Redsocks in Action

Certainly! I’ll guide you through setting up a **Shadowsocks** proxy using your Mac computer and configuring your **OpenWRT** router to route all connected devices’ traffic through this proxy. This setup involves the following main steps:

1. **Setting Up Shadowsocks Client on macOS (Using Shadowsocks-NG or Clash)**
2. **Configuring macOS to Allow External Proxy Connections**
3. **Assigning a Static IP to Your Mac**
4. **Installing and Configuring Redsocks on OpenWRT**
5. **Redirecting OpenWRT’ s Traffic Through the macOS Proxy**
6. **Testing the Proxy Setup**

Let’ s dive into each step in detail.

---

1. Setting Up Shadowsocks Client on macOS

You can use either **Shadowsocks-NG** or **Clash** as your Shadowsocks client. Below are instructions for both.

Option A: Using Shadowsocks-NG

Shadowsocks-NG is a popular and user-friendly Shadowsocks client for macOS.

Step 1: Download and Install Shadowsocks-NG

1. **Download Shadowsocks-NG:**
   - Visit the Shadowsocks-NG GitHub Releases page.
   - Download the latest `.dmg` file.
2. **Install the Application:**
   - Open the downloaded `.dmg` file.
   - Drag the **ShadowsocksX-NG** app to your **Applications** folder.
3. **Launch Shadowsocks-NG:**
   - Open **ShadowsocksX-NG** from your **Applications** folder.
   - You might need to grant the app necessary permissions in **System Preferences**.

Step 2: Configure Shadowsocks-NG

1. **Open Preferences:**
   - Click on the **ShadowsocksX-NG** icon in the menu bar.

- Select “**Open ShadowsocksX-NG**” > “**Preferences**”.

2. **Add a New Server:**
   - Navigate to the “**Servers**” tab.
   - Click the “+” button to add a new server.
3. **Import the Shadowsocks URL:**
   - **Copy Your Shadowsocks URL:**

     `ss://[ENCRYPTED_PASSWORD]@xxx.xxx.xxx.xxx:xxxxx/?outline=1`
   - **Import Method:**
     - Click “**Import**”.
     - Paste your Shadowsocks URL.
     - Shadowsocks-NG should automatically parse and fill in the server details.
4. **Set Up the Local Proxy:**
   - Ensure that “**Enable SOCKS5 Proxy**” is checked.
   - Note the **Local Port** (default is usually `1080`).
5. **Save and Activate:**
   - Click “**OK**” to save the server.
   - Toggle the “**Enable Shadowsocks**” switch to **ON**.

Option B: Using Clash

Clash is a versatile proxy client that supports multiple protocols, including Shadowsocks.

Step 1: Download and Install Clash

1. **Download Clash for macOS:**
   - Visit the Clash GitHub Releases page.
   - Download the latest **Clash for macOS** binary.
2. **Install the Application:**
   - Move the downloaded **Clash** application to your **Applications** folder.
3. **Launch Clash:**
   - Open **Clash** from your **Applications** folder.
   - You might need to grant necessary permissions in **System Preferences**.

Step 2: Configure Clash

1. **Access Configuration File:**
   - Clash uses a YAML configuration file. You can create or edit it using a text editor like **TextEdit** or **Visual Studio Code**.
2. **Add Your Shadowsocks Server:**

- Create a configuration file (e.g., `config.yaml`) with the following content:

  ```yaml
  port: 7890
  socks-port: 7891
  allow-lan: true
  mode: Rule
  log-level: info

  proxies:
    - name: "MyShadowsocks"
      type: ss
      server: xxx.xxx.xxx.xxx
      port: xxxxx
      cipher: chacha20-ietf-poly1305
      password: "xxxxxx"

  proxy-groups:
    - name: "Default"
      type: select
      proxies:
        - "MyShadowsocks"
        - "DIRECT"

  rules:
    - MATCH,Default
  ```

  Notes:

  - `port` and `socks-port` define the HTTP and SOCKS5 proxy ports Clash will listen on.
  - `allow-lan: true` allows LAN devices to use the proxy.
  - `proxies` section includes your Shadowsocks server details.
  - `proxy-groups` and `rules` determine how traffic is routed.

3. **Start Clash with Configuration:**
   - Launch Clash and ensure it uses your `config.yaml` file.
   - You might need to specify the configuration path when starting Clash.
4. **Verify Proxy is Running:**
   - Ensure Clash is active and connected to your Shadowsocks server.
   - Check the menu bar icon for status.

---

2. Configuring macOS to Allow External Proxy Connections

By default, Shadowsocks clients bind the proxy to `localhost` (`127.0.0.1`), meaning only the Mac can use the proxy. To allow your OpenWRT router to use this proxy, you need to bind the proxy to the Mac’ s LAN IP.

For Shadowsocks-NG:

1. **Open Preferences:**
   - Click the **ShadowsocksX-NG** icon in the menu bar.
   - Select “**Open ShadowsocksX-NG**” > “**Preferences**” .
2. **Go to the Advanced Tab:**
   - Navigate to the “**Advanced**” tab.
3. **Set the Listening Address:**
   - Change the “**Listen Address**” from `127.0.0.1` to `0.0.0.0` to allow connections from any interface.
   - Alternatively, specify the Mac’ s LAN IP (e.g., `192.168.1.xxx`).
4. **Save and Restart Shadowsocks-NG:**
   - Click “**OK**” to save changes.
   - Restart the Shadowsocks-NG client to apply the new settings.

For Clash:

1. **Edit Configuration File:**
   - Ensure that the `allow-lan: true` setting is enabled in your `config.yaml`.
2. **Bind to All Interfaces:**
   - In the configuration, setting `allow-lan: true` typically binds the proxy to all available interfaces, including the LAN.
3. **Restart Clash:**
   - Restart the Clash client to apply the changes.

---

3. Assigning a Static IP to Your Mac

To ensure consistent connectivity between your OpenWRT router and the Mac, assign a static IP to your Mac within your local network.

Steps to Assign a Static IP on macOS:

1. **Open System Preferences:**

- Click the **Apple** menu and select “**System Preferences**”.

2. **Navigate to Network Settings:**
   - Click on “**Network**”.
3. **Select Your Active Connection:**
   - Choose “**Wi-Fi**” or “**Ethernet**” from the left sidebar, depending on how your Mac is connected to the router.
4. **Configure IPv4 Settings:**
   - Click “**Advanced…**”.
   - Go to the “**TCP/IP**” tab.
   - Change “**Configure IPv4**” from “**Using DHCP**” to “**Manually**”.
5. **Set Static IP Address:**
   - **IP Address:** Choose an IP outside your router’s DHCP range to prevent conflicts (e.g., `192.168.1.xxx`).
   - **Subnet Mask:** Typically `255.255.255.0`.
   - **Router:** Your router’s IP address (e.g., `192.168.1.1`).
   - **DNS Server:** You can use your router’s IP or another DNS service like `8.8.8.8`.
6. **Apply Settings:**
   - Click “**OK**” and then “**Apply**” to save the changes.

---

4. Installing and Configuring Redsocks on OpenWRT

Redsocks is a transparent socks redirector that allows you to route network traffic through a SOCKS5 proxy. We’ll use Redsocks to redirect OpenWRT’s traffic through the Shadowsocks proxy running on your Mac.

Step 1: Install Redsocks

1. **Update Package Lists:**

   ```
   ssh root@<router_ip>
   opkg update
   ```

2. **Install Redsocks:**

   ```
   opkg install redsocks
   ```

   If Redsocks is not available in your OpenWRT repository, you might need to compile it manually or use an alternative package.

Step 2: Configure Redsocks

1. **Create or Edit Redsocks Configuration File:**

```
vi /etc/redsocks.conf
```

2. **Add the Following Configuration:**

```
base {
    log_debug = on;
    log_info = on;
    log = "file:/var/log/redsocks.log";
    daemon = on;
    redirector = iptables;
}

redsocks {
    local_ip = 0.0.0.0;
    local_port = 12345;  # Local port for Redsocks to listen on
    ip = xxx.xxx.xxx.xxx;  # Mac's static IP
    port = xxxxx;          # Shadowsocks-NG's local SOCKS5 proxy port
    type = socks5;
    login = "";           # If your proxy requires authentication
    password = "";
}
```

Notes:

- **`local_port`**: The port Redsocks listens on for incoming connections from iptables redirects.
- **`ip`** and **`port`**: Point to your Mac' s Shadowsocks SOCKS5 proxy (`xxx.xxx.xxx.xxx:xxxxx` based on previous steps).
- **`type`**: Set to `socks5` as Shadowsocks provides a SOCKS5 proxy.

3. **Save and Exit:**

- Press `ESC`, type `:wq`, and press `Enter`.

4. **Create Log File:**

```
touch /var/log/redsocks.log
chmod 644 /var/log/redsocks.log
```

Step 3: Start Redsocks Service

1. **Enable Redsocks to Start on Boot:**

   ```
   /etc/init.d/redsocks enable
   ```

2. **Start Redsocks:**

   ```
   /etc/init.d/redsocks start
   ```

3. **Verify Redsocks is Running:**

   ```
   ps | grep redsocks
   ```

   You should see a Redsocks process running.

---

5. Redirecting OpenWRT' s Traffic Through the macOS Proxy

Now that Redsocks is set up on OpenWRT, configure iptables to redirect all outbound TCP traffic through Redsocks, which in turn routes it through your Mac' s Shadowsocks proxy.

Step 1: Configure iptables Rules

1. **Add iptables Rules to Redirect Traffic:**

   ```
   # Redirect all TCP traffic to Redsocks (except traffic to the proxy itself)
   iptables -t nat -N REDSOCKS
   iptables -t nat -A REDSOCKS -d xxx.xxx.xxx.xxx -p tcp --dport xxxxx -j RETURN
   iptables -t nat -A REDSOCKS -p tcp -j REDIRECT --to-ports 12345

   # Apply the REDSOCKS chain to all outgoing traffic
   iptables -t nat -A OUTPUT -p tcp -j REDSOCKS
   iptables -t nat -A PREROUTING -p tcp -j REDSOCKS
   ```

   Explanation:

   - **Create a New Chain:** `REDSOCKS`
   - **Exclude Proxy Traffic:** Ensure traffic destined for the proxy itself is not redirected.
   - **Redirect Other TCP Traffic:** Forward other TCP traffic to Redsocks' listening port (`12345`).

2. **Save iptables Rules:**

   To make these rules persistent across reboots, add them to the firewall configuration.

   ```
   vi /etc/firewall.user
   ```

Add the iptables Rules:

```
# Redirect all TCP traffic to Redsocks (except proxy)
iptables -t nat -N REDSOCKS
iptables -t nat -A REDSOCKS -d xxx.xxx.xxx.xxx -p tcp --dport xxxxx -j RETURN
iptables -t nat -A REDSOCKS -p tcp -j REDIRECT --to-ports 12345

# Apply the REDSOCKS chain
iptables -t nat -A OUTPUT -p tcp -j REDSOCKS
iptables -t nat -A PREROUTING -p tcp -j REDSOCKS
```

Save and Exit:

- Press `ESC`, type `:wq`, and press `Enter`.

3. **Restart Firewall to Apply Changes:**

```
/etc/init.d/firewall restart
```

Step 2: Verify Traffic is Being Redirected

1. **Check Redsocks Logs:**

```
cat /var/log/redsocks.log
```

You should see logs indicating that traffic is being processed through Redsocks.

2. **Test from a Client Device:**

- Connect a device to your OpenWRT router.
- Visit a website or perform an action that uses the internet.
- Verify that the traffic is routed through the Shadowsocks proxy by checking the external IP address (e.g., via WhatIsMyIP.com) to see if it reflects the proxy’s IP.

---

6. Testing the Proxy Setup

Ensure that the entire setup works as intended by performing the following tests.

Step 1: Verify Shadowsocks Connection on Mac

1. **Check Shadowsocks Client Status:**
   - Ensure that Shadowsocks-NG or Clash is actively connected to the Shadowsocks server.
   - Verify that the local proxy (e.g., `xxx.xxx.xxx.xxx:xxxxx`) is accessible.

2. **Test the Proxy Locally:**
   - On your Mac, open a browser and configure it to use `localhost:1080` as the SOCKS5 proxy.
   - Visit WhatIsMyIP.com to confirm the IP matches the Shadowsocks server.

Step 2: Verify OpenWRT’ s Traffic is Routed Through the Proxy

1. **Check OpenWRT’ s External IP:**
   - From a device connected to OpenWRT, visit WhatIsMyIP.com to see if the IP reflects the Shadowsocks server’ s IP.
2. **Monitor Redsocks Logs:**
   - On OpenWRT, monitor Redsocks logs to ensure traffic is being redirected.

   ```
   tail -f /var/log/redsocks.log
   ```
3. **Troubleshoot if Necessary:**
   - If traffic isn’ t being routed correctly:
     - Ensure Shadowsocks client on Mac is running and accessible.
     - Verify iptables rules are correctly set.
     - Check firewall settings on both Mac and OpenWRT.

Additional Considerations

1. Security

- **Secure Your Proxy:**
  - Ensure that only trusted devices can access the proxy. Since you’ re redirecting all traffic through Redsocks, ensure that your Mac’ s firewall only allows connections from your OpenWRT router.

  On macOS:

  - Go to **System Preferences** > **Security & Privacy** > **Firewall**.
  - Configure the firewall to allow incoming connections on the proxy port (`xxxxx`) only from the OpenWRT router’ s IP.
- **Authentication:**
  - Shadowsocks already provides some level of security via encryption. Ensure strong passwords and encryption methods.

2. Performance

- **Router Resources:**

    - Running proxy services like Redsocks can consume additional CPU and memory on your OpenWRT router. Ensure your router has sufficient resources.
- **Mac Performance:**
    - Ensure your Mac remains powered on and connected to the network to maintain proxy availability.

3. Maintenance

- **Monitor Logs:**
    - Regularly check Redsocks and Shadowsocks logs for any unusual activity or errors.
- **Update Software:**
    - Keep OpenWRT, Redsocks, and your Shadowsocks client updated to benefit from security patches and performance improvements.

4. Alternative Approaches

While using a Mac as an intermediary proxy server is feasible, consider the following alternatives for potentially simpler setups:

- **Directly Configure OpenWRT as a Shadowsocks Client:**
    - OpenWRT supports Shadowsocks directly through packages like `shadowsocks-libev`. This approach eliminates the need for a Mac intermediary.
- **Use a Dedicated Proxy/VPN Device:**
    - Devices like Raspberry Pi can run proxy services and act as dedicated gateways.

Conclusion

By following the steps outlined above, you' ve set up your Mac as a Shadowsocks proxy server and configured your OpenWRT router to route all connected devices' traffic through this proxy. This setup enhances your network' s privacy and control by leveraging the Shadowsocks protocol.

Key Points Recap:

1. **Shadowsocks Client on Mac:**
    - Installed and configured Shadowsocks-NG or Clash with your provided Shadowsocks URL.
    - Configured the client to listen on the Mac' s LAN IP.
2. **Proxy Accessibility:**
    - Assigned a static IP to the Mac to ensure consistent proxy access.
    - Configured macOS firewall to allow incoming proxy connections.
3. **OpenWRT Configuration:**

    - Installed and configured Redsocks to redirect all outbound TCP traffic through the Shadowsocks proxy.
    - Applied necessary iptables rules to enforce traffic redirection.
4. **Testing:**
    - Verified that traffic from connected devices routes through the Shadowsocks proxy by checking external IP addresses.

Recommendations:

- **Ensure Stability:** Keep your Mac powered and connected to maintain proxy availability.
- **Security First:** Regularly update all components and monitor logs to safeguard against unauthorized access.
- **Consider Direct Router Configuration:** For a more streamlined setup, explore configuring Shadowsocks directly on your OpenWRT router.

If you encounter any issues or need further assistance with specific configurations, feel free to reach out!

```
base {
    log_debug = on;
    log_info = on;
    log = "file:/var/log/redsocks.log";
    daemon = on;
    redirector = generic;
}

redsocks {
    local_ip = 0.0.0.0;
    local_port = 7891;
    ip = xxx.xxx.xxx.xxx;
    port = xxxxx;
    type = http-connect;
    login = "";
    password = "";
}
```