

EXHIBIT 150

UNREDACTED VERSION OF DOCUMENT SOUGHT TO BE LODGED UNDER SEAL

From: Konstantinos Papamiltiadis </O=THEFACEBOOK/OU=EXTERNAL (FYDIBOHF25SPDLT)/CN=RECIPIENTS/CN=942ACCD3D3C54FBEA8B7253E97A8A6D7>
Sent: Monday, November 18, 2013 9:55 AM
To: Simon Cross
Cc: Monica Walsh
Subject: Re: The Linkedin scenario

Great feedback, Simon! Thanks a lot.

You are right, this process is about the Permission Review for access to data available via Public APIs and as such updating it based on your feedback makes total sense.

Unfortunately I am about to hit the road now and will be traveling for the next two days (Seedcamp in Berlin), is it ok if we synch up on Thursday?

Thanks again,
kp

From: Simon Cross <si@fb.com>
Date: Monday, November 18, 2013 at 5:21 PM
To: Konstantinos Papamiltiadis <kpapamiltiadis@fb.com>
Cc: Monica Walsh <mwalsh@fb.com>
Subject: Re: The Linkedin scenario

+ Monica

Hi KP – Awesome that you've been thinking about this.

In parallel, Monica and I have been working on a subset of your framing: specifically around providing a clear framework for making decisions about whitelisting apps for the newly privatized APIs. The way we're doing this is to divide up every app which is currently using to-be-privatized APIs into four buckets, and to subdivide into segments from there:

- 1/ Games – Namita, Gareth etc
- 2/ Mobile/Carrier/Device/OEM – Peter Kang, Vikas etc – Samsung, Nokia, HTC, Vodafone etc
- 3/ PMD/Marketers – solutions engineering / PMD team etc – Nanigans, Coke etc
- 4/ Non-games – essentially everything else – i.e. Our Team – Messaging apps

We've scoped this work specifically to the to-be-privatized APIs (read_stream, friends_* perms) - we're not covering the permissions/APIs which continue to be public (friends, login etc).

We're at the stage where each of four groups is going off and defining their criteria for exempting or extending an app from PS12n deprecations. Monica is leading this work.

It feels to me that we should separate the work around whitelisting apps for newly private APIs from the work to decide who gets access to the public features of platform: login, friends list. To me, the people we'll be considering for private APIs are those developer who are allowed to use the public APIs.

To that end, I'd rescope this deck specifically around the Unified Review process – i.e. A framework for making decisions about how we decide who gets access to which PUBLIC APIs.

Lets also not frame using the word 'partner' as that is a phase we should reserve for those apps which have access to private APIs – lets just call this Apps.

Given that this is really about Unified Review policies, suggest you make this a co-production with someone from DevOps/Policy - who from that team have you worked with on this yet?

Want to do a quick call today to sync?

S

--

Simon Cross

Product Partnerships

www.facebook.com/sicross @sicross

Facebook, 1601 Willow Road, Menlo Park, CA 94025, USA

From: Konstantinos Papamiltiadis <kpapamiltiadis@fb.com>

Date: Monday, November 18, 2013 at 7:54 AM

To: Simon Cross <si@fb.com>

Subject: FW: The Linkedin scenario

Hello Simon,

As you know with PS12n, the intention is to review (and approve/reject) all permissions a developer requests from users.

I think it will be helpful to create a framework that our devops would use to classify the app under one of the following buckets:

- 1/ Competitive
- 2/ Potentially competitive
- 3/ Aligned
- 4/ Partner (contract)

Given you have been running point on PS12n and have a much better context, I was wondering if you have time to review the process/classification criteria suggested in the attached before we can circulate it with product for their input. Eddie suggested this would be useful for him when he talks to Doug about who should be approved for certain permissions and who shouldn't.

Thanks a lot,
kp

From: Konstantinos Papamiltiadis <kpapamiltiadis@fb.com>

Date: Friday, November 15, 2013 at 1:08 PM

To: Eddie O'Neil <ekoneil@fb.com>

Cc: Simon Cross <si@fb.com>

Subject: Re: The Linkedin scenario

+ Simon as the running point in our team with anything related to PS12n (Simon see below for some context)

Eddie, do let me know if it would be more convenient to have a quick chat this morning instead since you are off the hook from the developer conference.

Cheers,
kp

From: Konstantinos Papamiltiadis <kpapamiltiadis@fb.com>
Date: Tuesday, November 12, 2013 at 3:05 PM
To: Eddie O'Neil <ekoneil@fb.com>
Subject: Re: The Linkedin scenario

Hello Eddie,

I was wondering if could share your thoughts on this one below.

I need to make a recommendation to Ime/Chris on how to deal with Linkedin and your input is super valuable to formulate our approach.

Thanks a lot,
konstantinos

From: Konstantinos Papamiltiadis <kpapamiltiadis@fb.com>
Date: Sunday, November 10, 2013 at 9:56 AM
To: Eddie O'Neil <ekoneil@fb.com>
Subject: The Linkedin scenario

Hello Eddie,

As you may recall, we had a chat a few days back on how best to deal with Linkedin going forward. I need to understand a few things with regards to your current thinking about Unified Review, PS12n and how the “competitive businesses” will be treated, in order to decide whether we should take action now or leave it for later PS12n comes to light. Namely:

1. Are we deprecating the ability for developers to access work history generally?

My current understanding is that this won't be deprecated, but depending on the use case under which the data is used, we can allow devs to access work history as part of the Unified Review.

2. Is PS12N changing the definition of what is competitive today?

This is grey area for me. Last time we talked, the plan was not ready to be distributed to a wider audience, but could you give me an idea of what constitutes competitive and how are we planning to treat them differently? I.e less permissions, restrict access to APIs, block them all together, etc.

3. How specifically will PS12n impact LinkedIn?

If none of the above 2 will apply to Linkedin, do you foresee them being affected somehow?

Just fyi, when the DevOps team reviewed Linkedin recently for me, there could not find a noticeable Policy Violation that would allow us to block them from accessing sensitive data. In terms of the plan, my current thinking is to allow them to use the data but push them into a direction where:

- 1/ User changing jobs and announcing that on Linkedin will be able to share on FB
- 2/ User's being promoted and changing titles on Linkedin will be able to share on FB
- 3/ Other lifetime events, such as Education History, Qualifications, etc should be shared on FB

At this point of time, sharing a status update posted on LinkedIn to FB is not good enough for me, but is deemed good enough in terms of reciprocity.

Thanks a lot,
kp