Router

Router Purpose: Gives IDs to hosts and allows peers (hosts and clients) to agree on how they will bypass NAT. The router server must have a public IP address.

1. **Installing the router**

   Windows:

   Run aspia-router-2.0.0-x86.msi and follow the instructions on the screen.

   Linux:

   ```
   sudo apt install ./aspia-router-2.0.0-x86_64.deb
   ```

2. **Creating a default configuration**

   Windows:

   ```
   cd /d "C:\Program Files (x86)\Aspia\Router"
   aspia_router --create-config
   ```

   Linux:

   ```
   sudo aspia_router --create-config
   ```

3. **Open configuration file and fill in the fields (if necessary; the default configuration does not need to be edited in most cases).**

   Windows:

   C:\ProgramData\aspia\router.json

   Linux:

   /etc/aspia/router.json

   3.1. **PrivateKey:** If you already have a private key, then write it here. If this is your first time installation, then leave the field blank. The key will be generated automatically upon first launch.

   3.2. **Port:** The port on which incoming connections will be accepted. You can leave the default value.

   3.3. **MinLogLevel:** Log level. Decreasing the value increases the number of messages in the log. The minimum value is 0, the maximum value is 3. You can leave the default value.

   3.4. **ClientWhiteList:** The IP address list of clients who are allowed to connect to the router. Addresses are separated by semicolons. If the list is empty, then connections from all clients are allowed. If the list contains items, then only the clients specified in this list can connect.

   3.5. **HostWhiteList:** The IP address list of hosts who are allowed to connect to the router. Addresses are separated by semicolons. If the list is empty, then connections from all hosts are allowed. If the list contains items, then only the hosts specified in this list can connect.

   3.6. **AdminWhiteList:** The IP address list of admins who are allowed to connect to the router. Addresses are separated by semicolons. If the list is empty, then connections from all admins are allowed. If the list contains items, then only the admins specified in this list can connect.

   3.7. **RelayWhiteList:** The IP address list of relays who are allowed to connect to the router. Addresses are separated by semicolons. If the list is empty, then connections from all relays are allowed. If the list contains items, then only the relays specified in this list can connect.

4. **Service/daemon starting**

   Windows:

   ```
   net stop aspia-router
   ```

```
net start aspia-router
```

Linux:

```
sudo systemctl enable aspia-router
sudo service aspia-router start
```

5. **Open public key file and copy the public key. It will come in handy for configuring the relay and hosts.**

   Windows:

   C:\ProgramData\aspia\router.pub

   Linux:

   /etc/aspia/router.pub

Router file locations

1. **Logs**

   Windows:

   C:\Windows\Temp\aspia\aspia_router-*.log

   Linux:

   ```
   sudo journalctl -u aspia-router
   ```

2. **Configuration**

   Windows:

   C:\ProgramData\aspia\router.json

   Linux:

   /etc/aspia/router.json

3. **Data base**

   Windows:

   C:\ProgramData\aspia\router.db3

   Linux:

   /var/lib/aspia/router.db3

NOTES:

1. Hosts and relays connect to the router using a public key.
2. Clients and the console connect using a username and password. You can add additional users when managing routers in the console.
3. It is recommended that you set up regular backups of your configuration files and database.
4. Don't forget to add rules in your firewall to access the router. The router does not add rules automatically.
5. It is recommended to limit the list of relays that can be connected to the router. Whitelist the required relays.

Relay

Relay Purpose: Passes traffic between peers (hosts and clients) through itself. The relay server must have a public IP address.

1. **Installing the relay**

   Windows:

   Run aspia-relay-2.0.0-x86.msi and follow the instructions on the screen.

   Linux:

   ```
   sudo apt install ./aspia-relay-2.0.0-x86_64.deb
   ```

2. **Creating a default configuration**

   Windows:

   ```
   cd /d "C:\Program Files (x86)\Aspia\Relay"
   aspia_relay --create-config
   ```

   Linux:

   ```
   sudo aspia_relay --create-config
   ```

3. **Open configuration file and fill in the fields**

   Windows:

   C:\ProgramData\aspia\relay.json

   Linux:

   /etc/aspia/relay.json

   3.1. **RouterAddress**: Router address. It can be equal to localhost if the router is installed on the same computer.

   3.2. **RouterPort**: If you did not change the port in the router configuration file, then the field must be left with the default value. If you changed the configuration of the router, then write the required value.

   3.3. **RouterPublicKey**: Should contain the public key of the router that you received when installing it.

   3.4. **PeerAddress**: The address that peers will receive to connect to the relay server.

   WARNING! This address must be accessible to all participants in the connection. You should keep in mind that both peers (host and client) must be able to connect to this address. Consider this when setting up your network hardware if you are setting up port forwarding on your router.

   3.5. **PeerPort**: The port through which peers will connect to the relay server. You can leave the default value.

   3.6. **PeerIdleTimeout**: Time in minutes. If during this time no data comes from the peers, the connection is terminated. You can leave the default value.

   3.7. **MaxPeerCount**: The maximum number of simultaneous connections established between peers. You can leave the default value.

   3.8. **MinLogLevel**: Log level. Decreasing the value increases the number of messages in the log. The minimum value is 0, the maximum value is 3. You can leave the default value.

4. Service/daemon starting

   Windows:

   ```
   net stop aspia-relay
   net start aspia-relay
   ```

Linux:

```
sudo systemctl enable aspia-relay
sudo service aspia-relay start
```

Relay file locations

1. **Logs**

   Windows:

   C:\Windows\Temp\aspia\aspia_relay-*.log

   Linux:

   ```
   sudo journalctl -u aspia-relay
   ```

2. **Configuration**

   Windows:

   C:\ProgramData\aspia\relay.json

   Linux:

   /etc/aspia/relay.json

NOTES:

1. Don't forget to add rules in your firewall to access the relay. The relay does not add rules automatically.

Console

Console Purpose: Allows you to create address books, add computers to them and group them. It also allows you to manage computers and routers.

1. **Installing the console**

   Windows:

   Run aspia-console-2.0.0-x86.msi and follow the instructions on the screen.

   MacOS X:

   Open aspia-console-2.0.0.dmg and move “Aspia Console” to “Applications”.

   Linux:

   ```
   sudo apt install ./aspia-console-2.0.0-x86_64.deb
   ```

2. **Create a new address book and configure the router in its properties.**

   Default username: admin

   Default password: admin

   WARNING: Don't forget to change your password! Connect to your router and change the default password. You can also add additional users.

Client

Client Purpose: Allows you to connect to and control hosts.

1. **Installing the client**

   Windows:

      Run aspia-client-2.0.0-x86.msi and follow the instructions on the screen.

   MacOS X:

      Open aspia-client-2.0.0.dmg and move “Aspia Client” to “Applications”.

   Linux:

   ```
   sudo apt install ./aspia-client-2.0.0-x86_64.deb
   ```

Host

Host Purpose: Allows Accepts incoming connections from clients and consoles to manage the computer on which it is installed.

1. **Installing the host**

   The host is only available for Windows.

   Run aspia-host-2.0.0-x86.msi and follow the instructions on the screen.

2. **Enabling the router in the settings**

   2.1. Go to settings (Aspia -> Settings… -> Router)

   2.2. Enable the use of the router

   2.3. Write the address of your router

   2.4. Write your router's public key

APPENDIX

If you are configuring port forwarding on your network router, then an example of setting for the Mikrotik router may come in handy:

```
/ip firewall nat

add action=netmap chain=dstnat comment="Aspia Relay" dst-port=8070 in-interface=WAN protocol=tcp to-addresses=RELAY_IP to-ports=8070

add action=netmap chain=dstnat comment="Aspia Router" dst-port=8060 in-interface=WAN protocol=tcp to-addresses=ROUTER_IP to-ports=8060

add action=dst-nat chain=dstnat comment="Aspia Relay" dst-address=EXTERNAL_IP dst-port=8070 protocol=tcp src-address= LOCAL_NETWORK to-addresses=RELAY_IP to-ports=8070

add action=dst-nat chain=dstnat comment="Aspia Router" dst-address=EXTERNAL_IP dst-port=8060 protocol=tcp src-address= LOCAL_NETWORK to-addresses=ROUTER_IP to-ports=8060

add action=masquerade chain=srcnat comment="Aspia Relay" dst-address=SERVER_IP dst-port=8070 protocol=tcp src-address=LOCAL_NETWORK

add action=masquerade chain=srcnat comment="Aspia Router" dst-address=SERVER_IP dst-port=8060 protocol=tcp src-address=LOCAL_NETWORK
```

Replace the following with your data:

ROUTER_IP - IP address of the computer on which the router is installed.

RELAY_IP - IP address of the computer on which the relay is installed.

LOCAL_NETWORK – your local network (for example 192.168.1.0/24).

EXTERNAL_IP - your external IP address