

EasyConfig

O que é uma configuração que você pode se perguntar?

Uma configuração [Configuração] é um verificador reduzido para um programa como o OpenBullet, onde ele pode ler as instruções incorporadas!

Como começar [O que você precisa]

Tudo que você vai precisar para este e-book é o seguinte:

- *OpenBullet*
- *Violinista*
- *Nox App Player*
- *0 Conhecimento*

Parte 1. Blocos

REQUEST

- *Solicitar ou a parte do site é onde faremos todas as partes do nosso site, então faça o login e recebendo a página da web*

PARSE

- *Analisar é assim que você pegará o conteúdo que deseja da página ou resposta da página*

FUNCTION

- *Função é onde você encontrará todas as suas funções, como hashing de uma palavra, tradução ou Conversão de Base64*

KEY CHECK

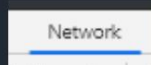
- Key Check esta é a parte mais importante de uma configuração para que funcione, aqui determinaremos se uma conta está funcionando a partir de sua resposta à solicitação

Parte 2. Fazendo uma configuração básica

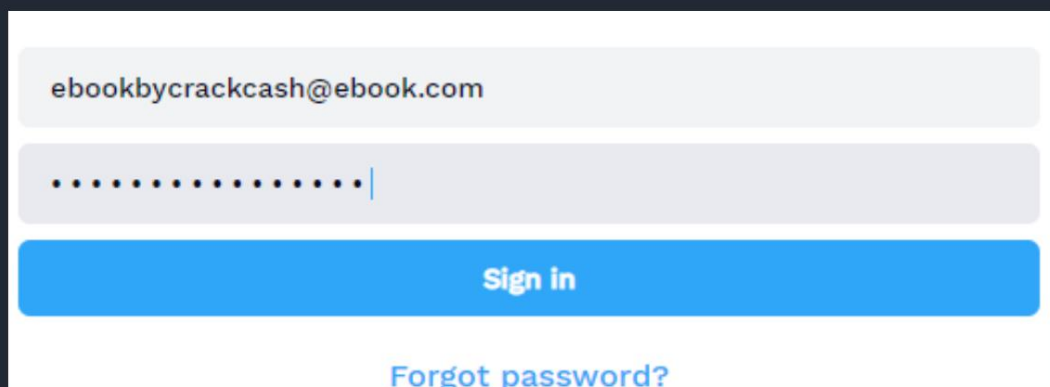
Para este ebook para a configuração básica, vou fazer uma configuração para este site:

<https://account.hotspotshield.com/sign-in/>

Então, antes de tudo, precisamos ver o que está acontecendo quando fazemos login, então tudo o que precisamos fazer é inspecionar o elemento da página de login e ir para esta guia:



Agora podemos inserir quaisquer detalhes de login aleatórios nos campos de e-mail e senha:

A screenshot of a login interface. It features two input fields: the first contains the email 'ebookbycrackcash@ebook.com' and the second contains masked characters (dots) followed by a vertical cursor. Below these fields is a large blue button with the text 'Sign in'. At the bottom of the form, there is a blue link that says 'Forgot password?'.

E pressione **Entrar**.

Agora você deve ver algo assim na guia de rede.

The screenshot shows a web browser with a 'Sign in' form. The form has a text input field containing 'bookbycrackcash@ebook.com', a password input field with dots, and a blue 'Sign in' button. Below the button are links for 'Forgot password?' and 'New member? Create account'. To the right, the browser's developer tools are open, showing the 'Network' tab with a list of requests. The first request is 'sign-in', which is selected. The 'Headers' tab is active, showing the 'Name' and 'Value' columns. The 'Name' column lists various cookies and headers, and the 'Value' column shows their corresponding values.

Vamos nos concentrar no login, no início pode ser um pouco confuso, então vou detalhar o que você precisa:

Request URL: `https://account.hotspotshield.com/sign-in`
Request Method: `POST`

- URL Para solicitação, método de solicitação para o método de solicitação em nossa solicitação de url no openbullet

content-type: `application/x-www-form-urlencoded; charset=UTF-8`

- Tipo de conteúdo para solicitação Openbullet e todos os últimos cabeçalhos

origin: `https://account.hotspotshield.com`
referer: `https://account.hotspotshield.com/sign-in/`
sec-fetch-mode: `cors`
sec-fetch-site: `same-origin`
user-agent: `Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/76.0.3809.132 Safari/537.36`
x-requested-with: `XMLHttpRequest`

- para nossos cabeçalhos de solicitação

▼ **Form Data** view source view URL encoded

login: `bookbycrackcash@ebook.com`
pwd: `bookbycrackcash`
remember: `0`
ajax: `true`
vxstok: `5f21568197131:J3iDgePwDxQd_8jCozbrwEv2q5KvrMQo9HiwmnMma7I`

- E por último nossos dados de postagem

CLIQUE NISSO:

view source

```
login=ebookbycrackcash%40ebook.com&pwd=ebookbycrackcash&remember=0&ajax=true&vxstok=5f21568197131%3AJ3iDgePwDxQd_8jCozbRwEv2q5KVrMQo9HiWmnMma7I
```

Agora deve ser legível para OpenBullet:

[Eu recomendaria salvar todas essas informações em um documento de texto]

```
Request URL: https://account.hotspotshield.com/sign-in
Request Method: POST
content-type: application/x-www-form-urlencoded; charset=UTF-8
origin: https://account.hotspotshield.com
referer: https://account.hotspotshield.com/sign-in/
sec-fetch-mode: cors
sec-fetch-site: same-origin
user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/76.0.3809.132 Safari/537.36
x-requested-with: XMLHttpRequest
login=ebookbycrackcash%40ebook.com&pwd=ebookbycrackcash&remember=0&ajax=true&vxstok=5f21568197131%3AJ3iDgePwDxQd_8jCozbRwEv2q5KVrMQo9HiWmnMma7I
```

Agora, muitas vezes você verá coisas adicionadas aos dados do post, como csrf_token e muito mais

```
e&vxstok=5f21568197131%3AJ3iDgePwDxQd_8jCozbRwEv2q5KVrMQo9HiWmnMma7I
```

Portanto, precisaremos obter isso, o primeiro local a ser verificado será a página de login, então clique com o botão direito na página e clique em visualizar fonte da página.

Ele irá redirecioná-lo para uma página como esta:

 view-source:account.hotspotshield.com/sign-in/

Agora vamos procurar o termo, neste caso [vxstok] :



E nós encontramos!

```
ajax: true,
vxstok: '6b21568197595:YARbn7N8ABiypPUnkeWt9RxJzUc4FfuZ2Fv0Y58p6xA'
```

Então copie e cole a linha inteira em nosso documento de texto:

```
Request URL: https://account.hotspotshield.com/sign-in
Request Method: POST
content-type: application/x-www-form-urlencoded; charset=UTF-8
origin: https://account.hotspotshield.com
referer: https://account.hotspotshield.com/sign-in/
sec-fetch-mode: cors
sec-fetch-site: same-origin
user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/76.0.3809.132 Safari/537.36
x-requested-with: XMLHttpRequest
login=ebookbycrackcash%40ebook.com&pwd=ebookbycrackcash&remember=0&ajax=true&vxstok=5f21568197131%3AJ3iDgePwDxQd_8jCozbRwEv2q5KVrMQo9HiWmnMma7I
vxstok: '6b21568197595:YARbn7N8ABiypPUnkeWt9RxJzUc4FfuZ2Fv0Y58p6xA'
```

Agora estamos prontos para começar a fazer a configuração:

- Então, antes de tudo, precisaremos obter esse token [vxstok] .

REQUEST	URL: https://account.hotspotshield.com/sign-in/
	Method: GET ☒ Auto Redirect

Agora vamos analisar o token

estamos querendo analisar esta parte da linha

vxstok: '6b21568197595:YARbn7N8ABiypPUnkeWt9RxJzUc4FfuZ2Fv0Y58p6xA'

Agora vamos definir isso:

Mode: ☒ LR ☐ CSS ☐ JSON ☐ REGEX
Left string:
Right string:
☐ Recursive ☐ Enc. Output ☒ Create Empty ☐ Use Regex
Automatic mode:

À esquerda temos - vxstok: '

E à direita temos -

REQUEST	Parse: <SOURCE>
	☐ Is Capture Var/Cap Name:
PARSE	Prefix: Suffix:
	Mode: ☒ LR ☐ CSS ☐ JSON ☐ REGEX
	Left string: vxstok: '
	Right string: '

Agora podemos clicar em iniciar para verificar se está funcionando

Como você pode ver funcionou:

☐ Start	☐ SBS	Step	Data:	MailPass
Proxy: ☐	OFF			Http
Data	Log	HTML View		
BOT STATUS: NONE VARIABLES: (Single) = a671568197926:y7GTjuFANnYUco0ApZWGlbeCwnczSuInLnVIjpe1LP4				

Mas não tem nome, então daremos um nome ao token que acabamos de analisar.

Is Capture

Var/Cap Name: token

fix:

Suffix:

de:

☒ LR ☐ CSS ☐ JSON ☐ REGEX

ht string:

vxsf tok: '

ht string:

'

☐ Recursive ☐ Enc. Output ☒ Create Empty ☐ Use Regex

Agora estamos lendo para fazer a parte de login, então voltando às nossas notas

No entanto, desta vez na solicitação, precisaremos alterar o método para o nosso método em nossas notas:

Method:

GET

▼

Request URL: https://a

Request Method: POST

Agora precisamos preencher todas as nossas outras informações:

URL:

https://account.hotspotshield.com/sign-in

Method:

POST

▼

☒ Auto Redirect

☒ Read Resp. Source ☐ Parse GET Query ☐ Encode Content

☒ Accept-Encoding

Request Type:

Standard

▼

POST Data:

Content-Type:

application/x-www-form-urlencoded

▼

Custom Cookies:

Custom Headers:

origin: https://account.hotspotshield.com
referer: https://account.hotspotshield.com/sign-in/
sec-fetch-mode: cors
sec-fetch-site: same-origin
user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/76.0.3809.132 Safari/537.36
x-requested-with: XMLHttpRequest

Response Type:

String

▼

Agora veremos a caixa POST Date.

Aqui precisaremos colocar em nosso post que recebemos do login.

```
login=ebookbycrackcash%40ebook.com&pwd=ebookbycrackcash&remember=0&ajax=true&vxstok=5f21568197131%3A731DgePWdXQd_8jCozbRwEv2q5KvRmQo9HiWmnMma7I
```

No entanto, precisamos substituir o e-mail, senha e token

Para isso, substituiremos o e-mail por <USER> e a senha por <PASS> .

login=<USER>&pwd=<PASS>&remember=0&ajax=true&vxstok=5f21568197131%3AJ3iDgePWDxQd_8 jCozbRwEv2q5KVrMQo9HiWmnMma7I

Agora precisamos substituir o token, então precisamos fazer o mesmo, mas colocar nosso nome de análise que é **token**:

```
login=<USER>&pwd=<PASS>&remember=0&ajax=true&vxstok=<token>
```

Request Type: Standard

POST Data: login=<USER>&pwd=<PASS>&remember=0&ajax=true&vxsfok=<token>

Agora estamos ordenados. Agora podemos colocar qualquer login aqui:

Start ☐ SBS Step Data: MailPass

E veja que resposta podemos obter. Então, como você pode ver, esta é a nossa resposta:

HTML View

```
{"result":0,"error":"The username and/or password is incorrect","debug":""}
```

Isso usaremos como nossa chave de falha.

Então vamos fazer um **bloco de verificação de chave**.

REQUEST

PARSE

REQUEST

KEY CHECK

☐ Insta Ban 4xx

☒ Ban if no key found

Keychains: +

Como você pode ver, não há nada lá.

Então, precisamos clicar neste botão:

Keychains: +

Type: Success ▼ Mode: OR ▼

Precisamos mudá-lo de **SUCESSO** PARA **FALHA**.

Type: Failure ▼ Mode: OR ▼

Keys +

Agora precisamos clicar neste botão:

Keys +

Agora vai aparecer isso:

Keys +

<SOURCE> ▼ Contains ▼

Vamos colocar a resposta que recebemos na caixa à direita:

Keys +

<SOURCE> ▼ Contains ▼ ie and\\or password is incorrect","debug":""}

Agora vamos clicar em iniciar novamente para ter certeza de que funciona:

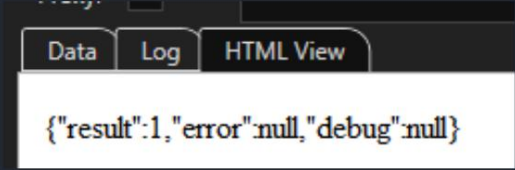
BOT STATUS: FAIL
VARIABLES:

Como você pode ver, funciona enquanto lê nossa chave de falha!

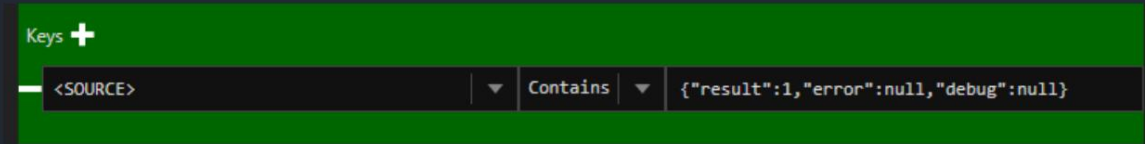
Agora tudo o que precisamos fazer é adicionar uma chave de sucesso.

Então tudo que você precisa fazer é criar uma conta para o site.

Agora, quando colocamos em uma conta de trabalho, a resposta muda.



Portanto, esta será a nossa chave de sucesso:



Agora esta configuração estará funcionando! Parabéns você fez uma configuração funcional!

Agora vamos rachar com ele para ter certeza de que funciona!

- Observe se a verificação está lenta, isso é normal, pois o site possui proteção anti-craqueamento cloudflare

Parte 3. Configurações avançadas

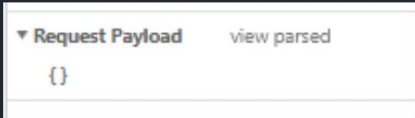
O que é uma configuração avançada?

Uma configuração avançada é uma configuração muito mais difícil de fazer do que uma configuração básica para que possa conter informações em hash nos cabeçalhos ou postdata.

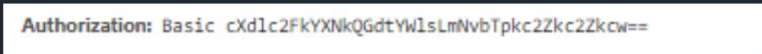
Então, para esta configuração avançada, faremos o [Uplay!](#)

Então, já sabemos como obter todos os nossos dados de postagem e outras coisas, no entanto, o Uplay é um pouco diferente:

Estes são os dados da postagem do Uplay:



Porém com o uplay o login fica nos cabeçalhos conforme mostrado abaixo:



Se o login não estiver nos dados do post, ele estará nos cabeçalhos normalmente assim. Agora não se preocupe, será sempre o mesmo e é muito simples.

Este é um hash base64, que pode ser descriptografado muito facilmente com o openbullet.

Então precisamos fazer um bloco de funções

FUNCTION	Variable Name:	
	Input string:	
	☐ Is Capture	
Function:		Base64Decode

Agora na entrada podemos colocar o hash base64:

Input string:	cXd1c2FkYXNkQGdtYW1sLmNvbTpkc2Zkc2Zkcw==
---------------	--

Agora vamos clicar em iniciar e ele irá decodificá-lo:

```

BOT STATUS: NONE
VARIABLES:
  (Single) = qwesadasd@gmail.com:dsfdsfds
  
```

Então agora sabemos que é isso que está criptografando, para que possamos reverter a função para codificação da decodificação, no entanto, colocaremos:

<USUÁRIO>:<PASS>

Variable Name:	
Input string:	<USER>:<PASS>
☐ Is Capture	
Function:	Base64Encode

Agora vamos clicar em iniciar:

```

BOT STATUS: NONE
VARIABLES:
  (Single) = amV6YWlsczEyM0BnbWFPbC5jb206TDIzMWVkbMTIzZDI=
  
```

Agora sua codificação.

Precisamos dar a isso um nome que usaremos com Base64.

Variable Name:	Base64
Input string:	<USER>:<PASS>

Agora podemos fazer nosso pedido. No entanto, o tipo de conteúdo é:

Content-Type: application/json

Precisamos mudar:

Content-Type:

application/json

Authorization: Basic cXd1c2FkYXNkQGdtYWlsLmNvbTpkc2Zkc2Zkcw==

Sec-Fetch-Mode: cors

Ubi-AppId: c5393f10-7ac7-4b4f-90fa-21f8f3451a04

Ubi-RequestedPlatformType: uplay

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)

Chrome/76.0.3809.132 Safari/537.36

Na Autorização, precisamos alterar o base64 para o nome da nossa variável:

Custom Headers:

Authorization: Basic <Base64>

Sec-Fetch-Mode: cors

Agora podemos clicar em iniciar e obter nossos dados de resposta:

{"message":"Invalid credentials","errorCode":3,"httpCode":401,"errorContext":"Profiles Client","moreInfo":"","transactionTime":"2019-09-11T11:01:19.1623076Z","transactionId":"9786ede0-a262-49b7-afe2-2099cd1b8878","environment":"PROD"}

Então, como você pode ver, esta é a nossa chave de falha!

No entanto, não precisamos de tudo isso apenas a parte principal

`{"message":"Credenciais inválidas","`

REQUEST

KEY CHECK

Type: Failure Mode: OR

Keys +

<SOURCE>

Contains

{"message":"Invalid credentials","

Agora precisamos fazer uma conta válida e obter a chave do sucesso! Depois disso terminamos.

Type: Success Mode: OR

Keys +

<SOURCE>

Contains

{"platformType":"uplay","ticket":

Parte 4. Capturar!

Para tutorial de captura é este ebook que estaremos usando o **Spotify!**

Então, depois de fazer sua configuração, vamos querer adicionar captura!

Portanto, na guia Rede, mantenha-o aberto e vá para a página onde deseja capturar.

Para Spotify é:

<https://www.spotify.com/uk/account/subscription/>

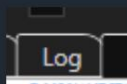
Então vamos fazer um pedido e pegar a página.

Queremos analisar a assinatura.

Subscription and payment

Spotify Free

Então vamos nessa aba:



E procure por **Spotify Free** :

```
"><h3 class="product-name">Spotify Free</h3><p
```

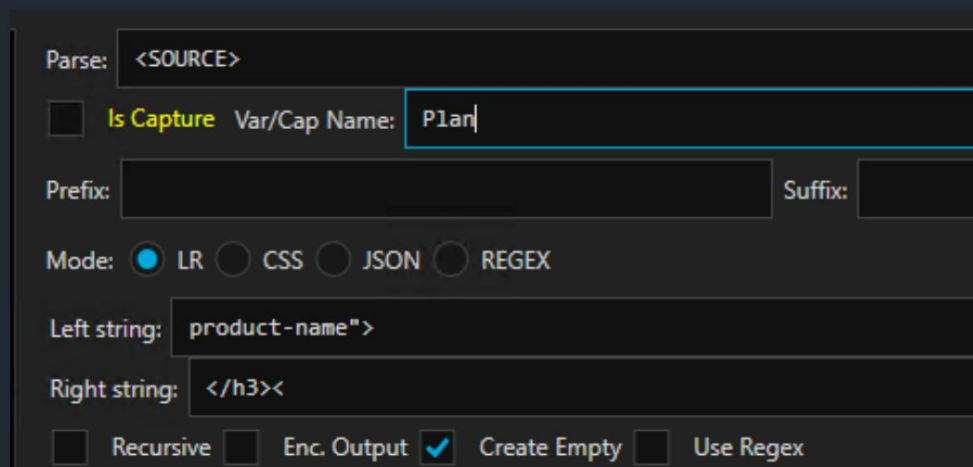
Como antes, queremos capturar a parte intermediária, então vamos para um bloco de análise.

À esquerda temos:

```
=<i>"nome do produto">
```

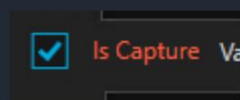
E à direita temos:

```
</h3><
```



The screenshot shows the 'New Capture Rule' dialog in Fiddler. The 'Parse' field contains '<SOURCE>'. The 'Is Capture' checkbox is checked, and the 'Var/Cap Name' field contains 'Plan'. The 'Prefix' and 'Suffix' fields are empty. The 'Mode' is set to 'LR'. The 'Left string' field contains 'product-name">' and the 'Right string' field contains '</h3><'. At the bottom, the 'Recursive' checkbox is unchecked, 'Enc. Output' is unchecked, 'Create Empty' is checked, and 'Use Regex' is unchecked.

Agora, porque é uma configuração de captura, precisamos marcar isso:



Agora podemos clicar em iniciar para ver se funciona!

```
BOT STATUS: SUCCESS
VARIABLES:
Plan (Single) = Spotify Free
```

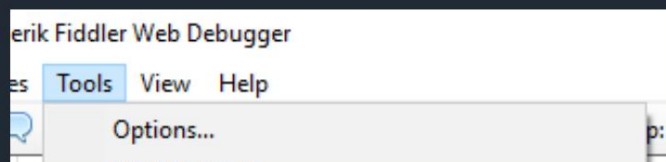
Como você pode ver , funciona!

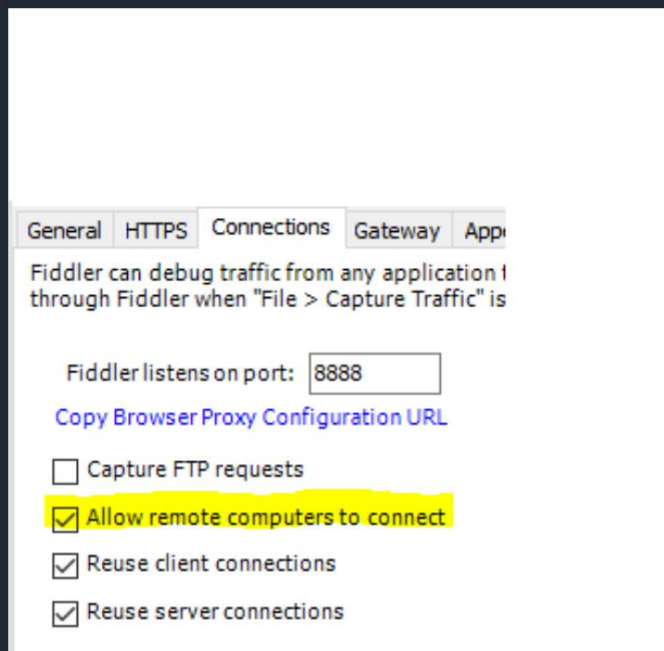
Parabéns, agora você pode adicionar captura!

Parte 5. Obtendo a API para dispositivos móveis

Para isso, você precisará do Fiddler e do NOX baixados [Você precisará de uma placa gráfica]

Depois de baixar o Fiddler, clique em Ferramentas e depois em Opções:

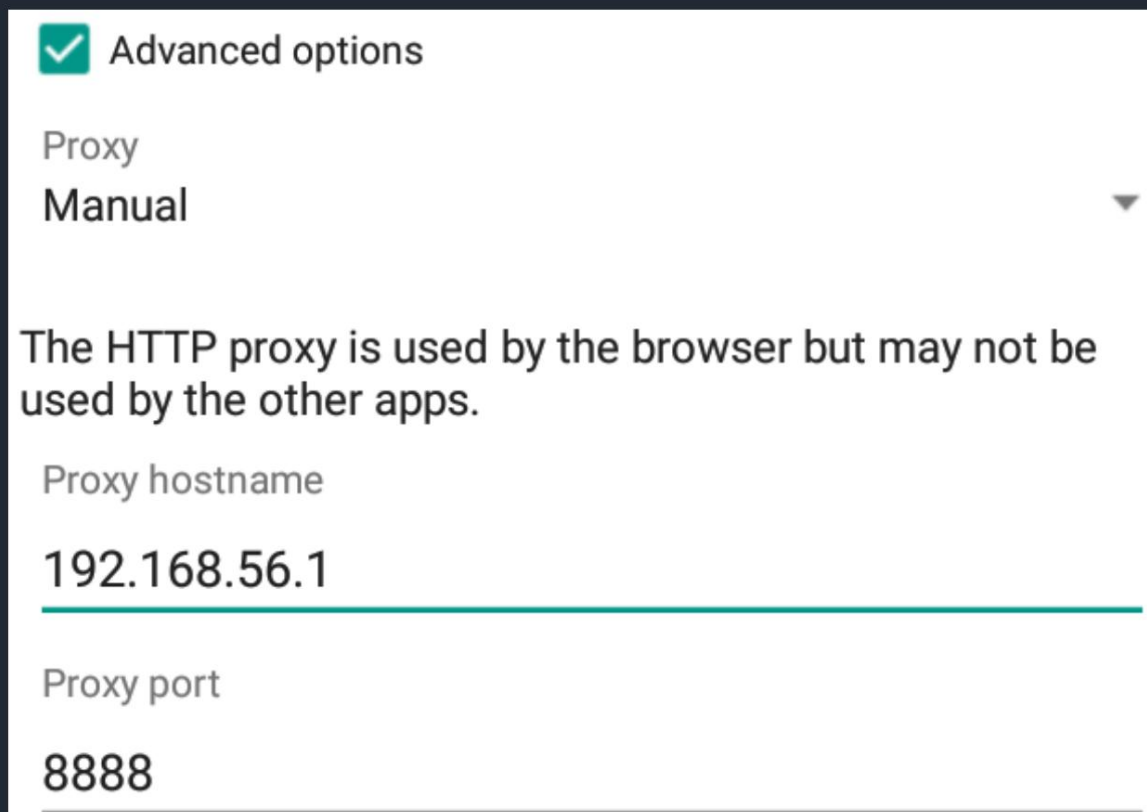




Clique aqui e reinicie o Fiddler

Agora abra NOX.

Vá para configurações e clique em WI-FI.

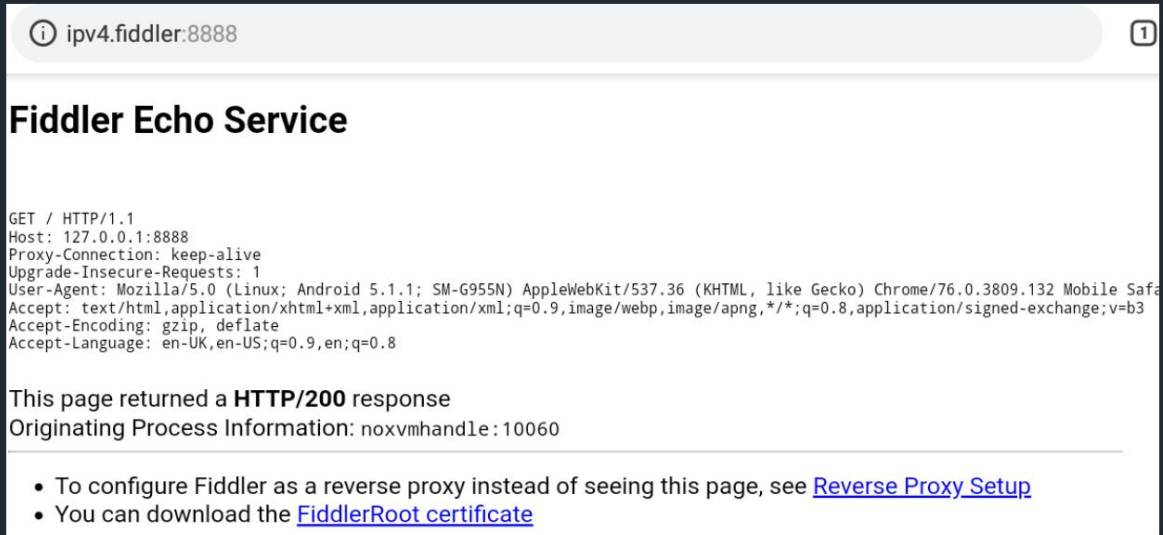


Para obter o nome do host no Fiddler, passe o mouse sobre este ícone:



Agora abra o navegador no nox e digite esta url:

ipv4.fiddler:8888

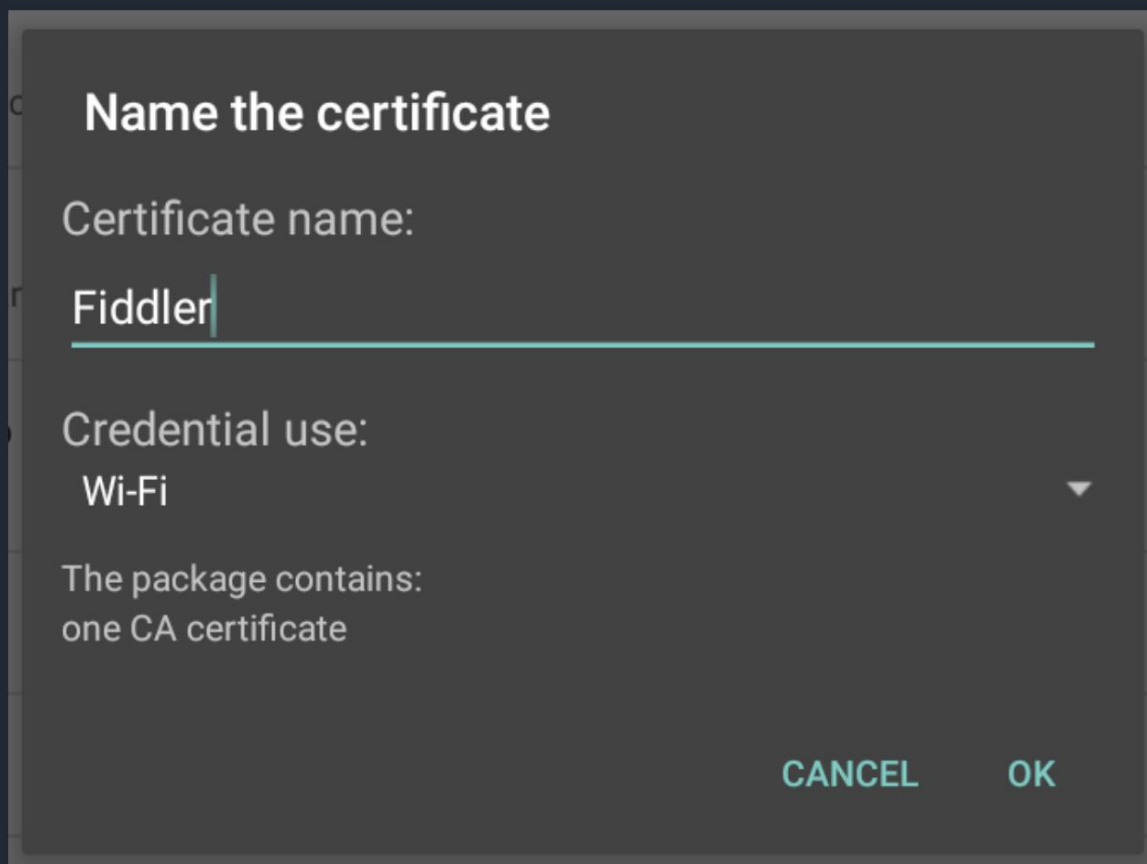


Isso vai trazer isso.

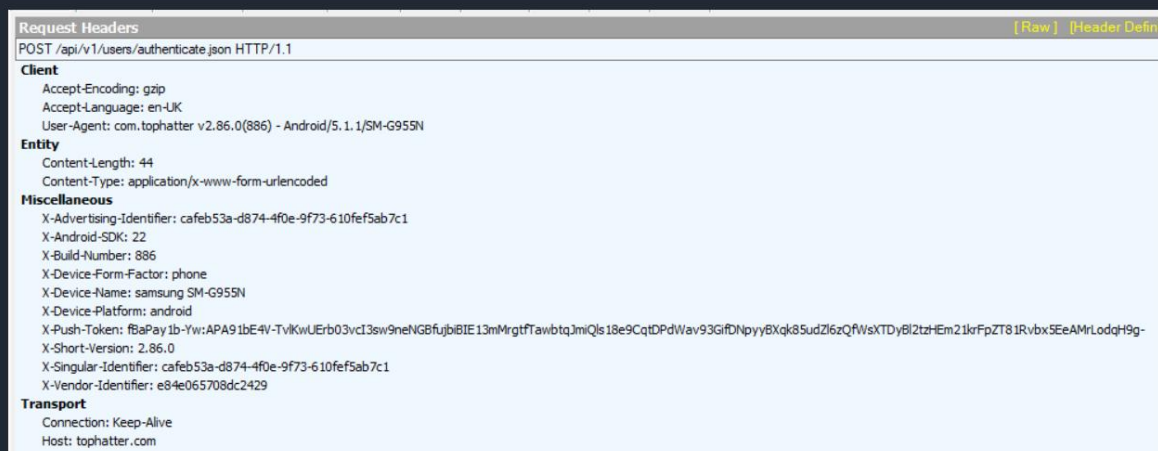
Clique neste:

- You can download the [FiddlerRoot certificate](#)

E instale o certificado.



Agora seremos capazes de começar a sniffar a solicitação dos aplicativos.



Agora siga o tutorial de criação de configuração avançada! e em breve você saberá o que fazer!

Parte 6. Vendendo configurações!

Existem muitas maneiras de vender configuração, mas a maneira mais rápida é vender solicitações de configuração personalizadas!

De qualquer forma espero que tenha gostado do ebook e obrigado por comprar!