# Report a Security Vulnerability - Deanonymizing Facebook users by exploiting CSP header functionality

Fall #207164694
Der Fall ist geschlossen

Wir haben dir eine Nachricht geschickt.
Vor 22 Stunden
That's fine, I'd just ask that if you do include it you include the conversation verbatim :-)

Neal
Security
Facebook
**Der Fall wurde noch nicht geschlossen, daher musst du nichts weiter unternehmen. Danke, dass du uns kontaktiert hast.**
Fallverlauf

Du hast dich an uns gewendet.
1. August 2014 16:57
Great, thx :)

Can I also provide a copy this conversion publicly?

(sorry for reopening, didn't know how I could ask that otherwise)

Cheers
Pascal

Wir haben dir eine Nachricht geschickt.
1. August 2014 16:42
Hey Pascal,

That sounds fine with me.

Neal
Security
Facebook

Du hast dich an uns gewendet.
1. August 2014 11:23
Hey Neal,

thank you for your timely reply and the detailed information. Let me just reiterate two things:

- You're right about the ~250KB restriction for the CSP Header in Chrome, but only if it is sent _via HTTP Header_. By using a <meta http-equiv="Content-Security-Policy" content=""> it's possible to use much more data. In my POC i'm using ~5MB (100.000 profiles).

- You're also right about the identification of a completely random user (hardly possible). But if you can narrow it down to a certain group (as outlined in "Prequalification of Facebook profile URLs" - I guess at least the matching from IP Location to Hometown will provide a good start) it'll work. Of course it's arguable if that's still a problem :)

In any case, I guess you've already discussed most of the possible scenarios and decided to let the problem solve itself by latest CSP standard.

So I would like to ask for your permission to go ahead and publish the article I sent you.

Cheers
Pascal

Wir haben dir eine Nachricht geschickt.
1. August 2014 02:22
Hi Pascal,

I appreciate you writing in with this report. As you noted, there has been a fair bit of research and discussion into CSP (and path-based rules specifically) in the past 12 months or so.

- Back in January, Egor Homakov outlined a brute-forcing attack very similar to the one you describe: http://homakov.blogspot.com/2014/01/using-content-security-policy-for-evil.html
- In February there was a public discussion of the potential for abuse of this functionality on a W3C mailing list: http://lists.w3.org/Archives/Public/public-webappsec/2014Feb/0036.html
- Another researcher recently published a post discussing this behavior and how it can be used to detect whether or not a user is logged in to a site: http://words.zemn.me/csp

Along with those external discussions we've had discussions about this issue within the security team. There are a few points which I think help contextualize this attack:

- As you point out in your PDF, this behavior has actually been addressed by the W3C in the latest version of the standard (http://www.w3.org/TR/CSP11/#source-list-paths-and-redirects) and will naturally cease to function as browsers support the latest version of the standard.
- As you also, mentioned, the particular attack vector you describe can not feasibly be used to identify an arbitrary person since it requires you to enumerate a list of usernames which you then attempt to search through.
- There are technical limitations on the amount of data you can send in an individual header: I believe Chrome's limit is roughly 250 KB. That further reduces the number of user accounts you can target in a single request.

Again, thank you for sending this in. Please let me know if you have any additional questions.

Best,

Neal
Security
Facebook

Wir haben dir eine Nachricht geschickt.
31. Juli 2014 11:03
Hallo,

Danke für deine Meldung. Wenn du einen Sicherheitsfehler meldest, reagieren wir auf deine Meldung so schnell wie möglich. Deine Meldungsnummer lautet 207164694. Wenn du uns nicht auf Englisch schreibst, sondern in einer anderen Sprache, können wir dir derzeit nur auf Englisch antworten. Wir möchten uns für jegliche damit verbundenen Unannehmlichkeiten entschuldigen.

Wenn du ein anderes Problem meldest, lies dir bitte die unten stehenden Informationen durch, wie du hierbei Hilfe erhältst.

- Falls dein Konto oder das eines Freundes verdächtige Links verschickt:
https://www.facebook.com/help/hacked
- Meldung von Missbrauch: https://www.facebook.com/help/reportlinks
- Falls du Fragen oder Bedenken hast, besuche bitte unseren Hilfebereich:
https://www.facebook.com/help

Mit freundlichen Grüßen
Das Facebook-Team

Du hast dich an uns gewendet.
31. Juli 2014 11:00
Vulnerability Type
Privacy / Authentication
Vulnerability Scope
Main Site (www.facebook.com)
Title
Deanonymizing Facebook users by exploiting CSP header functionality
Product / URL
https://www.facebook.com/
Description and Impact
I think I have discovered a possible privacy breach exploit on Facebook, currently only working in Google Chrome. By bruteforcing profile urls as a restriction in the Content-Policy-Header it's possible to identify if a user is currently logged in and what his exact profile url is.
Reproduction Instructions / Proof of Concept
Please see http://myseosolution.de/test/csp-deanonymisierung.php?network=Facebook&lang=en&token=Qzfrklwlmvsszknigmt3 for a POC.

1. Open Google Chrome (25 to 36)

2. Make sure not to block Third-Party-Cookies
3. Log into Xing
4. Enter up to 750 profiles in the text field and make sure to include your own profile *
5. Hit "Start deanonymization"
You should get an output similar to this:
• Starting deanonymization...
• Check: CSP implementation blocks access on path level... true
• Check: CSP implementation allows access on path level... true
• Check: Logged into Xing... true
• Profile found in test group 0? false
• Profile found in test group test-set? true
• 251 profiles left... decision-tree: [0]
• 126 profiles left... decision-tree: [0,1]
• 63 profiles left... decision-tree: [0,1,0]
• 32 profiles left... decision-tree: [0,1,0,0]
• 16 profiles left... decision-tree: [0,1,0,0,0]
• 8 profiles left... decision-tree: [0,1,0,0,0,0]
• 4 profiles left... decision-tree: [0,1,0,0,0,0,1]
• 2 profiles left... decision-tree: [0,1,0,0,0,0,1,0]
• 1 profiles left... decision-tree: [0,1,0,0,0,0,1,0,0]
• Profile: https://www.facebook.com/[YOUR USERNAME]

* This step is required for the POC to work, because I did not want to scrape "real" user profiles - not sure about the legal issues that might be involved in mass scraping. In a real world scenario one would use a targeting method to find a predefined set of profiles to check.

I attached a document with a more detailed explanation of what's going on and why I think it's dangerous. See section "Prequalification of Facebook profile URLs" for some practical approaches to find predefined user profiles.

Before I publish it at http://www.myseosolution.de/deanonymizing-facebook-users-by-csp-bruteforcing I wanted to check back with you guys if that 'exploit' is something for you to worry about or if it's more or less a "won't fix, go ahead".

Other networks are vulnerable to this exploit as well but before I provide them with the detailed information I attached, I wanted to get your approval (since it's kinda third party and the article is tailored to facebook)

Cheers,
Pascal Landau
Is this bug public or known by third parties?
No
Can you reproduce this issue every time?
Yes
How did you find this bug?
Manually / Other
Anhänge:
Deanonymizing Facebook Users By CSP Bruteforcing.pdf