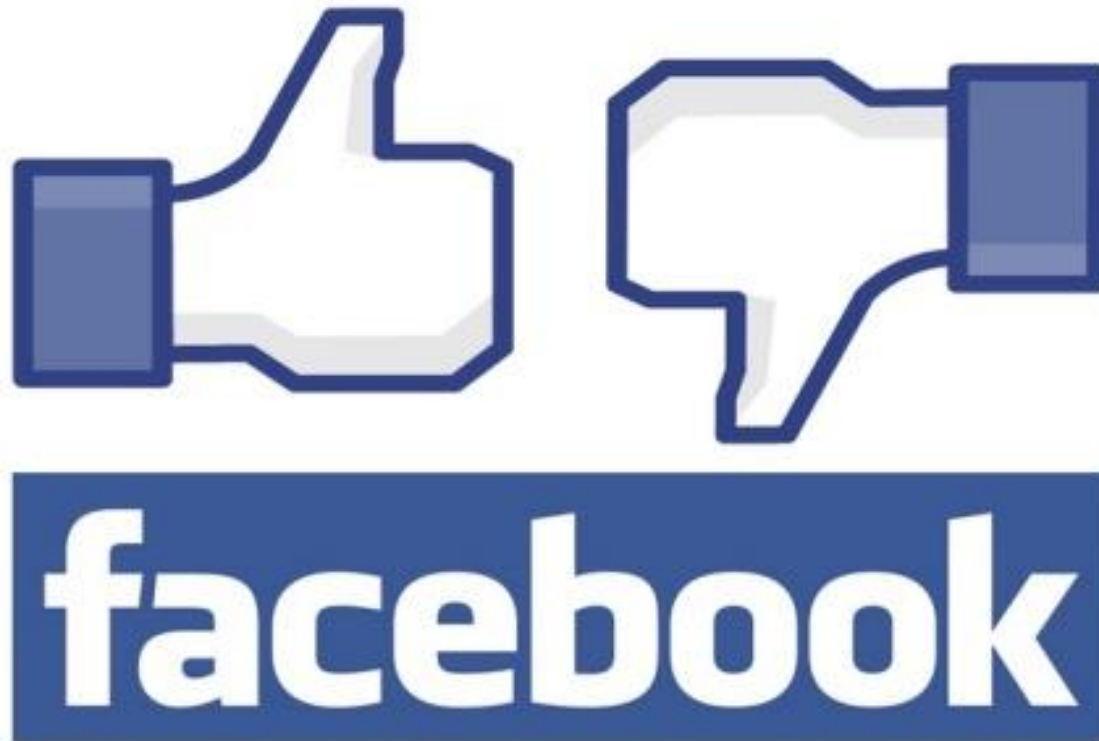




# Hackeando o Facebook

fmasanori@gmail.com

# Facebook Hacking



# Facebook Hacking



Hackers não são “do mal”?

# Facebook Hacking

*“O Facebook é dirigido por hackers. A maioria das pessoas pensa de um hacker como alguém que invade um sistema de computador. Mas nós vemos isso como uma filosofia. Aqui, os hackers assumem que sempre há uma maneira melhor, mais eficiente de resolver problemas.” [facebook.com/careers](https://www.facebook.com/careers) (2012)*

# Facebook Profile

```
import urllib.request
import json
from pprint import pprint
```

```
url = 'https://graph.facebook.com/fmasanori'
resp = urllib.request.urlopen(url).read()
data = json.loads(resp.decode('utf-8'))
pprint (data)
```

```
>>>
```

```
{ 'first_name': 'Fernando',
  'gender': 'male',
  'id': '1183621847',
  'last_name': 'Ashikaga',
  'locale': 'en_US',
  'middle_name': 'Masanori',
  'name': 'Fernando Masanori Ashikaga',
  'username': 'fmasanori' }
```

<https://gist.github.com/4667205>

# Facebook Profile Photo

```
import urllib.request
import json

user = 'fmasanori'
url = 'https://graph.facebook.com/'+user+'/picture?type=large'
figura = urllib.request.urlopen(url).read()

arquivo = user + '.jpg'
f = open(arquivo, 'wb')
f.write(figura)
f.close()

print(arquivo, 'gravado no seu diretório...')
```



<https://gist.github.com/4667293>

# Graph API

- <https://developers.facebook.com/docs/reference/api/examples/>
- **Objetos JSON**
- **Links autenticados**
- **Obs.: access\_token expira**
- **Autenticar novamente neste caso**

# Facebook Friends

facebook developers

Search Facebook Developers

Docs Tools Support News Apps

Fernando

Objects

- Achievement(Instance)
- Album
- Application
- Checkin
- Comment
- Domain
- Errors
- Event
- FriendList
- Group
- Insights
- Link
- Message
- Note
- Offer
- Order
- Page
- Photo
- Pictures

Connections

- Friends: [https://graph.facebook.com/me/friends?access\\_token=...](https://graph.facebook.com/me/friends?access_token=...)
- News feed (this is an outdated view, does not reflect the News Feed on facebook.com): [https://graph.facebook.com/me/home?access\\_token=...](https://graph.facebook.com/me/home?access_token=...)
- Profile feed (Wall): [https://graph.facebook.com/me/feed?access\\_token=...](https://graph.facebook.com/me/feed?access_token=...)
- Likes: [https://graph.facebook.com/me/likes?access\\_token=...](https://graph.facebook.com/me/likes?access_token=...)
- Movies: [https://graph.facebook.com/me/movies?access\\_token=...](https://graph.facebook.com/me/movies?access_token=...)
- Music: [https://graph.facebook.com/me/music?access\\_token=...](https://graph.facebook.com/me/music?access_token=...)
- Books: [https://graph.facebook.com/me/books?access\\_token=...](https://graph.facebook.com/me/books?access_token=...)
- Notes: [https://graph.facebook.com/me/notes?access\\_token=...](https://graph.facebook.com/me/notes?access_token=...)
- Permissions: [https://graph.facebook.com/me/permissions?access\\_token=...](https://graph.facebook.com/me/permissions?access_token=...)
- Photo Tags: [https://graph.facebook.com/me/photos?access\\_token=...](https://graph.facebook.com/me/photos?access_token=...)
- Photo Albums: [https://graph.facebook.com/me/albums?access\\_token=...](https://graph.facebook.com/me/albums?access_token=...)
- Video Tags: [https://graph.facebook.com/me/videos?access\\_token=...](https://graph.facebook.com/me/videos?access_token=...)
- Video Uploads: [https://graph.facebook.com/me/videos/uploaded?access\\_token=...](https://graph.facebook.com/me/videos/uploaded?access_token=...)
- Events: [https://graph.facebook.com/me/events?access\\_token=...](https://graph.facebook.com/me/events?access_token=...)
- Groups: [https://graph.facebook.com/me/groups?access\\_token=...](https://graph.facebook.com/me/groups?access_token=...)
- Checkins: [https://graph.facebook.com/me/checkins?access\\_token=...](https://graph.facebook.com/me/checkins?access_token=...)
- Objects with Location: [https://graph.facebook.com/me/locations?access\\_token=...](https://graph.facebook.com/me/locations?access_token=...)

Copiar o link

<https://developers.facebook.com/docs/reference/api/examples/>



# Facebook Friends

```
import urllib.request
import json

url = 'copie aqui o link Connections Friends'
resp = urllib.request.urlopen(url).read()
data = json.loads(resp.decode('utf-8'))
for amigo in data['data']:
    print (amigo['name'])
```

<https://gist.github.com/4674792>

# Facebook Friends Photos

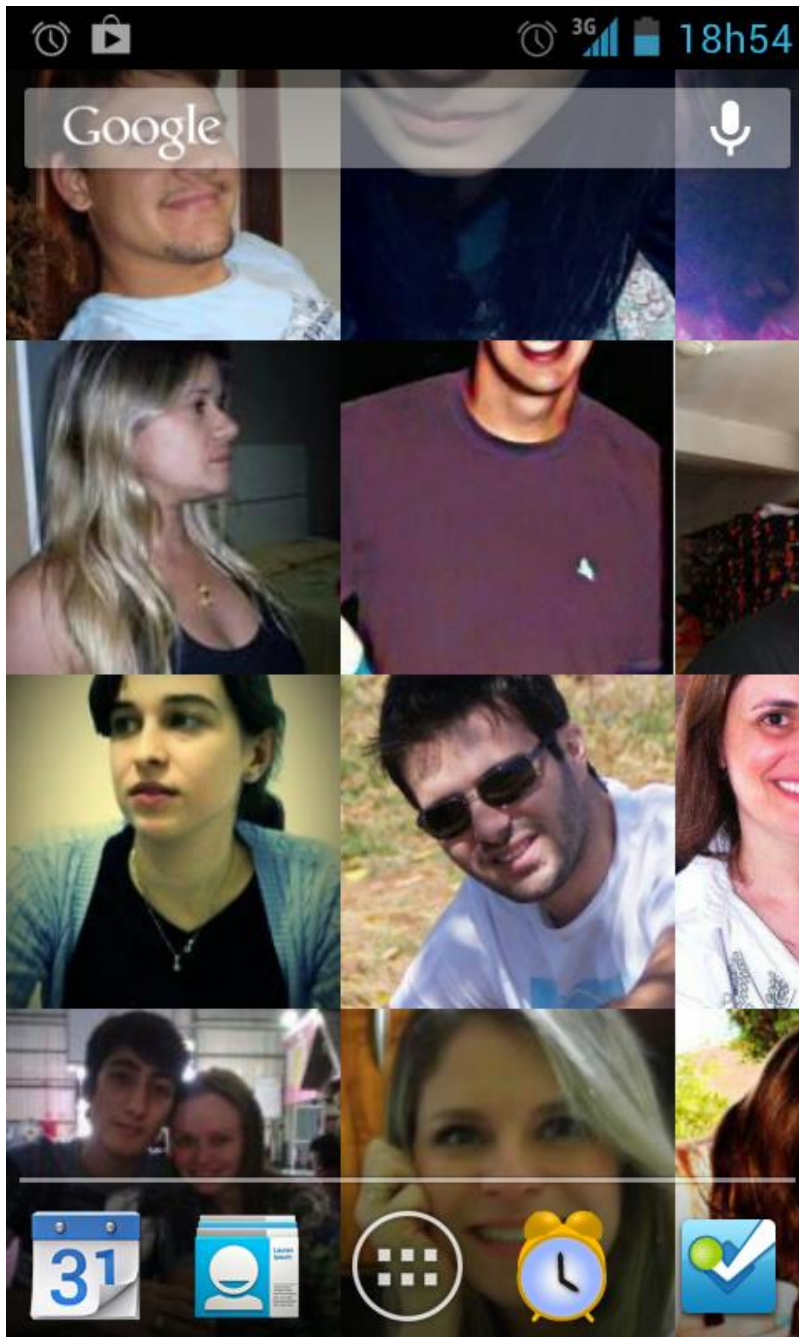
```
import urllib.request
import json

def grava_imagem(amigo):
    size = '/picture?width=200&height=200'
    url = 'https://graph.facebook.com/' + amigo['id'] + size
    figura = urllib.request.urlopen(url).read()
    f = open(amigo['name'] + '.jpg', 'wb')
    f.write(figura)
    f.close()
    print (amigo['name'] + '.jpg impresso')

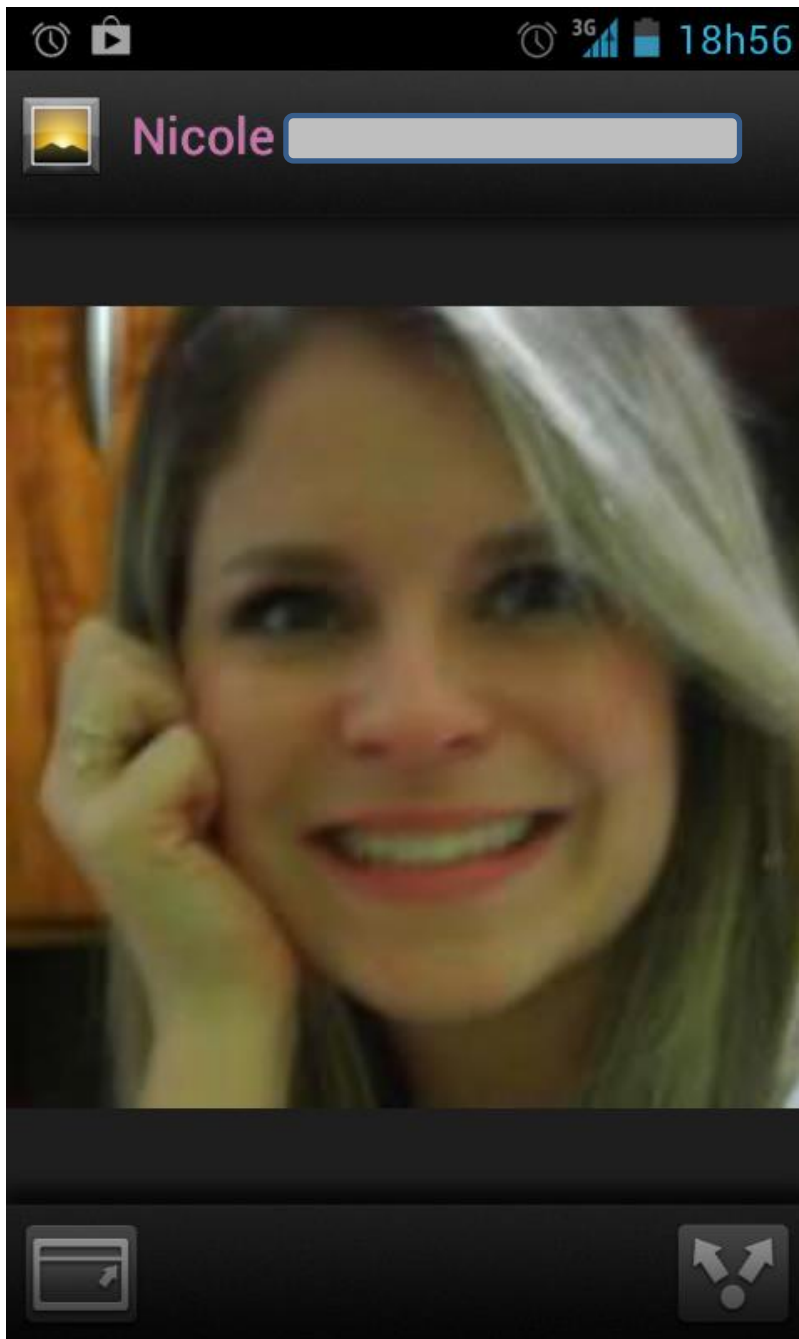
url = 'copie aqui o link Connections Friends'
resp = urllib.request.urlopen(url).read()
dados = json.loads(resp.decode('utf-8'))

for amigo in dados['data']:
    grava_imagem(amigo)
```

<https://gist.github.com/4684752>



# Android Wallpaper



# **Android Wallpaper**

# Facebook Search

```
import urllib.request
import json

def search(texto):
    url = 'https://graph.facebook.com/search?q='
    tail = '&type=post&access_token=<copie aqui o access_token>'
    resp = urllib.request.urlopen(url+texto+tail).read()
    data = json.loads(resp.decode('utf-8'))
    return data['data']

for resp in search('cpbr6'):
    if 'message' in resp:
        print (resp['from']['name'] + ': ' + resp['message'] + '\n')
```

<https://gist.github.com/4684949>

# Facebook Search

Piadas Nerds: Agora entendo pq todo ano a Angela Bismarck volta a ser virgem..  
. é só pra poder entrar na Campus Party. #CPBr6 (Via @FabioGuere)

Eu Compraria! Shop: Estamos no #CPBR6, na frente do palco Pitágoros e Arquimedes! procurem a TV com fotos dos nossos produtos ;)

Mônica Custódio: Aos designers de plantão, fica a dica de uma palestra da CPBR 6 :)

Design de Interação e Inovação / Artes digitais.

CoxinhaNerd: Coxinhos da #cpbr6, o Panda está rodando a campus neste momento tirando fotos para a primeira gincana virtual da Coxinha Nerd! Fiquem ligados para participar! ;) Em breve no ar.

Wagner Wakka Alves: Fazendo stand-up agora às 16h no BarCamp da #cpbr6 quero todo mundo lá!!

# Sentiment Analysis (Facebook)

- Sabe o que é ter vontade de acordar cedo no domingo para estudar?!?
- Bom, eu também não sei explicar, mas é o que estou sentindo no momento .-.
- Bora Casdinho!



# Participantes de um evento qualquer

- Hackear participantes a partir do evento criado no Facebook
- Inserir link para o perfil pessoal no Facebook
- Obs.: somente alguns participantes e com a foto na data do evento





# Reddit MongoDB Client

(tribute to Aaron Swartz, cofounder of Reddit, nov-1986, jan-2013)



{ name: mongo, type: DB }

# Reddit Client

```
import urllib.request
import json

url = 'http://www.reddit.com/r/Python/.json'
resp = urllib.request.urlopen(url).read()

parsed = json.loads(resp.decode('utf-8'))

for item in parsed['data']['children']:
    doc = item['data']
    print (doc['title'])
    print ('#comments: %d' % doc['num_comments'])
    print (doc['url'])
    print ()
```

(by 10gen)

<https://gist.github.com/4691077>

# Reddit Client

A cool little gravity simulator I made with pygame.

#comments: 4

<http://youtu.be/93FH4oXfTlY>

Tornado web server internals

#comments: 5

<http://www.slideshare.net/pgollakota/tornado-web-server-internals>

I need to extract an array from a multiblock dataset block.

#comments: 1

[http://www.reddit.com/r/Python/comments/17nlso/i\\_need\\_to\\_extract\\_an\\_array\\_from\\_a\\_multiblock/](http://www.reddit.com/r/Python/comments/17nlso/i_need_to_extract_an_array_from_a_multiblock/)

python user detected :P

#comments: 70

<http://i.imgur.com/ScMSvvD.png>



# MongoDB == JSON



**Flexible Schema**

# Relational DB



# Reddit MongoDB Client

```
import json
import urllib.request
import pymongo

connection = pymongo.Connection('mongodb://localhost', safe = True)

db = connection.reddit
stories = db.stories

url = 'http://www.reddit.com/r/Python/.json'
reddit_page = urllib.request.urlopen(url)

parsed = json.loads(reddit_page.read().decode('utf-8'))

for item in parsed['data']['children']:
    print (item['data'])
    stories.insert(item['data'])
```

# Reddit MongoDB Client

```
> use reddit
switched to db reddit
> db.stories.findOne()
{
  "_id" : ObjectId("50a7dfcbe31cec02380b2924"),
  "domain" : "robgolding.com",
  "banned_by" : null,
  "media_embed" : {
    },
  "subreddit" : "Python",
  "selftext_html" : null,
  "selftext" : "",
  "likes" : null,
  "link_flair_text" : null,
  "id" : "13cxm9",
  "clicked" : false,
  "title" : "Django Class-Based View Mixins: Part 2 (Sorting &
ng)",
  "num_comments" : 0,
```