Role of Apps in Undoing of Privacy Policies on Facebook

Vishwas T. Patil, Nivia Jatain, R. K. Shyamasundar

Role of Apps in Undoing of Privacy Policies on Facebook

Vishwas T. Patil, Nivia Jatain, and R.K. Shyamasundar

Indian Institute of Technology Bombay, Mumbai 400076, India
{ivishwas, jatain.nivia, shyamasundar}@gmail.com

Abstract. Facebook allows its users to specify privacy settings for the information they share with other users and Apps. Apps seek a set of permissions from the user at the time of installation. There is no check that is performed to evaluate any possible adverse implications of App's permissions on the *in-force* privacy settings of an user. In this paper, we have investigated FB's platform for access to users' data by Apps and Advertisers. By signing up with FB, users implicitly trust the platform, which they believe can be held accountable in case of a breach. However, similar expectation of accountability from Apps is hard to imagine and difficult to ensure. At times, Apps have as much access to user data as FB and such a common access to user data undermines provenance of data leakage. Recently, though FB has reduced the extent of data access for Apps by deprecating certain APIs, a systematic design approach is missing for platform-wide access policy specification and conformance. We have presented several scenarios where App permissions are violating user privacy policies. Our findings have been presented with the help of experiments using Facebook Developer Platform.

Keywords: Social network · Privacy · Linkability

1 Introduction

Facebook is the largest social network. Maintaining 1.5 billion daily active users, their connections and updates in real-time is a tremendous engineering feat. However, it appears that the guiding principles in the evolution of Facebook's data platform have been: real-time response [2] and features to users, app developers, and advertisers. The recent revelations [3] have forced Facebook to acknowledge that data privacy is an important *feature!* The platform's design choices, for speed and features, will hinder it from coherently enforcing privacy policies anytime soon in the near future.

Facebook's platform allows users to establish and organize their relationships with other users using social relationship categories like "Friends", "Close Friends", "Family", etc. An update in user's personal life is more relevant to members of "Family" than "Friends" and the platform does such a prioritization intelligently. Similarly, among the categories of relationships further prioritization of updates is done based on the interests of the users that are at the other end of the connection. That is, a friend from school falls in sub-category school and likewise a friend from university. Furthermore, friends from school who have interest in history are distinguished from the friends who have interest in finance. Such a segmentation of categories helps the platform to build relevant audiences for a user's updates. Users are given a control to decide which segment should

see what updates. Facebook (FB) organizes all these information about its users and their interactions as a graph – called social graph. Users (nodes) are free to form new relationship (edge) and update the old ones. Social graph is a continuously evolving graph and this type of organization of users and their data helps FB in segmenting users with similar interests so that they can be introduced to a new post or an advertisement.

FB platform allows developers to write Apps, which users can install. An App serves a specific function to its users. When a user installs an App (represented by an edge between the App and the user on social graph), it signifies that user's interest in the functionality provided by that App. Thus, users get a functional convenience and FB automatically gets contextual insights about users. Both, the App and the platform will have an access to users' interactions within the administrative sphere of the App. FB can build an accurate context about an user than an App because it has other insights about the user. Thus an App, through its functional category, helps the platform to segment users in a specific category so that it can be used in profiling the users. For example, a flower delivery App can help identify users who are single, male, within a specific geographical area, and who have purchased flowers last year on Valentine's day. In order to build audiences of such type, FB needs to build, maintain a detailed profile for each of its users. Higher the interactions of a user, richer the profile. Connectivity and interactions are important objectives of the platform, and FB does it very well in its ecosystem of users, Apps, content and interactions among them. This ecosystem of interacting nodes is depicted as a pyramid, in Fig. 2), to highlight their access privileges (either explicit or implicit) on the platform. Each layer (user layer, app layer, advertisement layer) serves a different purpose and has a different access control mechanism to control access to users' information. In [23], we have analyzed privacy claims of the platform at the user level alone. In this paper, we analyze conformance of user privacy settings in the presence of Apps. **We will show that there is no coherence in policy enforcement across the layers, which undermines the privacy of its users.** We have validated our observations through experiments on Facebook's developer platform v2.12 and Facebook Audience Network. While FB does profiling of users for varieties of reasons, one of the trusting factors of FB is that it shall not divulge intentionally or for price the data that violates its committed privacy setting with its users. However, this cannot be said about the app developers or the advertisers on the app. Thus, our findings show the challenges to plug the leaks, due to apps/advertisers, FB should undertake.

In the following section, we present the somewhat hybrid, ad-hoc nature of access control mechanisms employed by FB. In Section 3, we analyze the platform and trace the flow of user information beyond the layers of its policy sphere. In Section 4, we present a few scenarios where defined privacy settings of a user are violated due to Apps. Section 5 discusses related work followed by conclusion in Section 6.

2 Access Control in Facebook

At the different layers of the platform, FB employs different types of access control mechanisms. At the user layer, user content and user attributes are protected by a discretionary access control. At the App layer, user content and user attributes are protected by capability lists. The other entities of the platform are not governed by any policy that

user can influence. Also, the metadata the platform collects about user is not controlled by the user in any way. The platform organizes all of its entities and content in a graph, which has a sub-graph that can be traversed by users/Apps according to their respective permissions. The platform owner can traverse the whole graph without any restriction and acts as a proxy to its collaborators (the advertisers).

Social graph - Reachability as the condition for access: Social graph in Facebook is a representation of user information on Facebook. Two user nodes have an edge between them if the users are friends with each other. Having an edge between two nodes establishes connectivity between them and in turn extends their reachability: that is, a user can access posts of her friend because there is a path present on the graph between the user and her friend's post via the friend node. Now, if the user likes her friend's post, this will be reflected in the social graph by putting an edge of type like between the user and her friend's post. Thus, each and every action or event created by Facebook's users is consumed by the social graph. The graph continuously changes its state reflecting its users' actions and interactions. Updates to social graph happen by adding/deleting nodes (or updating fields of nodes), and adding/deleting/updating the labelled edges – all such updates are due to a user's and app's *interactions* with their reachable nodes. Passive nodes like posts, photos, et al, do not interact on their own. Social graph also allows its nodes to be queried [23]. A user is allowed to compose a query by specifying a particular node (of type *root* [9]) about which the requester needs information. It is very likely that different sets of information about a node are presented based on who the requester is.

Lists as access policies for users: Each user is provided with pre-defined relationship categories, called lists, along which users organize their relationships with others. Then there is a category of lists that FB creates for a user based on her social affiliations. And a user is also allowed to create and manage her own private lists. Given below is a typical set of labels provided to express access control policies:

- Only Me: is a label/list in which user herself is the only member
- Public: is a label, when used, the associated object is accessible publicly
- Friends: is the primary list under which all friendship relations are enlisted
- Restricted: is a list of friends to whom only Public labelled information is allowed
- Family: is a list of friends who are assigned as family members
- Close Friends: is a list of friends who are assigned as close friends
- Acquaintances: is a list of friends who are assigned as acquaintances
- Friends of friends: list of users who have friendship relation with "Friends"
- *University*: is a social list of friends who are also members of Smart List *University*
- *School*: is a social list of friends who are also members of Smart List *School*
- *Cycling*: is a Private List to which user has assigned a set of friends
- Custom: is a custom policy constructed using the label types described above.

Access control of objects in FB is a simple check on associated list's membership. If a requester of an object is a member of the list with which the object is protected, the requester gets access. Tagging is a positive exception to the membership check. There are two negative exceptions to the membership check: "Restricted" list and "Blocked" list. If a requester of an object is member of one of these lists, access is denied even when the requester is member of the list with which the object is protected.

In Fig. 1, User2 can reach & access Post1 because there is a path and the access policy for Post1 is set as *friends* by its owner User1. Therefore, User2 could interact with Post1 by *like* action. User1 & User2 can access Post3 because User1 is a *friend of friend* of User3 and User2 is *friend* of User3. Post2 cannot be accessed by User1 because the custom policy allows access to all friends of User2 except User1. The Event created by User1 cannot be accessed by anyone except User1 because the access policy is *only me*. Thus, labels or lists are used to control access to the content owned/posted by Facebook users.

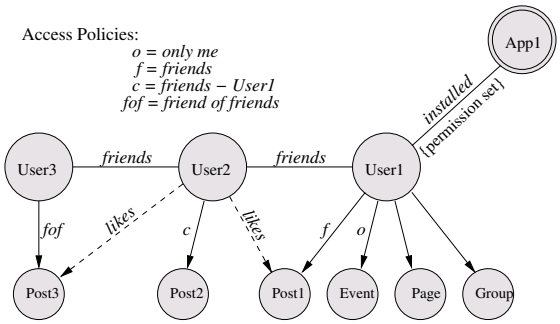

Fig. 1. Reachability and access in social graph of Facebook

Capabilities as access policies for Apps: FB Apps too are represented by nodes on social graph. However, Apps' traverse-ability on the social graph is limited to the immediate neighborhood of the user node consisting only the object nodes. In other words, the App can neither reach the friends of the user nor the other Apps installed by that user. What interactions the App can do in the user's neighborhood is determined by the set of permissions the user has allowed at the time of establishing the *installed* relationship with the user. There are 48 such permissions an App can obtain from its user. This is similar to capability lists in access control paradigm [17]. In later sections we shall discuss which of these permissions to an App undermines user's privacy.

The utility of social graph is not limited to representation of subjects, objects and their relationship but to also provide real-time updates about the changes in the neighborhood of the subject. Prioritization of updates according to their relevancy to a user based on users' past interactions on social graph is handled by NewsFeed algorithm; a core function of FB platform. How the App ecosystem helps it in achieving precision is explained below along with the other important components of platform.

3 Architecture of Facebook Platform

Fig. 3 gives a schematic architecture of Facebook platform depicting the relationships between the major entities of this platform. In the following we describe the entities and their functionalities. The platform is logically divided into two: public space & private space. The entities in public space are the users and applications. They are said to be in public space because, having an account on FB, these types of nodes can query and interact among each other based on the access policies. Though the entities from private space can influence and have a richer view of the graph topology, they cannot perform any of the operations available to nodes in public space without being a node in the public space. Fig. 2 depicts the access-hierarchy in the social graph of Facebook. The primary objective of the platform is to build accurate user profiles (behavioral, psychometric, etc.) so that advertisers can be accurately matched to their audience.

The platform has been quite successful in micro-targeting users in real-time so that it artificially puts limits on advertisers while building their target audiences. An advertiser cannot compose a target audience whose size is less than 100. Similarly, an advertiser cannot request audience-tracking for audience size less than 100. To understand the design of this platform let us describe the role and functionality of its individual entities.

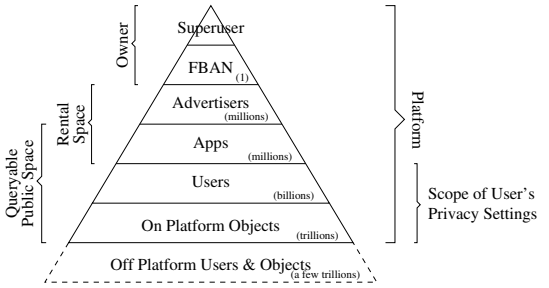

Fig. 2. Access-hierarchy in the social graph

NewsFeed: Facebook has an intelligent algorithm to prioritize the updates to a user, which is called NewsFeed. If we assume that each object/content on the social graph has a category type associated with it, like: education, finance, food, sarcasm, celebrity, etc., then a subject's interaction with these objects determine the probability of interest the subject may have in such categories. Each interaction of a subject with its neighborhood node improves the confidence level of subject-category mapping. The objective of NewsFeed algorithm is to increase subjects' interaction with varying categories [12] of content so that a rich user profile can be built. Such a user profile is necessary to determine relevancy of updates to the user and also to match the user with an advertiser interested in particular category [24]. If we assume the nodes in the graph are labelled with categories and edges are weighted proportional to the confidence level of the category, then we can think of an *influence* function over two nodes. A node with higher confidence value influences the confidence value of its peer. Thus the utility of NewsFeed function is incite the user to interact with content from its neighborhood and also from other influential nodes with whom the user does not have relationship (either *friend* or *follow*) yet. Higher the engagement of the user, more are the interaction, and thus higher the confidence value to categorize the user.

Users: Users are the largest part of the platform. Their *interactions* within their reachable neighborhood and with the nodes introduced by the NewsFeed builds their individual user profiles. Users interactions with content outside the platform also helps in building the profile.

Apps: The platform gives a general purpose connectivity and interaction mechanism to the users, whereas the Apps give a context to user profile. Apps serves a specific functionality (e.g., finance, education, dating, et al.) to its users and that functionality is a stronger measure to categorize users. Apps can opt for monetization of their functionality by serving advertisements to the users via the App. Apps obtain analytics over their users interactions. The analytics information contains attributes (like mobile advertisement ID, Facebook UID, email, phone, Device info, location, etc.) that can uniquely measure interactions of App users. To advertise itself, or to persuade its existing users the App may share its analytics with advertisers to target the existing and new users.

Advertisers: Advertisers are the paid interfaces to the platform's ability to find precise audiences for a specific category/issue. Advertisers build advertisement campaigns by requesting specific audience type from the platform against a fee. To build the audience

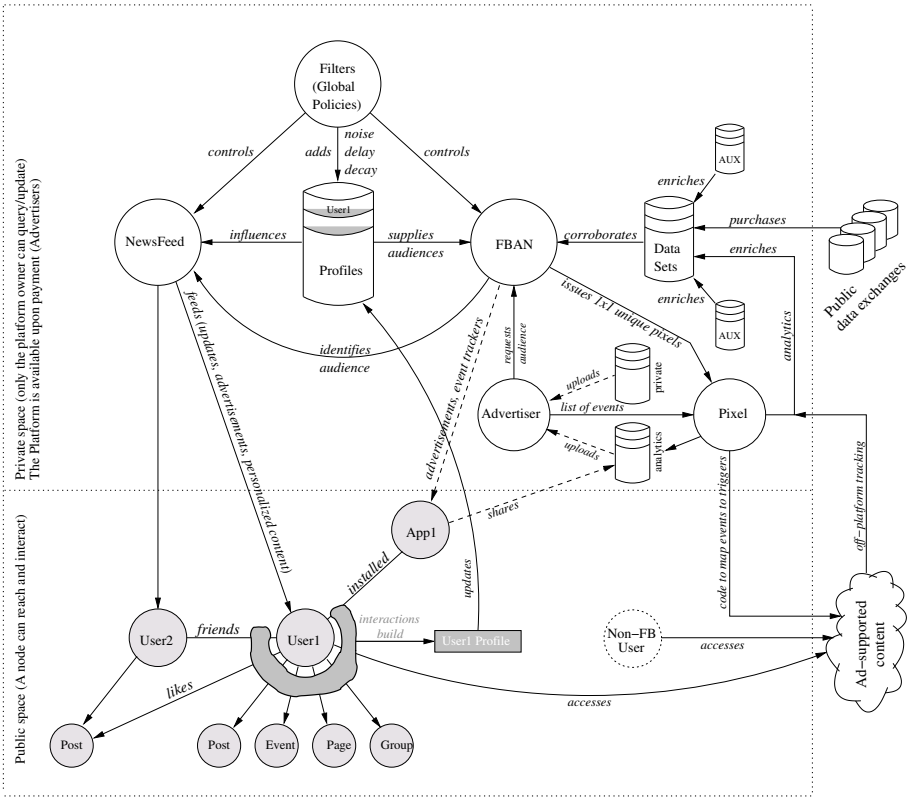

Fig. 3. Facebook's Schematic Architecture

request, advertisers upload data fields that are compared against the user profiles that are built by the platform. Upon evaluating the scope of campaign targeting based on the uploaded data by the advertiser, the platform either accepts or rejects the request. Advertisers are allowed to micro-target a specific audience that is already engaged with it. Advertisers do so by defining events inside the Apps and trigger actions via Pixel for those events' realization. For example, list of users who have browsed a product but did not checkout.

Pixel: It is a micro-targeting framework https://fb.com/business/learn/facebook-ads-pixel that uniquely identifies users of the platform and also the users off-the-platform. This is a script that generates a unique tracking number each time a defined event occurs. The events could be as simple as loading a website or a user selecting a product in her cart. The unique number concatenated with cookie at user side tracks the user event by event. These user behavior analytics are shared by the platform with the advertisers so that advertisers can measure the impact of their advertising campaigns.

FBAN: Facebook Audience Network (https://fb.com/audiencenetwork) is the core component of the platform and has access to users profiles generated by the platform. It has its own data-set that is built from user tracking (analytics) and other associated plat-

forms' meta-data information (like WhatsApp, Messenger, Instagram). It accepts audience requests from advertisers and based on the corroboration with its data-sets and user profiles, it identifies the target audience for a campaign. There exist public data-exchanges for user information, which can help enriching the profile attributes of users that come in contact with the platform.
Profiles: All individual user profiles are further enriched and attributed by the insights obtained from platform analytics and plausibly external public/private data-sets [6] (For Indian users, Facebook tried to link their Aadhaar numbers with their profiles. Aadhaar numbers are not secret but are used in various financial and public services delivery).
Filters: These determine the general access policy of the platform. For example, Facebook recently decided not to allow querying of its users (nodes) by their email/phone. This is also responsible for guiding the behavior of the platform in general. For example, to suppress a specific category of nodes appearing in the NewsFeed. Facebook had made an understanding with a large government (Project Colorful Balloons) to ensure a specific category of nodes is identified, tracked and controlled.

Having understood the roles various entities play in the FB ecosystem and keeping in mind those entities' access hierarchy, the question we ask is the following:
Assuming users explicitly trust Facebook to handle their private data against the free services, and assuming that Facebook desensitizes user data before making use of it for advertisement: what privacy & leakage assurances can we expect from the platform?

As Apps are only loosely coupled with the ecosystem as compared to the other entities in the ecosystem, it is difficult to assume that (smaller) Apps will strive for achieving the same level of trust with users as Facebook *may* have. In the following we present a few scenarios in which Apps violate users' privacy settings. In [23], we have presented whether Facebook users really preserve their privacy as they understand it or certain of their innocuous actions leak information contrary to their privacy settings. We would like to list those findings (at user-object layer of the platform) here:

1. Nonrestrictive change in policy of an object risks privacy of others,
2. Restrictive change in policy of an object suspends other's privileges,
3. "Share" operation is privacy-preserving,
4. Policy composition using intensional labels is not privacy-preserving,
5. "Like", "Comment" operations are not privacy-preserving.

In this paper, we extended the scope of our investigation to higher layers in the platform: that is, App layer and advertiser layer.

4 Experimental Scenarios of Access by Apps

In this section we list out our experiments using apps and advertisement facility of Facebook and highlight their potential in undermining user's privacy and security. The experiments are carried out using Facebook APIs (v2.12) and our findings are reproducible as of April 13, 2018. This sort of gap analysis in privacy policy conformance across platform is ignored [8], and precisely due to the lack of a platform-wide, coherent, privacy policy enforcement, rouge apps are tracking and siphoning off user data.

4.1 App Finds out User's Friends

Facebook has deprecated Apps to access its user's friend list. Consider a scenario as shown in Fig. 4, in which Alice has set her list of friends to private in her privacy settings. This setting sets an expectation that Alice's friend list will not be available to others. Alice installs App1 with permission `user_posts`. This permission allows App1 to reach all of Alice's posts and their fields (comments, reactions, post privacy settings). Fig. 5 is the list of posts retrieved by App1 from Alice's timeline. Fig. 6 shows the retrieval of comment & reaction on the first post in the list shown in Fig. 5. FB's NewsFeed function presents updates from Alice's timeline to her friends (Bob). When a friend interacts with the post, App1 can observe it and deduce with high probability that Bob is Alice's friend. The probability of such an inference is 1 when Alice has given App1 permission to post with post's access policy as "Friends". Similarly, depending on post's permission policy setting, App1 can reason about Family et al.

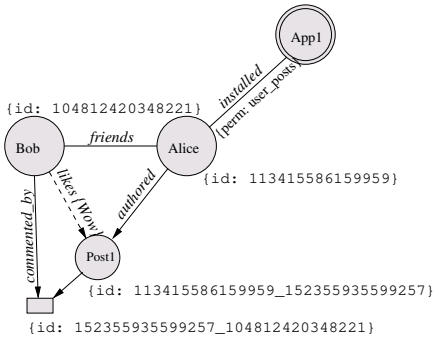

Fig. 4. Scenario: Alice has installed App1. Bob is Alice's friend

```
data:Array(7)
0:{message: "Test post for comment",
  created_time: "2018-04-12T10:34:40+0000",
  id: "113415586159959_152355935599257"}
1:{message: "Post by TestApp",
  created_time: "2018-04-12T06:47:07+0000",
  id: "113415586159959_152253442276173"}
2:{message: "Vtp",
  created_time: "2018-04-10T08:29:11+0000",
  id: "113415586159959_150842215750629"}
3:{message: "Modified Post",
  created_time: "2018-04-07T07:14:44+0000",
  id: "113415586159959_148447715990079"}
4:{message: "test post",
  created_time: "2018-04-03T08:21:11+0000",
  id: "113415586159959_145280909640093"}
5:{message: "Test post for Comments",
  created_time: "2018-03-20T04:16:11+0000",
  id: "113415586159959_133157927519058"}
6:{message: "Test post for Alice",
  created_time: "2018-03-20T04:15:24+0000",
  id: "113415586159959_133157680852416"}
length:7
```

Fig. 5. List of posts retrieved by App1 from Alice's timeline

```
data:Array(1)
  0:
    created_time:"2018-04-12T10:36:14+0000"
    from:
      id:"104812420348221"
      name:"Bob Greenewitz"
      __proto__:Object
    id:"152355935599257_152356518932532"
    message:"Test comment by Bob"
    __proto__:Object
  length:1

data:Array(1)
  0:
    id:"104812420348221"
    name:"Bob Greenewitz"
    type:"WOW"
    __proto__:Object
  length:1
```

Fig. 6. retrieval of comment & reaction on the first post in the list shown in Fig. 5

4.2 App Can Access User Objects Despite "Only Me" Policy

Consider in Fig. 4, Alice changes the access policy of her post P1 to "Only Me". This implies that only she can access this post. However, App1 can still access the post P1 even when Alice sets the policy to "Only Me", see Fig. 7.

```
created_time:"2018-04-07T07:14:44+0000"
id:"113415586159959_148447715990079"
message:"Modified Post"
privacy:
  allow:""
  deny:""
  description:"Only me"
  friends:""
  value:"SELF"
```

Fig. 7. Results of App1's query to Post1

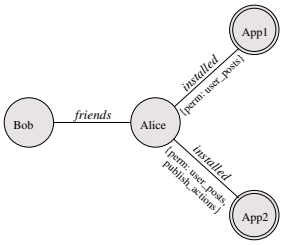

Fig. 8. Scenario: Alice has installed App1 and App2. Bob is Alice's friend

4.3 App Can Find out Other Apps Installed by the User

Consider the scenario shown in Fig. 8. Both the apps have permission `user_posts`. App2 (i.e., anshx.ananx as its real name in our experiments) has one additional permission `publish_actions` as shown in the figure. Let us assume that App2 publishes a post on Alice's timeline. App1 can observe this event and can obtain the post ID. Fig. 9 shows the query composed by App1 and its result, through which App1 deduces that Alice has also installed App2. Such a knowledge is useful is various ways.

4.4 App & Advertiser Can Identify Users: Linkability

Fig. 11 is the analytics report for a campaign we designed for a Page under our control. The analytics is available in real-time. The campaign was to invite users to follow our page on "Online Privacy". We could correlate the Likes (by Facebook users) on our page with the feed sequence report and find out which user has accessed the advertisement from what type of device and device OS version. This information greatly narrows down the types of attack payloads one can design to compromise a device. We could also access App user's Device Information.

```
FB.api( "/{post-id}?fields=application,created_time,message",
    function (response) {
      if (response && !response.error) {
        /* handle the result */
      }});
```

```
application:
  category:"Education"
  id:"335320743592541"
  link:"https://apps.facebook.com/anshxananx/"
  name:"anshx.ananx"
  namespace:"anshxananx"
  __proto__:Object
created_time:"2018-04-11T13:07:29+0000"
id:"108348820021337_108175360038683"
message:"Post by Anshx"
```

Fig. 9. Query composed by App1 and its result

```
["devices"]=>
    object(Facebook\GraphNodes\GraphNode)#16 (1) {
      ["items":protected]=>
      array(1) {
        [0]=>
        object(Facebook\GraphNodes\GraphNode)#15 (1) {
          ["items":protected]=>
          array(1) {
            ["os"]=>
            string(7) "Android"}}}}
```

Fig. 10. Retrieving App user's device information

A summary of privacy violations & data leaks from the above scenarios is given below:

1. App finds out user's friends despite user setting it private.
2. App can access user objects with "Only Me" policy.
3. App can find out what other apps are installed by its users.
4. Linkability: App and advertiser can identify their audience from the analytics data.

4.5 Analysis

Given that the trust level of FB and the app are not comparable, the question is how can FB control such data leaks? Some of the broad ways to contain these data leaks are:

1. By increasing the user's privacy policy specification scope from current user-object layers (refer Fig. 2) to all the layers of the platform, except the owner's layer. The current approach is fragmented and incoherent – that is, impact of changes at app layer on *in-force* settings at user layer is not communicated to users. The use of naturally understandable labels like "Friends", "Family" should be devised to categorize apps and advertisers, using which user can define her access policies.
2. By encrypting the analytics available to apps and advertisers such that per campaign a distinct but ciphered string is generated for each measurable event that cannot be used to track users across campaigns. Only the platform owner should link the events across campaigns. Thus, only one entity takes the accountability.
3. It appears that FB is trying to address this issue of linkability through the concept of `scope_id`. A user is assigned a unique local ID, whose scope is limited to the context (App, Page) for which it is generated. For example, App1 will generate a `scope_id`, which is different from the `scope_id` generated by App2. Thus App1 and App2 or their parent cannot link users. However, we observed that, as of now, these scope IDs are resolving to the real user ID for whom the scope IDs were generated. For example, https://fb.com/100007460080360, https://fb.com/2051781625080487, and https://fb.com/1708004396124880 reveal the actual user.

5 Related Work & Discussion

Social networks like Facebook, Twitter, Snapchat have come to prominence in last decade because of their ability to engage users online such that users can carry out

	A	C	D	G	H	I	J	L	N
1	Reporting Starts	Campaign Name	Region	Results	Result Indicator	Reach	Impressions	Amount Spent (INR)	Page Likes
2	01/04/18		Unknown	6	actions:like	32	32	13.18	6
3	01/04/18	Followers for Online Privacy Page	Assam	1	actions:like	7	8	2.16	1
4	01/04/18	Followers for Online Privacy Page	Haryana	2	actions:like	2	2	2.92	2
5	01/04/18	Followers for Online Privacy Page	Himachal Pradesh			1	1	0.47	
6	01/04/18	Followers for Online Privacy Page	Jammu and Kashmir			2	2	0.32	
7	01/04/18	Followers for Online Privacy Page	Maharashtra	1	actions:like	1	1	0.14	1
8	01/04/18	Followers for Online Privacy Page	Odisha			1	1	0.1	
9	01/04/18	Followers for Online Privacy Page	Punjab region	1	actions:like	1	1	2.35	1
10	01/04/18	Followers for Online Privacy Page	Rajasthan			1	1	0.22	
11	01/04/18	Followers for Online Privacy Page	West Bengal	1	actions:like	9	10	2.83	1
12	01/04/18	Followers for Online Privacy Page	Bihar			1	1	0.07	
13	01/04/18	Followers for Online Privacy Page	Madhya Pradesh			1	1	1.14	
14	01/04/18	Followers for Online Privacy Page	Uttar Pradesh			1	1	0.07	
15	01/04/18	Followers for Online Privacy Page	Jharkhand			2	2	0.39	

Fig. 11. Campaign measurement report

their social discourse 24x7, around the world. As the users get convenience and real-time engagement with their connections for free, the platform gets user insights. The platform recovers its operational costs by sharing the insights in plausibly privacy-preserving fashion with advertisers https://fb.com/ads/about/. The rich data-sets generated by such social networks have ushered: advertising into a real-time persuasion industry [18,27,26], communication into a precision tracking system [1,7], and social network platform into a rich user/content/relation labelling platform. All of these transformations have brought in tremendous challenges [19] in terms of privacy of users.

Privacy in social networks has been studied for quite some time and the research community had been highlighting privacy implication of connectivity [14] even before the Cambridge Analytica fiasco. In [13], a survey on security and privacy in social networks is presented that touches upon properties like: anonymization, de-anonymization, link predictability [11,15], information leakage, trust [22], and link privacy [20]. In [10], a privacy-preservation model for Facebook-style social network is proposed. Concepts for privacy-preservation in an app ecosystem, presented in [16] for mobile platforms, can be borrowed in FB's platform. FB's infrastructure [2] is a unique and not much is available in public. It remains interesting to see how Facebook adopts to the forthcoming European GDPR [5] regulation. The data generated across layers of Facebook platform is interlinked and once a data-tuple is associated with personal data, it becomes tainted and the tainted attributes propagate user's identity further. Under GDPR, when a Facebook user invokes her right to be forgotten/erased, it will be interesting to see how far the data deletion chain goes; since the data is linked across the ecosystem. We believe that Facebook will have to define context and scope of user information and the deletion of user data will happen within that pre-defined scope.

6 Conclusion

We presented the role Apps play in tracking and profiling users on Facebook platform. We have shown a few instances of App configurations that violated the underlying primary privacy settings of the user. Apps may use such shortcomings in policy enforcement for various reasons that can seriously undermine not only the privacy of users but also their security. From the study of ecosystem on FB's platform we showed that Apps potentially have as much visibility of its users' objects, connections, and interactions as FF itself. If a coherent access control model across layers of FB ecosystem is not deployed, then FB with its ad-hoc approach will remain a sophisticated surveillance system available to any user. People, including lawmakers, around the world are asking FB should it really be expanding into influencing people based on what it has captured as their profile? This conundrum is multiplied in the presence of millions of Apps on its platform. App permission management need to be made understandable and available as extensional/intensional labels similar to permission management at users layer. It is not hard to see why our recommendations based on our analysis demands expansion of the scope of user privacy policies across user layer, app layer, and beyond.

Acknowledgments: This work is carried out as part of research at ISRDC (Information Security Research and Development Center), supported by Ministry of Electronics and Information Technology, Govt. of India (15DEITY00-004). The authors would like to thank Anshu S. Anand, Abhishek Behra, Ankush Dubey for their participation in discussions and experiments.

References

1. Acar, G., Alsenoy, B.V., Piessens, F., Diaz, C., Preneel, B.: Facebook Tracking Through Social Plug-ins. Tech. rep., KU Leuven (06 2015)
2. Bronson, N., et al.: TAO: Facebook's Distributed Data Store for the Social Graph. In: USENIX ATC 13. pp. 49–60 (2013)
3. Cadwalladr, C.: 'I made Steve Bannon's psychological warfare tool': meet the data war whistleblower. *online* (2018), The Guardian
4. Carminati, B., Ferrari, E., Perego, A.: Enforcing access control in web-based social networks. ACM Trans. Inf. Syst. Secur. **13**(1), 6:1–6:38 (Nov 2009)
5. European Union: Data Protection - Rules for the protection of personal data inside and outside the EU. *online* (2018)
6. Facebook: Data policy. *online* (2016)
7. Facebook: About facebook pixel. *online* (2018)
8. Facebook: Cracking down on platform abuse. *online* (2018)
9. Facebook: Graph API overview. *online* (2018)
10. Fong, P.W.L., Anwar, M., Zhao, Z.: A privacy preservation model for facebook-style social network systems. In: ESORICS'09. pp. 303–320. Springer-Verlag (2009)
11. Gilbert, E., Karahalios, K.: Predicting tie strength with social media. In: Proc. of the SIGCHI Conference on Human Factors in Computing Systems. pp. 211–220. CHI '09, ACM (2009)
12. International Personality Item Pool: The 3,320 IPIP items in alphabetical order. *online* (2018)
13. Joshi, P., Kuo, C.C.J.: Security and privacy in online social networks: A survey. In: 2011 IEEE International Conference on Multimedia and Expo. pp. 1–6 (July 2011)
14. Juels, A.: Targeted advertising ... and privacy too. In: CT-RSA 2001. pp. 408–424 (2001)
15. Kahanda, I., Neville, J.: Using transactional information to predict link strength in online social networks. In: International AAAI Conference on Web and Social Media (2009)
16. Lee, S., Wong, E.L., Goel, D., Dahlin, M., Shmatikov, V.: Box: A platform for privacy-preserving apps. In: NSDI 13. pp. 501–514. USENIX (2013)
17. Levy, H.M.: Capability-Based Computer Systems. Digital Press (1984)
18. Matz, S.C., et al.: Psychological targeting as an effective approach to digital mass persuasion. PNAS **114**(48), 12714–12719 (2017)
19. Michal, K., et al.: Facebook as a research tool for the social sciences: Opportunities, challenges, ethical considerations, and practical guidelines. American Psychology **70**(6), 543–556 (2015)
20. Mittal, P., Papamanthou, C., Song, D.: Preserving link privacy in social network based systems. CoRR **abs/1208.6189** (2012)
21. Patil, V.T., Jatain, N., Shyamasundar, R.K.: Role of apps in undoing of privacy policies on facebook. Tech. rep., ISRDC, IIT Bombay (2018), *online*
22. Patil, V.T., Shyamasundar, R.K.: Privacy as a currency: Un-regulated? In: Volume 4: SECRYPT'17. pp. 586–595 (2017)
23. Patil, V.T., Shyamasundar, R.K.: Undoing of privacy policies on facebook. In: IFIP WG 11.3 Conference, DBSec 2017. pp. 239–255 (2017)
24. ProPublica Data Store: Facebook ad categories. *online* (2016)
25. Roosendaal, A.: We Are All Connected to Facebook ... by Facebook! European Data Protection: In Good Health? (2012)
26. Sam Biddle: Facebook uses artificial intelligence to predict your future actions for advertisers, says confidential document. *online* (2018)
27. Youyou, W., Kosinski, M., Stillwell, D.: Computer-based personality judgments are more accurate than those made by humans. PNAS **112**(4), 1036–1040 (2015)