**Hiding Traffic from C2 Servers**

**The problem:** Clients notice suspicious traffic and trace it back to your server.

**The solution(s):**

1. Ngrok. Ngrok (<https://ngrok.com>) is a free service that provide secure tunnels to a local host. This means the traffic cannot be traced back to you.

Pros

* Free SSL certificate
* Un-traceable
* Free, easy to use
* Supports forwarding http, tcp

Cons

* Traffic to ngrok still looks suspicious
* Unless you pay, you are only allowed one tunnel at a time (non-http)
* For non-http forwarding, you must sign up for a free account

1. Google Cloud Console. Google provide a free cloud console. However, changes aren’t saved over sessions and you need a Google account.

Pros

* Free (with Google account)
* 20 GB storage (session is removed after 20 minutes)
* Fast internet connection
* Fast
* Traffic sent to Google which is usually unblocked

Cons

* Google probably tracks you
* Sessions get deleted after 20 minutes inactivity

1. Hiding traffic through other services.
2. Dropbox. A program called DBC2 (DropBox C2) was developed as a Command and Control server that communicates through Dropbox.

Pros

* Free
* Un-traceable
* All traffic is sent to Dropbox
* The Client program does not need the API key

Cons

* You need a dropbox account
* Can be slow sometimes

1. Twitter. A program (written by me) called TwitterC2 (<https://github.com/AgeOfMarcus/TwitterC2>) was developed as a Command and Control program that communicated via Twitter.

Pros

* + All traffic is sent through twitter (which tends to be unblocked for marketing)

Cons

* + You need 2 twitter accounts
  + The API key is needed in the Client program (for the slave account)
  + Speed can sometimes be slow
  + Files can’t be transferred (as far as I’ve tried)