EPP-IP: A Flexible and Automatic IP Address Registration Mechanism

Yang Liu
Computer Network Information Center, Chinese Academy of Sciences
Graduate University, Chinese Academy of Sciences
4, South 4th Street, Zhongguancun, Haidian district, Beijing 100190, China
liu-yang@cnnic.cn

Ye Tian, Shuo Shen, Wei Mao
Computer Network Information Center, Chinese Academy of Sciences
4, South 4th Street, Zhongguancun, Haidian district, Beijing 100190, China
{tianye, shenshuo, mao}@cnnic.cn

Abstract—The inevitability of IPv6 has come to fore. How to resolve the problem of IP address registration has been a great challenge for network management. The existing solution uses manual registration mechanism, not suitable for dealing with massive IPv6 addresses. Based on the Extensible Provisioning Protocol, we propose an “EPP-IP” mechanism to achieve a flexible and automatic registration management for hierarchical IP addresses. It is compatible with the current address allocation strategy, and can be continued to extend as needed. Complete and consistent registration information will contribute greatly to other network security research.

Keywords—IP address; registration management; flexible; automatic; hierarchical; EPP-IP

I. INTRODUCTION

In early February of 2011, the central pool of available IPv4 addresses managed by the Internet Assigned Numbers Authority (IANA) was depleted averagely [1] and each Regional Internet Registry (RIR) received its last /8 blocks of addresses, marking the existing IPv4 address resources have been fully allocated on a global scale. But demand for IP address due to some new applications (e.g., Internet of Things and Mobile Internet) is still growing quickly.

According to the China Academy of Telecommunication Research (CATR) of the Ministry of Information Industry Technology (MIIT)’s report [2], in the next 5 years, the demand for IP address of China will be: 500 million for fixed Internet, 1 billion for Mobile Internet, and 10 billion for the Internet of Things. If we estimate with the utilization of 33%, the total demand will be 34.5 billion, which is 8 times the capacity of IPv4 addresses.

Obviously, how to manage the vast amounts of IPv6 addresses would be a critical issue. Among the six goals in IPv6 address space management [3], registration is one of the most important to ensure the efficiency and security of the Next Generation Internet (NGI).

In this paper we describe the current situation for IP addresses registration, and design an “EPP-IP” mechanism to unify the communication among multi-level agencies for registering IP address resources. Through analysis, we can infer that such a solution is better for flexible and automatic registration management of IP addresses, thereby enhancing the efficiency and accuracy. It is also helpful to preserve more detailed information about registering IP address.

The rest of this paper is organized as follows: In Section II, we describe the current situation. In Section III, we propose the “EPP-IP” mechanism. In Section IV, we present our prototype system and experiment environment. An corresponding analysis is given in Section V. Finally, we provide a summary of this paper in Section VI.

II. RELATED WORK

IP addresses (Both IPv4 and IPv6 addresses) are generally distributed in a hierarchical manner as depicted in figure 1.

Top level of this architecture is IANA, which is responsible for global coordination of the Internet Protocol addressing systems. According to RIRs’ needs as described by global policy and to document protocol assignments made by the IETF [4], IANA allocates IP addresses from the central pool of available addresses to the RIRs.

There are five Regional Internet Registries (RIRs): African Network Information Centre (AfriNIC), American Registry for Internet Numbers (ARIN), Asia-Pacific Network Information Centre (APNIC), Latin America and Caribbean Network Information Centre (LACNIC), Réseaux IP Européens Network Coordination Centre (RIPE NCC). When a RIR requires more IP addresses, the IANA makes an additional
allocation to the RIR. Because of its cultural and linguistic complexity, APNIC further established National Internet Registries (NIRs) to manage IP addresses of a country or a region, such as CNNIC, JPNIC, KRNIC, etc.

After obtaining IP addresses from RIR or NIR, Internet Service Provider (ISP) can allocate those IP addresses to smaller ISPs or end-organizations. Finally, IP addresses are assigned to each end-user (EU) through DHCP or DHCPv6, etc.

Such a “request–response” interactive process for IP address application and allocation between different levels is called address registration. Next, we will describe the current registration method and its shortcomings.

A. Manual Registration and Limitation

Currently, no matter applying IP addresses from RIR directly, or taking NIR as the agent to apply indirectly, an ISP must submit applications to its superior management agency by email. Both the application and allocation are mainly operated by network administrators. The same method is used between ISP and smaller ISPs or end-organizations.

Depending too much on manual operation, such an inefficient mechanism is difficult to ensure a real-time and reliable communication for IP addresses registration. Especially, to deal with masses of IPv6 address in the future, application and allocation will become very frequent and may increase explosively. For example, in accordance with the provisions [5] of APNIC, a member that has an IPv4 allocation is eligible for a /32 IPv6 address block. There is no specific policy for allocating IPv6 address space to subordinate members, but usually the maximum allocation size is /48, which means possibly millions of subnets will emerge. Then, the traditional method relying on email cannot overcome such challenges.

In addition, complete details about registration of IP address is very important, which can not only facilitate the management and maintenance of the Internet, but also provide an effective way to ensure network security. Such information can still play a big role on many Internet services attributed to its sensitivity to IP address. Presently, NIRs record registry information into their WHOIS [6] database, but most of the lower management levels don’t record any more. Even if few of them conserve some relative information, yet there is no convenient and accurate query interface. Due to various email templates used in each IP address agency, and different requirements for recording, these registry data in WHOIS database is incomplete, inconsistent and insecure [7]. So it is difficult to be applied for solving IP traceback, IP hijacking and other network security issues [8].

B. Automatic Registration and Limitation

Internet Engineering Task Force (IETF) designed the Extensible Provisioning Protocol (EPP) [9] for the provisioning and management of objects stored in a shared central repository.

It was originally intended to provide a mechanism for the registration of domain name resources. EPP unifies the communication between domain name registrars and domain name registries. At present, EPP has been adopted by a large number of major ccTLD (country code top-level domain) and gTLD (generic top-level domain) registries, such as .com, .net, .org, .asia, .eu, .cn, .uk, .us, etc. Undoubtedly, EPP has played a positive and effective role to facilitate the registration management of domain names.

It is expected that this protocol can be used beyond domain name registration. So EPP provides an extension framework at the protocol, object, and command-response levels. Now it can be mapped for the provisioning and management of objects, including domain [10], host [11], and contact [12].

Domain names and IP addresses have a lot in common:

- **Fundamental**: They are both key elements that keep the Internet running smoothly.
- **Huge**: Just like IP address, domain name also has a huge amount of registration. Until the end of the fourth quarter of 2010, registered top-level domain names have been beyond a total of 205.3 million. [13]
- **Hierarchy**: They both require multi-level registration and management.

But EPP cannot support the registration management of IP address until now. A simple mapping from EPP to IP address resource cannot work. Because there are still big differences between IP address and domain name:

- **Domain name itself reflects its hierarchy.** A domain name consists of one or more labels. Each label is concatenated and delimited by dots, such as “www.example.com”. But an IP address is just a bunch of meaningless numbers. Different from domain names, we are difficult to find where the IP address is stored and who owns it.
- **Domain name resources are stored in some central integrated databases, while IP address resources are preserved in a large number of distributed IP address pools.** Once the resource is inadequate in the address pool at higher levels, some recursive operations will be needed.

III. EPP-IP

Through the description above, we can conclude that the current manual registration of IP address is inefficient and unreliable. In the era of IPv6, it will face much more challenges. As an automatic resource registration mechanism, EPP cannot meet the specific needs for IP address registration. It is urgent to design a unified communications protocol among multi-level agencies to solve the problem of IP address registration management.

Next, we will first provide a definition of IP-address object and kinds of interactive commands, and then introduce the procedure of communication. We also refer to offline review of requested actions. Finally, an example will be used to explain the format of message.
A. IP-address Object

An IP-address object is used to encapsulate the relevant information during the communication for registering IP address. It may contain the following six kinds of attributes and associated values:

1) IP addresses

Such a name is composed by address prefix and its prefix length, just like "2001: FEE:: /64", which can be used to uniquely identify an IP address block in the global scale.

2) Contact and Client Identifiers

Character strings with a specified minimum length, a specified maximum length, and a specified format is used to server-uniqely identify the sponsoring client and contacts, including registrant, administrator, financial officer and technical director.

3) Status Values

At least one associated value is needed, to indicate the current status of an IP-address object. For example, “OK”, “pendingCreate”, “clientDeleteProhibited”, etc.

4) Dates Times

Including the date and time when an IP-address object is created, deleted, modified, or updated.

5) Validity Period

The validity period is defined when an IP-address object is created, and it can still be extended.

6) Authorization Information

These attributes contain information required for the security and authentication, including username, password, etc.

B. Interactive Command

Two kinds of commands for operating IP-address object are provided: one is the query commands, including <check> command, <info> command. The other is the transform commands, including <create> command, <delete> command, <renew> command, <update> command and <transfer> command. The latter would possibly change attributes and associated values of the IP-address object.

All these commands are atomic and designed so that they can be made idempotent:

- Atomic: there is no partial success or partial failure.
- Idempotent: if successfully implemented, executing a command once or several times has the same effect.

It means either succeeding completely or failing completely, and absolutely no ambiguous results because of repeated executions.

C. Communication Procedure

Under this mechanism, the registration management of IP address should comply with certain procedures as follows.

1) allocation

Because there is only one level (Domain Registrar) between the customers and Domain Registries, the Domain Registrar only need to retrieve the central database in a specified Domain Registry for registering a domain name. If no one had registered such a domain name before, then a customer can register it.

However, this situation is distinctly different from dealing with IP addresses. IP address management hierarchy is much more complex. From an applicant (just like a small company or an organization) to the management agency who actually own the IP address resources, there are possibly many middle levels among them. Lots of them do not actually own IP address pool, while just play the rule as an agent for further application to the superior level.

Once receiving a client’s application for IP address, the server will check its own IP address pool (suppose that it has such a pool). If there is enough free address space, the server can directly allocate IP address blocks to the customer. Otherwise, it will forward this application to its superior management agency. Obviously, such an application and allocation of IP addresses may be recursive.

2) modification

Considering the customers’ need, most attributes of an IP-address object can be changed. Customers can extend the validity period to achieve renew function. They can also modify information about contacts, even the registrant, to change the ownership of IP addresses for achieving transfer function.

3) recycle

If customer determine to stop using an IP address block and return to its superior management agency. Such an IP-address object maybe destroyed and relative information in database will be modified to reuse this IP address resource in the future.

4) query

We can provide three kinds of query pattern. The first pattern is to determine whether an IP-address object can be provisioned within a repository or not. The second pattern is to retrieve information associated with an IP-address object, including the name of an object, identification for storing, IP address prefix, etc. The last pattern is to query the real-time status of pending and completed transfer requests.

Because of different server policy and clients’ access authority, the response of query is also various.

D. Offline Review of Requested Actions

Theoretically, once receiving a client’s application for IP address, the server should process that request immediately. But as a matter of fact, considering the service needs, some manual review or supervision from the third party is also very meaningful. Therefore, a server operator probably performs an offline review before completing the requested action.

Then giving back an immediate response to confirm receipt is necessary. The server must clearly note that it has received the transform command and is processing such a request; however, the requested action is pending because of manual review. The client can query the processing of the pending action through the status of the corresponding IP-object. After completing the offline processing of the action, the server must inform the client.
E. Message Format

In order to explain the message format required for communication between the two interaction sides, we take the <create> command to create a new IP-address object as an example.

We use <IP: create> element to identify the request command, which contains the following child elements:

- <IP: name> element: contains the fully qualified name of the IP-address object.
- <IP: period> element: optional, indicates how long the customer wants to rent such an IP address block.
- <IP: len> element: contains the length of the IP-address block which the client wants to rent.
- <IP: registrant> element: optional, contains the identifier of the contact (maybe a person or an organization) who will actually own the IP address block.
- <IP: contact> element: optional, one or more, contains the identifiers for other contact objects except registrant, just like administrator, financial officer or technical director.
- <IP: authInfo> element: contains related authorization information, just like username or password.

The sample depicted in figure 2 shows that the client “sh8013” apply for a /48 IPv6 address block for 2 years.

Similarly, we use <IP: creData > element to identify the response, which contains the following child elements:

- <IP: name> element: contains the fully qualified name of the IP-address object.
- <IP: prefix> element: contains the prefix of the IP address block.
- <IP: prefix_len> element: contains the length of the prefix to indicate how many IP address resources that superior agent actually allocated to the client.
- <IP: ParentIP> element: contains the fully qualified name of the IP-address object on the upper level, from which this IP-address object is depleted.
- <IP: crDate> element: contains the date and time when the IP-address object was created.
- <IP: exDate> element: contains the date and time when the IP-address object should be returned back or renewed.

After authentication, once the request is reasonable, a corresponding IP-address object will be created. As is shown in figure 3, a /48 IPv6 address block is allocated to user “sh8013”.

IV. EXPERIMENT

We designed a prototype system to verify the “EPP-IP” mechanism described in section 3. The system is based on C/S mode as the same as the main framework of EPP. “C” means client, located on the lower level; “S” means sever, and located on the upper level. Of course, any intermediate agent may also play a dual role of both server and client. Improved GAP algorithm is used for server to allocate IPv6 addresses [14].

Architecture of this system is depicted in figure 4.
We built an experiment environment using 12 hosts, and the specific configuration parameters are shown in figure 5.

For example, a small company (A1) needs a /64 IPv6 address block, and submits this application to its superior agency (A). Suppose that A does not actually own IP address resource. Then, A will forward this application to A’s superior agent (Root). Luckily, Root has enough IP addresses. Considering possible need in the future, Root depletes a /58 IPv6 address block to A, which is larger than A’s application. Finally, A allocates a /64 IPv6 address block to the initial applicant (A).

Among these different agencies, both application and allocation for IP addresses are managed according to the “EPP-IP” mechanism.

V. ANALYSIS

In this section, we evaluate this mechanism through three aspects to confirm its performance, availability and security.

1) convenience

This mechanism can be used to register IP addresses more easily, including application and allocation, etc. All of these operations will be automatic as much as possible, which is not only necessary for dealing with the huge amounts of IP address data, but also can reduce the problems brought by error-prone manual processing.

2) flexibility

The communication format among multi-level agencies is unified in our proposal. But the actual implementation at both ends of the communication is not restricted. Meanwhile, specification based-XML is also conducive to future extension.

In addition, automatic response to confirm receipt is immediate, but manual review or supervision can be finished offline.

3) security

Under the premise of ensuring the efficiency of present Internet, it is necessary for detailed records of allocation, recycling and other information about IP addresses. Our proposal can achieve such a destination through its complete and consistent specification.

VI. CONCLUSION

In this paper, through extending the Extensible Provisioning Protocol and mapping to IP-address object, we propose a new approach for IP address registration.

This approach can not only reduce the problems brought by manual processing, but also meet the needs of handling vast amounts of IPv6 addresses. It can be extended without too many adjustments by IP address management agency to adapt.

Detailed information provided by this mechanism will be used to ensure the security of the Internet.

REFERENCES