

New issue

[Jump to bottom](#)

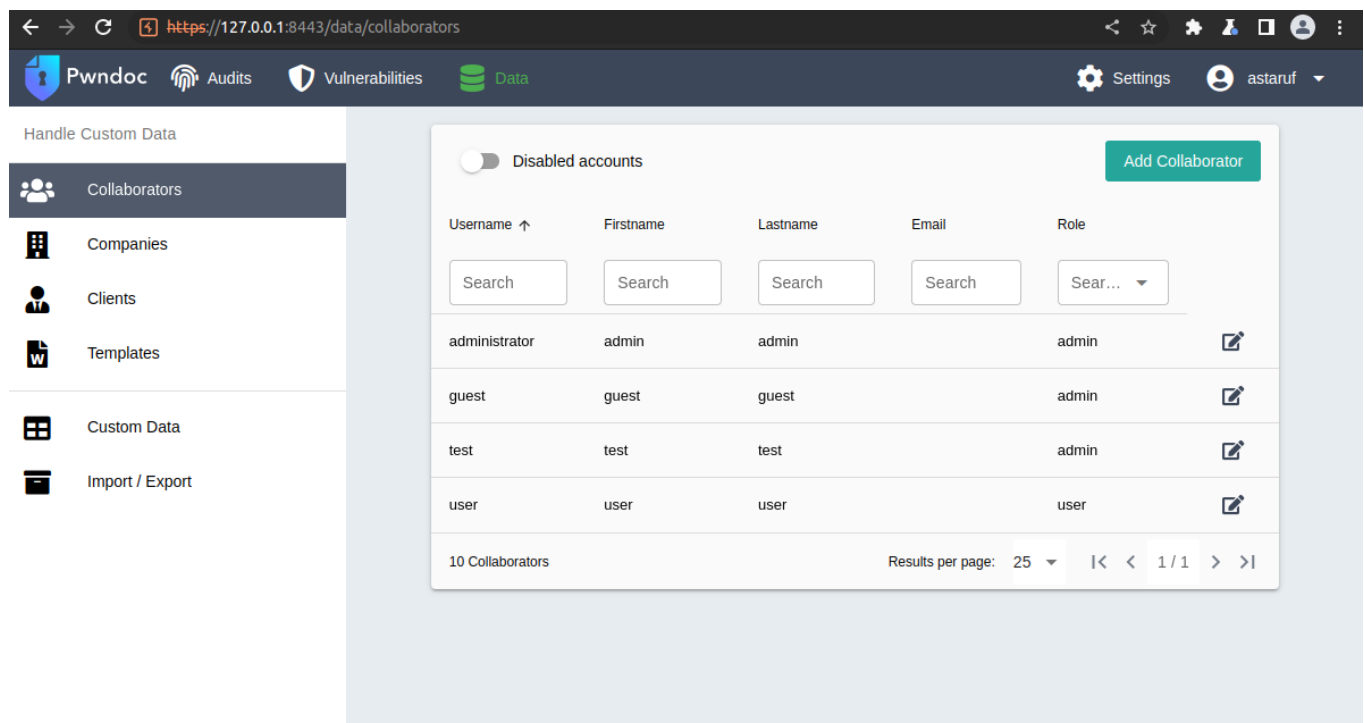
## (Vulnerability) Disabled user account enumeration via different responses #382

Open Astaruf opened this issue on Oct 24 · 1 comment

Astaruf commented on Oct 24 • edited

It is possible to enumerate "disabled account" usernames in PwnDoc (tested on 0.5.3 - 2022-07-19) observing the web server responses to login requests.

For example, let's suppose these users were registered on PwnDoc and then disabled:



Handle Custom Data

- Collaborators
- Companies
- Clients
- Templates
- Custom Data
- Import / Export

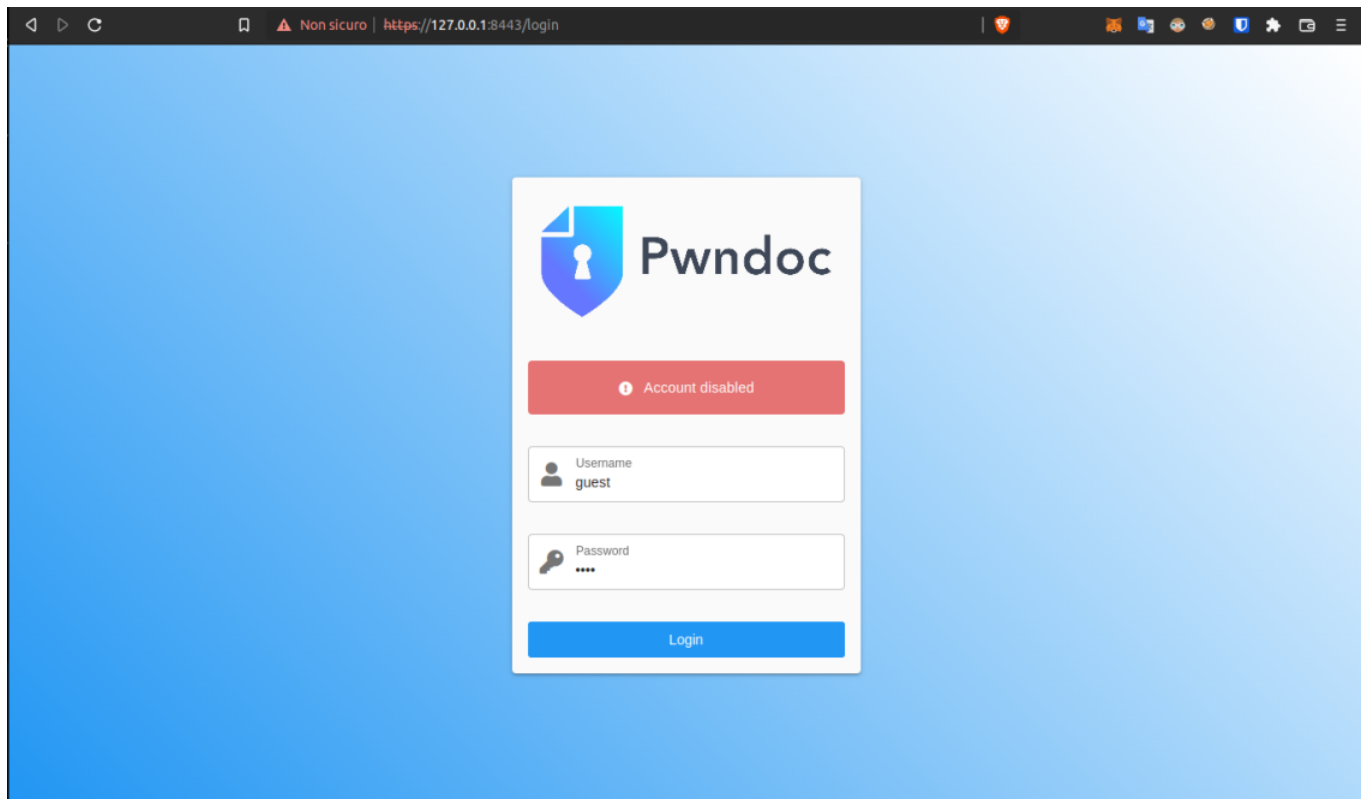
Disabled accounts

Add Collaborator

Username ↑	Firstname	Lastname	Email	Role
Search	Search	Search	Search	Sear... ▾
administrator	admin	admin		admin
guest	guest	guest		admin
test	test	test		admin
user	user	user		user

10 Collaborators Results per page: 25 ▾ |< < 1 / 1 > >|

Trying to log in with one of these disabled users in fact the application responds with the message "Account disabled".



Client request and server response:

Request		Response	
Pretty	Raw	Pretty	Raw
<pre>1 POST /api/users/token HTTP/1.1 2 Host: 127.0.0.1:8443 3 Content-Length: 52 4 Sec-Ch-Ua: "Not;A=Brand";v="99", "Chromium";v="106" 5 Accept: application/json, text/plain, */* 6 Content-Type: application/json; charset=UTF-8 7 Sec-Ch-Ua-Mobile: ?0 8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)   AppleWebKit/537.36 (KHTML, like Gecko) Chrome/106.0.5249.62   Safari/537.36 9 Sec-Ch-Ua-Platform: "Linux" 10 Origin: https://127.0.0.1:8443 11 Sec-Fetch-Site: same-origin 12 Sec-Fetch-Mode: cors 13 Sec-Fetch-Dest: empty 14 Referer: https://127.0.0.1:8443/login 15 Accept-Encoding: gzip, deflate 16 Accept-Language: it-IT,it;q=0.9,en-US;q=0.8,en;q=0.7 17 Connection: close 18 19 {   "username": "guest",   "password": "asd",   "totpToken": "" }</pre>		<pre>1 HTTP/1.1 401 Unauthorized 2 Server: nginx/1.22.0 3 Date: Thu, 06 Oct 2022 21:14:50 GMT 4 Content-Type: application/json; charset=utf-8 5 Content-Length: 45 6 Connection: close 7 X-Powered-By: Express 8 Access-Control-Allow-Methods: GET,POST,DELETE,PUT,OPTIONS 9 Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,   Accept 10 Access-Control-Expose-Headers: Content-Disposition 11 Etag: W/"2d-db7A6m1NjDBPAJcJH+rLevoFsg" 12 13 {   "status": "error",   "datas": "Account disabled" }</pre>	

Trying to log in with a user who does not exist, the application responds with "Invalid credentials":

Request		Response	
Pretty	Raw	Pretty	Raw
<pre> 1 POST /api/users/token HTTP/1.1 2 Host: 127.0.0.1:8443 3 Content-Length: 54 4 Sec-Ch-Ua: "Not;A=Brand";v="99", "Chromium";v="106" 5 Accept: application/json, text/plain, */* 6 Content-Type: application/json;charset=UTF-8 7 Sec-Ch-Ua-Mobile: ?0 8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64)   AppleWebKit/537.36 (KHTML, like Gecko) Chrome/106.0.5249.62   Safari/537.36 9 Sec-Ch-Ua-Platform: "Linux" 10 Origin: https://127.0.0.1:8443 11 Sec-Fetch-Site: same-origin 12 Sec-Fetch-Mode: cors 13 Sec-Fetch-Dest: empty 14 Referer: https://127.0.0.1:8443/login 15 Accept-Encoding: gzip, deflate 16 Accept-Language: it-IT,it;q=0.9,en-US;q=0.8,en;q=0.7 17 Connection: close 18 19 {   "username":"astaruf",   "password":"asd",   "totpToken":"" } </pre>		<pre> 1 HTTP/1.1 401 Unauthorized 2 Server: nginx/1.22.0 3 Date: Thu, 06 Oct 2022 21:14:15 GMT 4 Content-Type: application/json; charset=utf-8 5 Content-Length: 48 6 Connection: close 7 X-Powered-By: Express 8 Access-Control-Allow-Methods: GET,POST,DELETE,PUT,OPTIONS 9 Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type,   Accept 10 Access-Control-Expose-Headers: Content-Disposition 11 ETag: W/"30-cK6GtT3+HJH6fNzEwf5Ds7FDTBM" 12 13 {   "status":"error",   "datas":"Invalid credentials" } </pre>	
<div> <div>0 matches</div> <div>Search...</div> </div>		<div> <div>0 matches</div> <div>Search...</div> </div>	

This server behavior can be exploited to enumerate disabled users on the platform, who may be re-enabled by an admin and used again in the future.

By performing a brute force dictionary attack, a defined list of users can be provided via login POST request to detect all the "Account disabled" server's responses and exclude the "Invalid credentials" ones.

Attack Save Columns							
Results Positions Payloads Resource Pool Options							
Filter: Showing all items							
Request	Payload	Status	Error	Timeout	Length	Account disabled ▾	Comment
3	test	401	<input type="checkbox"/>	<input type="checkbox"/>	473	1	
4	guest	401	<input type="checkbox"/>	<input type="checkbox"/>	473	1	
8	user	401	<input type="checkbox"/>	<input type="checkbox"/>	473	1	
9	administrator	401	<input type="checkbox"/>	<input type="checkbox"/>	473	1	
24	test	401	<input type="checkbox"/>	<input type="checkbox"/>	473	1	
0		401	<input type="checkbox"/>	<input type="checkbox"/>	476		
1	root	401	<input type="checkbox"/>	<input type="checkbox"/>	476		
2	admin	401	<input type="checkbox"/>	<input type="checkbox"/>	476		
5	info	401	<input type="checkbox"/>	<input type="checkbox"/>	476		
6	adm	401	<input type="checkbox"/>	<input type="checkbox"/>	476		
7	mysql	401	<input type="checkbox"/>	<input type="checkbox"/>	476		
10	oracle	401	<input type="checkbox"/>	<input type="checkbox"/>	476		
11	test	401	<input type="checkbox"/>	<input type="checkbox"/>	476		

Request	Response
Pretty	Raw Hex Render
	<pre> 1 HTTP/1.1 401 Unauthorized 2 Server: nginx/1.22.0 3 Date: Thu, 06 Oct 2022 21:17:52 GMT 4 Content-Type: application/json; charset=utf-8 5 Content-Length: 45 6 Connection: close 7 X-Powered-By: Express 8 Access-Control-Allow-Methods: GET,POST,DELETE,PUT,OPTIONS 9 Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept 10 Access-Control-Expose-Headers: Content-Disposition 11 ETag: W/"2d-db7A6milNjDBPAJcJH+rLevoFsg" 12 13 </pre>


0 matches

Finished

The standard recommendation to mitigate this vulnerability is to return identical responses for “valid user/wrong password” and “invalid user” login requests.

Let me know if I can help you in any way so, once fixed I would like to get a CVE from mtre.org

1

 **Astaruf** changed the title ~~Disabled user account enumeration via different responses (Vulnerability)~~ Disabled user account enumeration via different responses last month

Astaruf commented 26 days ago • edited ▾

Author

A CVE-ID has been reserved by Mitre.org for this vulnerability [CVE-2022-44023](#).

1

Assignees

No one assigned

---

Labels

None yet

---

Projects

None yet

---

Milestone

No milestone

---

Development

No branches or pull requests

---

1 participant

