Universal Plug and Play Protocol

Presented by
Mr. Swapnil Suryakant Patil
Roll no: 305128 Class: B
Under Guidance of
Prof. S. S. Pawar

Introduction

- Architecture developed by **Microsoft**
- It was created to integrate computers connected to the network for sharing purpose
- Now used to integrate all sorts of devices such as Smart televisions, refrigerators, IGD (Internet gateway devices), Smartphones, printers.
- These devices can talk to each other all **automatically** as we will discuss shortly

Alternate Technologies

- **Internet Of Things** : It is great achievement for industry level automation methods
- **Miracast (Chromecast)** : Supports easy **mirroring functionality** with wireless communication support
- **RFID** : It has two components **receiver and transmitter** with an active or passive tag
- **NAS (Network Attached Server)** : A local server that stores all kind of files to share among different network devices
- **Cloud** : Three types of service **SAAS, PAAS, IAAS**

Alternate Technologies

Technology	Advantages	Disadvantages
UPNP	A pure P2P networking architecture with large no of applications	Security and Power dissipation
IOT	Ad hoc network for embedded devices	Suitable Only for embedded devices Scalability is not possible Also UPNP+ is emerging technology for IOT
Cloud based networking	Useful when network is mobile and not local	Bandwidth Limitations for simpler applications UPNP can be also used in cloud networking
Chromecast	Easier to use	Cost Scalability
RFID	Also used for embedded devices	Outdated , Absolute
Physical means such as pen drives ,hard disks	No need of networking	Not feasible for automation

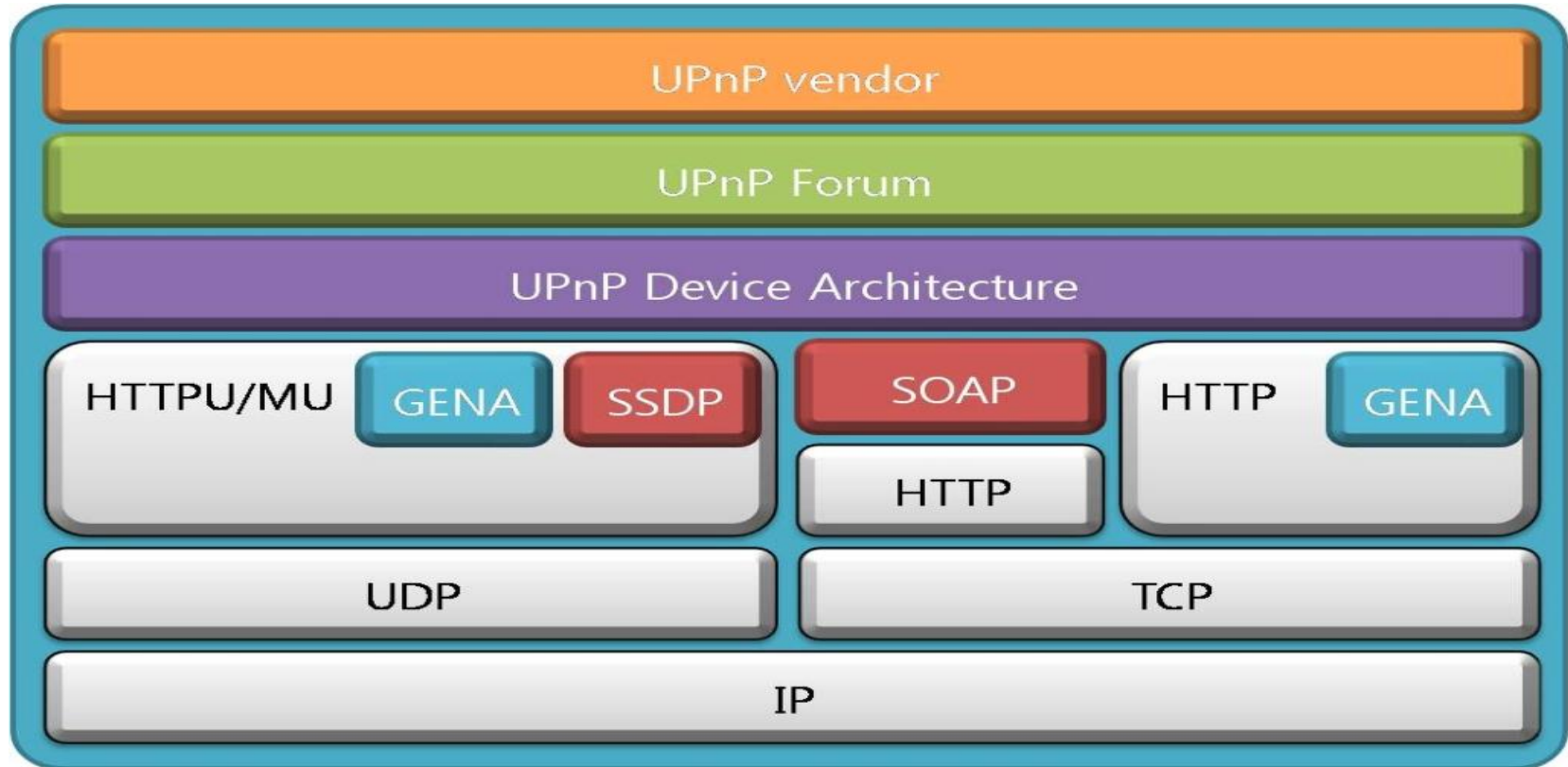# "Interpolability" and "Configurability"

UPNP supports Interpolability

And it is "ZeroConf"

Architecture.

The Main Question

How does it work then?

UPNP Architecture

UPNP Overview

- **UPNP** is a simple protocol as it uses **TCP/IP suite** as its underlying layer as shown in previous dig.
- Due to this fact, It makes UPNP a ***robust architecture***
- **HTTP** : This protocol is used to get the files that are essential for UPNP to work such as XML file that lists various services provided by the device node
- **SSDP** : This protocol is heart of UPNP , Device Discovery is the main responsibility of this protocol
- **UDP for Multicasting** : UPNP uses UDP for communication and discovery of all the devices connected to the network with the help of **HTTPMU**.

UPNP Overview

- **GENA (Generic Event Notification Architecture) :** A Simple *subscribe and notify* protocol which uses HTTP as a backend, This makes UPNP capable devices to subscribe to different events and get notifications when they occur
- **SOAP (Simple Object Access Protocol):** provides **ability to exchange parameters** that are required by actions or commands executed on different device nodes
- **HTTPU/HTTPMU (HTTP with UDP multicasting):** variants of http having support for **udp multicasting**

Step One:
Addressing

Every device needs a ip address to communicate:

1. By means of **DHCP** (Dynamic Host Configuration Protocol)

2. If DHCP is not available it **link local address range** is chosen (169.254.0.0/16)

Step Two:

Device Discovery

Discover available devices on the network that supports UPNP:

1. **SSDP** Does this job

2. It sends a **udp multicast** to 239.255.255.250 on port 1900

3. All the available devices responds this message about their profile to the device sending discovery msg

Step Three:

Description

Every device maintains its **profile** in a **XML file**.

1. XML file contains **device info** such as **services** it provide, **icons**, **event subscription urls** and lot more.
2. **HTTP** transfers this XML file to **control point** (device which sends discovery msg)
3. Now the **CP will know every device**

Step Four:

Control

The services available on various devices can be accessed in this step:

1. **Services** such as port forwarding, music and video streaming etc.

2. **Requesting a service** can be done by **SOAP** with sending **correct parameters** as **RPC** mechanism with **XML**.

3. Req is sent to the control url in **<service>** tag of device profile

Step Five:

Eventing

The events such as playback aborted is notified to CP's or subscribed devices for that event.

1. **GENA** is handy for eventing job.
2. Concept of **state variables**
3. These variables get changed upon **occurrence** of particular events and are **notified** to **subscribed devices** for that event.

Advantages
Great features of
UPNP

- Platform Independent (Interpolable)
- Simplicity as TCP/IP suite used as a backend
- No usage overhead i.e User is free from technicality and overall complexity known as "Autoconfig" or "Zeroconf"
- Home Automation is possible due to services available such as Coffee machine makes coffee and when it is ready user is notified by Eventing capability of UPNP
- Large developer community ex. OCF and UPNP forum
- Evolving nature i.e. UPNP+ is developing for efficient sensor systems that will save power.

Disadvantages
Ok!! great features
But ....

- Power inefficiency (Solution UPNP+)
- Security : It is major drawback of UPNP and many developers and researchers are trying to make UPNP more secure

A Case Study (Application)

The Master Switch:

Mr. Alan arrive at home tired of all day's work. Most of us will turn on the light, put some breakfast in the oven and so on. But with the help of UPNP Mr. Alan has a master switch that can automate these things. The Switch is simply a UPNP service defined by a state variable and when the variable changes, preset scripts will run on a pc device…

A Case Study (Application)

Now the Script will request the refrigerator to fill a cold glass of water, a smart tv will turn on and will play a favourite music, HVAC system will be turned on to maintain the home environment and so on all easily possible due to UPNP.

Conclusion

UPNP has some great features that will make it one day a industry standard for home automation, But security is major setback to this architecture that will hinder its usage.

UPNP has a wide scope of applications, research areas and a large supporting community for its future development and I intend to contribute for evolution of this amazing protocol.

Thank You!!