Case Study - IPv6 based building automation solution integration into an IPv4 Network Service Provider infrastructure

Nikolay Milovanov, Ivan Bogomilov

***Abstract**: The case study presents a case study describing an Internet Protocol (IP) version 6 (v6) introduction to an IPv4 Internet Service Provider (ISP) network infrastructure. The case study driver is an ISP willing to introduce a new "killer" service related to Internet of Things (IoT) style building automation. The provider and cooperation of third party companies specialized in building automation will provide the service. The ISP has to deliver the network access layer and to accommodate the building automation solution traffic throughout its network infrastructure. The third party companies are system integrators and building automation solution vendors. IPv6 is suitable for such solutions due to the following reasons. The operator can't accommodate large number of IPv4 embedded devices in its current network due to the lack of address space and the fact that many of those will need clear 2 way IP communication channel.*

The Authors propose a strategy for IPv6 introduction into operator infrastructure based on the current network architecture present service portfolio and several transition mechanisms. The strategy has been applied in laboratory with setup close enough to the current operator's network. The criterion for a successful experiment is full two-way IPv6 application layer connectivity between the IPv6 server and the IPv6 Internet of Things (IoT) cloud.

Keywords: Internet of Things, IPv6, embedded systems, building automation

INTRODUCTION

The current global network - Internet has stormed through the word of telecommunications from 1983 when it was born until now. All those years it has been based on IPv4. It is the building block of the current Internet. It had great success and really pushed the industry during the last 30 years. Nowadays almost anybody's life is linked to Internet and IP network services.

Despite that success, IPv4 is not perfect and has its limitations. Most of those have been overcome during the years except one - the limited address space. Despite all the efforts including private IP addresses and various kinds of network address translation the last /8 IPv4 address blocks are already allocated [1]. In [2] is written "*To support the large number of emerging applications for smart objects, the underlying networking technology must be inherently scalable, interoperable and have a solid standardization base to support future innovation as the application space grows*." It can be stated that the TCP/IP model is the only communication technology that could be described as end-to-end, stable, open, lightweight, scalable, versatile and well standardized. The smart objects could be described as Internet of Things cloud. As per [3] Internet of Things refers to "*uniquely identifiable objects (things) and their virtual representations in an Internet-like structure*".

The current Internet is IPv4 based and IPv4 has already an address shortage. From here could be concluded that the lack of more real IPv4 addresses is one of the burdens for future "Internet of Things" growth.

That impediment has became the reason why the client – well known Internet Service Provider is looking for a long term strategy how an IoT based services and solutions could be integrated in its current network and be part of future service offerings.

The Service Provider has already an established market position and a large number of residential and also business customers.

IoT building automation solution offering will complement well its current value added services portfolio and will give him a competitive advantage among its rivals. The operator has realized that it can't step up directly on that market since his personal does not have the needed knowledge and is trying to provide an integrated solution with a third party companies.

Building automation solution providers are willing to provide managed services to their customers. The managed services solution is based on a common management platform shared between many customers and reliable two-way network communication channel between the platform and the IoT cloud. The platform might be part of a public or private service cloud (Fig.1).

BUILDING AUTOMATION NETWORK ARCHITECTURE

IoT based building automation solutions extnend the network access layer by adding new "micro" access layer into the customer's building. That layer is usually introduced through a separate access device called home gateway. The current vision is that the home gateway will be placed between the current CPE and the IoT sensors cloud. However there might be also another solution in which the home gateway may merge with the CPE. If this happens arrise the interesting question who will manage the CPE/Home Gateway. Will that be the network operator or the building automation solution provider? The home gateway has to support different communication standards from the current home CPE. Examples for such are Zigbee [18], UPnP [19], KNX [20]. They can be TCP/IP based or based on another transport stack. The IoT clould has to be able to communicate with the management service cloud. The clould could be private (part of the ISP Service Zones) or public (part of the global Internet).

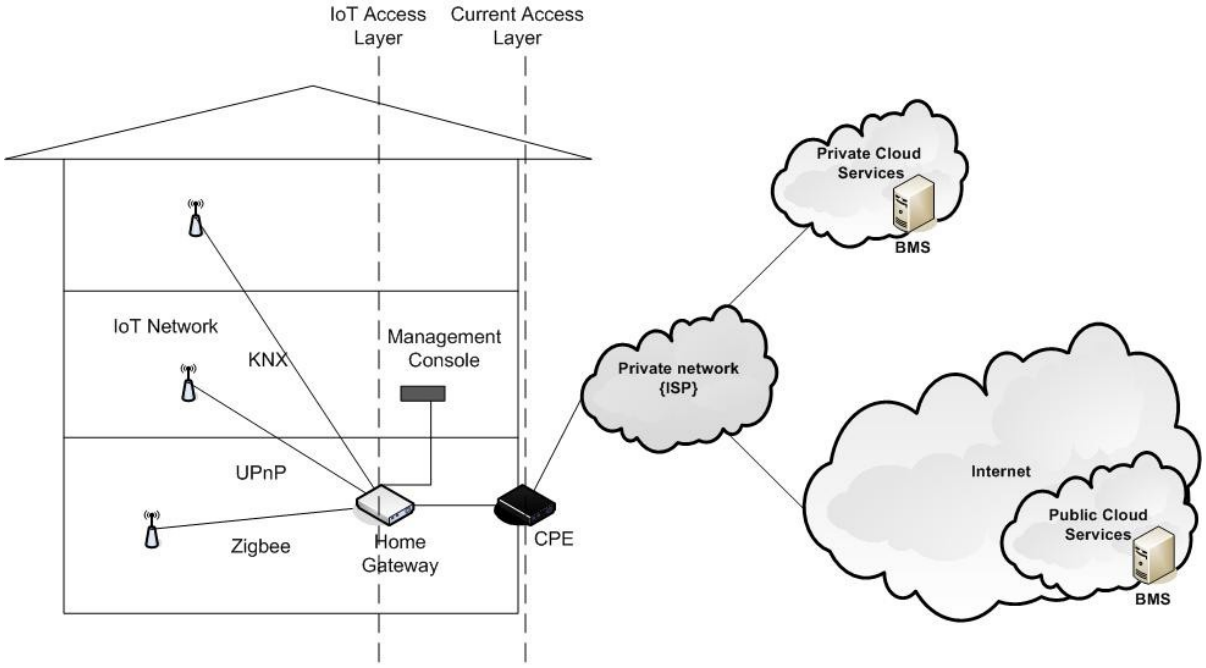

Figure 1. Network architecture

The case study is focused only home automation management platfrom placed in a private clould. This scenario is considered as the most interesting one from ISP point of view.

The traffic flow between the IoT sensors and the service cloud at the moment when the case study has been performed was still based on proprietary Representational state transfer (REST) protocol. It has been build on top of the current level 7 Hyper Text Transfer Protocol (HTTP). Therefore we consider it as layer 7+ protocol. It is important to

be noted that HTTP is the protocol caring most of the traffic in the current Internet. Despite the fact that the IP layer will change from IPv4 to IPv6 and most likely this will require an introduction of a new TCP/IP stack on operation system level no change is expected to happen on HTTP and above layers. Most likely this is the reason why the embedded system manufacturers considered a solution based on REST. In IPv4 to IPv6 context it is likely other manufacturers in other business domains to choose a similar approach. Now in the embedded systems domain are emerging new standards (still in draft stage) describing exactly the same topic [17]. UPnP is already following that architecture model, KNX also and Zigbee is going into that direction.

QUALITY OF SERVICE CONSIDERATIONS

The traffic flow between the BMS and the IoT cloud could be considered as a higly valuable one for the building automation solution customers. Through the BMS they will know what is happening in their houses and will be able to interact with the solution. The Internet Service Provider can adapt its current Quality of Service policy in order to accommodate the new traffic flow.

From network perspective the traffic between the Sensor cloud and the Service Cloud could be considered as symetric 2-way HTTP communication flow between the home gateway and the BMS.

Modern network operators have relatively complex QoS policies based on differentiated services markings different queuing and scheduling mechanisms for the different kinds of traffic and MPLS traffic engineering. In our case the operator had a similar kind of policy. It could be described as 6 different traffic levels.

VOICE – % of the traffic stripped from the layer-2 interface bandwidth and serviced by the priority queue

DATA – Calculated as % of the remaining Layer 2 Interface bandwidth – VOICE. This traffic has been serviced by a Class-Based Weighted Fair Queuing (CBWFQ) mechanism. Each queue has been serviced by a Weighted Random Early Detection (WRED) mechanism. The following traffic classes were identified. It is important to note that they are ordered by their priority.

Network Protocol Traffic – 10% of the DATA. Used for all caring the control traffic from Internet protocols marked with IP precedence 6).

Business1 – Caring the traffic from critical business applications.

Business2 – Caring the traffic from the not so critical business applications.

Streaming - Caring the traffic from all streaming applications.

Internet – Internet Access traffic

Other – Junk (traffic from torrents and other p2p applications).

Based on that QoS map our proposal was the traffic from the building automation to be placed in a new CBWFQ queue with priority higher then the Business1 traffic and lower then the Network Protocol Traffic. In the end if certain fire alarm has not been signalized properly and the office building burns down the business traffic might not be so important.

IPv4 TO IPv6 TRANSITION MECHANISMS

There are many methods for IPv6 introduction in an IPv4 network. In IoT there will be many devices with limited resources that could be considered from network perspective as end hosts and resource rich network that will contain many kinds of equipment as switches, routers and servers. The devices have to be identifiable through a unique network address. The resource and feature rich network is usually a nice to have advantage but actually is quite a challenge. Many of the popular features available for IPv4 are still just bullets in the roadmaps of the network equipment vendors. Despite that fact since the end hosts are the one with the most limited resources has to be considered a migration strategy suitable to the particular current network setup and the limited end

device capabilities. In IoT the uniquely identifiable objects could be split to three major groups.

Objects with an IPv4 network address - such are all embedded systems part of the current IPv4 Internet. Examples are most of the home routers, mobile phones and many other IPv4 enabled devices. For those that have to join an IPv6 based Internet of Things several approaches are possible. If the embedded device resources are enough for handling the new IPv6 TCP/IP and there is a new firmware the easiest possible way will be a firmware upgrade to a version that has IPv6. For those devices that do not offer such opportunity could be used 4to6 network address translation (NAT) into the core of operator's network or 6bed4 tunneling methodology [7]. The translation could be done with Carrier Grade NAT (CGN) [13] Network Address Translation/Protocol Translation (NAT-PT) [12] or NAT46 [14].

Objects with IPv6 network address only - this is the preferred case for the Internet of Things deployments. If those devices have to communicate with the current IPv4 Internet this could be achieved through NAT64/DNS64 [15],[16] or tunneling mechanism such as 6bed4.

Objects with IPv4 and IPv6 - dual stack ensures that each device will have connectivity in IPv4 and IPv6 domains. This approach brings additional complexity to embedded systems development process and requirements for more processor and memory resources.

IPv6 Automatic Address Assignment

IP address could be manually configured or automatically obtained. Automatic configuration is statefull and is performed through Dynamic Host Configuration Protocol (DHCP) client implementation in end host device and DHCP server implemented into the server/router side. In IPv6 the options are the same with one small but very important difference. Automatic configuration could happen in two ways - statefull (e.g. again through DHCP version 6 [12]) or stateless [11].

The stateless approach is useful when there is no particularly concern about the exact addresses that the hosts will use as long as they are unique and properly routable. On the other hand, DHCPv6 is used when a site requires tighter control over exact address assignments. Both stateless address auto-configuration and DHCPv6 may be used simultaneously. It is also important to mention that the DHCPv6 process will consume more resources and will increase the cost of the devices.

In Internet of Things the strict control won't be so important, manual configuration is impossible and the manufacturing cost of the device is a substantial factor. With those considerations in mind, the authors propose an IPv6 stateless mechanism implemented into the embedded device in combination with address information advertised by local routers for mass network deployment of IoT devices with limited resources.

STRATEGY

The goal of the authors was to develop an IPv6 introduction strategy in the context of the current network operator infrastructure and current and future planned service offerings. As per the requirements the strategy has to reflect the current network architecture and future growth plans. The strategy has to ensure smooth introduction of IPv6 and the customers should not experience any unplanned service outage. The strategy has to be executed during the maintenance windows mentioned in the Service Level Agreements (SLA) signed between the ISP and its customers.

Based on the analysis of the transition mechanisms and the options for address autoconfiguration our intention was to introduce a fully functional dual IP stack network infrastructure, IPv6 stateless autoconfiguartion for the IoT cloud of devices and NAT64/DNS64 in case any IPv6 traffic has to exit from the IoT cloud to the current IPv4 Internet. The IoT building automation system responsible for management of sensor

clouds is also dual stack based and accessible from the internal and external through IPv4 or IPv6.

In order to prove that the strategy is working our team performed an experiment in a laboratory environment. For the purpose operator's personal has prepared a setup with hardware and software similar or same as the one in their production network.

Experiment setup

The network laboratory has been built from several equipment vendors. Among them there are devices from CISCO Systems, Juniper Networks and Huawei Technologies. As Embedded systems have been used aj-200 development kits [8]. The kits were able to support IPv4, IPv6 or both. As services they were able to perform ICMP echo and echo reply and also to perform simple Representational State Transfer (REST) http GETs and POSTs. Those two were vital for having a realistic enough scenario and an ability to meet the criteria for successful test.

The experiment is focused only on the traffic flow between the IPv6 BMS into the private cloud and the embedded devices.

Experiment scenario

The experiment has been split to several steps:

Step 1: **Identify the initial network setup**. Since the network has been pre arranged our task was to try to reason about it based on the configurations created by the ISP personal and the network design guides that we possessed. Also we had to gather as detailed as possible information about the devices, modules, topology, IP address allocation, routing protocols and security settings. For the purpose our team has used a custom made SNMP discovery tool [4] able to discover the network topology and to gather the required inventory information.

Step 2: **Analyze and compare the network discovery results with the provided documentation**. On that step has been made a comparisson between the real network setup and the documentation. As a result many deviations were discovered. Some of the examples were:

- Software versions and hardware modules used on some of the lab devices were actually different (more or less left to default) from the one recommended by the vendors in the design guides.
- IP Address allocation rules were not strictly followed and deviated a lot from the one in the guides.
- Differences related to typos (wrong or misleading interface descriptions)
- Wrong policy rules for routing protocol-to-protocol redistribution.
- The recommended Security Best Practices were partially implemented.

The result of Step 2 audit was so disturbing for the Project Stakeholders that a separate unplanned audit step 2' was conducted against several sites of the real network. The results were not that much different then the one in the lab environment. So after 2' the goals of our team was not only to define a strategy for IPv6 network transformation and IP allocation rules for IPv6 embedded devices but also to inline the current network setup to the level described into the design documentation.

In order to achieve that in a large-scale network some form of network automation has to be used. The custom discovery tool was a good start but was not sufficient. We needed an automation solution also in direction device configuration and device operation system upgrades.

Step 3: – **Network alignment to the documentation Best Practices**. The allignment procedure has to ensure that all network devices are with the right operation system images and with configuration that is allined to the documentation best practices. During that step were developed custom scripts for device software verification and

upgrade. Also were developed scripts that check configuration for conformance with certain config templates.

Step 4: – **IPv6 network introduction**. Step 4 could be considered as the real start of the current project. Our strategy could be summarized as dual stack enabling through the network operator's network. Step 4 consists of several substeps.

1. IPv6 forwarding enablement throughout the network core, aggregation and access layers.
2. IPv6 configuration on the currently enabled IPv4 interfaces.
3. IPv6 core routing protocol introduction next to the current IPv4 core routing protocol. In the current case the IPv4 routing protocol has been OSPF. OSPF is an IP based protocol so a completely new OSPF version 3 has to be configured in order to follow the current IPv4 OSPF setup. Other network operators might save a lot of effort on that step if they use Intermediate System to Intermediate System (IS-IS) routing protocol. IS-IS is an ISO protocol that is independent from the IP layer and just "carries" the IP routing information. So there is no need for almost any IS-IS reconfiguration on protocol layer in order to accommodate IPv6.
4. Verifies end-to-end IPv6 network connectivity between any two PE routers part of the current lab setup.

Step 5: – **IPv6 in the access network**. The building automation solution has as a target both business customers and home subscribers. Business customers were terminated as Layer 2 MPLS VPNs and Layer 3 MPLS VPNs. Home subscribers get Internet Access Service plus additional value added services as IPTV, Telephony and Video on Demand. Each customer case is a separate subscenario of step 5.

1. **MPLS L2 VPN customers** - there are two major approaches for L2 VPN service delivery – point-to-point (virtual private wire service) and point-to-multipoint (Virtual Private LAN Service – VPLS). The service provider has three times more customers on point-to-point pseudowires then on point to multipoint. In either case the building automation solution was provided as a separate VLAN VPLS service terminated in the building automation service zone as a Layer 3 IPv6 interface. The IPv6 address prefix used for stateless IPv6 autoconfiguration has been configured for each separate vlan interface. Each of the embedded systems under test was able to obtain correctly both the local link and the global IPv6 address.
2. **VPLS L3 VPN customers**. The deceison was to provide building automation solution as a separate L3 VPN next to the current customer VPNs. As a configuration the building automation L3 MPLS VPN is a classical hub and spoke. First was configured a Virtual Routing and Forwarding (VRF) instance on the hub and on the spokes. The hub is the site where the building automation solution server resides and the spokes are the customer sites. The hub site has to export its own route-target and to import the customer site router-targets. The customer sites have to import server route-target and to export their own one. The IPv6 global prefix that has been used for stateless address autoconfiguraiton has been configured on the CPE interfaces facing the embedded device cloud. Between the CPE and the PE has been configured a separate "global" routing subnet. On the CPE has been configured a default route pointing towards the PE. On the PEs has been configured a static route matching the global IPv6 prefix for the IoT cloud and pointing towards the CPE. Finally the static route on the PE has been redistributed into the Building automation BGP address family.
3. **Small Office Home Office (SOHO) customers**. The current SOHO subscribers receive a basic Internet Access Service + optional supplementary services as Voice over IP Telephony or IPTV. In that context the building automation is just

yet another service. Currently each service has been provided through a separate device (IP phone or set-top-box) connected to the main SOHO access device. In the case of building automation there will be a yet another IPv6 enabled box between the sensors and the system. The only additional change needed was to enable IPv6 on the Internet Customer Access device and to configure IPv6 global address prefix on the UNI interface.

Step 6: **Verification**. Two verification mechanisms were used to test the IPv6 connectivity between the server and the devices in the IoT cloud. The first mechanism verifies IPv6 layer three connectivity. It was based on IPv6 ping echo test between the embedded devices and the server. The ping echoes were performed through an automated script. Each 10^{th} minute were executed 30 IPv6 pings from the server to the IoT devices. The criteria for success were 98% successfully received echo-replys.

The second mechanism was a full HTTP test between the IoTs and the BMS. The criteria for success were successful bi-directional HTTP GETs and POSTs between the server and the embedded devices.

After executing steps 1-6 the experiment goals were achieved and the lab network was able to support two-way unicast IPv6 connectivity between the embedded devices and the server. With this, the project team has executed its initial goals related to integrating IPv6 Building automation service in the context of multivendor ISP network architecture.

Experiment Problems

The first burden was to harmonize the current network setup as per commonly agreed Best Practices. The only important quality attribute that has to be achieved was consistency throughout the network. Unfortunately this was not the case. The audit has clearly proved that a lot of effort has to be spent just for reaching this point.

The real work was to determine the IPv6 introduction strategy itself. The strategy proposed by us is rather simple. It has introduced IPv6 into ISP network infrastructure and has made possible the deployment of IPv6 based building automation solutions. It does not include any complex transiton mechanisms or tunneling. Our advice is to avoid those except if they are not 100% necesarry. However if there is such necessity it could be solved on a per service bases and should not mess up with the overall NAT free and tunnel free strategy.

Finally there were numerous problems experienced during the strategy application. Basically despite that the network was with the proper configuration as per the design documentation and with the proper software versions there were a number of missing features needed for the successful strategy application. That whole process has triggered several device software upgrades in order to be found an image supporting each and every feature required for the current network opperation. After each upgrade a regression testing was performed to verify that all the other features really work as expected with the new operation system.

CONCLUSIONS AND FUTURE WORK

Internet of Things will be a great achievement from any perspective of view for the human society. With all those objects equipped with minuscule identifying devices, daily life on Earth would undergo a wonderful transformation. Companies would not run out of stock or waste products, as all involved parties would know exactly which products are required and consumed. Mislaid and stolen items would be easily tracked and located. Office Buildings and our homes will be much smarter, will spent less electricity and the CO2 emissions will be decreased.

In order all that to happen all the "things" has be uniquely identified in a global network infrastructure. The only feasible way this to become reality is to be reused the current Internet. However Internet also has its own limitations. It is built around IPv4 and

IPv4 address space has already been used. Therefore the only real option for Internet of Things is to go towards IPv6.

The case study is based on the author's experience from a real project related to IPv6 introduction into IPv4 ISP infrastructure conducted by building automation business opportunity. Others could use our approach and design decisions for deploying similar solutions in other network infrastructures.

It does not cover the deep technical details about how such transformation will be executed but it reveals the rational and the strategy followed by the authors. The article also summarizes the problems experienced during the IPv6 introduction process. As a conclusion some of those could be avoided if the network infrastructure has a setup that comply to some commonly agreed one and if the operator has a propper software tools for network inventorization, configuration management and software images management.

REFERENCES

[1] IANA IPv4 Address Space Registry, http://www.iana.org/assignments/ipv4-address-space/ipv4-address-space.xml

[2] Ashton K., "That 'Internet of Things' Thing" In: RFID Journal, 22 July 2009.

[3] Dunkels A., Vasseur JP, "IP for Smart Objects", Internet Protocol for Smart Objects (IPSO) Alliance, September 2008

[4] Bogomilov I., Milovanov N., Slavinski A., "4to6TRANS USE CASE – Discovering Internet Service Provider topology, MOTSP 2011

[7] Rein. R,. IPv6 Tunneling for Embedded Systems (6bed4)- draft-vanrein-v6ops-6bed4-00, IETF, 7.2011

[8] Ajile Systems, http://www.ajile.com/downloads/aJ-200V1.pdf

[9] Oliveira, L.M.L., Rodrigues, J.J.P.C., Macao, B.M., Nicolau, P.A., Lei Wang; Lei Shu, "End-to-end connectivity IPv6 over wireless sensor networks", ICUFN 2011

[10] Aoun C., Davis, E., "Reasons to Move the Network Address Translator - Protocol Translator (NAT-PT) to Historic Status", RFC 4966, IETF, July 2007

[11] Thomson S., Narten T., "IPv6 Stateless Address Autoconfiguration" RFC4862, IETF, September 2007

[12] Bound K., Droms Ed., "Dynamic Host Configuration Protocol for IPv6 ", RFC 3315 IETF, July 2003

[13] Jiang S., Carpenter B., "An Incremental Carrier-Grade NAT (CGN) for IPv6 Transition", RFC 6264, IETF, June 2011

[14] Deng H., Liu D., "NAT46 considerations", draft-liu-behave-nat46-02, IETF March 2010

[15] Bagnulo M., Sullivan A.," DNS64:DNS Extensions for Network Address Translation from IPv6 Clients to IPv4 Servers", RFC 6147, IETF, April 2011

[16] Bagnulo M., Matthews P.,"Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", RFC 6146, IETF, April 2011

[17] Luimula M., Peintner D., Shelby Z., "Efficient XML Encoding and 6LowApp", Internet Draft, IETF, October 16, 2009

[18] ZigBee Alliance, "ZigBee Home Automation Public Application Profile", 2010

[19] UPnP Forum, "UPnP Device Architecture v.1.1", 2010, http://www.upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.1.pdf

[20] KNX Association, "KNX Specifications", http://www.knx.org

ABOUT THE AUTHOR

Phd. Student, Nikolay Milovanov, Msc., Department of Telecomunications, New Bulgarian University, Phone: +359 898 76 3322, E-mail: nmilovanov@nbu.bg

Phd. Associate Professor Ivan Bogomilov, Department of Telecommunications, New Bulgarian University, Phone: +359 888 777 175, E-mail: ibogomilov@nbu.bg