EXHIBIT 6

NOT SO SECRET HISTORY

Why you shouldn't share links on Facebook

Inti De Ceukelaire | June 29, 2016

Your messages aren't so secret. (Reuters/Stephen Lam)

Recently, security researchers at Checkpoint discovered a vulnerability that would have allowed attackers to change messages and links sent through Facebook Messenger. Facebook quickly patched the bug … but did you know links sent privately through Messenger can be read by anyone? Moreover, Facebook knows about this and has no plans to fix the issue.

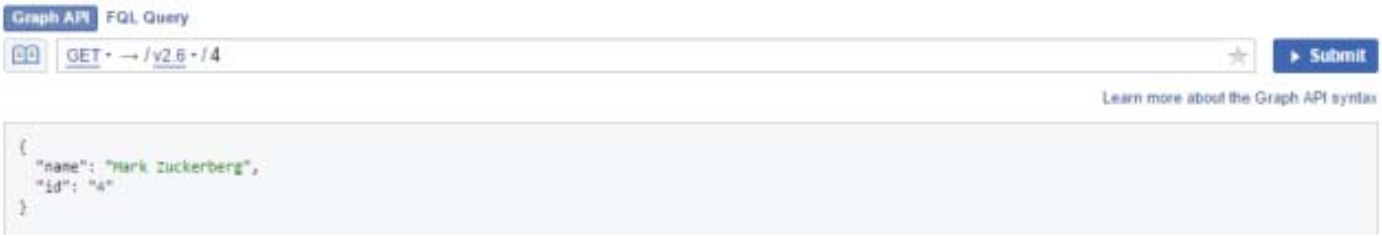
How Facebook links work

The first time a specific link is shared on Facebook, Facebook's crawler takes a look at the shared page, extracts the title, the description, and the thumbnail image, assigns an unique identifier, and then stores this information. The next time Facebook displays the link, it simply fetches this information from the database. There's absolutely nothing wrong with this. At least when this data is kept secret.

Workings. (Facebook)

The number game

All objects stored on Facebook, whether it's a picture, a status, or a link, are given a unique, non-chronological identification number. Mark Zuckerberg is object number four:



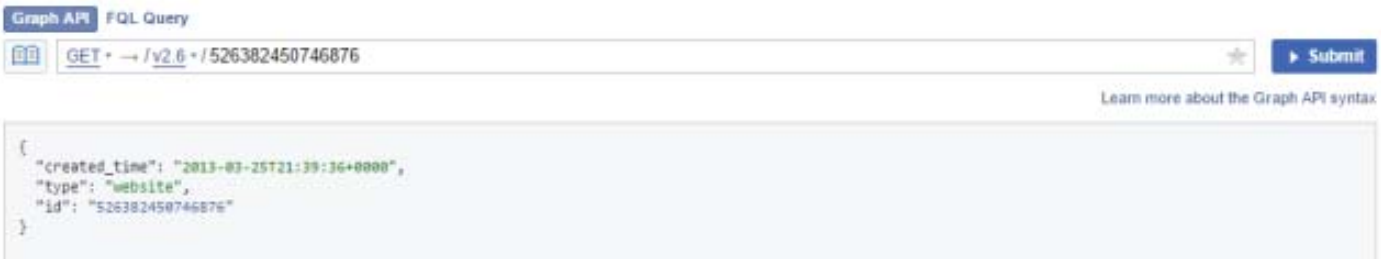
Now I'm wondering who's number one. (Provided by author)

A developer can request an object by its number through the Facebook API, an interface for developers to connect with Facebook, which will return the corresponding information **only** if you have permission to access it. This means that you can't simply access someone else's private status update without obtaining their permission. Seems logical, right?

I started playing around with this feature. Most of the time, I got an error message that the object either did not exist or that I did not have permission to view it:

No permission to access 39402139014. (Provided by author)

As I was about to give up, a URL popped up. Cool, but this left me none the wiser: you can't do much with a timestamp and a string that says: "website."

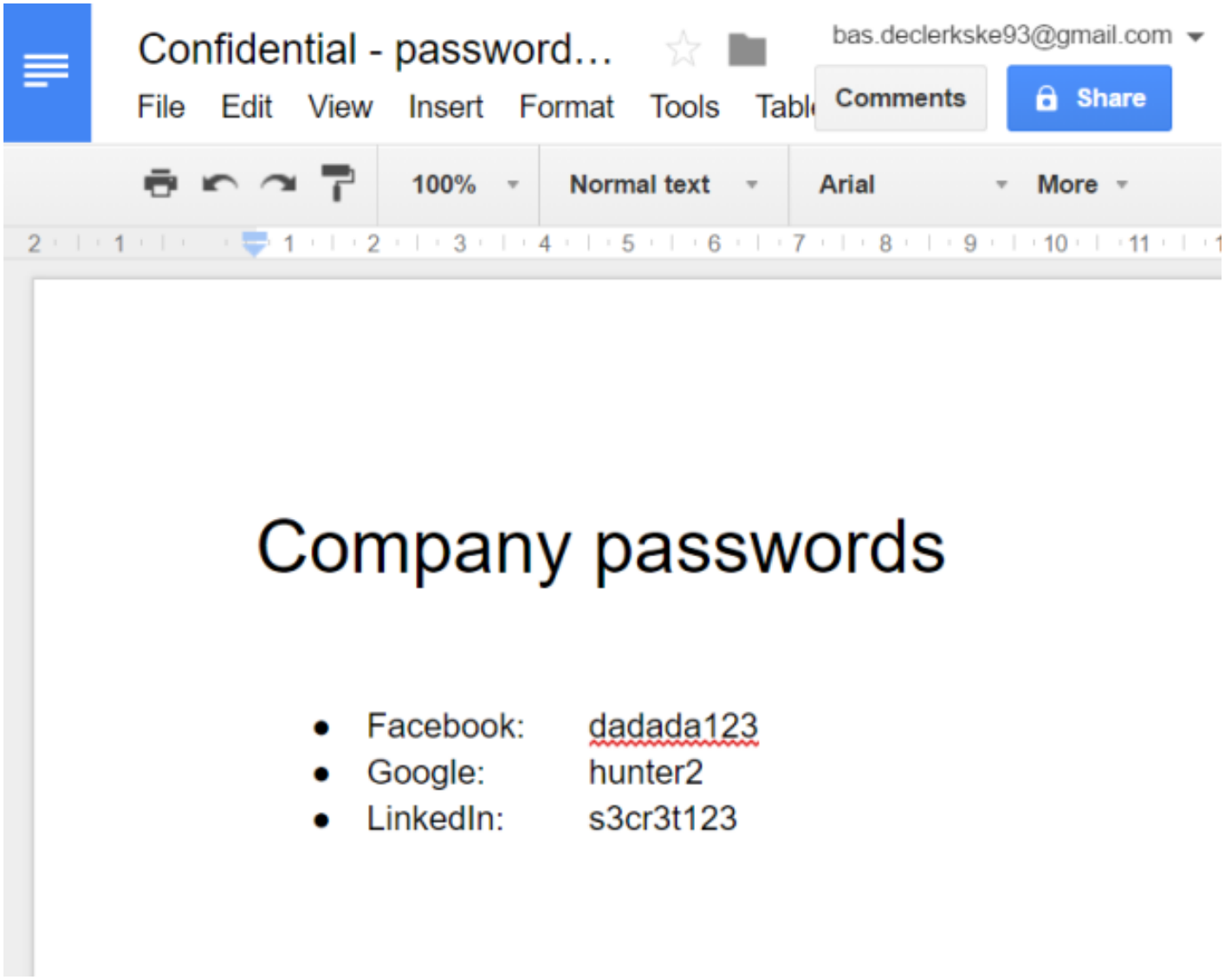


A website? How's that an object in Facebook? (Provided by author)

Then I appended "url" to my initial request, asking Facebook if it would be so nice as to display the link address as well. To my surprise, this worked:

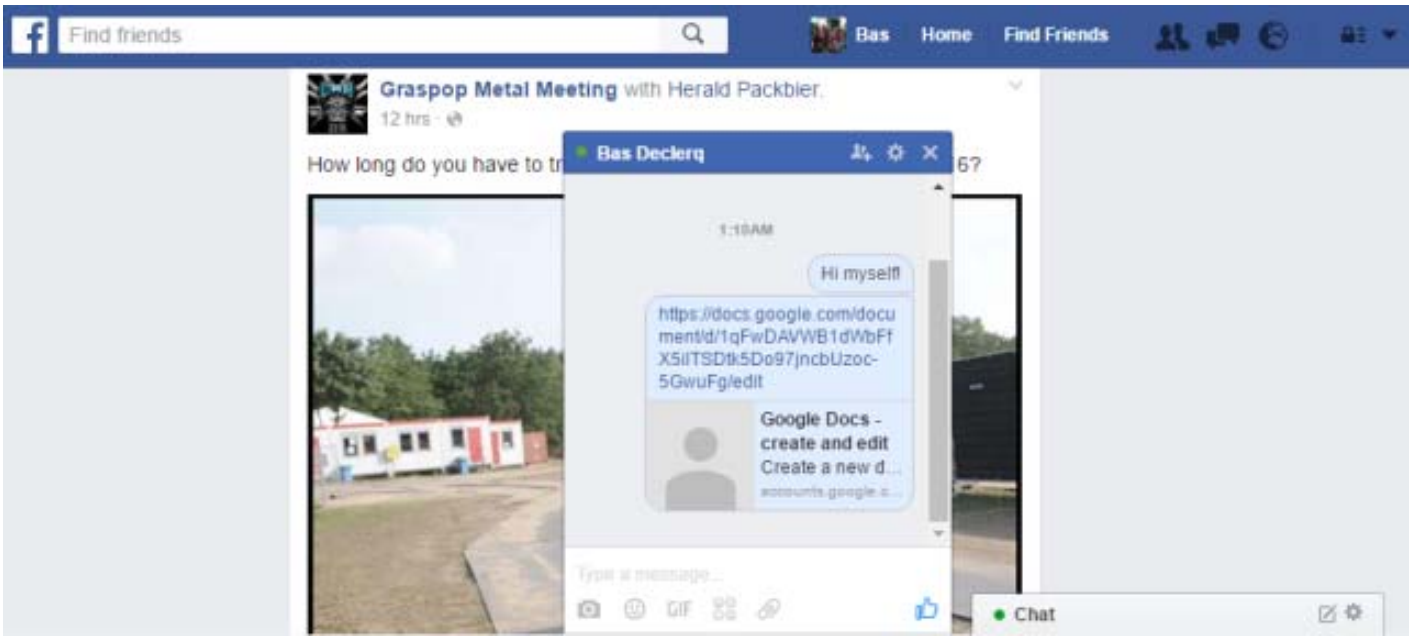
Apparently someone visited this URL from Facebook back in 2013 when it was still online. (Provided by author)

At this point, I wondered if I could also use this to view links users privately shared, so I asked my friend Bas to help me out, create a Google Doc and privately share the link. Here's what I received:



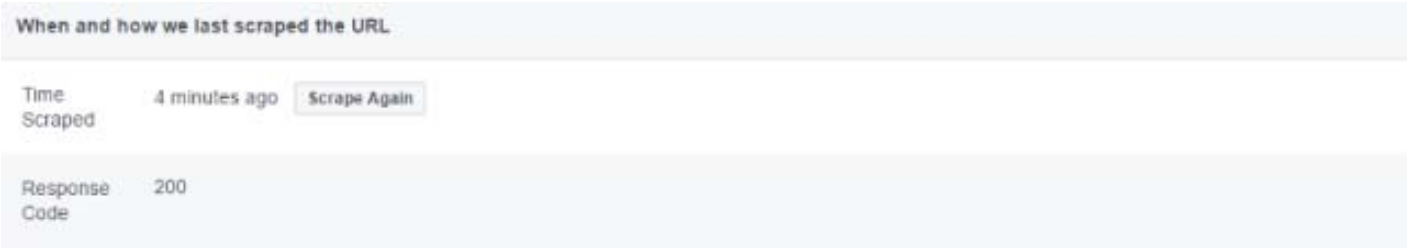
Bas' side: An example of a Google Document with confidential information. (Provided by author)

Then I asked Bas to use Facebook Messenger to send the link to himself and click on it:



Bas' side: Sharing the link with himself on Messenger. Harmless, right? (Provided by author)

Since the link was saved to Facebook's database when Bas opened the link via Messenger, I asked him to use Facebook debugger tool to get the object identification number and provide it to me:

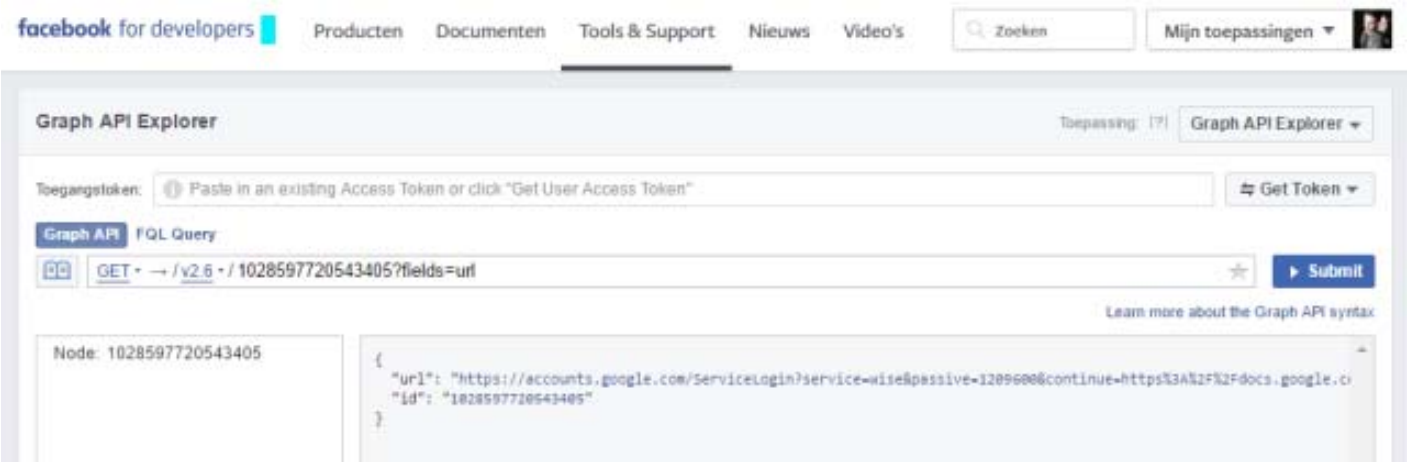


Bas side: The URL was scraped when it got clicked. (Provided by author)

The tool also showed the number identifier attached to the post:

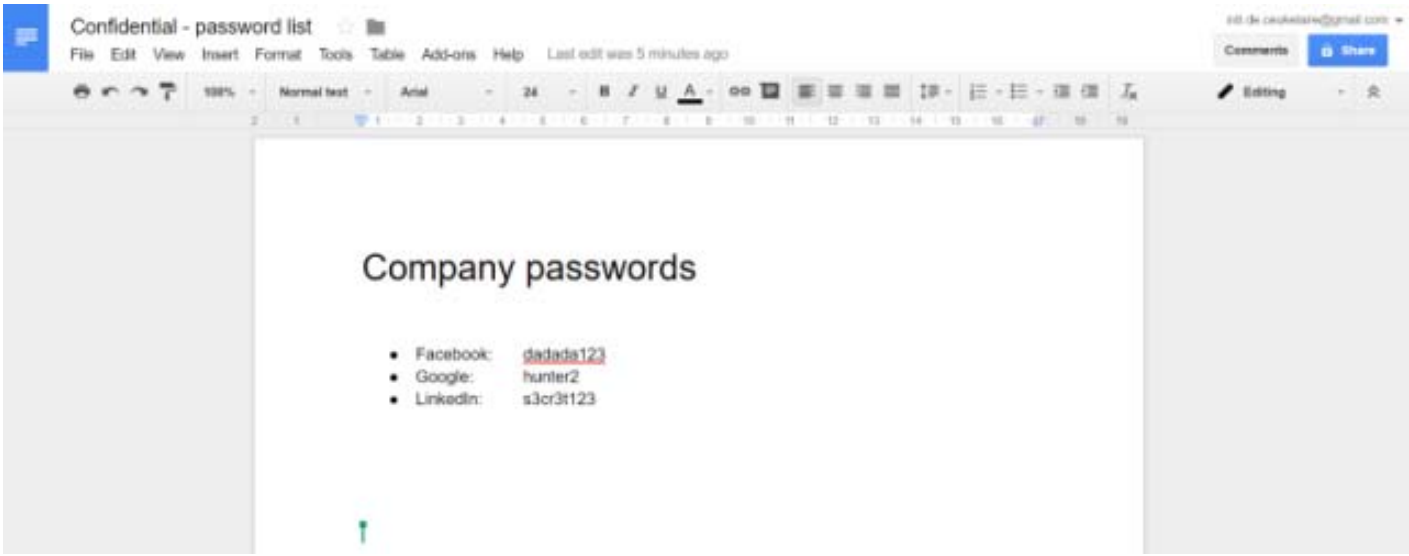
The number identifier was another number someone could have guessed or stumbled upon. (Provided by author)

Back to my account. When I tried to access the object attached to the number above, the URL popped up:



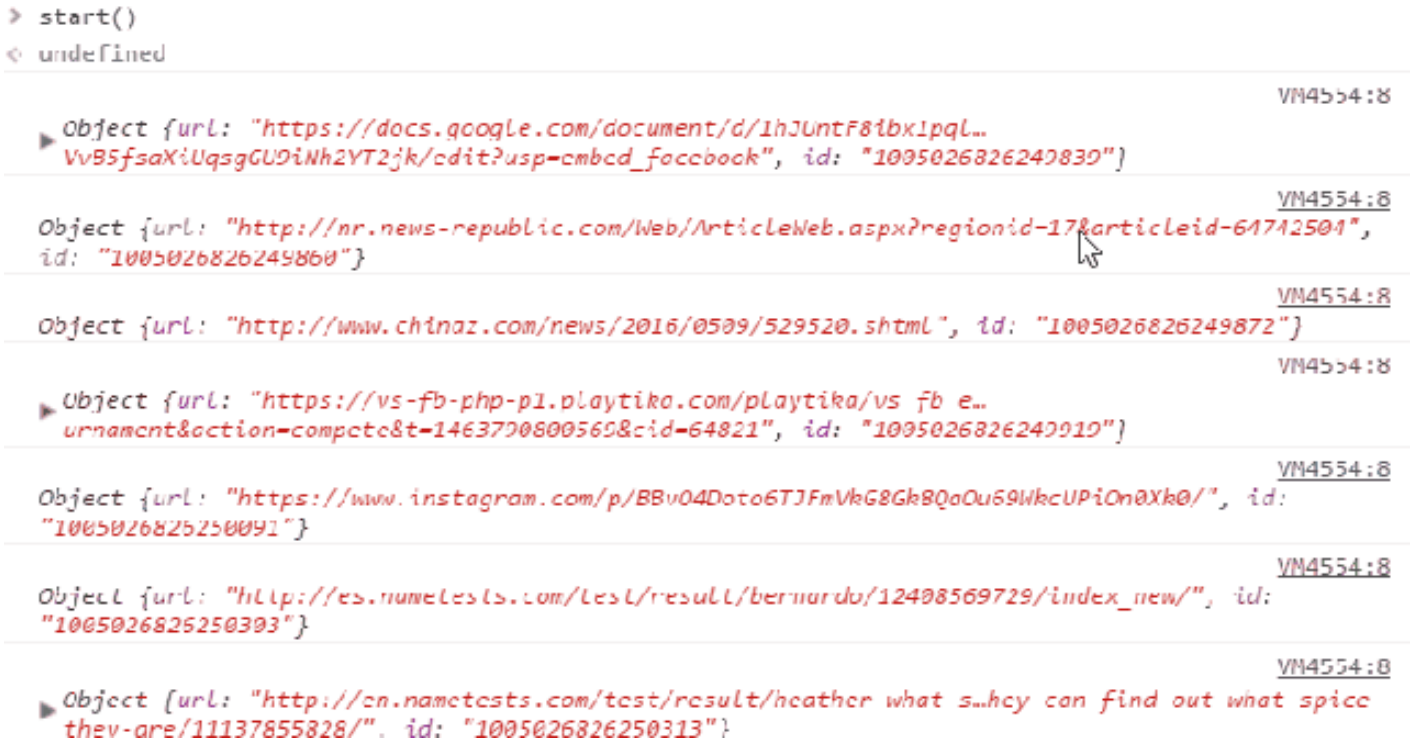
On my side: Enter a number, get a link. (Provided by author)

Moments later, I was able to access the confidential Google Document:



My view: Successfuly "hacked" into Bas' secret document. (Provided by author)

I tried to reproduce this a couple of times until I decided to find out whether it was also possible to get other people's (private) links exploiting this. I wrote a quick script that would take any identification number and increment it gradually to discover other links. It worked:



A script I wrote to extract links from Facebook. For this example, I checked all the links to make sure they weren't confidential. (Provided by author)

While the results did not include an ID for the user who shared the link, I was able to identify some because their user ID, the number corresponding to their account, was included in the link.

Why this is a big deal

While you may only share links to funny cat videos with your friends, you should still be worried about this exploit. Sometimes, sensitive information (personal data, secret keys, …) are included in links without you even noticing. Just take a look at the redacted and quite innocent looking links from earlier:

(Provided by author)

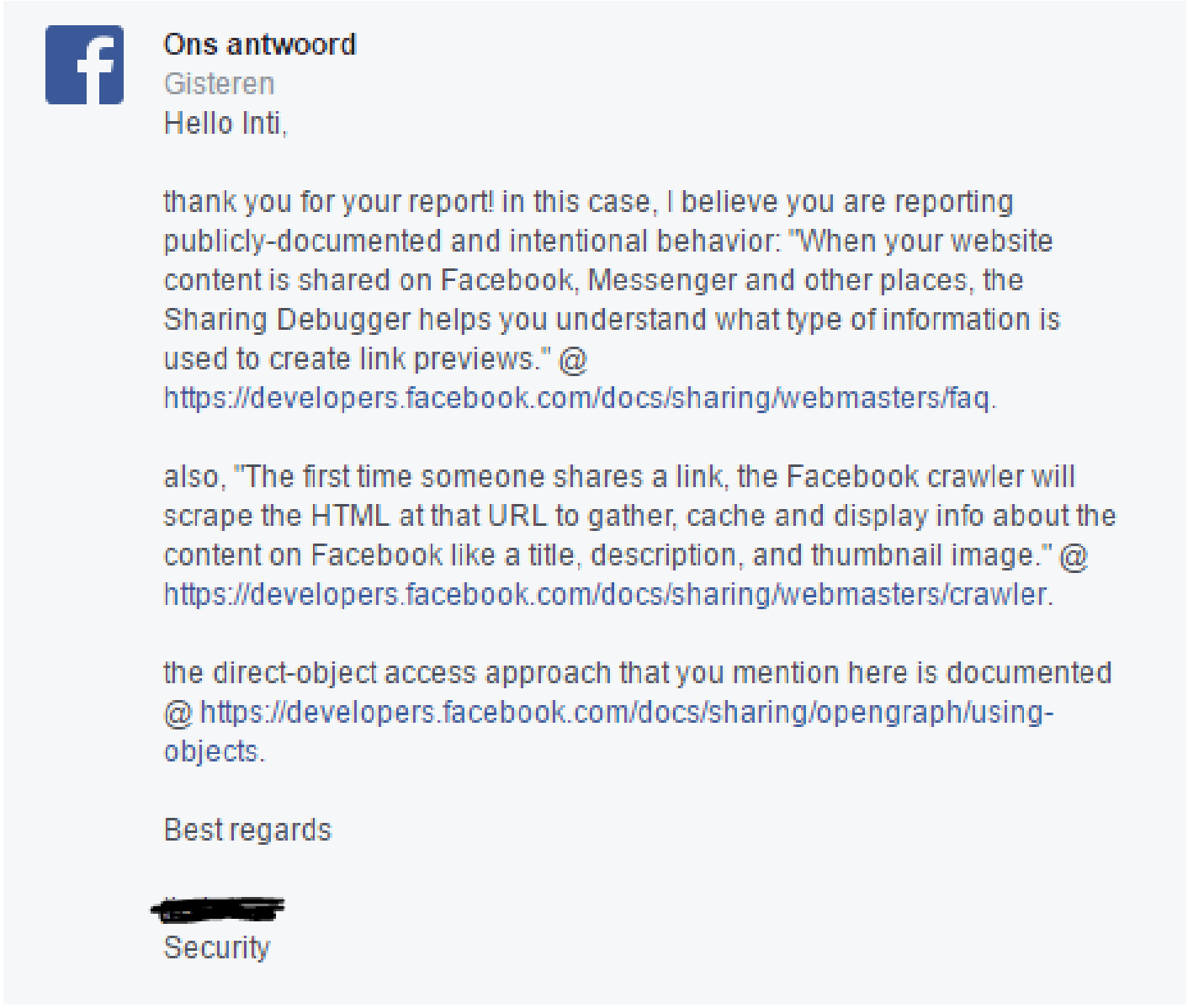
In this small set of extracted URL's, I've already found some interesting info:

- **Names**: Heather, Jenny, Paula, Yollanda, Bernardo, …
- **Location** or **language**.
- **Attachments or pictures from the FB CDN**: Direct link that sometimes allows access bypassing privacy restrictions.
- **Application or game data**: Some parameters are friend_level, friend_chips, user_name, group, steal_amount, …
- **Secret links** or **hidden keys:** Such as the editable Google Drive links or links to hidden pages, websites, and beta environments.

..and these aren't mutually exclusive; some URLs include multiple parameter types listed above in one single link, thereby allowing a total stranger to gain personal information about you. Hello NSA?

Facebook's response

I reported this issue to Facebook under their responsible disclosure program, which I've had successful experiences with before. Here's their official response:



Ons antwoord
Gisteren
Hello Inti,

thank you for your report! in this case, I believe you are reporting publicly-documented and intentional behavior: "When your website content is shared on Facebook, Messenger and other places, the Sharing Debugger helps you understand what type of information is used to create link previews." @ https://developers.facebook.com/docs/sharing/webmasters/faq.

also, "The first time someone shares a link, the Facebook crawler will scrape the HTML at that URL to gather, cache and display info about the content on Facebook like a title, description, and thumbnail image." @ https://developers.facebook.com/docs/sharing/webmasters/crawler.

the direct-object access approach that you mention here is documented @ https://developers.facebook.com/docs/sharing/opengraph/using-objects.

Best regards

[redacted]
Security

Apparently, Facebook has no problems with our privately shared links being accessible. (Provided by author)

Friendly and descriptive as always (I *love* participating in Facebook's bounty program), they told me this would be deemed a won't-fix, and is actually intentional behavior. I was puzzled: How can Facebook let this happen? Whilst it's not possible to get links for a specific user, you could easily run through results all day* until you find something interesting.

Timeline

> *29th of May: Me: Reported through Facebook's whitehat tool.*
< *31th of May: Reply from Facebook: Needs more information.*
> *31th of May: Me: I provide the information asked for.*
> *2nd of June: Me: I send a follow-up as I think this is a critical issue.*
> *7th of June: Me: Another friendly ping from my side.*
< *7th of June: Reply from Facebook: Needs more information.*
> *7th of June: Me: I provide the information asked for.*

< 8th of June: Reply from Facebook: This is intentional behavior.
> 9th of June: Published this blog article.

*Yes, Facebook does block excessive requests but there are ways to bypass that, e.g., using multiple access tokens and if needed, VPNs. Rate limiting won't stop someone who is determined.

Are the links we send being tracked this way? I have absolutely no idea, but now at least we know they could be.

FAQ

- **When is a link scraped and stored in Facebook's database?**
  From my testing I assume that a link is stored in Facebook's database from the moment someone actually clicks it on Facebook. This does not apply to links shared through Facebook which no one clicks on.
- **Does it matter where or how we share a link?**
  As far as I know, this does not matter: Links shared through messenger, private groups, status updates, or by using the mobile application seem to be vulnerable to the methods described.
- **Do links even matter? I don't care if someone sees the links I shared.**
  Links sometimes include personal stuff without you even knowing. See "Why this is a big deal" above.
- **Why are you making this issue public?**
  Facebook clearly stated that this is an intended behavior and I respect their decision, however, I think it is our right to know who can see the data we share. Are the links we send being tracked this way? I have absolutely no idea, but now at least we know they could be. Just keeping my mouth shut won't help.
- **If I share a link on Facebook, what's the actual chance someone will actually see it using this "intended behavior?"**
  There are lots of objects on Facebook and it would be a really hard, if not impossible, to scan all of them using the API. You would be really unlucky if an attacker stumbled upon the number linking to your secret link, but the odds increase if attackers start monitoring these numbers on a regular basis. In about ten minutes, I was able to extract 70 links. Facebook does have some rate limiting in place to prevent this type of abuse but as mentioned above, it is possible to bypass that.
- **Are you mad at Facebook?**

Not at all. Facebook has one of the best bug bounty programs available to hackers. I respect their decision, but I also think it's our right to be informed of the design decisions which may impact our privacy.

- **I found a vulnerability in Facebook. Where do I start?**
  Cool! Make sure you read their bug bounty rules and are reporting a valid bug. After reporting the issue here, they may decide to honor you in their Hall of Fame and reward you with a bounty starting at $500.

- **Who are you?**
  I'm Inti and I live in Oilsjt, Belgium—the country known for its beer, fries, chocolate, and terrorists. As a kid, I was extremely skilled at breaking stuff. I'm 21 now, a student, and still doing more or less the same being an ethical hacker with references at Google, Facebook, Microsoft, Yahoo, and so on.

This post originally appeared at Medium.