FIREWALL:

- **Allow only admin IPs(let's say 192.168.1.3) to access SSH/RDP.**

```
iptables -A INPUT -p tcp -s 172.17.0.1 --dport 22 -j ACCEPT
iptables -A INPUT -p tcp --dport 22 -j DROP
iptables -A INPUT -p tcp -s 172.17.0.1 --dport 3389 -j ACCEPT
iptables -A INPUT -p tcp --dport 3389 -j DROP
```

-A INPUT → Appends the rule to the INPUT chain
-p tcp → Specifies the protocol as TCP.
-s 192.168.1.3 → Restricts the rule to traffic originating from IP 192.168.1.3.
--dport 22 → Applies the rule to port 22 (SSH).
-j ACCEPT → Allows the matching traffic

The given iptables rules restrict SSH (port 22) and RDP (port 3389) access to only the admin IP (172.17.0.1). The first rule allows SSH connections from 172.17.0.1, while the second rule blocks all other SSH access. Similarly, the third rule permits RDP connections from 172.17.0.1, and the fourth rule denies RDP access for all other IPs. However, if the intended admin IP is 192.168.1.3, you should replace 172.17.0.1 with 192.168.1.3.

- **Allow HTTP/HTTPS but block access from blacklisted IP range 103.25.231.0/24.**

```
iptables -A INPUT -s 103.25.231.0/24 -p tcp --dport 80 -j DROP
iptables -A INPUT -s 103.25.231.0/24 -p tcp --dport 443 -j DROP
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
```

These iptables rules block HTTP (port 80) and HTTPS (port 443) access from the blacklisted IP range 103.25.231.0/24 while allowing all other traffic. The DROP rules ensure that requests from the blacklisted range are denied before the general ACCEPT rules permit access for everyone else.

- **Allow only internal IPs to access the database server(consider default port for databases).**

```
iptables -A INPUT -p tcp -s 192.168.0.0/16 --dport 3306 -j ACCEPT
iptables -A INPUT -p tcp -s 192.168.0.0/16 --dport 5432 -j ACCEPT
iptables -A INPUT -p tcp -s 192.168.0.0/16 --dport 27017 -j ACCEPT
iptables -A INPUT -p tcp --dport 3306 -j DROP
iptables -A INPUT -p tcp --dport 5432 -j DROP
iptables -A INPUT -p tcp --dport 27017 -j DROP
```

These iptables rules ensure that only internal IPs (192.168.0.0/16) can access the database server while blocking external access. The first three rules allow incoming connections from the internal network to MySQL (3306), PostgreSQL (5432), and MongoDB (27017). The last three rules explicitly drop any traffic trying to reach these ports from outside the internal network, enhancing security by preventing unauthorized access.

- **Enable logging for debugging and tracking unauthorized access.**

```
sudo iptables -A INPUT -m limit --limit 10/second --limit-burst
100 -j LOG --log-prefix "[UNAUTHORIZED ACCESS] Iptables Drop: "
--log-level 4
```

This iptables rule enables logging for debugging and tracking unauthorized access attempts. It logs dropped packets with the prefix [UNAUTHORIZED ACCESS] Iptables Drop: at log level 4 (warning). The -m limit --limit 10/second --limit-burst 100 prevents log flooding by restricting logs to a maximum of 10 entries per second with an initial burst of 100, ensuring efficient monitoring without overwhelming the system logs.

- **Configure rate limiting to prevent excessive HTTP requests from a single IP.**

  ```
  sudo iptables -A INPUT -p tcp --dport 80 -m conntrack
  --ctstate NEW -m
  hashlimit \
  --hashlimit-name http_limit \
  --hashlimit-above 5/second \
  --hashlimit-mode srcip \
  --hashlimit-burst 10 \
  --hashlimit-htable-expire 60000 \
  -j DROP
  ```

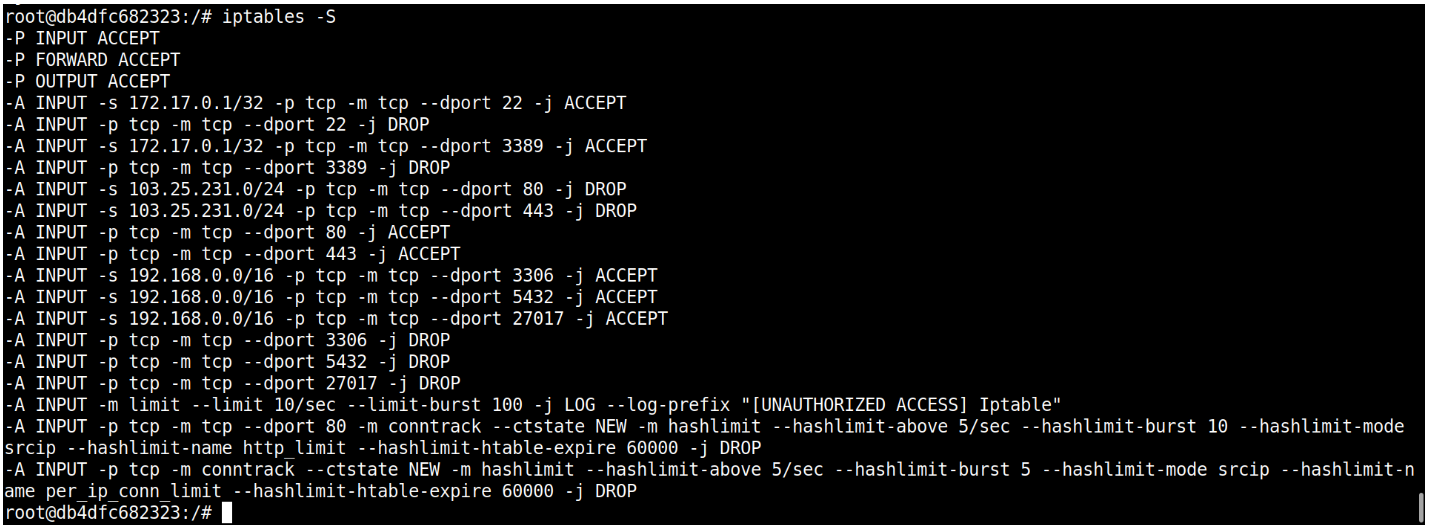
This iptables rule implements rate limiting to prevent excessive HTTP requests from a single IP. It uses the hashlimit module to restrict new connections to port 80 (HTTP) to a maximum of 5 requests per second per IP (--hashlimit-above 5/second). A burst of up to 10 requests is allowed before throttling (--hashlimit-burst 10), and inactive entries expire after 60 seconds (--hashlimit-htable-expire 60000). If an IP exceeds the limit, further requests are dropped, protecting against DoS attacks and abusive traffic.

- **Set a limit of 5 connections per second per IP to prevent overloading.**

  ```
  sudo iptables -A INPUT -p tcp -m conntrack --ctstate NEW -m
  hashlimit \
  --hashlimit-name per_ip_conn_limit \
  --hashlimit-above 5/second \
  --hashlimit-mode srcip \
  --hashlimit-burst 5 \
  --hashlimit-htable-expire 60000 \
  -j DROP
  ```

This iptables rule prevents server overload by limiting new TCP connections to 5 per second per IP. The hashlimit module tracks connections and drops requests exceeding 5second (--hashlimit-above 5/second). A burst of up to 5 connections is allowed before enforcement (--hashlimit-burst 5), and inactive IP entries expire after 60 seconds (--hashlimit-htable-expire 60000). This helps mitigate abuse, brute-force attacks, and excessive connection attempts.

Firewall rules added:



```
root@db4dfc682323:/# iptables -S
-P INPUT ACCEPT
-P FORWARD ACCEPT
-P OUTPUT ACCEPT
-A INPUT -s 172.17.0.1/32 -p tcp -m tcp --dport 22 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 22 -j DROP
-A INPUT -s 172.17.0.1/32 -p tcp -m tcp --dport 3389 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 3389 -j DROP
-A INPUT -s 103.25.231.0/24 -p tcp -m tcp --dport 80 -j DROP
-A INPUT -s 103.25.231.0/24 -p tcp -m tcp --dport 443 -j DROP
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
-A INPUT -s 192.168.0.0/16 -p tcp -m tcp --dport 3306 -j ACCEPT
-A INPUT -s 192.168.0.0/16 -p tcp -m tcp --dport 5432 -j ACCEPT
-A INPUT -s 192.168.0.0/16 -p tcp -m tcp --dport 27017 -j ACCEPT
-A INPUT -p tcp -m tcp --dport 3306 -j DROP
-A INPUT -p tcp -m tcp --dport 5432 -j DROP
-A INPUT -p tcp -m tcp --dport 27017 -j DROP
-A INPUT -m limit --limit 10/sec --limit-burst 100 -j LOG --log-prefix "[UNAUTHORIZED ACCESS] Iptable"
-A INPUT -p tcp -m tcp --dport 80 -m conntrack --ctstate NEW -m hashlimit --hashlimit-above 5/sec --hashlimit-burst 10 --hashlimit-mode
srcip --hashlimit-name http_limit --hashlimit-htable-expire 60000 -j DROP
-A INPUT -p tcp -m conntrack --ctstate NEW -m hashlimit --hashlimit-above 5/sec --hashlimit-burst 5 --hashlimit-mode srcip --hashlimit-n
ame per_ip_conn_limit --hashlimit-htable-expire 60000 -j DROP
root@db4dfc682323:/#
```