25 Most Frequently Used Linux IPTables Rules Examples

by Ramesh Natarajan on June 14, 2011

At a first glance, IPTables rules might look cryptic.

In this article, I've given 25 practical IPTables rules that you can copy/paste and use it for your needs.

These examples will act as a basic templates for you to tweak these rules to suite your specific requirement.

For easy reference, all these 25 iptables rules are in shell script format: iptables-rules

1. Delete Existing Rules

Before you start building new set of rules, you might want to clean-up all the default rules, and existing rules. Use the iptables flush command as shown below to do this.

```
iptables -F
(or)
iptables --flush
```

2. Set Default Chain Policies

The default chain policy is ACCEPT. Change this to DROP for all INPUT, FORWARD, and OUTPUT chains as shown below.

```
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT DROP
```

When you make both INPUT, and OUTPUT chain's default policy as DROP, for every firewall rule requirement you have, you should define two rules. i.e one for incoming and one for outgoing.

In all our examples below, we have two rules for each scenario, as we've set DROP as default policy for both INPUT and OUTPUT chain.

If you trust your internal users, you can omit the last line above. i.e Do not DROP all outgoing packets by default. In that case, for every firewall rule requirement you have, you just have to define only one rule. i.e define rule only for incoming, as the outgoing is ACCEPT for all packets.

Note: If you don't know what a chain means, you should first familiarize yourself with the IPTables fundamentals.

3. Block a Specific ip-address

Before we proceed further will other examples, if you want to block a specific ip-address, you should do that first as shown below. Change the "x.x.x.x" in the following example to the specific ip-address that you like to block.

```
BLOCK_THIS_IP="x.x.x.x"
iptables -A INPUT -s "$BLOCK_THIS_IP" -j DROP
```

This is helpful when you find some strange activities from a specific ip-address in your log files, and you want to temporarily block that ip-address while you do further research.

You can also use one of the following variations, which blocks only TCP traffic on eth0 connection for this ip-address.

```
iptables -A INPUT -i eth0 -s "$BLOCK_THIS_IP" -j DROP
iptables -A INPUT -i eth0 -p tcp -s "$BLOCK_THIS_IP" -j DROP
```

4. Allow ALL Incoming SSH

The following rules allow ALL incoming ssh connections on eth0 interface.

```
iptables -A INPUT -i eth0 -p tcp --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -o eth0 -p tcp --sport 22 -m state --state ESTABLISHED -j ACCEPT
```

Note: If you like to understand exactly what each and every one of the arguments means, you should read How to Add IPTables Firewall Rules

5. Allow Incoming SSH only from a Sepcific Network

The following rules allow incoming ssh connections only from 192.168.100.X network.

```
iptables -A INPUT -i eth0 -p tcp -s 192.168.100.0/24 --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -o eth0 -p tcp --sport 22 -m state --state ESTABLISHED -j ACCEPT
```

In the above example, instead of /24, you can also use the full subnet mask. i.e “192.168.100.0/255.255.255.0”.

6. Allow Incoming HTTP and HTTPS

The following rules allow all incoming web traffic. i.e HTTP traffic to port 80.

```
iptables -A INPUT -i eth0 -p tcp --dport 80 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -o eth0 -p tcp --sport 80 -m state --state ESTABLISHED -j ACCEPT
```

The following rules allow all incoming secure web traffic. i.e HTTPS traffic to port 443.

```
iptables -A INPUT -i eth0 -p tcp --dport 443 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -o eth0 -p tcp --sport 443 -m state --state ESTABLISHED -j ACCEPT
```

7. Combine Multiple Rules Together using MultiPorts

When you are allowing incoming connections from outside world to multiple ports, instead of writing individual rules for each and every port, you can combine them together using the multiport extension as shown below.

The following example allows all incoming SSH, HTTP and HTTPS traffic.

```
iptables -A INPUT -i eth0 -p tcp -m multiport --dports 22,80,443 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -o eth0 -p tcp -m multiport --sports 22,80,443 -m state --state ESTABLISHED -j ACCEPT
```

8. Allow Outgoing SSH

The following rules allow outgoing ssh connection. i.e When you ssh from inside to an outside server.

```
iptables -A OUTPUT -o eth0 -p tcp --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A INPUT -i eth0 -p tcp --sport 22 -m state --state ESTABLISHED -j ACCEPT
```

Please note that this is slightly different than the incoming rule. i.e We allow both the NEW and ESTABLISHED state on the OUTPUT chain, and only ESTABLISHED state on the INPUT chain. For the incoming rule, it is vice versa.

9. Allow Outgoing SSH only to a Specific Network

The following rules allow outgoing ssh connection only to a specific network. i.e You an ssh only to 192.168.100.0/24 network from the inside.

```
iptables -A OUTPUT -o eth0 -p tcp -d 192.168.100.0/24 --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A INPUT -i eth0 -p tcp --sport 22 -m state --state ESTABLISHED -j ACCEPT
```

10. Allow Outgoing HTTPS

The following rules allow outgoing secure web traffic. This is helpful when you want to allow internet traffic for your users. On servers, these rules are also helpful when you want to use wget to download some files from outside.

```
iptables -A OUTPUT -o eth0 -p tcp --dport 443 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A INPUT -i eth0 -p tcp --sport 443 -m state --state ESTABLISHED -j ACCEPT
```

Note: For outgoing HTTP web traffic, add two additional rules like the above, and change 443 to 80.

11. Load Balance Incoming Web Traffic

You can also load balance your incoming web traffic using iptables firewall rules.

This uses the iptables nth extension. The following example load balances the HTTPS traffic to three different ip-address. For every 3th packet, it is load balanced to the appropriate server (using the counter 0).

```
iptables -A PREROUTING -i eth0 -p tcp --dport 443 -m state --state NEW -m nth --counter 0 --every 3 --packet 0 -j DNAT --to-destination 192.168.1.101:443
iptables -A PREROUTING -i eth0 -p tcp --dport 443 -m state --state NEW -m nth --counter 0 --every 3 --packet 1 -j DNAT --to-destination 192.168.1.102:443
iptables -A PREROUTING -i eth0 -p tcp --dport 443 -m state --state NEW -m nth --counter 0 --every 3 --packet 2 -j DNAT --to-destination 192.168.1.103:443
```

12. Allow Ping from Outside to Inside

The following rules allow outside users to be able to ping your servers.

```
iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT
iptables -A OUTPUT -p icmp --icmp-type echo-reply -j ACCEPT
```

13. Allow Ping from Inside to Outside

The following rules allow you to ping from inside to any of the outside servers.

```
iptables -A OUTPUT -p icmp --icmp-type echo-request -j ACCEPT
iptables -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT
```

14. Allow Loopback Access

You should allow full loopback access on your servers. i.e access using 127.0.0.1

```
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT
```

15. Allow Internal Network to External network.

On the firewall server where one ethernet card is connected to the external, and another ethernet card connected to the internal servers, use the following rules to allow internal network talk to external network.

In this example, eth1 is connected to external network (internet), and eth0 is connected to internal network (For example: 192.168.1.x).

```
iptables -A FORWARD -i eth0 -o eth1 -j ACCEPT
```

16. Allow outbound DNS

The following rules allow outgoing DNS connections.

```
iptables -A OUTPUT -p udp -o eth0 --dport 53 -j ACCEPT
iptables -A INPUT -p udp -i eth0 --sport 53 -j ACCEPT
```

17. Allow NIS Connections

If you are running NIS to manage your user accounts, you should allow the NIS connections. Even when the SSH connection is allowed, if you don't allow the NIS related ypbind connections, users will not be able to login.

The NIS ports are dynamic. i.e When the ypbind starts it allocates the ports.

First do a rpcinfo -p as shown below and get the port numbers. In this example, it was using port 853 and 850.

```
rpcinfo -p | grep ypbind
```

Now allow incoming connection to the port 111, and the ports that were used by ypbind.

```
iptables -A INPUT -p tcp --dport 111 -j ACCEPT
iptables -A INPUT -p udp --dport 111 -j ACCEPT
iptables -A INPUT -p tcp --dport 853 -j ACCEPT
iptables -A INPUT -p udp --dport 853 -j ACCEPT
iptables -A INPUT -p tcp --dport 850 -j ACCEPT
iptables -A INPUT -p udp --dport 850 -j ACCEPT
```

The above will not work when you restart the ypbind, as it will have different port numbers that time.

There are two solutions to this: 1) Use static ip-address for your NIS, or 2) Use some clever shell scripting techniques to automatically grab the dynamic port number from the "rpcinfo -p" command output, and use those in the above iptables rules.

18. Allow Rsync From a Specific Network

The following rules allows rsync only from a specific network.

```
iptables -A INPUT -i eth0 -p tcp -s 192.168.101.0/24 --dport 873 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -o eth0 -p tcp --sport 873 -m state --state ESTABLISHED -j ACCEPT
```

19. Allow MySQL connection only from a specific network

If you are running MySQL, typically you don't want to allow direct connection from outside. In most cases, you might have web server running on the same server where the MySQL database runs.

However DBA and developers might need to login directly to the MySQL from their laptop and desktop using MySQL client. In those case, you might want to allow your internal network to talk to the MySQL directly as shown below.

```
iptables -A INPUT -i eth0 -p tcp -s 192.168.100.0/24 --dport 3306 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -o eth0 -p tcp --sport 3306 -m state --state ESTABLISHED -j ACCEPT
```

20. Allow Sendmail or Postfix Traffic

The following rules allow mail traffic. It may be sendmail or postfix.

```
iptables -A INPUT -i eth0 -p tcp --dport 25 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -o eth0 -p tcp --sport 25 -m state --state ESTABLISHED -j ACCEPT
```

21. Allow IMAP and IMAPS

The following rules allow IMAP/IMAP2 traffic.

```
iptables -A INPUT -i eth0 -p tcp --dport 143 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -o eth0 -p tcp --sport 143 -m state --state ESTABLISHED -j ACCEPT
```

The following rules allow IMAPS traffic.

```
iptables -A INPUT -i eth0 -p tcp --dport 993 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -o eth0 -p tcp --sport 993 -m state --state ESTABLISHED -j ACCEPT
```

22. Allow POP3 and POP3S

The following rules allow POP3 access.

```
iptables -A INPUT -i eth0 -p tcp --dport 110 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -o eth0 -p tcp --sport 110 -m state --state ESTABLISHED -j ACCEPT
```

The following rules allow POP3S access.

```
iptables -A INPUT -i eth0 -p tcp --dport 995 -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -o eth0 -p tcp --sport 995 -m state --state ESTABLISHED -j ACCEPT
```

23. Prevent DoS Attack

The following iptables rule will help you prevent the Denial of Service (DoS) attack on your webserver.

```
iptables -A INPUT -p tcp --dport 80 -m limit --limit 25/minute --limit-burst 100 -j ACCEPT
```

In the above example:

- -m limit: This uses the limit iptables extension
- –limit 25/minute: This limits only maximum of 25 connection per minute. Change this value based on your specific requirement
- –limit-burst 100: This value indicates that the limit/minute will be enforced only after the total number of connection have reached the limit-burst level.

24. Port Forwarding

The following example routes all traffic that comes to the port 442 to 22. This means that the incoming ssh connection can come from both port 22 and 422.

```
iptables -t nat -A PREROUTING -p tcp -d 192.168.102.37 --dport 422 -j DNAT --to 192.168.102.37:22
```

If you do the above, you also need to explicitly allow incoming connection on the port 422.

```
iptables -A INPUT -i eth0 -p tcp --dport 422 -m state --state NEW,ESTABLISHED -j ACCEPT
```

```
iptables -A OUTPUT -o eth0 -p tcp --sport 422 -m state --state ESTABLISHED -j ACCEPT
```

25. Log Dropped Packets

You might also want to log all the dropped packets. These rules should be at the bottom.

First, create a new chain called LOGGING.

```
iptables -N LOGGING
```

Next, make sure all the remaining incoming connections jump to the LOGGING chain as shown below.

```
iptables -A INPUT -j LOGGING
```

Next, log these packets by specifying a custom "log-prefix".

```
iptables -A LOGGING -m limit --limit 2/min -j LOG --log-prefix "IPTables Packet Dropped: " --log-level 7
```

Finally, drop these packets.

```
iptables -A LOGGING -j DROP
```

All of the above 25 iptables rules are in shell script format: iptables-rules

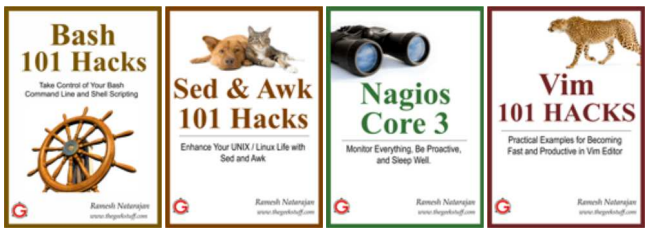
Previous articles in the iptables series:

- Linux Firewall Tutorial: IPTables Tables, Chains, Rules Fundamentals
- IPTables Flush: Delete / Remove All Rules On RedHat and CentOS Linux
- Linux IPTables: How to Add Firewall Rules (With Allow SSH Example)
- Linux IPTables: Incoming and Outgoing Rule Examples (SSH and HTTP)

If you enjoyed this article, you might also like..

1. 50 Linux Sysadmin Tutorials
2. 50 Most Frequently Used Linux Commands (With Examples)
3. Top 25 Best Linux Performance Monitoring and Debugging Tools
4. Mommy, I found it! – 15 Practical Linux Find Command Examples
5. Linux 101 Hacks 2nd Edition eBook Free

- Awk Introduction – 7 Awk Print Examples
- Advanced Sed Substitution Examples
- 8 Essential Vim Editor Navigation Fundamentals
- 25 Most Frequently Used Linux IPTables Rules Examples
- Turbocharge PuTTY with 12 Powerful Add-Ons



Tagged as: IPTables Block IP, IPTables Block IP Address, IPTables Block Port, IPTables DNAT, IPTables HowTo, IPTables Log, IPTables NAT, IPTables Tutorial, IPTables Ubuntu

{ 46 comments… add one }

- kgas June 14, 2011, 3:42 am

  Good one and expecting all your iptables related writings in a single pdf file. Thanks

  Link
- diptanu June 14, 2011, 3:57 am

  Hi,Thanks alot for the above info.However would like to know that if the blocking or allowing through iptables is possible for specific MAC address over internet, as because if my eth0 is using a local ip 10.10.10.10 which is natted via public ip eg 100.100.100.100 and connected to internet via ISP, then someone from internet with specific MAC id (allowed in iptables) should be able to ssh to my public ip (100.100.100.100) and the rest should be dropped.

  Is that possible over the internet

  Link
- Manoj June 14, 2011, 4:25 am

  Keep Writing 🙂 Gr8 Work!!!

  Link
- Gareth Williams June 14, 2011, 11:16 am

  Diptanu you cannot filter packets based on mac addresses. Even if that was possible that host is not on your subnet therefore you do not communicate to it or it to you using MAC address. When communicating to that host all traffic is routed via your default gateway.

  Link
- Marcin Rybak June 14, 2011, 2:26 pm

  First ruleset should go deep deep , if somebody has default policy to drop – first rule:
  iptables -F, will cut him out. You have been warned 🙂

  Link
- Kishore June 14, 2011, 2:59 pm

  Very Useful 🙂

Link

- Vonskippy June 14, 2011, 3:03 pm

  A default rule should always be block ALL outbound traffic on TCP25 except to your own email server.

  Link
- vierupro June 16, 2011, 3:07 am

  Good collection of iptables rules!
  Just wanted to mention that ” -m limit ” does match packets not connections, so in your example you will match 25 packets per minute, which I think is not what you want to.
  The solution to limit the number of connections is to use connlimit match.
  an example:
  iptables -A INPUT -p tcp –syn –dport 80 -m connlimit –connlimit-above 15 –connlimit-mask 32 -j REJECT –reject-with tcp-reset
  that will reject connections above 15 from one source IP – a very good rule to defend a web server.

  Also it will be great to add to your list of rules an example with “hashlimit” match, which has more options when you want to protect from a DDoS attack.
  With “limit” match you can limit the global rate of packets per time interval, but with “hashlimit”, you can limit them per IP, per combination IP + port, etc.
  So an example for a web server will be something like that:
  iptables -A INPUT -p tcp –dport 80 -m hashlimit –hashlimit 45/sec –hashlimit-burst 60 –hashlimit-mode srcip
  –hashlimit-name DDOS –hashlimit-htable-size 32768 –hashlimit-htable-max 32768 –hashlimit-htable-gcinterval 1000
  –hashlimit-htable-expire 100000 -j ACCEPT

  Hope this will help someone.

  Anyway, thanks for a good article!

  Link
- Paul M June 30, 2011, 7:59 am

  it makes sense to have a generic rule permitting established/related for input and forwarded, allow all outbound traffic from the firewall itself, and then control traffic in the forwarded or input chain, otherwise it gets too complex too quickly.

  and don’t forget to add rules for the lo interface.

  Link
- jalal hajigholamali October 21, 2011, 11:04 am

  very good… thanks

  Link
- Sharad December 12, 2011, 6:11 am

  Really helped me a lot…

  Just 1 question, after implementing some of the policies, My server is unable to resolve any domain name.

  Link
- Rod_Moncada January 16, 2012, 4:48 pm

  Hey Sharad,

  Actually need to open port 53 for INPUT. Something like this will work:
  # dnsserver=ipaddress
  # iptables -A INPUT -p udp –sport 53 –dport 1024:65535 -d $dnsserver -j ACCEPT

  That worked for me, please some advice if there’s a better or safer way 😉

  Link
- Stuart January 28, 2012, 2:17 pm

  Excellent. Thanks.

  Link
- de2x February 24, 2012, 11:13 pm

  Can i allow host/domain google on iptables? can you tell me, please help..

  Link
- tarani March 28, 2012, 7:36 am

  EXELLENT……

  Link
- Curtis Ruck May 8, 2012, 2:52 pm

  How can you forward traffic (like 443) to a hidden server (i.e. in a DMZ) from one NIC to another? I’ve tried setting up your rule #24, but the packets are getting logged and dropped from the FORWARD chain.

  Link
- Max May 27, 2012, 2:13 pm

  VERY USEFULL ARTICE….

  @vierupro, I have router with busybox and have defined many rules to set network limits. plz write down a rule which is capable to limit download speed to 20 kb /s for a specif IP.

  Link
- sriram r July 9, 2012, 5:51 pm

  Is there a rule so I can make my outbound https sessions ‘sticky’ using iptables? I am trying to achieve this on a dual WAN dd-wrt router (Linksys E2000)

  Link
- Kennedy August 25, 2012, 8:26 am

  “Error applying iptables rules. Exit code: 11.

iptables v1.4.10
iptables: No chain/target/ match by that name.”

Can any one tell what this error is and how to over come.
I have a android phone and i am tring to install a firewall in my rooted device and i am stuck with this error.

Link

- Renato September 13, 2012, 9:44 pm

  How do I compare two linux firewall, in terms of network security?

  Link

- Saed September 28, 2012, 10:01 pm

  that right Paul M …
  iptables –flush
  iptables –policy INPUT DROP
  iptables –policy OUTPUT DROP
  iptables –policy FORWARD DROP
  iptables –append INPUT -i lo -j ACCEPT
  iptables –append OUTPUT -o lo -j ACCEPT
  iptables –append INPUT -m state –state \
  ESTABLISHED,RELATED -j ACCEPT
  iptables –append OUTPUT -m state –state \
  NEW,ESTABLISHED,RELATED -j ACCEPT

  Link

- aleee January 30, 2013, 2:09 am

  thanks……..

  Link

- dp February 23, 2013, 7:24 am

  Hai Friends,

  Please let me know how to restrict the server access by using the iptables with mac addressses.

  I want to give permission only to 2 mac addresses and restict the remaining mac addresses to connect with the server.

  Thanks
  dp

  Link

- Amit Shrivastava April 9, 2013, 2:51 am

  Good collection…Thx

  Link

- Sumanta April 25, 2013, 1:42 am

  Hi,

  Thanks for this great article! I have one query regarding the deletion of a rule.
  I am creating a static nat rule in iptables using the command

  iptables -t nat -I POSTROUTING 1 -j SNAT -p ip -s 1.1.1.10/32 –to-source 2.2.2.30-2.2.2.30 -o eth2

  After executing the above command i am able to observe the source translation happening and also it is listed in the iptables nat list.

  After I remove the rule by executing the below command

  iptables -t nat -D POSTROUTING -j SNAT -p ip -s 1.1.1.10/32 –to-source 2.2.2.30-2.2.2.30 -o eth2

  The entry in the iptables nat is deleted, but i am still able to observe the source translation happening from 1.1.1.10 to 2.2.2.30.

  Please suggest what i need to do to delete the entry completely from the nat table or did i missed something.

  Thanks in advance!

  Regards,
  Sumanta.

  Link

- Daniel June 10, 2013, 12:25 am

  I would offer that you may want to consider changing the order you discuss the commands.

  MOST things in linux require a service x restart in order to put the change into effect.

  Not, apparently, iptables.

  Telling a VPS to drop ALL traffic while you’re SSHd into it is never a good idea.

  Link

- Br H August 5, 2013, 6:36 am

  Hi,
  Thanks for this wonderful list.
  Just with the emails – the only configuration that worked was with the following three rules:
  iptables -A OUTPUT -o eth0 -p tcp –sport 25 -m state –state NEW,ESTABLISHED -j ACCEPT
  iptables -A OUTPUT -o eth0 -p tcp –dport 25 -m state –state NEW,ESTABLISHED -j ACCEPT
  iptables -A INPUT -i eth0 -p tcp –sport 25 -m state –state ESTABLISHED -j ACCEPT

Many thanks again

Link

- iabc August 27, 2013, 9:10 am

Here is iptables for passive ftp if anyone is interested.
& Thanks for Ramesh for your wonderful blogs.

iptables -A INPUT -p tcp -m tcp –dport 21 -m state –state NEW,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -p tcp -m tcp –sport 21 -m state –state ESTABLISHED -j ACCEPT

iptables -A INPUT -p tcp -m tcp –sport 1024:65535 –dport 1024:65535 -m state –state NEW,RELATED,ESTABLISHED -j ACCEPT
iptables -A OUTPUT -p tcp -m tcp –sport 1024:65535 –dport 1024:65535 -m state –state NEW,RELATED,ESTABLISHED -j ACCEPT

Link

- iTommix September 2, 2013, 7:25 am

Hi,

nice tutorial. I have a following question:

I build a captive portal (hotspot) using iptables. from the beginning i block every port and routing to an splash page on the own apache server:

iptables -t mangle -A PREROUTING -i wlan0 -p tcp -m tcp –dport 1:65535 -j internet
iptables -t nat -A PREROUTING -i wlan0 -p tcp -m mark –mark 99 -m tcp –dport 1:65535 -j DNAT –to-destination 10.0.0.254

But i need an exception on a specific domain to load some css and javascript files into this splash page. How can i do this?

Regards an thanks for reply in advance.

Link

- tarvi October 16, 2013, 12:29 am

What is eth0 ?

Link

- zug October 16, 2013, 10:32 pm

Very helpful and a good introduction. Thanks.

Link

- saththiyan November 3, 2013, 10:18 pm

hey ,

I have two AWS instances , Ngx and another one call WP, WP running a as hosting server and Ngx is proxy. I need to configure WP to accept only HTTP request from Ngx. How do i do this?

Link

- saththiyan November 3, 2013, 10:23 pm

Hi,

I need to configure firewall (iptable) to accept HTTP request from a server, How do i configure this?
i have two instances on AWS, instance one need to accept HTTP request from Instance two only.

How do i configure ? Do i need to configure incoming and outgoing http request ?

Link

- Anon February 10, 2014, 6:13 am

It worked for me, Gj!

Link

- Anitha February 13, 2014, 12:54 am

We have to block the unused IPs through IPTABLE, for example one of our branch network has configured through this range 192.168.3.0/24 and many unused IPs are present in this series. For some reason we have to block the unused IP address.How could we have to block those IPs (Unused IPs like 192.168.3.5, 192.168.3.10 to 192.168.3.20, 192.168.3.24, 192.168.3.27, etc.,)

Link

- Mitchelle June 5, 2014, 5:04 am

Hello,

I am not able to access to imap and smtp ports 465 and 993 to access gmail using outlook express or ms outlook. Though i have accepted both ports in iptables. Still cant access.

Is there anything else our gateway that is ubuntu 14.04, proxy is on. two interface card one connecting to router and one to internal interface.

Any ideas?

Link

- Holger June 24, 2014, 4:15 pm

Example 24: you don’t have to open port 422, the prerouting is done before filter!

Link

- John Selbie November 9, 2014, 4:17 am

For #23 above, you have the following listed as the means to prevent a DOS attack:

———————————

23. Prevent DoS Attack

The following iptables rule will help you prevent the Denial of Service (DoS) attack on your webserver.

iptables -A INPUT -p tcp –dport 80 -m limit –limit 25/minute –limit-burst 100 -j ACCEPT

_____

Shouldn't the rule be specifying "-j DROP" at the end instead of "-j ACCEPT" ? Or am I confused on how this rule is supposed to behave?

Link

- Jon Syvertson March 4, 2015, 11:51 am

  Just so you know your postfix configuration doesn't work. I believe –sport and –dport are swapped. – Regards, Jon

  Link
- Dmitry Sandalov March 5, 2015, 5:11 pm

  My imaps and smtp didn't work too. Works perfectly with these modifications:

  # 20. Allow Sendmail or Postfix
  iptables -A OUTPUT -o eth0 -p tcp –dport 25 -m state –state NEW,ESTABLISHED -j ACCEPT
  iptables -A INPUT -i eth0 -p tcp –sport 25 -m state –state ESTABLISHED -j ACCEPT

  # 21. Allow IMAPS
  iptables -A OUTPUT -o eth0 -p tcp –dport 993 -m state –state NEW,ESTABLISHED -j ACCEPT
  iptables -A INPUT -i eth0 -p tcp –sport 993 -m state –state ESTABLISHED -j ACCEPT

  # Allow SMTPs
  iptables -A OUTPUT -o eth0 -p tcp –dport 465 -m state –state NEW,ESTABLISHED -j ACCEPT
  iptables -A INPUT -i eth0 -p tcp –sport 465 -m state –state ESTABLISHED -j ACCEPT

  Link
- Abdul Vadood March 12, 2015, 4:44 am

  Hi,

  I want to keep iptable enabled on my server, but even after added 443 to accept connection, https is not loading. It is loading when I disable iptable.

  selinux is on disabled state.

  Tried following rules, but no luck
  -A INPUT -p tcp -m tcp –dport 443 -j ACCEPT
  -A INPUT -p tcp -m state –state NEW,ESTABLISHED -m tcp –dport 443 -j ACCEPT

  Any idea?

  Thanks,
  Abdul vadood

  Link
- nithesh March 13, 2015, 7:20 pm

  It really helped
  Thanks a lot

  Link
- Peter Muia May 26, 2015, 11:39 am

  Hi
  If I run steps for Load Balancing incoming web traffic on Example 11 I get the following:
  [root@iptablesSrv ~]# iptables -A PREROUTING -i eth0 -p tcp –dport 443 -m state –state NEW -m nth –counter 0 –every 3 –packet 0 -j DNAT –to-destination 192.168.1.101:443
  iptables v1.4.21: Couldn't load match `nth':No such file or directory
  I am using centos 7 & seem not to find any info on `nth'

  Link
- Santiago February 25, 2016, 4:59 pm

  Hi. Hope you can help me.

  I have installed MongoDB in a CentOS server. I have the default configuration.

  I want to connect to a database and make some queries from an Android application. I was told that I should open a port in the server and forward that traffic to the MongoDB port (which by default is 27017).
  What I understand from "iptables" is that I can open directly that port (INPUT chain of filter table).
  Am I wrong? Are there security issues that could rise doing it?
  Should I better use the PREROUTING chain of NAT table?

  Thanks for your help!

  Link
- Jouni "rautamiekka" Järvinen February 29, 2016, 5:48 pm

  WARNING: If you don't have rules which allow access, running §iptables -A LOGGING -j DROP§ as part of #25 blocks access completely !

  Link
- Jo April 12, 2016, 8:06 am

  Hey guys. I need big help.
  I have a task to make difference between iptables and nftables.
  I should use iperf.
  To make a step to iperf first of all i should take some bash script to run ~ 50 – 100k rules with last rule possible to throw the iperf.
  How is it possible to create that kind of rules number ?

  Link

Leave a Comment