Zeroconf and UPnP techniques

Petri Palmila
Helsinki University of Technology
ppalmila@cc.hut.fi

Abstract

All the time we are buying new devices which are able to talk to each other within a network. However, to build up the network we need somebody who will add needed configuration or technique which will do it automatically. Because building the network is complicated and too difficult for most people there is a need for automatic connecting technology. Zeroconf and UPnP -techniques offer easy way to connect electronic devices on a network without any user configuration. The Zeroconf and the UPnP offer automatic network configuration by a different way but the result is the same. The purpose of this paper is to offer comparison of the Zeroconf and the UPnP which are the most popular automatic configuration techniques. Advantages and disadvantages of both techniques will be presented and each of these will be compared those of techniques.

KEYWORDS: Zeroconf, Bonjour, Rendezvous, Avahi, UPnP, DLNA

1 Introduction

Configuration of network devices seamlessly and without any user configuration is a principle of the Zeroconf and the UPnP techniques. The Zeroconf has several implementations and probably the most known is Bonjour that is product name of Apple Inc. [1]. Some of the Zeroconf implementations are open source software which offers some advantages against the UPnP technique. However, the Bonjour implementation is not open source software and that can be disadvantage for it. The UPnP(Universal Plug and Play)-implementations come from different manufacturers of electronic devices. These companies are joined to DLNA (Digital Living Network Alliance) alliance in order to be come more conspicuousness [5]. Unfortunately, this kind of splitting to different technique is normal in telecommunication area. It just takes time to see that which one will be the winner of the race. The aim of this paper is to present difference of these implementations and compare which one is more suitable for different environments. The paper is divided into the following parts. The first part introduces the Zeroconf technology and how it works. The Zeroconf part is based to Apple's Bonjour/Rendezvous implementation and documentation [6]. Then comes the part Zeroconf in details which introduce the Zeroconf more deeply. The third part introduces the UPnP technology and how it works. The UPnP in details will be presented after the UPnP part. The fourth part will compare the Zeroconf and the UPnP to the each other. After technical comparison of the Zeroconf and the UPnP the paper will focus on future of the Zeroconf and the UPnP. The last part of the paper is conclusion.

2 Zeroconf

The Zeroconf is technique which will create ip network without any manual configuration and support services for partners. The network is build by using three main tasks. The first, it will manage numeric network addresses for networked devices. The second, it will handle host names for devices. The third, it will offer and search services for devices. The Zeroconf technique can be offered using several implementations. Probably the most known is Apple's Bonjour. Before it was called Rendezvous but nowadays that the concept has disappeared totally because the name was registered by the other company [5]. For example, other implementations are Avahi that is for Linux and BSDs, LLMNR that is a solution of Microsoft for Windows CE. The Avahi is a free software implementation of the Zeroconf.

The most visible part of the Zeroconf is service discovery. The Zeroconf can be divided into three more technical part which are link-local addressing, naming which is done using multicast DNS and DNS service discovery [1]. The link-local addressing part is used to find IP address if DHCP server is not available. The Link-local addressing for IPv4 is described in RFC 3927. Multicast DNS is used when there is no working DNS server available. The last one DNS service discovery is used for browsing services if there is no neighbor who can offer that information [3]. One of the main benefits for end user is possibility to browse services easily. For example, in the computer world you have to get first hardware and after that configure it correctly before printing. The Zeroconf is based to the idea where the end user browses services which are able to print without configuring hardware or knowing hardware. The main idea in the Zeroconf is ease of use. There is no need to open any terminal window and perform any administrator's commands.

The Zeroconf is able to get its IPv4 address by using ARP requests if there is no DHCP server available. The ARP mechanism is described in RFC 826. The Zeroconf supports IP version 6 but there are still lot of peripherals which are not able to use that [1].

2.1 Zeroconf in details

The Zeroconf protocol can be divided in to the following part: Dynamic Configuration of IPv4 Link-Local Addresses [IPv4LL], Multicast DNS [mDNS], DNS-Based Ser-

vice Discovery [DNS-SD]. The Zeroconf include support also for wide area networks. For the WAN purpose the Zeroconf have parts: DNS Long-Lived Queries [DNS-LLQ], Dynamic DNS Update Leases [DNS-UL] and NAT Port Mapping Protocol [NAT-PMP].

The dynamic configuration of IPv4 link-local addresses allows device to specify its ipv4 address within network prefix 169.254/16 [6]. That prefix allows just local communications between devices. The multicast DNS handle the cases where devices start to send DNS queries to a multicast address. Format of the multicast DNS message is almost as in unicast DNS message. For example there are different packet size and in port number. The multicast message uses port 5353 instead of port 53 and packet size is limited to 9000 bytes instead of 512 bytes.

The DNS multicast uses only UTF-8 coding in information DNS-Based service discovery allows devices to discover for services by using standard DNS queries. The service discovery is asynchronous operation. In asynchronous operation speed of response for the request is not a constant. For example, it can vary from a second to a day.

The DNS long-lived queries makes possible to search services all the time. It is very useful because using that finding new services can happen faster than without it. The Dynamic DNS update leases can be used for removing old information of available services.

As said before the following mechanisms are for WAN purposes. The service discovery has own definition for the WAN. The DNS-SD is based on standard secure DNS Dynamic Update which is defined in RFC2136.

Request and response for the DNS-UL is described in RFC 2136 [6]. The RFC document proposes a method to update contain of resource records and allowing a server to garbage collect stale resource records.

The LLQ setup consists of four way handshake. The first step is initial request which is sent from client to server. The initial request is based to standard of DNS query with LLQ extensions. The second step is challenge which is a response for initial request. The third step is challenge response where the client tells that it received the challenge message successfully.

The final handshake consist acknowledgement from the server and the client's answer for it. The NAT port mapping protocol allow to create network address translation between public and private addresses. The process also takes care of port numbers and it makes possible that communication between services are working after address translation [1].

For the programmers the Zeroconf offer a possibility to program operations by using (Application Programming Interface) APIs. The programming can be done by using several different programming languages.

3 UPnP

The UPnP consists of six layers, which are addressing, discovering, description, controlling, eventing and presentation (Fig. 1). The addressing layer is the same as in the Zeroconf technique although the structure is different. The link-local addressing in the UPnP is based to draft of the RFC 3927 but it is compatible with the final version. The addressing is used to get ip address and it works without any special services for example dhcp server. The description layer offers list from devices to other equipments in the network. The rest of layers will be presented in the UPnP in the Details chapter.

The UPnP has no naming layer. It means that hostnames can't be used instead of ip addresses.

Architecture of the UPnP device consist device, services and control points. There can be zero or more services in the device and more services offer more functionality for the device. The Service can be divided into the part methods and state variables which have optional values for different purposes. For example, input, output or return values. All services are specified by the UPnP Forum working committee which also defines details for services, including parameters.

The control point works by using service which it got from a device. The control point can send requests for devices. The UPnP technique allows to build a device where is the control point inside. That kind of configuration allows to the device starts its negotiation with each others. The Control Point inside an UPnP Device (Fig. 1).

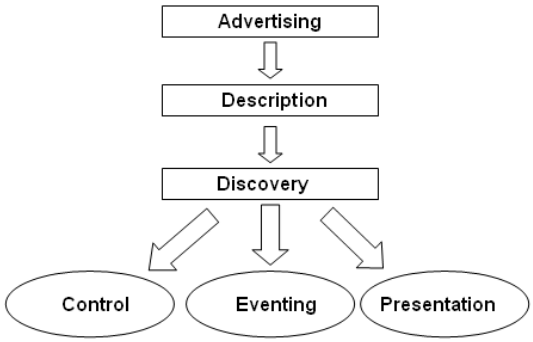

Figure 1: The UPnP Phases

3.1 UPnP in details

The UPnP device supports two kind of addressing protocols. The one is DHCP and another is dynamic configuration of link-local addresses, which is called auto-IP and every the UPnP device includes built-in dhcp-client. The DHCP protocol offers three different mechanisms to offer ip address for client. These mechanics are automatic allocation, manual allocation and dynamic allocation. The UPnP is using dynamic allocation because it does not need any administrative tasks. It gives an address for a limited period of time and automatic allocation support only permanent ip addresses, which is difference to the dynamic allocation. Client has to lease ip address for limited time and if needed the client can renew the lease. If the client does not need ip address any more then it is possible to release the lease. After releasing the dhcp server is allowed to give the address to another host. If the DHCP is not available for a UPnP device then the device starts auto-ip procedure. Normally this kind of situation happens in home because the dhcp-server needs someone who is able to manage it. In auto-ip mechanism the UPnP device starts to select a candidate address by using address space which is non-routable in gateways. There are

several address spaces which are for internal use only but the UPnP is using address space 169.254/16. The UPnP device will also configure subnet mask for configuration.

The UPnP device performs check for its ip candidate to be sure that the address is not in use. The check will be done by sending a ARP request for the address and if there is no response coming then the address is free. This kind of way to use the ARP is unusual but effective way to get a free address. Auto-ip mechanism allows building up connection between UPnP devices. There is also limitation for effective auto-ip mechanism and that is the non-routable address space. The auto-ip supports address which can not be used outside of local network segment. Probably the limitation is not relevant for some users and devices but anyway it should be known [7]. The UPnP device is always trying to get ip address first from dhcp server. If it does not succeed then the device will start auto-ip mechanism. The device will try to find a dhcp server again after a period. The default for re-check is 5 minutes for Ethernet implementations. If the UPnP device can find the dhcp server, then it starts to change address procedure. In the procedure the address can be changed only if all connections of the old address can be closed.

The UPnP device is able to perform discovery procedure after getting working ip address for the network. In discovery state the UPnP device is able to advertise its own services and discover available services from other devices. The main idea in the description is that it allows sharing of services between devices.

Simple Service Discovery (SSDP) is designed to be a simple discovery service for HTTP based resources [4]. It offers two way to discover service, service type and Unique Service Name (USN) which contains Universal Unique Identifier (UUID). The USN is a URL that is used to discover instance of service. The UUID is a 128-bit number which identified the target object.

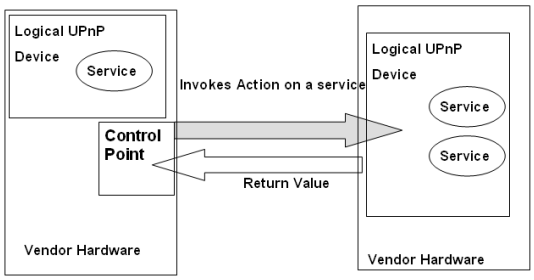

Figure 2: Control Point in a UPnP Device

The description layer is divided to eight parts which are device description, UPnP device template, service description, UPnP service template, non-standard vendor extensions, UPnP template for language for devices, UPnP template for language for services and retrieving a description. Control point collect description document from the UPnP device by using HTTP GET requests. The connection from control point to the another device (Fig. 2). The GET request includes URL, ip address, optional port and a language which is preferred by control point. The device sends responses to the request by sending device description. The answer should arrive to control point within 30 seconds from request. After getting the device description the control point is able convert service description request. Flow of request and response messages in service description (Fig. 3).

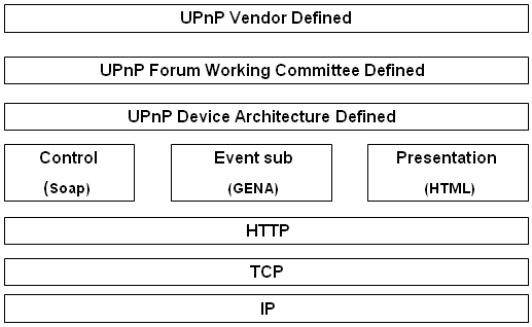

Figure 3: Control Point invoking an Action

The control point knows that device and service descriptions are valid because there is discovery advertising because of that. The UPnP device is able to change its description if needed and the change will be done with advertisement.

The device and the service description documents are based in xml-format and there is information for example of manufacturer, model of equipment, serial number, list of provided services. Every service in UPnP device description documents contains own row which shows URL of the service [8]. The control layer allows the control point to control services which are provided by the UPnP device. Control protocol that is in use in control layer is Simple Object Access Protocol (SOAP). For the developer the SOAP gives just information of control messages because the SOAP is implemented in SDK. The location of the SOAP in the UPnP architecture (Fig. 4).

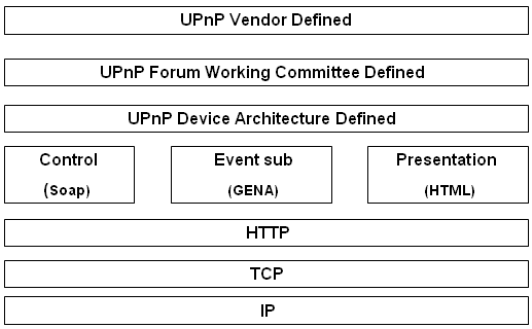

Figure 4: The UPnP Architecture Protocol Stack

The SOAP uses HTTP and XML protocols. The HTTP protocol is used for transfer purposes and XML is used for presentation of messages. Because of powerful XML format messages can be very complex which makes the SOAP more useful. The SOAP message is sent over the HTTP by using HTTP POST which includes type of content text and xml. The HTTP request includes SOAP Action header and the HTTP response includes xml documents where are envelope and body parts if the response was success. Success of response can be seen from HTTP code where range 2xx is for success. Code range of 3xx is used for redirection,

range 4xx for client error and range 5xx for server error purposes. These codes are standard codes from HTTP protocol [9]. The control point uses SOAP messages for controlling the UPnP device service. The SOAP sends control messages to the URL of each the service. In the case of action request, the control point sends request for service URL in action request format [4].

The query state variable can be used to make direct query for the service. With the query state variable control point is able to get single state variable from the service. The UPnP forums recommend using query state variable only for test purposes. The Query state variable will disappear from new version of UPnP devices [9]. The control point can keep track of the state of devices. Device will send a information of its new state only to subscribers through GENA.

The UPnP device include embedded web server because of presentation which is done by the SOAP based control messages and by GENA event messages. Presentation page of device can be get by using the HTTP GET request and the device will return it inside body of the HTTP response. Anyway, vendor of device is not required to offer device presentation page because it is optional. The presentation page is not standardized by the UPnP forum [4].

For multicast communication the UPnP architecture supports simultaneously communication. Discovery mechanism uses the communication to send messages to all devices on the network. The UPnP architecture uses HTTP which is sent by using UDP protocol. That protocol is called as HTTPMU. Devices will response to the request by sending directly message to the source. That protocol is called as HTTPU and it is based also to HTTP over UDP protocol. The location of the HTTPMU and the HTTPU in the UPnP architecture (Fig. 5).

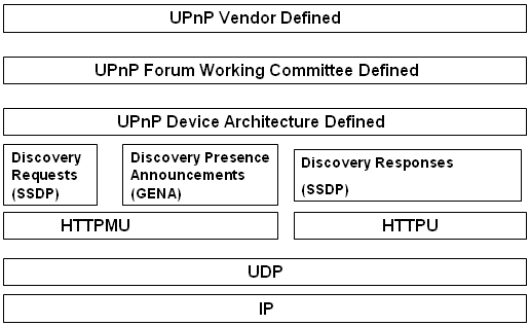

Figure 5: HTTPMU and HTTPU Protocol Stack

4 Comparison of Zeroconf and UPnP

The addressing layer of the Zeroconf has been defined by IETF Zeroconf Working Group. The group has defined RFC 3927 for the IPV4 Link-Local addressing. Because of that the addressing layer in the Zeroconf is defined officially contrast to the UPnP, where the addressing layer is based to early draft of the RPC. Anyway both are using the same link local addressing although the UPnP first started with the draft RFC. The naming Layer is working in the Zeroconf correctly and that supports translation of ip addresses to host names. In the UPnP there is no the same kind of multicast DNS service as in the Zeroconf [10]. Because of that hostnames can't be used instead of ip addresses.

For users the case can be difficult because ip addresses are randomly-assigned IPv4 Link-Local Addresses. Therefore you don't know what random address a device has picked for itself. The situation is much better in the Zeroconf than in the UPnP.

The Simple Service Discovery Protocol (SSDP) of the UPnP is still under development and it does not offer as simple service as Service Discovery (SD) offer in the Zeroconf. The SSDP needs more investigations and solutions to ensure that the traffic does not cause problem. Development of the UPnP's discovery was based to the draft RFC which was cancelled in October 1999 that is the main reason why the discovery is not dependable. In the Zeroconf the discovery is managed without troubles.

The application layer of the UPnP is needed to development for every a new device if existing application-layer protocol does not offer needed functions. Then a new application protocol needs own committee to manage the problem. The procedure ensures that implementations for the application layer will be standardized. In the Zeroconf implementation of the application layer is free for new development. Of course, the way offers easier and faster development process of new application but it lead to situation where application layers can be incompatible to others. The situation cause some risk to application development and because of that the way of the UPnP is better. The Zeroconf is based to three layer reliable ip networking which supports different application protocols and it allows to build whatever application layer protocol.

The UPnP will come more and more complicated because new devices with services will come all the time. There some serious problems with handling different layers and the correction for problems should be available as soon as possible to ensure bigger problems. The Zeroconf seems to be more clear and its layers are development with RFC without mistakes [11]. Support for wide area discovery is missing from the UPnP. However, it is included in the Zeroconf. That offers more possibilities to use. For example, services of iTunes music device can be used from far away. Of course, for home network the WAN discovery support can be needless but if you can get without extra charge then it cannot be a disadvantage.

5 Future of Zeroconf and UPnP

It seems that some layers are the same in the both technique. For example, it creates a possibility to connect the Zeroconf and the UPnP device to each other but just in low level. Unfortunately, it is not enough for the services that the addressing is working and probably discovery. There should be support for the UPnP layers in the Zeroconf technique or conversely. That could probably create new possibilities for the Zeroconf technique and strengthen its position in the race with the UPnP. For example the Zeroconf support lot of services which are presented in a list in (http://www.dns-sd.org/ServiceTypes.html) and there is the UPnP as a service but it seems that it is still under development [2]. The sup-

port of the UPnP in the Zeroconf could cause complexity and therefore it can be easier to support the Zeroconf in the UPnP. Alliance of the UPnP technique is very plausible. The most known manufacturers of electronic devices without the Apple are offering the UPnP within their own devices to the world wide market. Every device where the UPnP will be needs the first forum where the implementation will be negotiated.

The UPnP's link-local addressing layer is identical in the Zeroconf. Because of that some kind support for the both is implemented in every device which supports the Zeroconf or the UPnP. For this reason, it seems that all the time more and more devices are equipped with the UPnP technique than with the Zeroconf. But the number of manufacturer does not tell the truth. The Zeroconf is more suitable in wider are than the UPnP. For example, the Zeroconf is implemented almost in all printers in the world. The UPnP is based more complicated configuration and some devices has own stack which should be used with that device. There are some weaknesses in the both technique. In the UPnP technique there is need for special development almost with all new devices. Of course, that is some kind of weakness. Although, using that way it gives more special service possibilities for interface of the device[11].

The Apple is offering its famous music and video players with the Zeroconf and that is the second big area where the Zeroconf is in use after printers [6].

6 Conclusion

The both of technique's target is the same, to get a easier way to build IP network between devices. The Zeroconf is open source software but the Apple's own solution the Bonjour is not open source software. The Avahi implementation of the Zeroconf is totally free but it has limited support for different operating systems. Anyway, it supports easier way to implement a new device with a common technique than the UPnP which will start a new forum to handle implementation [6]. The race between the Zeroconf and the UPnP will continue.

The Zeroconf and the UPnP are focus a partly to different area. The Zeroconf is aimed to a horizontal direction and it is trying to be suitable for all IP-based networking protocols. The UPnP technique is going more to a vertical direction and it is aimed more at solving device specific vertical problems. The UPnP have limited solutions for all IP-based networking protocols and that is some kind of disadvantage.

It is usual that different companies offer different technique and does not give up to the others. The Apple is the company which does not give up to the UPnP. Behind the UPnP there is the alliance of 550 companies and these companies are very powerful together. The Zeroconf and the Apple are strong names in USA but there are also other countries were the situation is not the same. The UPnP does not offer the easiest way to implement a new device but the strong alliance does not give in the Zeroconf technique. The Avahi is one of the most interesting implementation because it offer free implementation of the Zeroconf but the solution is mainly for Linux operating system but it runs also in BSD, Solaris and Mac. Unfortunately, support for Windows OS is missing. It very difficult to know what will be happened but the time will show that. Several implementations are good in some area but convergence of different technique is still needed.

References

[1] Stuart Cheshire, Daniel H. Steinberg, *Zero Configuration Networking, the Definitive Guide* O'Reilly, 2005.

[2] RFC 2782. *D*NS SRV (RFC 2782) Service Types. `http://www.dns-sd.org/ServiceTypes.html`

[3] Zero Configuration Networking. *A*utoconfiguration for IP Networking. `http://www.zeroconf.org/w3onwire-zeroconf.pdf`

[4] Michael Jeronimo, Jack Weast *UPnP, Design by Example, A Software Developer's Guide to Universal Plug and Play* Intel Press, 2005.

[5] Digital Living Network Alliance. *D*LNA overview. `http://www.dlna.org/en/consumer/learn/overview/`

[6] Apple Ltd. *N*etworking Bonjour. `http://developer.apple.com/networking/bonjour/`

[7] UPnP Forum. *U*PnP Device Architecture , 2003. `http://www.upnp.org/resources/documents/CleanUPnPDA101-20031202s.pdf`

[8] UPnP Forum. *U*PnP Device Architecture , 2000. `http://www.upnp.org/download/UPnPDA10_20000613.htm`

[9] UPnP Forum. *U*PNP Vendor Implementation Guide, 2001. `http://www.upnp.org/download/UPnP_Vendor_Implementation_Guide_Jan2001.htm`

[10] UPnP Forum. *U*nderstanding UPNP. `http://www.upnp.org/download/UPNP_UnderstandingUPNP.doc`

[11] Zero Configuration Networking, Stuart Cheshire. *How does Zeroconf compare with Viiv/DLNA/DHWG/UPnP?*, 2004. `http://www.zeroconf.org/ZeroconfAndUPnP.html`