

# ? Making Monero Port 18080 Firewall Rule Persistent on DietPi (with ts-input Chain)

## ? Problem Summary

- Your DietPi uses a custom iptables chain called `ts-input` (e.g., created by Tailscale).
  - The Monero daemon needs TCP port 18080 open inbound for P2P connections.
  - Direct rule persistence (iptables-persistent, cron @reboot) **fails** because the `ts-input` chain is created **late** in boot or dynamically.
  - Applying `iptables -I ts-input 1 -p tcp --dport 18080 -j ACCEPT` too early errors out:  
`iptables: No chain/target/match by that name.`
- 

## ?? Workaround Overview

- Use a **custom script** that:
    - **Waits** for `ts-input` chain to exist (polls for up to 60 seconds)
    - **Removes duplicate** existing rules for port 18080
    - **Inserts the rule if missing**
  - Run this script via a **systemd service** triggered after the network and Tailscale daemon start.
- 

## ? Step 1 — Create the script

```
/root/scripts/fix-monero-fw.sh
```

```
#!/bin/bash
```

```
# Wait up to 60 seconds for ts-input chain to appear
for i in {1..60}; do
    if iptables -L ts-input &>/dev/null; then
```

```

    echo "$(date) - Chain ts-input exists, proceeding"
    break
else
    echo "$(date) - Chain ts-input does not exist yet, waiting..."
    sleep 1
fi
done

# Exit if chain never appears
if ! iptables -L ts-input &>/dev/null; then
    echo "$(date) - ERROR: Timeout waiting for ts-input chain. Exiting."
    exit 1
fi

# Remove duplicate rules for tcp port 18080 in ts-input, keep only first
RULE_COUNT=$(iptables -L ts-input -n --line-numbers | grep -c 'tcp dpt:18080')

while [ "$RULE_COUNT" -gt 1 ]; do
    echo "$(date) - Removing duplicate rule at position 2"
    iptables -D ts-input 2
    RULE_COUNT=$(iptables -L ts-input -n --line-numbers | grep -c 'tcp dpt:18080')
done

# Insert rule if missing
if ! iptables -C ts-input -p tcp --dport 18080 -j ACCEPT &>/dev/null; then
    echo "$(date) - Rule not found, inserting rule"
    iptables -I ts-input 1 -p tcp --dport 18080 -j ACCEPT
else
    echo "$(date) - Rule already exists, no action needed"
fi

exit 0

```

**Make executable:**

```
chmod +x /rootscripts/fix-monero-fw.sh
```

---

## ?? Step 2 — Create systemd service

```
/etc/systemd/system/fix-monero-  
fw.service
```

[Unit]

Description=Add iptables rule for Monero port 18080 after tailscaled starts

After=network-online.target tailscaled.service

Wants=network-online.target tailscaled.service

[Service]

Type=oneshot

ExecStart=/root/scripts/fix-monero-fw.sh

RemainAfterExit=yes

StandardOutput=append:/var/log/fix-monero-fw.log

StandardError=append:/var/log/fix-monero-fw.log

[Install]

WantedBy=multi-user.target

## ? Step 3 — Enable and start the service

systemctl daemon-reload

systemctl enable fix-monero-fw.service

systemctl start fix-monero-fw.service

---

## ? Step 4 — Verify after reboot

Check the rule applied:

```
iptables -L ts-input -n --line-numbers | grep 18080
```

Check logs:

```
tail /var/log/fix-monero-fw.log
```

---

## ? Optional — Log rotation

**Create** `/etc/logrotate.d/fix-monero-fw`

```
/var/log/fix-monero-fw.log {  
    daily  
    rotate 7  
    missingok  
    notifempty  
    compress  
    delaycompress  
    copytruncate  
}
```

```
}
```

**Test rotation:**

```
logrotate --force /etc/logrotate.d/fix-monero-fw
```

# ? Result

- Firewall rule for port 18080 in `ts-input` chain is applied reliably **after reboot**.
- Duplicate rules are automatically cleaned up.
- Logs are captured and rotated for easy troubleshooting.

# ? Additional Useful Commands

Command	Description
<code>lsof -i :18080</code>	Lists processes using TCP port 18080
<code>journalctl -u fix-monero-fw.service -b -n 50</code>	Shows last 50 logs from the fix-monero-fw.service for current boot
<code>cat /var/log/fix-monero-fw.log</code>	Displays the content of the fix-monero-fw.log file
<code>journalctl -u monerod.service -f</code>	Follows live logs of the Monero daemon service
<code>`ps aux`</code>	<code>grep monerod`</code>
<code>systemctl daemon-reexec</code>	Reloads the systemd manager configuration
<code>systemctl daemon-reload</code>	Reloads systemd units and configurations
<code>systemctl enable iptables-custom-rule.service</code>	Enables the iptables custom rule service
<code>iptables -L ts-input -n --line-numbers</code>	Lists all rules in <code>ts-input</code> chain with line numbers
<code>`iptables -L ts-input -n --line-numbers`</code>	<code>grep 18080`</code>
<code>sudo crontab -l</code>	Lists root user's cron jobs

•

---

Revision #9

Created 3 June 2025 23:00:40 by coolbaron

Updated 4 June 2025 01:02:07 by coolbaron