

# Network interfaces

by SuperBonBon

## 1. Network interfaces.

JAFS can use multiple network interfaces. All those interfaces are configured in the service config file with [network-interface](#) entries. Here is a small XML example :

```
<jafs-service xmlns="http://www.sbbi.net/jafs/1.0/jafs-service">
  <network-interfaces>
    <network-interface id="default" default="true" name="localhost"/>
  </network-interfaces>
  .
  .
  .
</jafs-service>
```

As you can see this network interface config entry for localhost is quite simple but the grammar is much more powerful and allows to :

- Define a firewall host if you are behind a router or firewall with the [fw-host-name](#) attribute
- Define a timer to detect dynamic IP changes and assure correct FTP server operation with the [dyn-ip-check-time](#) attribute
- Define an automatic NAT ports mapper with the [fw-nat-ports-mapper](#) tag
- Define dynamic DNS clients with the [dynamic-dns-client](#) tag
- Define the [input](#) and [output](#) buffer size.

The [id](#) attribute is used to identify the interface entry name. This identifier can be used in other configuration files such as the [networkInterface](#) FTP server attribute, used to define which network interface an FTP server will use.

The [name](#) attribute is used to determine the network interface. This attribute supports a generic IP address (127.0.0.1), an host name (localhost) or an OS interface identifier (eth0,ppp0...). If you use an OS interface name, and that the interface has multiple IP addresses, you can use the (*ipv4*) and (*ipv6*) keywords followed by the interface name ( (*ipv4*)eth0 (*ipv6*)eth0 ) to use the first matching IP address for the given type with the interface. Here is a full XML example with 4 interfaces defined :

```

<jafs-service xmlns="http://www.sbbi.net/jafs/1.0/jafs-service">
  <network-interfaces>
    <network-interface id="default" default="true" name="localhost"/>

    <network-interface id="ethernet" name="eth0" fw-host-name="ftp.mycomp.com"
      dyn-ip-check-time="10">

      <fw-nat-ports-mapper class-name="net.sbbi.jafs.nat.UPNPNatPortsMapper"
        mappings-monitoring-time="10">
        <setting name="deviceUDN">foo</setting>
        <setting name="discoveryTimeout">1500</setting>
      </fw-nat-ports-mapper>

      <dynamic-dns-client id="DynDNSClient"
        class-name="net.sbbi.jafs.dyndns.DynDnsOrgClient">
        <setting name="hostName">test.dyndns.org</setting>
        <setting name="backupMx">false</setting>
        <setting name="offline">false</setting>
        <setting name="password">test</setting>
        <setting name="userName">test</setting>
        <setting name="wildCard">false</setting>
      </dynamic-dns-client>

    </network-interface>

    <network-interface id="ethernet2" name="192.168.1.34"/>

    <network-interface id="dialup" name="ppp0"/>

  </network-interfaces>
  ...
</jafs-service>

```

## 2. Create you own NAT mapper

You can plug your own [NAT ports mapper](#) for automatic ports mappings if the UPNP mapper we provide (`net.sbbi.jafs.nat.UPNPNatPortsMapper`) is not compatible with your router.

Since most of routers have generally telnet access you can create your own mapper and use the telnet interface programatically to make the mappings. The only thing you'll need is to create a Java class implementing the `net.sbbi.jafs.nat.NatPortsMapper` interface (look at the API docs for more infos), put your compiled class into a jar file and copy it into the libs dir. Finally plug it in your desired network interface config entry :

```

<jafs-service xmlns="http://www.sbbi.net/jafs/1.0/jafs-service">

```

## Network interfaces

```
<network-interfaces>
  <network-interface id="ethernet" name="eth0" fw-host-name="ftp.mycomp.com"
    dyn-ip-check-time="10">
    <fw-nat-ports-mapper class-name="com.foo.bar.MyNATMapper"
      mappings-monitoring-time="10">
      <setting name="myTelnetHost">ftp.mycomp.com</setting>
      <setting name="myOtherSetting">bar</setting>
    </fw-nat-ports-mapper>
  </network-interface>
  ...
</network-interfaces>
...
</jafs-service>
```

### 3. Create you own Dynamic DNS client

We provide a [client](#) (`net.sbbi.jafs.dyndns.DynDnsOrgClient`) for `www.dyndns.org` based host names.

If you use another dynamic DNS provider, you can take a look at the provider docs to see how the host name can be updated and code a class implementing the `net.sbbi.jafs.dyndns.DynamicDNSClient` interface. Once you're done put your compiled class into a jar file and copy it into the `libs` dir. Finally plug it in your desired network interface config entry :

```
<jafs-service xmlns="http://www.sbbi.net/jafs/1.0/jafs-service">
  <network-interfaces>
    <network-interface id="ethernet" name="eth0" fw-host-name="ftp.mycomp.com"
      dyn-ip-check-time="10">
      <dynamic-dns-client id="DynDNSClient"
        class-name="com.foo.bar.MYDynDNSProvider">
        <setting name="hostName">myhost.DynDNSProvider.com</setting>
        <setting name="password">myUser</setting>
        <setting name="userName">myPassword</setting>
      </dynamic-dns-client>
    </network-interface>
    ...
  </network-interfaces>
  ...
</jafs-service>
```