MDNS/DNS-SD TUTORIAL

In this tutorial, we will describe how to use mDNS/DNS-SD on Raspberry Pi. mDNS/DNS-SD is a protocol for service discovery in a local area network. It is standardized under RFCs 6762 [1] and 6763 [2]. The protocol is also known by the *Bonjour* trademark by Apple, or *Zeroconf*. On Linux, it is implemented in the *avahi* package.


About mDNS/DNS-SD

There are several freely available implementations of mDNS/DNS-SD:

1. **avahi** – Linux implementation ([http://www.avahi.org/](http://www.avahi.org/))
3. **Bonjour** – MAC OS (installed by default)

During this course, we will use only **avahi**. However, any of the aforementioned implementations are compatible.

Avahi installation

**avahi** is available as a package for Raspbian. Install it with:

```
sudo apt-get install avahi-daemon avahi-utils
```

Avahi usage

**avahi-daemon** is the main process that takes care of proper operation of the protocol. It takes care of any configuration of the interfaces and network messaging. A user can control the deamon with command line utilities, or via D-Bus. In this document, we will describe the former option. For the latter one, please see [http://www.avahi.org/wiki/Bindings](http://www.avahi.org/wiki/Bindings).
Publishing services

`avahi-publish-service` is the command for publishing services. The syntax is:

```
avahi-publish-service SERVICE-NAME APPLICATION-PROTOCOL._TRANSPORT-PROTOCOL PORT "DESCRIPTION" --sub SUBPROTOCOL
```

For instance, the command:

```
avahi-publish-service light _coap._udp 5683 "/mylight" --sub _floor1._sub._coap._udp
```

will publish a service named 'light', which uses the CoAP protocol over UDP on port 5683. The same service will be discoverable under "_coap._udp.local" and "_floor1._sub._coap._udp.local".

Note: upon issuing `avahi-publish-service`, if successful, the process will remain active in the foreground. If it is killed (e.g. by issuing Control+C), it will stop publishing the advertisement. Therefore, when publishing such services, either run them in a separate console, or start them in the background (add & at the end). Don’t forget to kill any stale service advertisements!

Discovering services

Services can be discovered using the `avahi-browse` command. The syntax is:

```
avahi-browse [options] service-type
```

where some interesting options are:

- `-a`: browse for all service types, not just the ones listed.
- `-r`: resolve services automatically.
- `-p`: display output in parsable format.
- `-t`: terminate immediately after doing one discovery round. By default, `avahi-browse` remains active and keeps looking for matching services.

Examples:

1. Discover all CoAP servers and automatically resolve them:
   ```
   $ avahi-browse -rt _coap._udp
   +   eth1 IPv6 light
   _coap._udp      local
   ```
Discover the CoAP server at floor1, and show it in parsable format:

$ avahi-browse -rtp _floor1._sub._coap._udp
+;eth1;IPv6;light;_coap._udp;local
+;eth1;IPv6;light;_coap._udp;local
+;eth0;IPv6;light;_coap._udp;local
+;eth0;IPv4;light;_coap._udp;local
=;eth1;IPv6;light;_coap._udp;local;instant-contiki.local;fe80::20c:29ff:fe79:e439;5683;""/mylight"
=;eth0;IPv6;light;_coap._udp;local;instant-contiki.local;fe80::20c:29ff:fe79:e42f;5683;""/mylight"
=;eth1;IPv4;light;_coap._udp;local;instant-contiki.local;192.168.2.64;5683;""/mylight"
=;eth0;IPv4;light;_coap._udp;local;instant-contiki.local;192.168.10.130;5683;""/mylight""