[OSSA-2021-003] Account name and UUID oracles in account locking (CVE-2021-38155)

Bug #1688137 reported by Samuel de Medeiros Queiroz on 2017-05-03

This bug affects 2 people 268

Affects	Status	Importance	Assigned to	Milestone
OpenStack Identity (keystone)	Triaged	Medium	David Wilde	
OpenStack Security Advisory	Fix Released	Medium	Jeremy Stanley	

Bug Description

```
This relates to PCI DSS features added in the Newton release.
The involved PCI DSS requirements are 8.1.6 and 8.1.7, as described below:

8.1.6 Limit repeated access attempts by locking out the user ID after not
more than six attempts.
8.1.7 Set the lockout duration to a minimum of 30 minutes or until an
administrator enables the user ID.

The options lockout_failure_attempts and lockout_duration implement those
behaviors, respectively.

If those options are enabled in the keystone configuration file, for
example:

[security_compliance]
# Setting the account lockout threshold
lockout_failure_attempts = 2
lockout_duration = 10

All users in the cloud get exposed and can be subject of an attack.

The attacker could lock out an user account by:

1) Try to auth on a user's behalf:

POST /v3/auth/tokens

{ "auth": {
    "identity": {
      "methods": ["password"],
      "password": {
        "user": {
          "name": "sam",
          "domain": { "id": "default" },
          "password": "fake_password"
        }
      }
    }
  }
}

And after lockout_failure_attempts attempts, as the password is wrong (as
the attacker do not know it), the server returns:

{
  "error": {
    "code": 401,
    "title": "Unauthorized",
    "message": "The account is locked for user: 94ab353983174b0
4955fc9842779b085."
  }
}

And now the attacker even know the user's ID.

OR

2) Try to change a user's password on their behalf (would need to know the
user's ID):

POST /v3/users/<user_id>/password
{
 "user": {
  "original_password": "fake_password",
  "password": "new_password"
 }
}

As the original password is wrong (as the attacker do not know it), after
lockout_failure_attempts attempts that user account get locked out by
lockout_duration.

For both 1) and 2), before lockout_failure_attempts attempts, you get:

{
  "error": {
    "code": 401,
    "title": "Unauthorized",
    "message": "The request you have made requires authentication."
  }
}

After lockout_failure_attempts attempts, you get:

{
  "error": {
    "code": 401,
    "title": "Unauthorized",
    "message": "The account is locked for user: 94ab353983174b0
4955fc9842779b085."
  }
}
```

These approaches can be used by an attacker to lock out users indefinitely
by locking out users again and again after lockout_duration has passed.

See original description

Tags: in-stable-train in-stable-ussuri in-stable-victoria in-stable-wallaby

CVE References

2021-38155

Samuel de Medeiros Queiroz (samueldmq) wrote on 2017-05-03: #1

This can get even worse. An attacker can simply use a user name and user's
domain name to call /v3/auth/tokens on their behalf and get their account
locked.

For example:

```
POST /v3/auth/tokens

{ "auth": {
    "identity": {
      "methods": ["password"],
      "password": {
        "user": {
          "name": "sam",
          "domain": { "id": "default" },
          "password": "fake_password"
        }
      }
    }
  }
}
```

And after lockout_failure_attempts attempts, the server returns:

```
{
  "error": {
    "code": 401,
    "title": "Unauthorized",
    "message": "The account is locked for user: 94ab353983174b0
4955fc9842779b085."
  }
}
```

And now the attacker even know the user's ID.

Morgan Fainberg (mdrnstm) wrote on 2017-05-03: #2

Since this report concerns a possible security risk, an incomplete
security advisory task has been added while the core security reviewers
for the affected project or projects confirm the bug and discuss the scope
of any vulnerability along with potential solutions.

description:updated
Changed in ossa:
     status:New → Incomplete

Samuel de Medeiros Queiroz (samueldmq) on 2017-05-03

```
summary:- Attacker may use self-service password reset to lock out users
        - indefinitely
        + Attacker may use PCI-DSS 8.1.6 and 8.1.7 to lock out users indefinitely
```

Morgan Fainberg (mdrnstm) wrote on 2017-05-03: **Re: Attacker may use PCI-DSS 8.1.6 and 8.1.7 to lock out users indefinitely** #3

This bug is definitely a leak of information and if the PCI DSS features
are enabled, this could lead to user discovery. This bug is a possible DOS
for individual users, but does not allow an attacker to perform actions
that otherwise could not be done.

Likely the most correct fix is to ensure the "lockout" error is only shown
if the password is in-fact valid, eliminating the vector of user
discovery. As with any system that has lockouts, it is impossible to
determine the difference between a malicious user and a user with a poor
memory/forgotten password making many attempts. DOS for the individual
user is a known/expected aspect when lockouts are enabled.

This could be a Class A [0], if the user discovery leak is considered a
security vulnerability or a Class D [0] (a bug with security implications
and a hardening opportunity). Input from Keystone-Coresec will help to
identify the precise class of bug.

[0] https://security.openstack.org/vmt-process.html#incident-report-
taxonomy

Lance Bragstad (lbragstad) wrote on 2017-05-03: #4

These issues are valid, but when I think about them from the perspective
of other software there are a couple additional things in play. For
example, when using other services if an attacker is trying to log in as
me I usually get some sort for notification saying someone is attempting
to login from somewhere or that my account has been locked. Most of the
time this information is delivered through some sort of recovery contact
or address.

In keystone, I believe we emit notifications for things for failed
authentication and locked accounts (might need to double check that
though). In that case, one possible solution would be to write a consumer
that listens for those notifications and implements the recovery
notice/steps.

Samuel de Medeiros Queiroz (samueldmq) wrote on 2017-05-03: #5

I have updated the bug title and description to include what I stated on
comment #1, which goes beyond than just using the self-service password
API for the attack.

description:updated

Samuel de Medeiros Queiroz (samueldmq) wrote on 2017-05-04: #6

@Lance, the issue with your proposed possible solution is that we don't have such a consumer service in OpenStack, so we can't expect people have it in their deployments.

Lance Bragstad (lbragstad) wrote on 2017-05-04: #7

@Sam, correct. Expecting deployments to have that service would be unrealistic. It would be a work around for a deployment susceptible to the issues and want to mitigate out-of-band.

I'd be interested in investigating Morgan's proposal further.

Samuel de Medeiros Queiroz (samueldmq) wrote on 2017-05-04: #8

I like Morgan's proposal too.

If the attacker do not know it's working (since there is no message saying the user has been locked out), they are unlikely to continue.

Even if they do (they might know for sure that a given username exists), it will be up to the deployer to detect that a username has been under constant attack (we need to make sure we communicate well in the logs) and: 1) change the username or 2) ignore the lockout for that specific username until the attacker gets bored.

This can be combined with blacklisting the IPs from attackers, however I do not know how complex it can be to add such an automatic blacklisting mechanism into deployments.

Lance Bragstad (lbragstad) wrote on 2017-05-16: #9

I think our first course action is to implement Morgan's suggestion, where user information is only emitted if the password is correct.

Changed in keystone:
status:New → Triaged
importance:Undecided → Medium

Samuel de Medeiros Queiroz (samueldmq) wrote on 2017-06-15: #10

I am working on a patch for this bug and I have a question: is it okay to keep emitting a CADF notification with reason "The account is locked for user: <user_id>" ?

I assume it is, since that is not a message for final users (unless there is a system somewhere consuming it and giving it to users, but that is a different conversation and workflow).

Lance Bragstad (lbragstad) wrote on 2017-06-15: #11

I would agree - I think emitting notifications in this case is fine.

Samuel de Medeiros Queiroz (samueldmq) wrote on 2017-06-15: #12

1688137.diff (3.4 KiB, text/plain)

This patch fixes the issue with the solution proposed by Morgan, which had agreement from Lance and myself.

Tristan Cacqueray (tristan-cacqueray) wrote on 2017-08-15: #13

It seems like this warrants an advisory (class A according to VMT's taxonomy: https://security.openstack.org/vmt-process.html#incident-report-taxonomy).

@keystone-coresec, please review proposed patch in comment #12.

Is there a documented manual procedure to unlock accounts?

Jeremy Stanley (fungi) wrote on 2017-08-16: #14

Some interesting alternatives were floated in a NIST SP 800-63-3 update thread on the crypto ML this week: http://www.metzdowd.com/pipermail/cryptography/2017-August/032640.html (worth a read for anyone with their heads in this space currently).

Tristan Cacqueray (tristan-cacqueray) wrote on 2017-11-03: #15

Samuel, would you mind formatting the patch you proposed in #12 according to https://security.openstack.org/#how-to-propose-and-review-a-security-patch

Gage Hugo (gagehugo) wrote on 2018-07-26: #16

pci-dss-rocky.patch (5.0 KiB, text/plain)

Formatted the change from comment #12. I had to make an adjustment however, as a "reason" field was added to the audit notifications since this was reported I believe, which caused the notification to send a failure with no reason (Unauthorized) rather than AccountLocked. Please take a look to make sure this is correct.

The evasive-mode lockout from http://www.metzdowd.com/pipermail/cryptography/2017-August/032640.html was an interesting idea, perhaps that could be investigated in the future.

Vil Surkin
Vinu Pillai
Vishakha Agarwal
Xiang Hui
Xiaojun Lin
Xin Zhong
Xingchao Yu
Yahoo! Engineerin...
Yongqiang Yang
Zahid Hasan
ZhangNi
Ziv
ammarun
anndy
armyman420
avinashsau
brightson
bugtracker@devshe...
chaiwat wannaposop
chitu
congge
devin.li
dominic_chen
ekotkaj
fei Yang
galeido
gsccc
iopenstack
jeff wang
joel BELAFA
kalim khuang
kgrvamsi
lanpi
laoyi
lei zhang
liaonanhai
lololmarwa255
lpmqtt
maestropandy
manish
mershard frierson
miralaunchpad
mohit.048
nawawit kes
raja
satyanarayana pat...
satyanarayana pat...
sivagnanam C
sunilcn
tangfeixiong
truijllo
vivek.ys
wanghuagong
xiaoningli
xreuze
yangbo
yangzhenyu
zhangqinta
zzfancy

Patches

1688137.diff
pci-dss-rocky.patch
pci-dss.patch
Add patch

Jeremy Stanley (fungi) wrote on 2020-02-27: #17

In keeping with recent OpenStack vulnerability management policy changes,
no report should remain under private embargo for more than 90 days.
Because this report predates the change in policy, the deadline for public
disclosure is being set to 90 days from today. If the report is not
resolved within the next 90 days, it will revert to our public workflow as
of 2020-05-27. Please see http://lists.openstack.org/pipermail/openstack-
discuss/2020-February/012721.html for further details.

description:updated

Colleen Murphy (krinkle) wrote on 2020-04-27: #18

This seems to still be valid but the proposed patch doesn't apply any
more, can someone update the patch and can we move forward on this before
this is disclosed?

Gage Hugo (gagehugo) wrote on 2020-04-27: #19

pci-dss.patch (5.1 KiB, text/plain)

Update patch against master.

Colleen Murphy (krinkle) wrote on 2020-04-28: #20

Patch in comment 19 lgtm

Jeremy Stanley (fungi) on 2020-05-19

description:updated

Jeremy Stanley (fungi) wrote on 2020-05-27: #21

The embargo for this report has expired and is now lifted, so it's
acceptable to discuss further in public.

description:updated
information type:Private Security → Public Security

Jeremy Stanley (fungi) on 2020-10-23

summary:- Attacker may use PCI-DSS 8.1.6 and 8.1.7 to lock out users indefinitely
+ PCI-DSS account lock out DoS and account UUID lookup oracle

Jeremy Stanley (fungi) wrote on 2020-10-23: **Re: PCI-DSS account lock out DoS and account UUID lookup oracle** #22

So just to summarize, this report covers three possible vulnerabilities
related to the PCI-DSS account lock out feature:

1. If someone can guess a username they can prevent that user from
authenticating by repeatedly attempting to log in with an incorrect
credential.

2. Someone can identify valid usernames by trying to log in with candidate
strings with invalid passwords until the lock out is reached, at which
point the change in API response confirms the existence of that user.

3. The lock out response can be used as an oracle to determine the UUID
matching any known or guessed username.

OpenStack Infra (hudson-openstack) wrote on 2020-10-27: **Related fix proposed to keystone (master)** #23

Related fix proposed to branch: master
Review: https://review.opendev.org/759940

OpenStack Infra (hudson-openstack) wrote on 2021-05-06: **Related fix merged to keystone (master)** #24

Reviewed: https://review.opendev.org/c/openstack/keystone/+/759940
Committed: https://opendev.org/openstack/keystone/commit/ac2631ae3344587
7094cdae796fbcdce8833a626
Submitter: "Zuul (22348)"
Branch: master

commit ac2631ae33445877094cdae796fbcdce8833a626
Author: Gage Hugo <email address hidden>
Date: Tue Oct 27 15:22:04 2020 -0500

Hide AccountLocked exception from end users

This change hides the AccountLocked exception from being returned
to the end user to hide sensitive information that a potential
malicious person could gain insight from.

The notification handler catches the AccountLocked exception as
before, but after sending the audit notification, it instead
bubbles up Unauthorized rather than AccountLocked.

Co-Authored-By: Samuel de Medeiros Queiroz <email address hidden>

Change-Id: Id51241989b22c52810391f3e8e1cadbf8613d873
Related-Bug: #1688137

Jeremy Stanley (fungi) wrote on 2021-05-10: **Re: PCI-DSS account lock out DoS and account UUID lookup oracle** #25

It looks like the change which merged to master last week addresses
potential vulnerabilities #2 and #3 from comment #22. Is there any chance
for that to be backported to supported stable branches?

As for potential vulnerability #1, I don't really see a viable way to
address that, it's the intent of the feature that too many failed logins

lock the account. If a deployment considers that feature problematic, they
should disable it.

OpenStack Infra (hudson-openstack) wrote on 2021-05-10: **Related fix proposed to keystone (stable/wallaby)** #26

Related fix proposed to branch: stable/wallaby
Review: https://review.opendev.org/c/openstack/keystone/+/790440

OpenStack Infra (hudson-openstack) wrote on 2021-05-10: **Related fix proposed to keystone (stable/victoria)** #27

Related fix proposed to branch: stable/victoria
Review: https://review.opendev.org/c/openstack/keystone/+/790442

OpenStack Infra (hudson-openstack) wrote on 2021-05-10: **Related fix proposed to keystone (stable/ussuri)** #28

Related fix proposed to branch: stable/ussuri
Review: https://review.opendev.org/c/openstack/keystone/+/790443

OpenStack Infra (hudson-openstack) wrote on 2021-05-10: **Related fix proposed to keystone (stable/train)** #29

Related fix proposed to branch: stable/train
Review: https://review.opendev.org/c/openstack/keystone/+/790444

Gage Hugo (gagehugo) wrote on 2021-05-10: **Re: PCI-DSS account lock out DoS and account UUID lookup oracle** #30

Backports:

W - https://review.opendev.org/c/openstack/keystone/+/790440
V - https://review.opendev.org/c/openstack/keystone/+/790442
U - https://review.opendev.org/c/openstack/keystone/+/790443
T - https://review.opendev.org/c/openstack/keystone/+/790444

OpenStack Infra (hudson-openstack) wrote on 2021-06-03: **Related fix merged to keystone (stable/wallaby)** #31

Reviewed: https://review.opendev.org/c/openstack/keystone/+/790440
Committed: https://opendev.org/openstack/keystone/commit/f510c806de3e20c
dedd55291cd58dafa59398bec
Submitter: "Zuul (22348)"
Branch: stable/wallaby

commit f510c806de3e20cdedd55291cd58dafa59398bec
Author: Gage Hugo <email address hidden>
Date: Tue Oct 27 15:22:04 2020 -0500

    Hide AccountLocked exception from end users

    This change hides the AccountLocked exception from being returned
    to the end user to hide sensitive information that a potential
    malicious person could gain insight from.

    The notification handler catches the AccountLocked exception as
    before, but after sending the audit notification, it instead
    bubbles up Unauthorized rather than AccountLocked.

    Co-Authored-By: Samuel de Medeiros Queiroz <email address hidden>

    Change-Id: Id51241989b22c52810391f3e8e1cadbf8613d873
    Related-Bug: #1688137
    (cherry picked from commit ac2631ae33445877094cdae796fbcdce8833a626)

tags:added: in-stable-wallaby

OpenStack Infra (hudson-openstack) wrote on 2021-06-03: **Related fix merged to keystone (stable/victoria)** #32

Reviewed: https://review.opendev.org/c/openstack/keystone/+/790442
Committed: https://opendev.org/openstack/keystone/commit/4649fe6bfc749ab
48ec1905ca4dc2fc667914021
Submitter: "Zuul (22348)"
Branch: stable/victoria

commit 4649fe6bfc749ab48ec1905ca4dc2fc667914021
Author: Gage Hugo <email address hidden>
Date: Tue Oct 27 15:22:04 2020 -0500

    Hide AccountLocked exception from end users

    This change hides the AccountLocked exception from being returned
    to the end user to hide sensitive information that a potential
    malicious person could gain insight from.

    The notification handler catches the AccountLocked exception as
    before, but after sending the audit notification, it instead
    bubbles up Unauthorized rather than AccountLocked.

    Co-Authored-By: Samuel de Medeiros Queiroz <email address hidden>

    Change-Id: Id51241989b22c52810391f3e8e1cadbf8613d873
    Related-Bug: #1688137
    (cherry picked from commit ac2631ae33445877094cdae796fbcdce8833a626)

tags:added: in-stable-victoria

OpenStack Infra (hudson-openstack) wrote on 2021-06-04: **Related fix merged to keystone (stable/ussuri)** #33

Reviewed: https://review.opendev.org/c/openstack/keystone/+/790443
Committed: https://opendev.org/openstack/keystone/commit/8ab4eb27be4c13c
9bab2b3ea700f00a190521bf8
Submitter: "Zuul (22348)"
Branch: stable/ussuri

commit 8ab4eb27be4c13c9bab2b3ea700f00a190521bf8
Author: Gage Hugo <email address hidden>
Date: Tue Oct 27 15:22:04 2020 -0500

    Hide AccountLocked exception from end users

    This change hides the AccountLocked exception from being returned
    to the end user to hide sensitive information that a potential
    malicious person could gain insight from.

The notification handler catches the AccountLocked exception as
before, but after sending the audit notification, it instead
bubbles up Unauthorized rather than AccountLocked.

Co-Authored-By: Samuel de Medeiros Queiroz <email address hidden>

Change-Id: Id51241989b22c52810391f3e8e1cadbf8613d873
Related-Bug: #1688137
(cherry picked from commit ac2631ae33445877094cdae796fbcdce8833a626)

tags:added: in-stable-ussuri

OpenStack Infra (hudson-openstack) wrote on 2021-06-04: **Related fix merged to keystone (stable/train)** #34

Reviewed: https://review.opendev.org/c/openstack/keystone/+/790444
Committed: https://opendev.org/openstack/keystone/commit/1b573ae7d1c20e0ebfbde79bbe7538a09589c75d
Submitter: "Zuul (22348)"
Branch: stable/train

commit 1b573ae7d1c20e0ebfbde79bbe7538a09589c75d
Author: Gage Hugo <email address hidden>
Date: Tue Oct 27 15:22:04 2020 -0500

Hide AccountLocked exception from end users

This change hides the AccountLocked exception from being returned
to the end user to hide sensitive information that a potential
malicious person could gain insight from.

The notification handler catches the AccountLocked exception as
before, but after sending the audit notification, it instead
bubbles up Unauthorized rather than AccountLocked.

Co-Authored-By: Samuel de Medeiros Queiroz <email address hidden>

Change-Id: Id51241989b22c52810391f3e8e1cadbf8613d873
Related-Bug: #1688137
(cherry picked from commit ac2631ae33445877094cdae796fbcdce8833a626)

tags:added: in-stable-train

Jeremy Stanley (fungi) wrote on 2021-07-09: **Re: PCI-DSS account lock out DoS and account UUID lookup oracle** #35

A fix for the account name and UUID oracles has merged with backports
applied as far back as stable/train (so definitely covering all officially
maintained branches at this point). We should probably issue a security
advisory covering these points.

However, the other concern raised in this report is essentially with the
intent of PCI-DSS controls 8.1.6 and 8.1.7, which I think should not be
treated as a bug (if you don't want someone to be able to lock our another
user's account by repeatedly failing to log into it, don't enable that
feature in Keystone). I think the bug report should be retitled to focus
on the oracles, which were certainly unintended behaviors detrimental to
account security.

OpenStack Infra (hudson-openstack) wrote on 2021-08-05: **Fix proposed to ossa (master)** #36

Fix proposed to branch: master
Review: https://review.opendev.org/c/openstack/ossa/+/803640

Changed in ossa:
status:Incomplete → In Progress

Jeremy Stanley (fungi) wrote on 2021-08-05: **Re: PCI-DSS account lock out DoS and account UUID lookup oracle** #37

Please review the proposed security advisory linked above and let me know
if it correctly captures the fixed vulnerabilities.

Changed in ossa:
importance:Undecided → Medium
assignee:nobody → Jeremy Stanley (fungi)

Jeremy Stanley (fungi) on 2021-08-06

summary:- PCI-DSS account lock out DoS and account UUID lookup oracle
+ Account name and UUID oracles in account locking (CVE-2021-38155)

Jeremy Stanley (fungi) wrote on 2021-08-09: **Re: Account name and UUID oracles in account locking (CVE-2021-38155)** #38

Last call for reviews on the proposed https://review.opendev.org/803640
advisory content, I'm planning to approve and send copies to mailing lists
around this time tomorrow. Thanks!

Jeremy Stanley (fungi) on 2021-08-10

summary:- Account name and UUID oracles in account locking (CVE-2021-38155)
+ [OSSA-2021-003] Account name and UUID oracles in account locking
+ (CVE-2021-38155)

OpenStack Infra (hudson-openstack) wrote on 2021-08-10: **Fix merged to ossa (master)** #39

Reviewed: https://review.opendev.org/c/openstack/ossa/+/803640
Committed: https://opendev.org/openstack/ossa/commit/cf49e91bb4a6a663b960d65f87841f9ba589a8e4
Submitter: "Zuul (22348)"
Branch: master

commit cf49e91bb4a6a663b960d65f87841f9ba589a8e4
Author: Jeremy Stanley <email address hidden>
Date: Thu Aug 5 18:25:32 2021 +0000

Add OSSA-2021-003 (CVE-2021-38155)

Change-Id: Ic9c5d7a45be8a083931b2600adbc76c9e292d0ab
Closes-Bug: #1688137

Changed in ossa:
status:In Progress → Fix Released

Jeremy Stanley (fungi) wrote on 2021-08-10: #40

This advisory has been delivered to the usual mailing lists.

OpenStack Infra (hudson-openstack) wrote on 2021-08-16: **Related fix proposed to keystone (stable/stein)** #41

Related fix proposed to branch: stable/stein
Review: https://review.opendev.org/c/openstack/keystone/+/804718

OpenStack Infra (hudson-openstack) wrote on 2021-08-16: **Related fix proposed to keystone (stable/rocky)** #42

Related fix proposed to branch: stable/rocky
Review: https://review.opendev.org/c/openstack/keystone/+/804719

Jacolex (jacolex) wrote on 2022-03-11 (last edit on 2022-03-11): #43

I spent two days examining source code, why users are not receiving
explanation about locking account and finally I found this thread. I'm
operating various systems (AD, linuxes, LDAPs), where users have feedback
from system, why the logon not works in case of failure. I can't see
proper explenation why keystone developers loose balance between usability
and security. Of course account locked information is some kind of leak of
information about account names, but this is overkill for authentication
usability. If something wrong is happening with locking acocunts, the
administrator should take necessary steps to investigate and prevent the
attack. The logs should be analyzed continously to prevent the attacks,
but HOW TO DO IT IF THERE ARE NO LOGS!!!
Please consider once again such useless security standards. There is no
security if administrator and user has no information about logon
failures! After spending a lot of time on those problems I realized that I
have no tools to monitor failure logons and locking accounts. Even no
account names appearing keystone in logs. This is not what administrator
is expecting from secure authentication system!

My comments to the argumentation above:

>So just to summarize, this report covers three possible vulnerabilities
related to the PCI-DSS account lock out feature:

>1. If someone can guess a username they can prevent that user from
authenticating by repeatedly attempting to log in with an >incorrect
credential.

Yes this is kind of problem, but the attacker can guess the user name in
other ways and making such attack. In such cases the user should reported
abuse and then the openstack operator should prevent such attacks by
blocking IP address or changing account name. These issues shouldn't be
handled automatically but obscurity, but every case should be investigated
and prevented.

>2. Someone can identify valid usernames by trying to log in with
candidate strings with invalid passwords until the lock out is >reached,
at which point the change in API response confirms the existence of that
user.

> 3. The lock out response can be used as an oracle to determine the UUID
matching any known or guessed username.

These problems occur on every authentication systems. But lockout
threshold should prevent such attacks using for example throttling
policies. Another way: log lockout events to keystone log only (now there
are no information, even when insecure_debug is on).

I think that administrator should have possibility to choose the security
level of keystone authentication process, depending on company needs and
company security policy.

David Wilde (dave-wilde) on 2022-03-11

Changed in keystone:
assignee:nobody → David Wilde (dave-wilde)

Douglas Mendizábal (dougmendizabal) wrote on 2022-03-11: #44

I think it makes sense to provide more logging around this. Dave will take
this bug and work on a patch.

OpenStack Infra (hudson-openstack) wrote on 2022-07-15: **Change abandoned on keystone (stable/stein)** #45

Change abandoned by "Douglas Mendizábal <email address hidden>" on branch:
stable/stein
Review: https://review.opendev.org/c/openstack/keystone/+/804718
Reason: Abandoning unmerged stable/stein changes.

OpenStack Infra (hudson-openstack) wrote on 2022-07-15: **Change abandoned on keystone (stable/rocky)** #46

Change abandoned by "Douglas Mendizábal <email address hidden>" on branch:
stable/rocky
Review: https://review.opendev.org/c/openstack/keystone/+/804719
Reason: Abandoning unmerged stable/rocky changes.

See full activity log

To post a comment you must log in.