# 利用 Nginx、Tyk Gateway API 和 CloudFlare 防火墙隐藏 C2 设施 - 先知社区

> 先知社区，先知安全技术社区



## 0x01 前言

Cobalt Strike 的特征已经被各大安全厂商标记烂了，加上搜索引擎、空间测绘的扫描，在配置 C2 域名后，如果不做好基础设施的隐匿，很快会出现下图的情况。（图片截取自某情报社区）


(https://s2.loli.net/2022/08/25/oZv4UrYhA8yqLcK.png)

隐藏 C2 的手法有很多，比如早些时候的域前置和最近热门的利用云函数隐藏。但是如今许多 CDN 厂商都已经禁用了域前置技术，而云函数是有免费额度的，超过之后会开始计费，许多蓝队反制帖子已经开始分享消耗云函数额度的方法了。

最近看到一篇利用 Tyk Gateway API 隐藏 C2 流量的文章，通过配

置 Tyk Gateway API 转发恶意流量，以达到类似于域前置，或者腾讯云函数隐藏 C2 的效果。

在这个基础上，通过将域名托管到 CloudFlare，配置 Nginx 过滤不符合规则的请求，并通过配置 CloudFlare 防火墙只允许 Tyk Gateway API 的流量访问 C2 域名，可以达到隐藏 C2 域名，并且防止搜索引擎、空间测绘扫描和识别 Cobalt Strike 特征导致域名或 IP 被标记。

本文基于已经将域名托管到 CloudFlare 并配置 SSL 证书的情况，如果你不知道如何使用 CloudFlare 和配置 SSL 证书，请自行搜索相关资料。

# 0x02 配置 Nginx

将域名托管到 CloudFlare 后，可以配置 Nginx 反向代理来过滤部分请求，只让信标流量转发进服务器。

## 自定义 Nginx 配置文件

Nginx 的配置文件还是有点复杂的，可以使用 cs2modrewrite (https://github.com/threatexpress/cs2modrewrite) 进行生成，然后根据需求进行修改。

这里以使用 jquery-c2.4.5.profile (https://github.com/threatexpress/malleable-c2/blob/master/jquery-c2.4.5.profile) 作为 C2 配置文件的情况示例：

```
python3 ./cs2nginx.py -i jquery-c2.4.5.profile -c https://12
7.0.0.1:8443 -r http://www.baidu.com/ -H yourc2.domain > ./ng
inx.conf
```

- -i：指定 C2 配置文件
- -c：指定内部的监听端口
- -r：指定 302 跳转的地址
- -H：指定你的域名

通过该工具生成的 Nginx 配置文件的 `server` 块的部分配置如下：

```
server {
        set $C2_SERVER https://127.0.0.1:8443;
        set $REDIRECT_DOMAIN http://www.baidu.com/;
        server_name yourc2.domain;
        ......

        listen 80;
        listen [::]:80;

        listen 443 ssl;
        listen [::]:443 ssl;
        ......

        location ~ ^(/jquery-3.3.2.slim.min.js.*|/jquery-3.3.
1.min.js.*|/jquery-3.3.1.slim.min.js.*|/jquery-3.3.2.min.js.
*)$ {if ( $http_user_agent != "Mozilla/5.0 (Windows NT 6.3; T
rident/7.0; rv:11.0) like Gecko")
          {
            return 404;
          }
            proxy_pass          $C2_SERVER;
        ......
        }
        location @redirect {
                return 302 $REDIRECT_DOMAIN$request_uri;
        }

        }
```

使用 Nginx 加载该配置文件后。Nginx 将监听外部的 80、443 端口，并将符合规则的请求转发到内部的 8443 端口，不符合规则的请求将跳转到 `http://www.baidu.com/` 。

# 效果

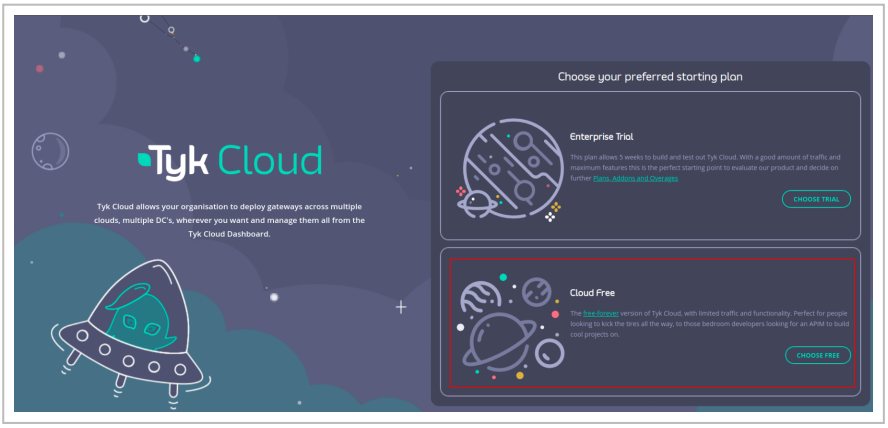
此时通过对服务器 IP 地址的扫描就无法获取到我们的 beacon stage 了。


(https://s2.loli.net/2022/08/26/rNz75bAiwQtkERq.png)

# 0x03 配置 Tyk Gateway API

## 注册账户

访问 注册地址 (https://account.cloud-ara.tyk.io/signup)，填写用户名、邮箱、密码等信息后点击注册，注册成功后选择免费版。

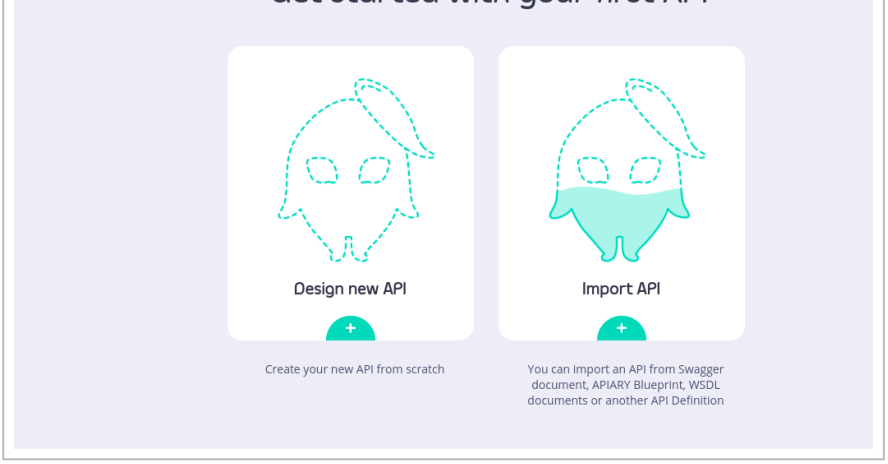



(https://s2.loli.net/2022/08/25/Q58cZ3YsdFaqMhu.png)

然后设置组织名称，设置好后会提示 `Deployment successful` 。

## 创建并配置 API

点击 `Manage APIs` 后会看到如下页面：



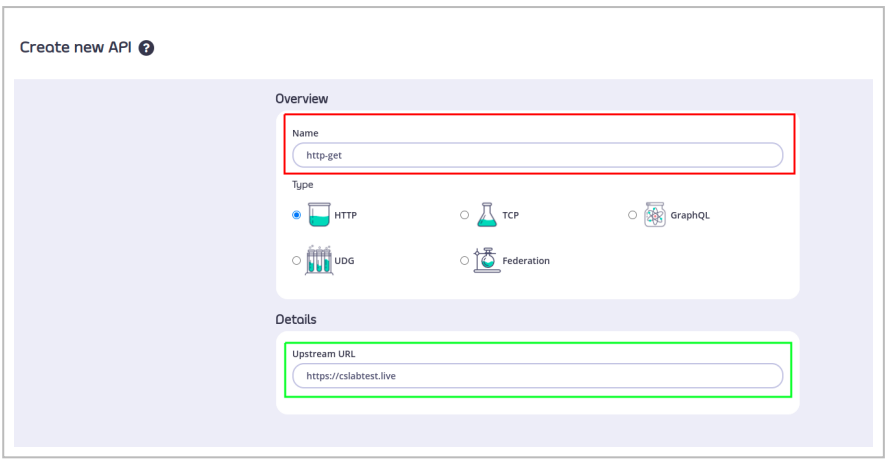

(https://s2.loli.net/2022/08/25/eUybjDunGM8oJzr.png)

接下来逐一创建并配置 http-get API、http-post API、Stager-x86 API、Stager-x64 API。以下以 http-get API 为例。

假设你的域名为 `cslabtest.live` ，且 C2 配置文件如下：

```
http-get {

    set uri "/api/v2/login";
   ....
}

http-post {

    set uri "/api/v2/status";
   ....
}
http-stager {
    set uri_x86 "/api/v2/GetProfilePicture";
    set uri_x64 "/api/v2/GetAttachment";
}
```

点击 `Design new API` 并填写 API 信息。

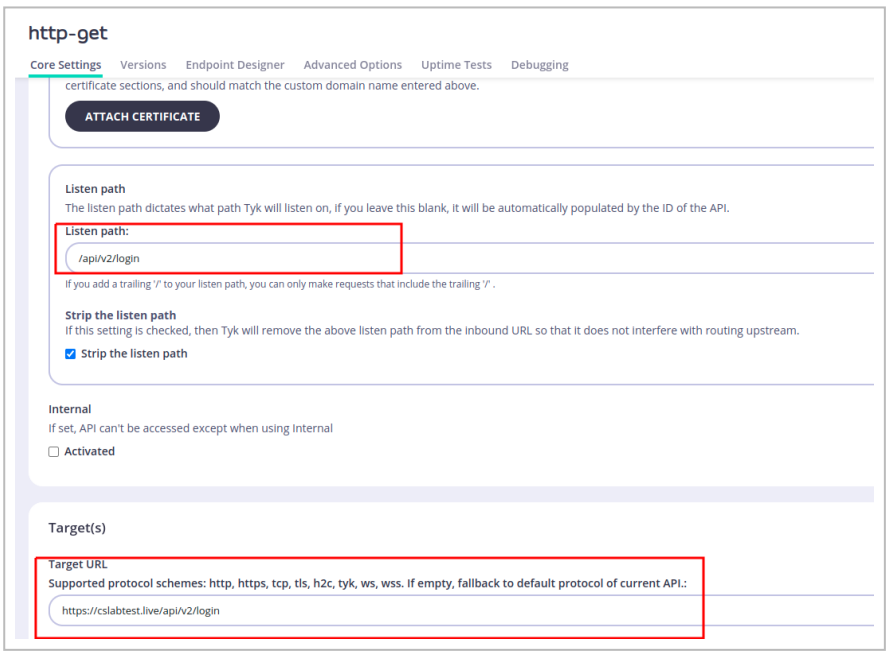


(https://s2.loli.net/2022/08/25/4m2H1xNwMij3Gu8.png)

创建好后进一步配置 API，来让它能够将请求转发到我们的 C2 服务器。

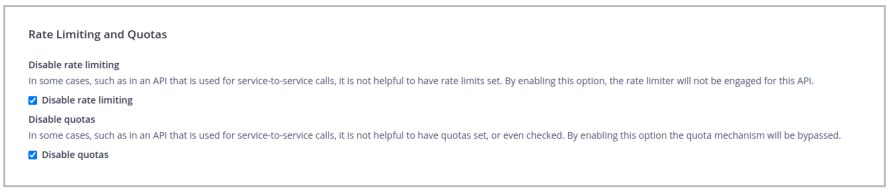
现在我们需要更改 `Listen path` 和 `Target URL` ，TYK 会监听 `Listen path` 的地址，并将请求转发到 `Target URL` 。

（注：若 C2 配置文件为 jquery-c2.4.5.profile，则将 `Listen path` 和 `Target URL` 的路径配置为相对应的 `.js` 的路径）




(https://s2.loli.net/2022/08/25/9u8Bsjh6cSMCfFQ.png)

为了能够上线 CS，还要配置 `Rate Limiting and Quotas` 。都选择 disable 即可。




(https://s2.loli.net/2022/08/25/sbpULn2YTPhvwyx.png)

然后来到 `Advanced Options` ，取消勾选 `Enable caching` 。


(https://s2.loli.net/2022/08/25/ax9wSqE2el6nRJ7.png)

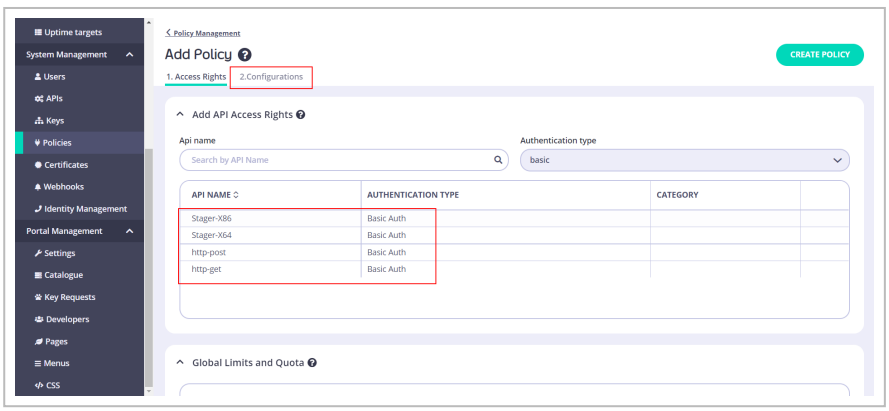
按这个步骤逐一新建 http-get API、http-post API、Stager-x86 API、Stager-x64 API。

## 设置访问验证策略

将上一步新建的 API 的 `Authentication` 更改为 `Basic Authentication`，如下所示：

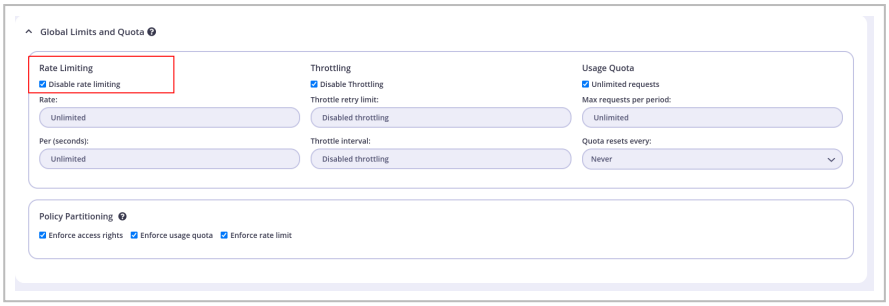

(https://s2.loli.net/2022/08/25/1KWilQdpevuoM7y.png)

然后来到 `Policies` 新建策略，选择你新建的四个 API。



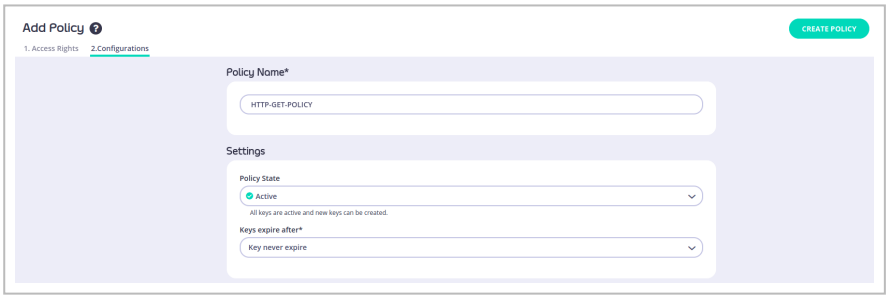

(https://s2.loli.net/2022/08/25/l1HnpmuLrQF6tIX.png)

然后点击 `Global Limits and Quota`，确认禁用 `Rate Limiting`。



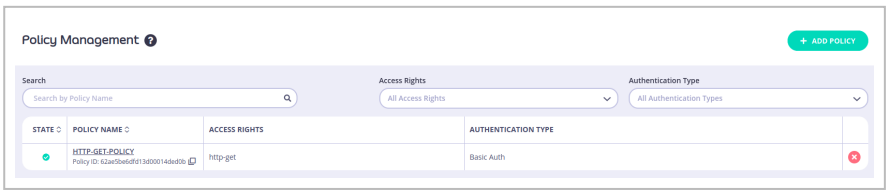

(https://s2.loli.net/2022/08/25/NRX1zoLMPQjOF9f.png)

接着配置策略名称并设置密钥过期时间。




(https://s2.loli.net/2022/08/25/3s257tH9WXQuK8B.png)

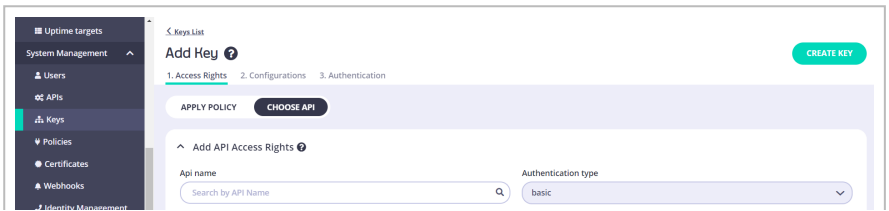
点击 `Create Policy` 以保存新策略，之后可以在 `Policies` 看到它：

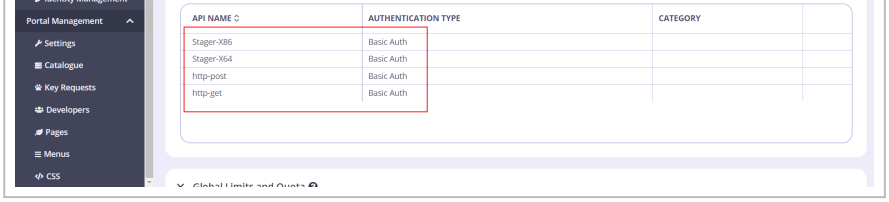



(https://s2.loli.net/2022/08/25/6GE1ygYCP753wdL.png)

# 配置访问验证 Key

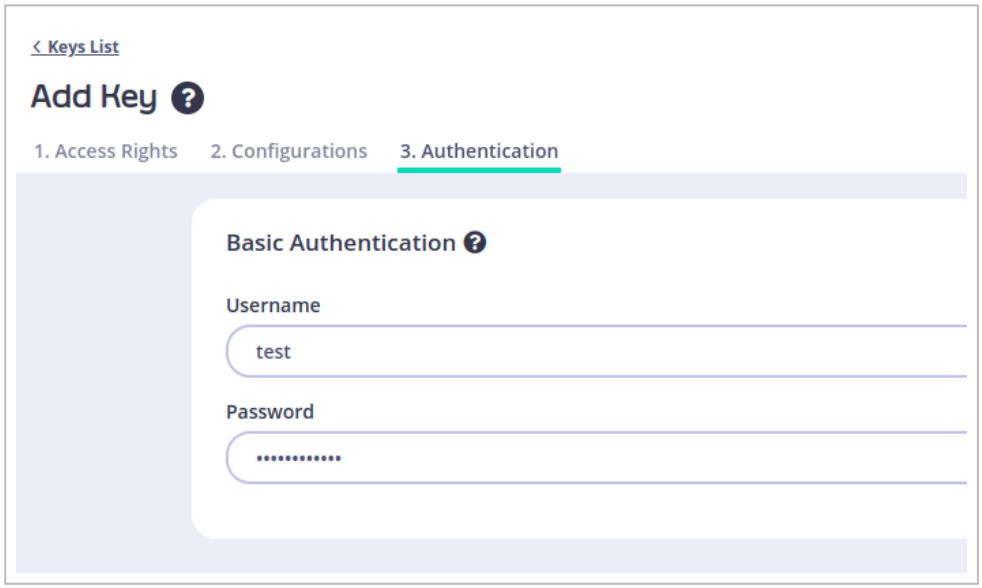
来到 `Keys` ，点击 `ADD KEY` ，然后在 `Apply policy` 选择我们之前创建的策略，并选择 API。





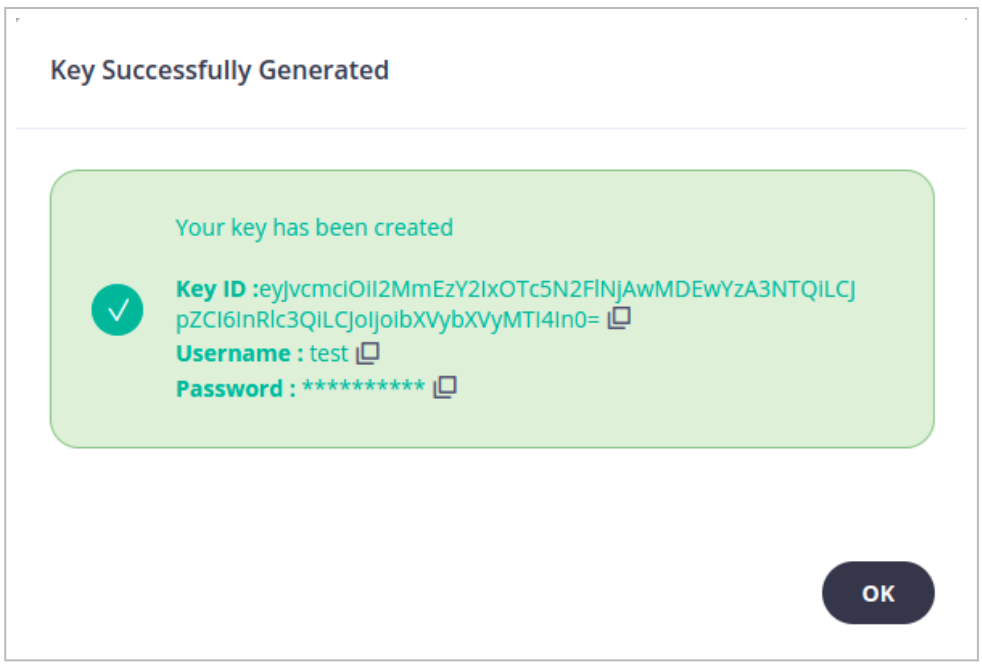

(https://s2.loli.net/2022/08/25/kMVjDTvq6t59m7x.png)

最后来到 `Authentication` 输入需要设置的用户名、密码，这里使用 `test:testtesttest` 作为用户名、密码。




(https://s2.loli.net/2022/08/25/lVfpUkga6KrndHM.png)

`Key` 创建成功后会有如下提示：



(https://s2.loli.net/2022/08/25/OuLAM4y9PdHwSG2.png)

## 配置 C2 配置文件

由于上一步我们设置了访问验证，所以要在 C2 配置文件中添加一个请求头才能正常上线 CS。

`Authorization` 的请求头设置格式如下：

```
Authorization: Basic base64(username:password)
```

所以按上一步添加的 `Key` ，我们要在 C2 配置文件中添加如下请求头：

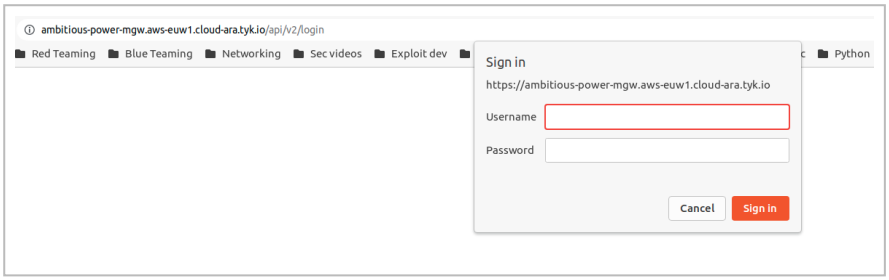
```
Authorization: Basic dGVzdDp0ZXN0dGVzdHRlc3Q=
```

最终配置文件如下：

```
http-get {

    set uri "/api/v2/login";

    client {
        header "Authorization" "Basic dGVzdDp0ZXN0dGVzdHRlc3Q
=";
}
   ....
}

http-post {

    set uri "/api/v2/status";

    client {
        header "Authorization" "Basic dGVzdDp0ZXN0dGVzdHRlc3Q
=";
}
   ....
}
http-stager {
    set uri_x86 "/api/v2/GetProfilePicture";
    set uri_x64 "/api/v2/GetAttachment";

    client {
        header "Authorization" "Basic dGVzdDp0ZXN0dGVzdHRlc3Q
=";
}
    ....
}
```

## 效果

此时直接访问我们设置的 API 的地址都是需要验证的，而 CS 是可以正常上线的。

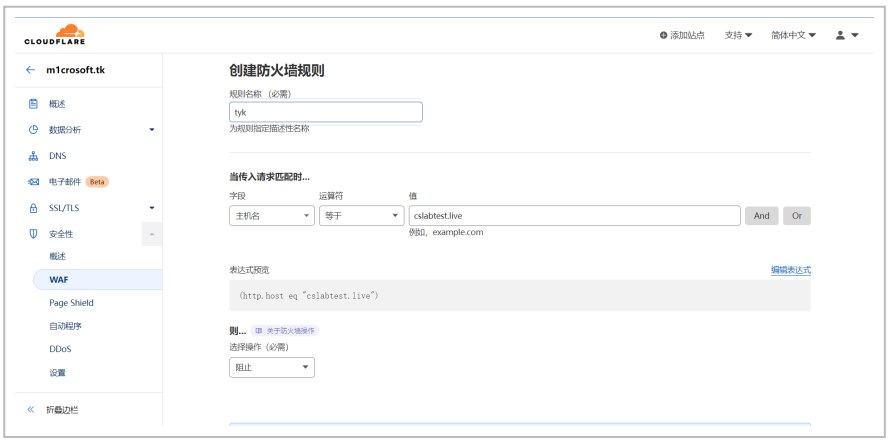



(https://s2.loli.net/2022/08/25/SvRAo7Zy4WBlYxT.png)

# 0x04 配置 CloudFlare 防火墙

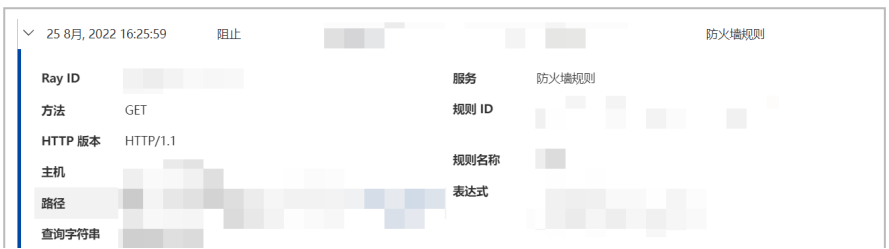
## 获得 tyk.io 特征

在 CF 的 `WAF` 中创建一条防火墙规则，如下：




(https://s2.loli.net/2022/08/25/e5ZWfulg9JhKOY2.png)

然后生成一个木马尝试上线 CS，这时肯定是无法上线的，来到 CF 的概述中可以看到所有拦截记录，点击单条拦截记录查看详细。

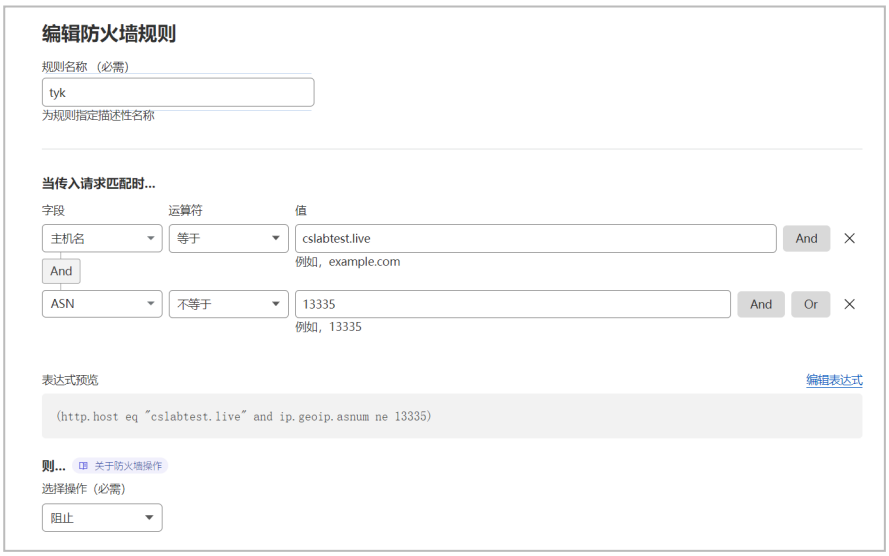



(https://s2.loli.net/2022/08/25/5X98QebcMgPF2Yv.png)

这里有很多的特征可以加到 `WAF` 拦截规则中，来实现只有 `Tyk Gateway API` 转发过来的流量才能允许访问，其他的流量都会阻止。

## 编辑防火墙规则

这里我以 `ASN` 为例子，配置如下：

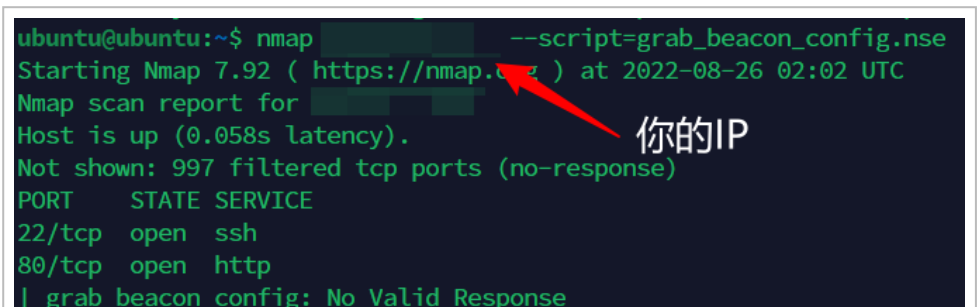


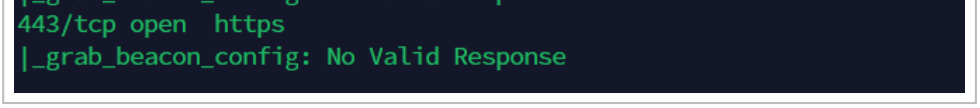

(https://s2.loli.net/2022/08/25/wodcvM8ufmgTYWQ.png)

保存防火墙规则即可。

# 0x05 最终效果

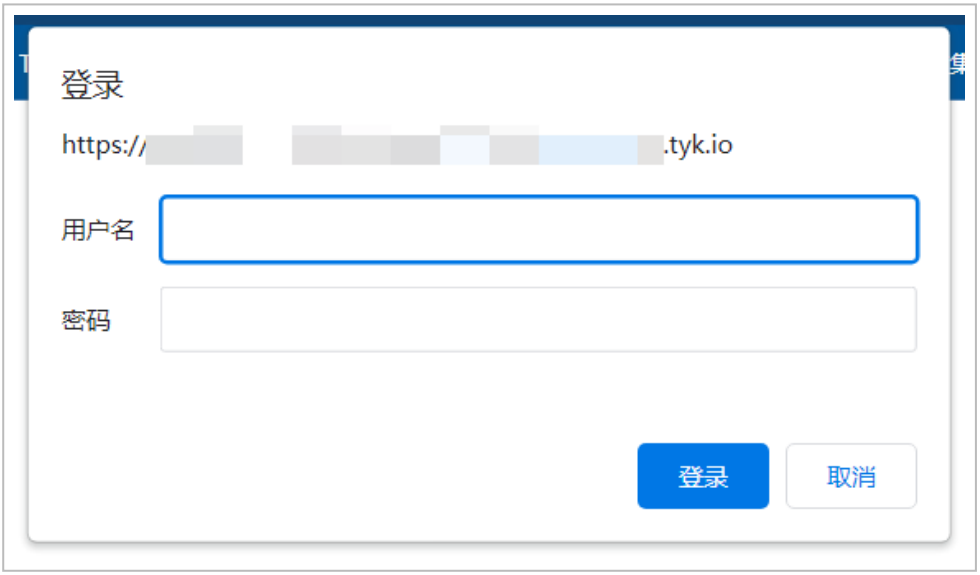
对服务器的扫描，无法获取到我们的 beacon stage。





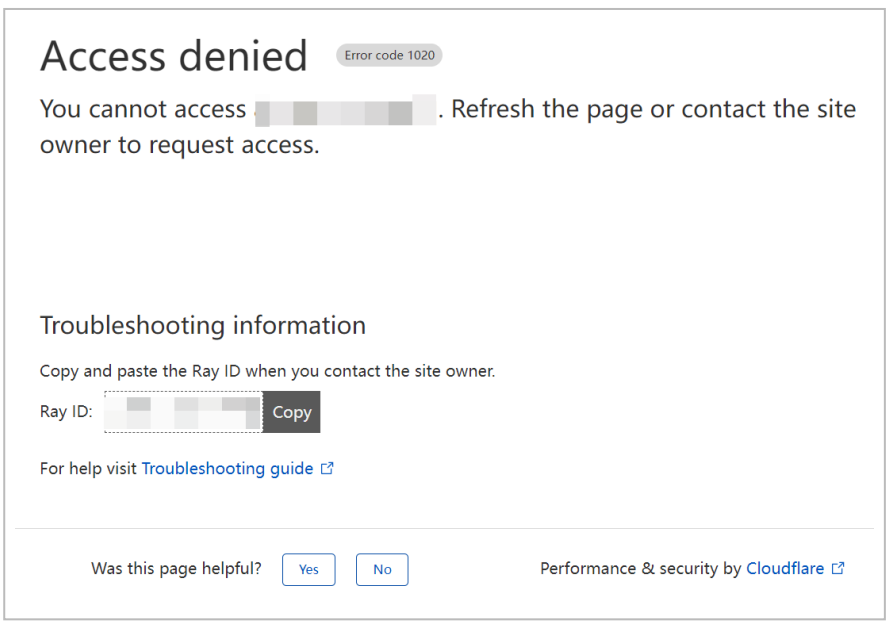

(https://s2.loli.net/2022/08/26/rNz75bAiwQtkERq.png)

直接访问设置的 Tyk Gateway API 的地址是需要验证的。



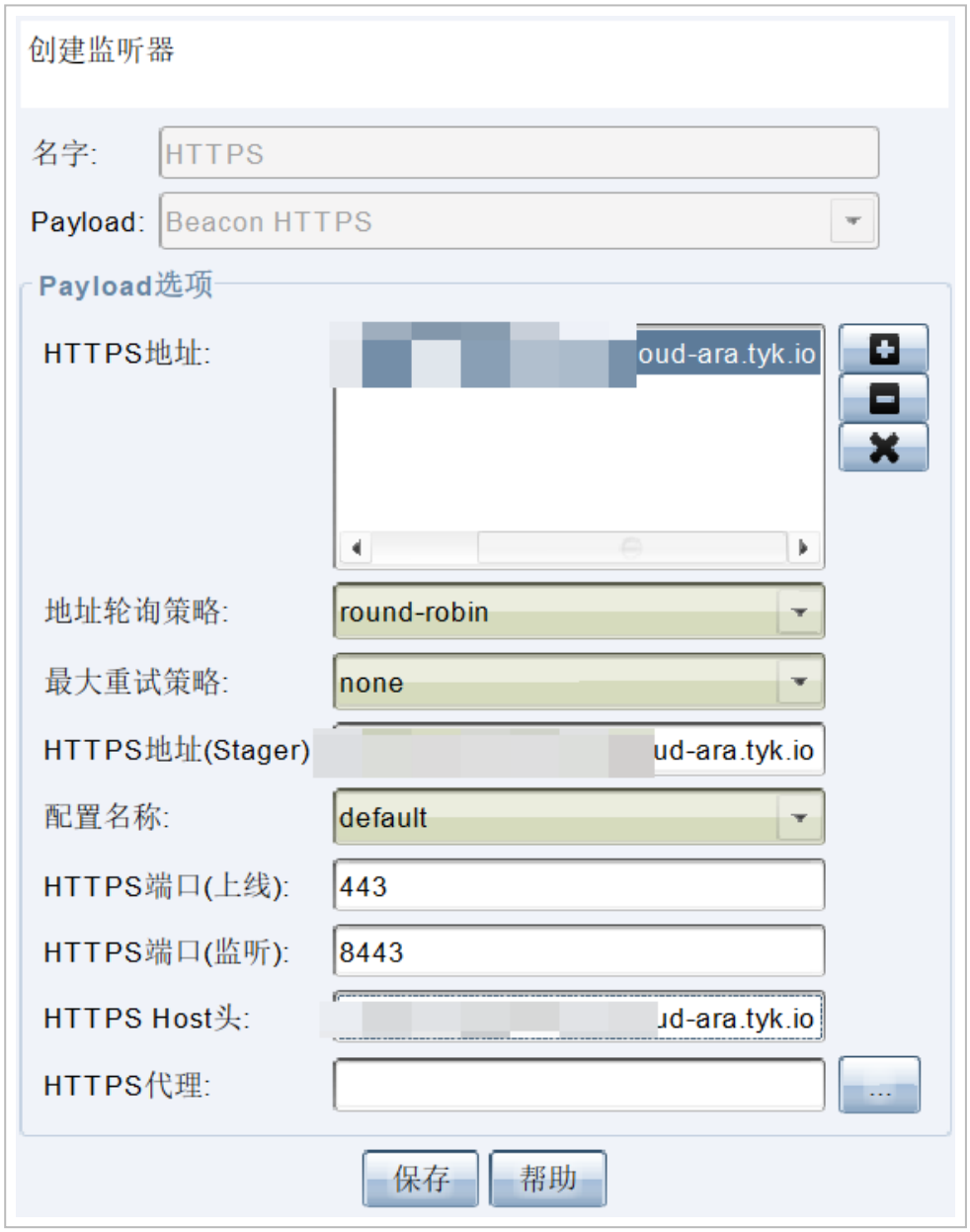

(https://s2.loli.net/2022/08/25/1W3mjGpAMBswySF.png)

直接访问我们的 C2 域名会被 CloudFlare 拦截。



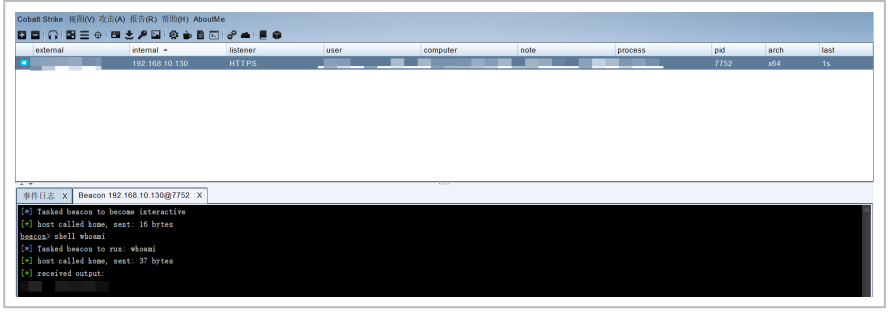
(https://s2.loli.net/2022/08/25/KUnbLxXzH5ECNvp.png)

CS 创建监听器，如下：




(https://s2.loli.net/2022/08/26/m6KodZ7iqONzRXL.png)

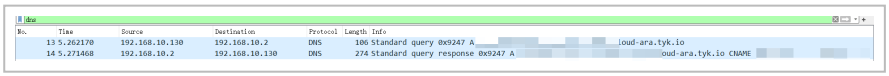
CS 生成木马，可以正常上线和执行命令。




(https://s2.loli.net/2022/08/25/1igUo76aJxB4lM5.png)

# 0x05 问题

- 由于流量经过多次转发，上线可能会有延迟。
- 通过配置 Tyk Gateway API 的域名，使用 HTTPS 的方式上线，流量中会出现 `*.tyk.io` 的 DNS 流量记录，算是一个比较明显的特征。




(https://s2.loli.net/2022/08/26/osOVWLZK19bT8mw.png)

# 0x06 参考

- Oh my API, abusing TYK cloud API management to hide your malicious C2 traffic (https://shells.systems/oh-my-api-abusing-tyk-cloud-api-management-service-to-hide-your-malicious-c2-traffic/)
- cobaltstrike 配置 nginx 反向代理 (https://www.freebuf.com/articles/others-articles/247115.html)
- cs2modrewrite (https://github.com/threatexpress/cs2modrewrite)