flyfly
(http://blog.leanote.com/ymhe)

flyfly

SSR如何选择正确协议和混淆加速连接

2019-07-03 18:08:39 15500 0 0

概要（看不懂的可以直接看最后的总结）

用于方便地产生各种协议接口。实现为在原来的协议外套一层编码和解码接口，不但可以伪装成其它协议流量，还可以把原协议转换为其它协议进行兼容或完善（但目前接口功能还没有写完，目前还在测试完善中），需要服务端与客户端配置相同的协议插件。插件共分为两类，包括混淆插件和协议定义插件。

简单点说明，懂的如何设置协议跟混淆的。有助于突破地区封锁，加速国外网站的访问、下载！这就是为什么都用同一个服务器节点，有的人速度很快，有的人速度很慢。

现有混淆介绍（在用户中心-资料编辑页面可以设置协议与混淆）

1.混淆插件
此类型的插件用于定义加密后的通信协议，通常用于协议伪装，部分插件能兼容原协议。

plain：表示不混淆，直接使用协议加密后的结果发送数据包

http_simple：并非完全按照http1.1标准实现，仅仅做了一个头部的GET请求和一个简单的回应，之后依然为原协议流。使用这个混淆后，已在部分地区观察到似乎欺骗了QoS的结果。对于这种混淆，它并非为了减少特征，相反的是提供一种强特征，试图欺骗GFW的协议检测。要注意的是应用范围变大以后因特征明显有可能会被封锁。此插件可以兼容原协议(需要在服务端配置为http_simple_compatible)，延迟与原协议几乎无异（在存在QoS的地区甚至可能更快），除了头部数据包外没有冗余数据包，支持自定义参数，参数为http请求的host，例如设置为cloudfront.com伪装为云服务器请求，可以使用逗号分割多个host如a.com,b.net,c.org，这时会随机使用。注意，错误设置此参数可能导致连接被断开甚至IP被封锁，如不清楚如何设置那么请留空。
本插件的高级设置（C#版、python版及ssr-libev版均支持）：本插件可以自定义几乎完整的http header，其中前两行的GET和host不能修改，可自定义从第三行开始的内容。例子：
baidu.com#User-Agent: abc\nAccept: text/html\nConnection: keep-alive
这是填于混淆参数的内容，在#号前面的是上文所说的host，后面即为自定义header，所有的换行使用\n表示（写于配置文件时也可直接使用\n而不必写成\n，换行符亦会转换），如遇到需要使用单独的\号，可写为\，最末尾不需要写\n，程序会自动加入连续的两个换行。

http_post：与http_simple绝大部分相同，区别是使用POST方式发送数据，符合http规范，欺骗性更好，但只有POST请求这种行为容易被统计分析出异常。此插件可以兼容http_simple，同时也可兼容原协议(需要在服务端配置为http_post_compatible)，参数设置等内容参见http_simple，密切注意如果使用自定义http header，请务必填写boundary。

random_head（不建议使用）：开始通讯前发送一个几乎为随机的数据包（目前末尾4字节为CRC32，会成为特征，以后会有改进版本），之后为原协议流。目标是让首个数据包根本不存在任何有效信息，让统计学习机制见鬼去吧。此插件可以兼容原协议(需要在服务端配置为random_head_compatible)，比原协议多一次握手导致连接时间会长一些，除了握手过程之后没有冗余数据包，不支持自定义参数。

tls1.2_ticket_auth（强烈推荐）:模拟TLS1.2在客户端有session ticket的情况下的握手连接。目前为完整模拟实现，经抓包软件测试完美伪装为TLS1.2。因为有ticket所以没有发送证书等复杂步骤，因而防火墙无法根据证书做判断。同时自带一定的抗重放攻击的能力，以及包长度混淆能力。如遇到重放攻击则会在服务端log里搜索到，可以通过grep "replay attack"搜索，可以用此插件发现你所在地区线路有没有针对TLS的干扰。防火墙对TLS比较无能为力，抗封锁能力应该会较其它插件强，但遇到的干扰也可能不少，不过协议本身会检查出任何干扰，遇到干扰便断开连接，避免长时间等待，让客户端或浏览器自行重连。此插件可以兼容原协议(需要在服务端配置为tls1.2_ticket_auth_compatible)，比原协议多一次握手导致连接时间会长一些，使用C#客户端开启自动重连时比其它插件表现更好。支持自定义参数，参数为SNI，即发送host名称的字段，此功能与TOR的meet插件十分相似，例如设置为cloudfront.net伪装为云服务器请求，可以使用逗号分割

flyfly
(http://blog.leanote.com/ymhe)

flyfly

多个host如a.com,b.net,c.org，这时会随机使用。注意，错误设置此参数可能导致连接被断开甚至IP被封锁，如不清楚如何设置那么请留空。推荐自定义参数设置为cloudflare.com或cloudfront.net。

2.协议定义插件
此类型的插件用于定义加密前的协议，通常用于长度混淆及增强安全性和隐蔽性，部分插件能兼容原协议。

origin：表示使用原始SS协议

verify_simple（已废弃）：对每一个包都进行CRC32验证和长度混淆，数据格式为：包长度(2字节)|随机数据长度+1(1字节)|随机数据|原数据包|CRC32。此插件与原协议握手延迟相同，整个通讯过程中存在验证及混淆用的冗余数据包，下载的情况下冗余数据平均占比1%，普通浏览时占比略高一些，但平均也不会超过5%。此插件不能兼容原协议，千万不要添加_compatible的后缀。

verify_deflate：对每一个包都进行deflate压缩，数据格式为：包长度(2字节)|压缩数据流|原数据流Adler-32，此格式省略了0x78,0x9C两字节的头部。另外，对于已经压缩过或加密过的数据将难以压缩（可能增加1~20字节），而对于未加密的html文本会有不错的压缩效果。因为压缩及解压缩较占CPU，不建议较多用户同时使用此混淆插件。此插件不能兼容原协议，千万不要添加_compatible的后缀。

verify_sha1（即原版OTA协议）：对每一个包都进行SHA-1校验，具体协议描述参阅One Time Auth，握手数据包增加10字节，其它数据包增加12字节。此插件能兼容原协议（需要在服务端配置为verify_sha1_compatible)。

auth_simple（已废弃）：首个客户端数据包会发送由客户端生成的随机客户端id(4byte)、连接id(4byte)、unix时间戳(4byte)以及CRC32，服务端通过验证后，之后的通讯与verify_simple相同。此插件提供了最基本的认证，能抵抗一般的重放攻击，默认同一端口最多支持16个客户端同时使用，可通过修改此值限制客户端数量，缺点是使用此插件的服务器与客户机的UTC时间差不能超过5分钟，通常只需要客户机校对本地时间并正确设置时区就可以了。此插件与原协议握手延迟相同，支持服务端自定义参数，参数为10进制整数，表示最大客户端同时使用数。

auth_sha1（不建议）：对首个包进行SHA-1校验，同时会发送由客户端生成的随机客户端id(4byte)、连接id(4byte)、unix时间戳(4byte)，之后的通讯使用Adler-32作为效验码。此插件提供了能抵抗一般的重放攻击的认证，默认同一端口最多支持64个客户端同时使用，可通过修改此值限制客户端数量，使用此插件的服务器与客户机的UTC时间差不能超过1小时，通常只需要客户机校对本地时间并正确设置时区就可以了。此插件与原协议握手延迟相同，能兼容原协议（需要在服务端配置为auth_sha1_compatible)，支持服务端自定义参数，参数为10进制整数，表示最大客户端同时使用数。

auth_sha1_v2（不建议）：与auth_sha1相似，去除时间验证，以避免部分设备由于时间导致无法连接的问题，增长客户端ID为8字节，使用较大的长度混淆。能兼容原协议（需要在服务端配置为auth_sha1_v2_compatible)，支持服务端自定义参数，参数为10进制整数，表示最大客户端同时使用数。

auth_sha1_v4（推荐）：与auth_sha1相似，包头次序调整，以抵抗抓包重放检测，使用较大的长度混淆，使用此插件的服务器与客户机的UTC时间差不能超过24小时，即只需要年份日期正确即可。能兼容原协议（需要在服务端配置为auth_sha1_v4_compatible)，支持服务端自定义参数，参数为10进制整数，表示最大客户端同时使用数。

auth_aes128_md5或auth_aes128_sha1（均推荐）：对首个包的认证部分进行使用Encrypt-then-MAC模式以真正免疫认证包的CCA攻击，预防各种探测和重防攻击，使用此插件的服务器与客户机的UTC时间差不能超过24小时，即只需要年份日期正确即可，针对UDP部分也有做简单的校验。此插件能兼容原协议，支持服务端自定义参数，参数为10进制整数，表示最大客户端同时使用数。

这样以来，将来只要简单的换一个混淆插件，让大家的特征各不相同，GFW就极难下手统一封锁了。推荐使用auth_aes128_md5插件，在以上插件里混淆能力较高，而抗检测能力最高，同时CPU占用稍微比auth_aes128_sha1低一些。同时如果要发布公开代理，以上auth插件均可严格限制使用人数（要注意的是服务端若配置为compatible，那么用户只要使用原协议就没有限制效果）。

混淆特性

flyfly
(http://blog.leanote.com/ymhe)

flyfly

name	encode speed	bandwidth	RTT	anti replay attack	cheat QoS	anti analysis
plain	100%	100%	0	No	0	/
http_simple	20%/100%	20%/100%	0	No	90	90
http_post	20%/100%	20%/100%	0	No	100	95
random_head (X)	100%	85%/100%	1	No	0	10
tls1.2_ticket_auth	98%	75%/ 95%	1	Yes	100	100

说明:
- 20%/100%表示首包为20%，其余为100%速度（或带宽），其它的 RTT 大于0的混淆，前面的表示在浏览普通网页的情况下平均有效利用带宽的估计值，后一个表示去除握手响应以后的值，适用于大文件下载时。
- RTT 表示此混淆是否会产生附加的延迟，1个RTT表示通讯数据一次来回所需要的时间。
- RTT 不为0且没有 anti replay attack 能力的混淆，不论协议是什么，都存在被主动探测的风险，即不建议使用random_head和tls_simple。 RTT 为0的，只要协议不是 origin，就没有被主动探测的风险。当然由于原协议本身也存在被主动探测的风险，在目前没有观察到主动探测行为的情况下，暂时不需要太担心。
- cheat QoS 表示欺骗路由器 QoS 的能力，100表示能完美欺骗，0表示没有任何作用，50分左右表示较为严格的路由能识别出来。
- anti analysis 表示抗协议分析能力，plain 的时候依赖于协议，其它的基于网友反馈而给出的分值。值为100表示完美伪装。

协议特性

name	encode speed	bandwidth	anti CPA	anti CCA	anti replay attack	anti mid-man detect	anti packet length analysis	anti packet time sequen analysis
origin	100%	99%	Yes	No	No	No	0	0
verify_simple	90%	96%	Yes	No	No	No	1	0
verify_deflate	30%	97%~110%	Yes	No	No	No	6	0
verify_sha1	85%	98%/99%	Yes	No	No	No	0	0
auth_simple (X)	85%	95%	Yes	No	Yes	No	1	0
auth_sha1 (X)	95%	97%	Yes	No	Yes	No	4	0
auth_sha1_v2	94%	80%/97%	Yes	No	Yes	No	10	0
auth_sha1_v4	90%	85%/98%	Yes	Yes?	Yes	No	10	0
auth_aes128_md5	80%	90%/99%	Yes	Yes	Yes	Yes	10	0
auth_aes128_sha1	70%	90%/99%	Yes	Yes	Yes	Yes	10	0

说明:

- 以上为浏览普通网页（非下载非看视频）的平均测试结果，浏览不同的网页会有不同的偏差
- encode speed仅用于提供相对速度的参考，不同环境下代码执行速度不同
- verify_deflate的bandwidth（有效带宽）上限110%仅为估值，若数据经过压缩或加密，那么压缩效果会很差
- verify_sha1的bandwidth意为上传平均有效带宽98%，下载99%

flyfly
(http://blog.leanote.com/ymhe)

flyfly

- auth_sha1_v2的bandwidth在浏览普通网页时较低（为了较强的长度混淆，但单个数据包尺寸会保持在1460以内，所以其实对网速影响很小），而看视频或下载时有效数据比率比auth_sha1要高，可达95%，所以不用担心下载时的速度。auth_sha1_v4及auth_aes128_md5类似
- 如果同时使用了其它的混淆插件，会令bandwidth的值降低，具体由所使用的混淆插件及所浏览的网页共同决定
- 对于抗包长度分析一列，满分为100，即0为完全无效果，5以下为效果轻微，具体分析方法可参阅方校长等人论文
- 对于抗包时序分析一列，方校长的论文表示虽然可利用，但利用难度大（也即他们还没能达到实用级），目前对此也不做处理

混淆与协议配置建议

- 如果你所在地区封锁不是很严重，推荐使用的协议：auth_aes128_md5或auth_aes128_sha1。混淆值：plain
- 地方封锁严重的，如校园网、企业网，推荐使用的协议：auth_aes128_md5或auth_aes128_sha1。混淆值：http_simple与tls1.2_ticket_auth

上一篇: (http://blog.leanote.com/post/ymhe/2dcbaceb5e08)

下一篇: 584游戏加速使用教程
(http://blog.leanote.com/post/ymhe/584%E4%B8%93%E7%BA%BF%E5%8A%A0%E9%80%9F%E4%BD%BF%E7%94

导航

主页 (http://blog.leanote.com/ymhe)
About Me (http://blog.leanote.com/single/ymhe/About-Me)
归档 (http://blog.leanote.com/archives/ymhe)
标签 (http://blog.leanote.com/tags/ymhe)

最近发表

MAC安装常见问题 (http://blog.leanote.com/post/ymhe/MAC%E6%8F%90%E7%A4%BA)
阅读赚代理说明 (http://blog.leanote.com/post/ymhe/72564195e2ad)
阅读赚广告合作 (http://blog.leanote.com/post/ymhe/5f716e92c7d7)
58pan使用教程 (http://blog.leanote.com/post/ymhe/58pan%E4%BD%BF%E7%94%A8%E6%95%99%E7%A8%8B)
樱花GMO代购 (http://blog.leanote.com/post/ymhe/ares)

友情链接

My Note (https://leanote.com/note)
Leanote Home (https://leanote.com)
Leanote BBS (http://bbs.leanote.com)
Leanote Github (https://github.com/leanote/leanote)