Improper Authorization in go-gitea/gitea

Valid Reported on Mar 6th 2022

0

Description

When Gitea is build and configured for PAM authentification it skips checking authorization completely. Therefore expired accounts and accounts with expired passwords can still login.

Proof of Concept

You can expire an account with `chage -E0 <username>` and still login.

Impact

Since disabling an account in PAM still allows to login via ssh-keys, it's common to set accounts to expire if you want to deny access. So accounts who technically don't have any privilege are still allowed to login. To circumvent this, after an successful call to `pam_authenticate` it is necessary to call `pam_acct_mgmt` .

References

- C3H2-CTF

CVE
CVE-2022-0905
(Published)

Vulnerability Type
CWE-285: Improper Authorization

Severity
Medium (6.5)

Visibility
Public

Status
Fixed

Fixed

Found by

ysf

@ysf

unranked

Fixed by

ysf

@ysf

unranked

This report was seen 1,119 times.

We are processing your report and will contact the **go-gitea/gitea** team within 24 hours.
9 months ago

ysf modified the report 9 months ago

ysf modified the report 9 months ago

ysf modified the report 9 months ago

ysf modified the report 9 months ago

ysf 9 months ago Researcher

This has also been reported and verified for Gogs. It makes sense to make both public at the same time.

ysf modified the report 9 months ago

We have contacted a member of the **go-gitea/gitea** team and are waiting to hear back
9 months ago

ysf submitted a patch 9 months ago

zeripath 9 months ago Maintainer

Hi @ysf

Thank you for reporting this could you show me your pam configuration when this occurs?

Gitea (and Gogs for that matter) expects that pam authorization eg. account checking would be done with a pam module.

We've deliberately not done authorization checking within the module as pam can do that itself - e.g. if unix expiry checks are required by the administrator the pam configuration should set required for these themself.

ysf 9 months ago Researcher

Hi,

I'm not at my desk for around a week now. But I used the /etc/pam.d/login service configuration as other apps also do. In the end this resolves to pam_unix.so - I entered "login" at the Pam Service prompt.

AFAIK if you set the account parameter in the pam.d/config you only instruct libpam what to check for if you call the pam_acct_mgmt() method, and this is what you skipped. So it doesn't matter what you put in the pam-configuration if you don't instruct libpam to check against.

This is normal because if a password is expired you still want the user to authenticate correctly and afterwards show a password-change prompt.

Also FYI gogs had the same problem too and validated it and accepted my fix.

ysf 9 months ago Researcher

English is not my mother language, don't want to sound rude or so. Thanks for any help

zeripath 9 months ago Maintainer

Sorry I've not been clear.

Gitea/Gogs was only using the authentication component of PAM.

The ostensible reason being two fold:

if you want to use authorization features of account management like expiry you should be able

to set those to be checked as part of the authentication PAM check.

Gitea manages its own users in its account management UI. If we were to add in this authorization/account mgt check we have no way of checking this in the admin UI or displaying in the admin UI. We have our own way of disabling users in the UI already.

NOW... this is not to say that we should not be doing this account management/authorization check - and in fact I considered adding it in 2 years ago - however, changing this needs to be considered carefully and perhaps should be optional. Certainly it's forms a breaking change.

I agree it's a kind of gotcha however, we need to think more clearly about what and why we're using PAM here. We currently offer AUTHENTICATION with PAM not AUTHORIZATION.

You suggest that "if a password is expired you still want the user to authenticate correctly and afterwards show a password-change prompt." but your fix does not provide for that - and nor can it. AFAIU this kind of disabling can be done at auth level - even if it's not done by default.

As this is a gotcha that is catching people out and you're right it could lead to people having access unexpectedly I'm of a mind that we should apply the fix. But we should consider if we need to make it optional or not.

zeripath 9 months ago Maintainer

I guess the question is

Can this expiry check be done at auth level or does it have to be done at account level?

My understanding is that it can be.

If it cannot be done at auth level - this is a definite security issue and it deserves a more quick release. If it can be - then this is a configuration gotcha issue - meaning we need to consider making it optional. (Although actually as the account checks are still pam configurable - perhaps we don't need the optional at all - people can just remove that bit from their configuration.)

zeripath modified the report 9 months ago

ysf 9 months ago Researcher

Hey,
thanks for clarifying. I've not been clear also. What I meant with the password c[illegible] was only to elaborate why pam_authenticate returns PAM_SUCCESSFUL even when the

passwords or accounts are expired. In that case a call to pam_acct_mgmt would return PAM_NEW_AUTHTOK_REQD or PAM_ACCT_EXPIRED.

There no explicit need to support password change via your app. You're just not allowed to let them login, this is why my fix is the smallest change possible.

AFAIK this can't be changed at PAM configuration level.

zeripath validated this vulnerability 9 months ago

ysf has been awarded the disclosure bounty ✔

The fix bounty is now up for grabs

zeripath 9 months ago Maintainer

Do you want to send a PR against Gitea or should I just do this?

ysf 9 months ago Researcher

You can cherry pick the commit from my branch here:

https://github.com/go-gitea/gitea/compare/HEAD...ysf:main

This would be quicker, as I'm only on phone right now.

zeripath 9 months ago Maintainer

We need to ensure that any resulting CVE is clear that this:

Only affects users who compile the pam module in.

Only affects users who expected Gitea to check account module pam authorization.

zeripath 9 months ago Maintainer

I have opened a PR using your branch https://github.com/go-gitea/gitea/pull/

The opening comment is as follows:

The opening comment is as follows:

The PAM module has previously only checked the results of the authentication module.

However, in normal PAM practice most users will expect account module authorization to also be checked. Without doing this check in almost every configuration expired accounts and accounts with expired passwords will still be able to login.

This is likely to represent a significant gotcha in most configurations and cause most users configurations to be potentially insecure. Therefore we should add in the account authorization check.

:warning: **BREAKING** :warning:

Users of the PAM module who rely on account modules not being checked will need to change their PAM configuration.

However, as it is likely that the vast majority of users of PAM will be expecting account authorization to be checked in addition to authentication we should make this breaking change to make the default behaviour correct for the majority.

I suggest we backport this despite the BREAKING nature because of the surprising nature of this.

Thanks to @ysf for bringing this to our attention.

ysf 9 months ago — Researcher

Yes absolutely. @admin can you edit the report to mention this?

zeripath marked this as fixed in **1.16.4** with commit **1314f3** 9 months ago

ysf has been awarded the fix bounty ✔

This vulnerability will not receive a CVE ✖

Jamie Slome 9 months ago — Admin

@ysf - can you give me the exact details that you would like me to copy and p
report, and tell me which section I should add it to?

A **go-gitea/gitea** maintainer 8 months ago Maintainer

I would argue about CWE, imho it would be more about CWE-303, not CWE-285

A **go-gitea/gitea** maintainer 8 months ago Maintainer

CVE score also needs to be updated: AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:L/A:N

A **go-gitea/gitea** maintainer 8 months ago Maintainer

Privileges required can not be none because you still need to know user password, none would be unauthorized exploitability. Confidentiality should be low as you would still be able to access to data that user already has access to not anything high'er

Lauris BH 8 months ago Maintainer

Did not realize I was writing it while unauthorized :)

ysf 8 months ago Researcher

Heyhey, I'm with you that the score can be lower. However, this is due to huntr.devs formular and missing impact and environment scoring.

The privileges have to stay none none because you're mixing authentication with authorization here. This issue is an unauthorizied exploitability but the user is not anonymous, this is a clear privilege of zero like it is in other CVEs too.

CWE-303 is wrong because, again, the authentication has been implemented correctly. The correct CWE is CWE-863: Incorrect Authorization, I did not find a way to choose it here.

The Confidentiality is not low because you can access data the user has no access to otherwise. It does not matter if the user had access to before that.

Jamie Slome 8 months ago Admin

Just for clarity, it would be good if we could have a consensus here, so will allo
and forth discussion to reach an agreement, but we do side with the maintai
not reach a consensus, I will update the report/CVE/CWE if needed!

Will keep an eye on this report 👁

ysf 8 months ago Researcher

What I'm trying to say is: I'm completely fine with a lower score. I'd think something around 5.5-6.5 is adequate. But this should not be achieved by stating wrong facts or because of huntr.devs lack of CVE formula. Even if @maintainers disagree on the the scoring facts, I truely don't mind, I just try to help both of you guys.

Stating this is a CWE-303 tho to me is non negotiably wrong and shows the main problem in the first place: not understanding the difference between authentication and authorization - which ironically is the reason we're all here now.

Lauris BH 8 months ago Maintainer

From Gitea perspective whole PAM/OpenID etc or anything else is **authentication** process, it does not in any way declares/influences what resources in Gitea is user able to access (except for LDAP where there is special additional attribute synchronization). The result of authentication is only is user is able to access Gitea or not.
This bug has no influence in anything related to authorization (checks for specific user level or access to specific resources).

Bug in authentication that user with expired password is still able to authorize does not somehow grant's him access to new resources etc. Yes he should not be able to login anymore but because he can, it's problem in authentication not authorization.

And let's keep discussion without accusations and keep the tone respectful - I know what is authentication and what is authorization and have known it for more than 20 years already ;)

ysf 8 months ago Researcher

As always I'm sorry when you took this off disrespectful, this was not intended. English is not my mother language and I try to communicate with a dictionary and translation services. I too am not new to the field. I'm hacking for more than 30 years now and work for more than 10 years in a CERT for high-security purposes. If we're talking about experience, let me tell you that I completely understand your point. There is a big "but": Even if in gitea you implemented your own authorization, it's wrong to categorize it that way because you have to see it from a pam perspective. This is why I said that the huntr.dev formular is broken, because in a normal CVE process you could explain this in the "impact". In the end you want to compare CVEs and find issues with CWEs also and it would be a bad sign to categorize this wrongly because you're using pam wrong. If you want this to be stated well, the correct behaviour wo
emphazize the impact in the report and not a wrong authentication in the ca
tldr: Having a CNA doing things completely different than all the other CNAs breaks unifing

CVEs and research in this field - especially if "pam_acct_mgmt" in the micro level is always authorization - no matter what you make out of.

ysf 8 months ago Researcher

@admin I think everything has been said here. I found it, I told it, I fixed it. I'm not going to fight against a project I'm not involved in, and with a startup I'm not earning anything from. If I stumble around problems with gitea again I'll use the classic cveform to save everyones time.

Lauris BH 8 months ago Maintainer

Gitea is not a startup, Gitea is completely community maintained project and maintainers also do not earn anything and do that in their free time.

Let's break it down about scoring (this is my reasoning):
On `AV:N`, `AC:H`, `UI:N`, `S:U` and `A:N` we agree on.
About others:

`PR:L` - it can not be none as you need to know password to be able to authorize

`C:L` - in my opinion it is low because attacker has no control over what data he will have access to when authorized

`I:L` - same as for `C`
So to me it results in `AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:L/A:N`

On hand I only found one CVE with similar vulnerability and it's scoring close the same as I'm proposing: https://www.cvedetails.com/cve-details.php?t=1&cve_id=CVE-2006-7108

About CWE it's always been kind a subjective and I did not even find any similar CVE with CWE assigned. If we can not agree on specific one, we can place it in more generic one :)

ysf 8 months ago Researcher

With startup I meant huntr.dev.

Lauris BH 8 months ago Maintainer

Sorry, copied wrong cvss - `AV:N/AC:H/PR:L/UI:N/S:U/C:L/I:L/A:N`

ysf 8 months ago Researcher

All fine Lauris. Thanks for your constructiveness. As I said, I did my job, now everything is up to you. I'm not going to fight and am fine with everything.

Jamie Slome 8 months ago Admin

@maintainer - seeing as we have reached a form of consensus, can you please confirm the updates that you would like me to make to the report + CVE?

@ysf - I have dropped you a message on Discord, as want to discuss some of your feedback here more with you 👍

Lauris BH 8 months ago Maintainer

CVE score should be updated to `AV:N/AC:H/PR:L/UI:N/S:U/C:L/I:L/A:N`

as for CWE I have no strong request to change, imho it's CWE-303 from Gitea as product point but I will let other maintainers to decide this @zeripath @maintainers?

Lauris BH 8 months ago Maintainer

@admin a bit offtopic - how does project maintainer label gets added, when I was writing as unauthorized I had `Maintainer`, now that I'm authorized it's not?

Jamie Slome 8 months ago Admin

The `Maintainer` label is added to chat messages when you have `write` permissions for a repository.

I can manually add you as a maintainer to this repository, and you should have full privilege capabilities.

Would you like me to go ahead and do this?

With regards to the CVE/CVSS adjustments, I will wait for the confirmation from the other maintainers before proceeding 👍

Lauris BH 8 months ago Maintainer

Yes, would be nice if you could add

Jamie Slome 8 months ago Admin

Sorted! ♥ You are now added as a maintainer to the repository 👍

Sign in to join this conversation