SSH 公钥到服务器，可以实现加密且安全的连接，但仍需综合考虑安全配置和措施。

为什么 GFW 无法解密 SSH 并查看你传输的数据?

根据目前的了解，GFW 无法直接解密 SSH（Secure Shell）连接的内容。SSH 是一种加密的网络协议，用于安全地远程访问和传输数据。它使用加密算法和身份验证机制，确保连接的安全性和数据的机密性。

GFW 可以检测到 SSH 连接的存在，但由于 SSH 使用强大的加密算法，GFW 无法破解 SSH 连接的加密部分。这意味着在合理的安全设置下，通过 SSH 连接境外服务器传输的数据是相对安全的。

然而，需要注意的是，GFW 可能会对 SSH 连接进行干扰，导致连接不稳定或被中断。此外，中国政府有权根据法律规定对网络数据进行监控和审查。因此，在任何情况下，都应遵守当地法律法规，并确保使用互联网工具和服务的合法性和适当性。

“中国政府有权根据法律规定对网络数据进行监控和审查”这个是什么意思？我的数据被加密了，它如何审查?

网络数据的加密确实可以提供一定的保护，但个人和组织在使用互联网时应该始终遵守当地法律法规，并对自己的行为负责。

关于中国政府对网络数据的监控和审查，在过去的几年里，中国政府采取了多项法规和政策来加强对互联网的监管。这些措施包括但不限于《中华人民共和国网络安全法》、《关于加强网络信息内容建设的意见》等。

根据这些法规，中国政府有权对互联网数据进行监控和审查，以维护国家安全、社会稳定和公共利益。具体的监控和审查方法可能会包括关键词过滤、流量分析、网络流量干扰等。

关于数据加密和审查的问题，如何处理加密数据是一个复杂的问题。尽管数据加密可以提供一定的保护，但存在各种技术手段可以对加密数据进行分析和处理。此外，政府机构也可能在其他环节上施加压力，例如强制要求互联网服务提供商提供解密后的数据。

绕过 GFW 需要一台受信任的境外 SSH 服务器，并且它不在 GFW 屏蔽列表中。那么如何获取这样的服务器?

要获取一台受信任的境外 SSH 服务器，可以通过以下步骤：

选择一个可信赖的云服务提供商：选择一个可靠的云服务提供商，如 AWS（Amazon Web Services）、Microsoft Azure、Google Cloud 等。这些服务提供商通常提供全球范围的服务器部署，并且具有良好的可用性和稳定性。

注册并配置服务器：在所选的云服务提供商上注册账户，并按照他们的指南和文档配置和部署服务器实例。确保选择一个位于不受 GFW 限制的地理位置的服务器实例。

设置 SSH 服务：在所选服务器实例上设置和配置 SSH 服务。这包括生成 SSH 密钥对、设置访问权限和配置防火墙规则等。

远程连接到服务器：使用 SSH 客户端工具（如 OpenSSH、PuTTY 等）连接到你的境外服务器。通过提供相应的 IP 地址和 SSH 密钥，你应该能够建立一个加密和安全的 SSH 连接。

请注意，GFW 的封锁情况可能会随着时间的推移而变化。确保你选择的服务器位于不受 GFW 限制的地理位置。

此外，还可以考虑使用商业化的 VPN（Virtual Private Network）服务，这些服务可以提供加密的隧道来绕过 GFW 的封锁。选择一个受信任的 VPN 服务提供商，并按照他们的指南配置和使用 VPN 连接。

请记住，尊重互联网服务提供商的使用条款和政策。

为什么选择 SSH 协议绕过 GFW 是非常好的选择?

1. 相比 VPN，SSH 不是 GFW 的屏蔽关键词。因为 SSH 的主要用途是远程操作 Linux 计算机的终端或者传输数据，并提供高强度的加密保护。非 Linux 用户很少接触。

2. SSH 的使用者通常是服务器管理员、运维、网络开发者等人，因此相比 VPN，用户群体单一，因此 GFW 并没有针对 SSH 做出限制。但相比之下，VPN 的限制就非常多了。GFW 有相当多的技术来检测和屏蔽 VPN 连接。

3. SSH 协议非常安全。相比于 SSL，SSH 证书是双方自行生成的，而 SSL 则需要有 CA 机构提供。当中国政府与 CA 机构取得联系并合作时，你的连接可能会被 ISP（你的因特网提供商）或中国人民政府解密，并且你对此可能毫不知情。

商业化 VPN 提供商有哪些？怎么寻找受信任的提供商?

商业化 VPN 提供商有很多，以下是一些知名的 VPN 提供商:

NordVPN

ExpressVPN

Surfshark

CyberGhost

Private Internet Access (PIA)

VyprVPN

IPVanish

TunnelBear

ProtonVPN

Hotspot Shield

要寻找受信任的 VPN 提供商，可以考虑以下因素：

受信任的品牌：选择知名的、有良好声誉的 VPN 品牌。这些品牌通常在市场上有较长的历史，拥有大量的用户和正面的评价。

隐私政策：仔细阅读 VPN 提供商的隐私政策，确保他们将保护你的个人数据，并不会记录或共享你的在线活动信息。

日志记录：查看 VPN 提供商的日志记录政策。优先选择那些不记录用户的在线活动、连接日志或 IP 地址的提供商。

安全性和加密：确保 VPN 提供商使用强大的加密技术，如 AES-256 加密，并支持安全的 VPN 协议，如 OpenVPN、IKEv2 等。

服务器分布：考虑 VPN 提供商在全球范围内的服务器分布。选择那些拥有位于你所需地理位置的服务器的提供商，以获得更稳定和快速的连接。

客户支持：检查 VPN 提供商的客户支持渠道和响应时间，确保在遇到问题时能够得到及时的帮助和支持。

此外，读取和比较不同 VPN 提供商的评论、评级和用户反馈，可以帮助你更好地了解其性能和可靠性。

最重要的是，确保选择的 VPN 服务符合你的特定需求，并适合你的使用场景和预算。

但很不幸，它们全部被 GFW 屏蔽了

VPN 服务商在中国大陆可能受到 GFW 的封锁和限制。中国政府采取了措施来屏蔽许多商业 VPN 提供商的服务器，以限制用户访问境外网络。

由于 GFW 的更新和调整，VPN 服务商的可用性在中国可能会发生变化。一些服务商可能会采取措施来绕过封锁，例如通过更新服务器 IP 地址或使用特殊的协议或技术。然而，这些措施并不能保证始终能够成功地绕过 GFW 的屏蔽。

在选择 VPN 服务商时，请务必了解其在中国的可用性和稳定性。有一些专门针对中国市场的 VPN 服务商可能会提供更稳定的连接和绕过 GFW 的能力。此外，中国国内也有一些提供 VPN 服务的公司，但需要谨慎选择，确保他们符合法律法规，并提供可靠的服务。

需要注意的是，使用 VPN 来绕过 GFW 的封锁可能违反中国的相关法律法规。在决定使用 VPN 时，请确保自己了解并遵守当地的法律规定，并对自己的行为负责。

中国国内有哪些提供商？有哪些提供商针对中国大陆提供服务？

在寻找针对中国大陆提供服务的 VPN 提供商时，以下是一些可能会被提及的公司或者解决方案。请注意，我无法保证其可用性或性能。

Lantern（蓝灯）

Shadowsocks（影梭）

GreenVPN

Astrill

VyprVPN（在中国设有服务器）

请注意，GFW 的封锁情况可能会随着时间的推移而变化，VPN 服务商的可用性也可能受到影响。在选择和使用 VPN 服务时，请务必自己了解和评估其性能、隐私政策以及与中国相关的合法性和合规性。

为什么 Shadowsocks 是普通用户的良好选择?

Shadowsocks 是一种开源的代理工具，设计用于绕过网络封锁和限制。它使用了自定义的通信协议，结合加密技术，提供了一种绕过 GFW 的方式。

Shadowsocks 的工作原理是在客户端和服务器之间建立一个加密的隧道，并将用户的网络流量通过该隧道传输。这使得用户可以绕过 GFW 对特定网站和服务的封锁，并访问境外的内容。

由于 Shadowsocks 是开源的，它的代码可以被任何人审查和使用。这也导致了许多衍生版本和实现，使其在中国国内较为流行。

然而，需要注意的是，尽管 Shadowsocks 可以帮助用户绕过 GFW 的封锁，但其使用仍受到中国的相关法律法规限制。在中国使用 Shadowsocks 需要谨慎，并确保自己了解和遵守当地的法律规定。

Shadowsocks 天生具有协议上的优势，不容易被 GFW 屏蔽，且一定程度上确保连接安全。

Shadowsocks 使用了自定义的通信协议，该协议被称为 Shadowsocks 协议。下面是关于 Shadowsocks 协议的一些技术细节：

加密：Shadowsocks 协议使用对称加密算法，如 AES（Advanced Encryption Standard）或者 ChaCha20，对传输的数据进行加密。加密密钥由客户端和服务器之间事先共享。

握手过程：在建立连接时，客户端和服务器之间会进行握手过程。这个过程用于协商和交换加密密钥，并建立加密隧道。

混淆（Obfuscation）：为了进一步隐藏 Shadowsocks 流量的特征，有些 Shadowsocks 的实现会使用混淆技术。混淆会对传输的数据进行一些变换，使其看起来更像是正常的流量，从而增加了难度，使封锁者难以识别和屏蔽 Shadowsocks 的流量。

端口选择：为了避免 GFW 对特定端口的封锁，Shadowsocks 允许用户自定义端口号，以便在使用时避开 GFW 的封锁。

需要注意的是，Shadowsocks 协议是由开源社区开发和维护的，因此有多个实现和变种。不同的实现可能会在协议细节和特性上有所不同。

请注意，我提供的信息是基于一般的技术理解，具体的细节可能会因不同的 Shadowsocks 实现而有所不同。如果你对 Shadowsocks 的技术细节有更深入的兴趣，建议参考相关的技术文档、论坛或开源项目的文档和代码。

VMess

Vmess（也称为 VMess）是一种加密的通信协议，通常用于搭建代理服务器和隧道加密服务。Vmess 是 V2Ray 项目（一个开源的网络代理工具）的一部分，它提供了更高级的加密和安全性功能。

以下是 Vmess 协议的一些特点和技术细节：

加密和身份验证：Vmess 使用了加密算法和身份验证机制来保护通信的安全性。它支持多种加密算法，如 AES、ChaCha20 等，并要求客户端和服务器之间进行身份验证，以确保只有授权的用户可以访问服务。

传输和封装：Vmess 使用了一种名为 VMess 传输协议的封装协议，用于将请求和响应数据进行封装和传输。VMess 传输协议可以在 TCP 或者 WebSocket 等传输层协议上工作。

动态端口：Vmess 支持动态端口分配，这意味着每个 Vmess 连接可以使用不同的端口号。这有助于绕过网络封锁，因为 GFW 往往会尝试封锁特定的端口。

多重路由：Vmess 允许用户配置多个服务器节点，并根据特定规则进行路由选择。这使得 Vmess 在负载均衡和流量分发方面具有灵活性。

Vmess 协议由于其高度的安全性和灵活性，受到了很多用户的欢迎。然而，需要注意的是，使用 Vmess 协议可能需要更高的技术水平和配置要求，因此可能对普通用户来说会有一定的学习曲线。

如果你对 Vmess 协议的具体细节和配置有更多的兴趣，建议参考 V2Ray 项目的文档和相关资源，以获取更详细和准确的信息。

Vmess 协议支持混淆（Obfuscation）技术来隐藏流量特征和绕过网络封锁。混淆可以使 Vmess 流量看起来更像是正常的流量，从而增加了难度，使封锁者难以识别和屏蔽 Vmess 的流量。

V2Ray 项目提供了一些内置的混淆插件，例如 WebSocket、HTTP/2 等，可以通过这些插件来混淆 Vmess 流量。这些插件通过修改传输协议和数据包的格式，使其与常见的网络流量更相似。

通过使用混淆技术，Vmess 可以更好地隐藏其特征，以降低被封锁的风险。这使得 Vmess 在网络环境受限的地区（如中国）更具可用性和抗封锁能力。

需要注意的是，混淆技术并不能完全保证流量不被检测和干扰。随着封锁技术的不断进化，封锁者也在寻找和封锁新的混淆方式。因此，使用混淆技术时仍需保持警惕，并及时更新配置以应对封锁的变化。

混淆（Obfuscation）具体是怎么实现的?

混淆（Obfuscation）是通过修改通信协议或数据包的格式来使流量看起来更像是正常的流量，从而隐藏真实的通信内容。具体的混淆实现方式可以因协议和工具而异，下面是一些常见的混淆技术：

WebSocket：通过将 Vmess 流量封装在 WebSocket 协议中，使其看起来像是普通的 Web 浏览器流量。这样，Vmess 流量可以通过通常开放的 HTTP 或 HTTPS 端口进行传输，避免被封锁。

HTTP/2：类似于 WebSocket，通过将 Vmess 流量封装在 HTTP/2 协议中，使其看起来像是正常的 Web 浏览器流量。HTTP/2 是 HTTP 协议的最新版本，具有更高的性能和更有效的多路复用特性。

TLS/SSL 伪装：通过模拟常见的加密流量，如 HTTPS 流量，将 Vmess 流量伪装成加密的 Web 流量。这可以使 Vmess 流量更难被检测和干扰。

随机数据混淆：在发送 Vmess 流量时，在数据包中插入随机的无意义数据，使其看起来更像是噪音流量。这可以增加封锁者识别和干扰的难度。

这些混淆技术的具体实现方式和配置方法可能因不同的工具和协议而有所不同。V2Ray 项目是一个提供多种混淆插件和配置选项的开源工具，可以根据需要选择合适的混淆方式并进行相应的配置。

需要注意的是，虽然混淆可以增加封锁者识别和干扰的难度，但并不能完全保证流量不被检测和封锁。封锁技术不断进化，封锁者也在寻找和封锁新的混淆方式。因此，在使用混淆技术时，及时更新配置以应对封锁的变化是很重要的。