<Working Group Name> A.Y.Chen
Internet Draft Avinta Communications, Inc.
Intended status: <e.g., Informational> May 19, 2015
Expires: November 2015

Preliminary

IPv4 with 40 bit Address Space

draft-<ietf-wgname-docname>-00.txt

Abstract

This document describes a solution, called Extended IPv4 (ExIP), to the Internet address depletion problem by utilizing the existing option mechanism in the IPv4 header. Because it fully conforms to IPv4 protocol, it maximizes backward compatibility while providing sufficient address space for the immediate and mid-term future needs, allowing IPv6 time to mature and then take over for the long term. Unlike other similar proposals, ExIP focuses only on software enhancement to Internet edge routers. Care was taken to avoid costly changes and requirements to the core routers and the deployed mass quantity of business LAN Gateways and residential HAN Gateways. That is, ExIP packets will pass through the former transparently (unprocessed) and will not be presented to the latter. Furthermore, the edge routers need be enhanced only if ExIP feature is offered to its subscribers.

In addition, because it is IPv4, the legal and operational complexities can be mostly postponed or avoided.

Table of Contents

1. Introduction

ExIP (Extended IPv4) is a scheme to extend the assignable public IPv4 address pool for relieving the exhaustion issue due to the number of population and IoTs (Internet of Things) that are expected to reach 7.6 billion and 50 billion, respectively, by Year 2020 according to a recent Cisco online paper [1].

The IPv4 dot-decimal address format, consisting of four octets each made of 8 binary bits, results in the maximum number of assignable public addresses of 4.295 billion (calculated by 256 x 256 x 256 x 256, to be 4,294,967,296 - decimal exact). Using the binary notation of 64K representing 256 x 256 (decimal 65,536), the full IPv4 address pool of 64K x 64K may be expressed as 4,096M, or 4.096B.

To minimize affecting the existing equipment operation characteristics, the ExIP scheme focuses on reclaiming the third octet of the reserved private network address block 192.168/16 to be publicly routable at the edge of the Internet. By making use of this octet as semi-public addresses, the number of assignable public addresses is multiplied by 256 times to become 1048B which is 20 times more than that of the expected IoTs.

This document describes a method of making use of this newly identified resource while minimizing the perturbation to the existing IPv4 based Internet operation characteristics:

A. Only the edge routers serving the subscribers, when needed, will be enhanced to make use of this scheme by creating and recognizing IP headers containing the ExIP information for performing the extra routing function. This modified IP header will be referred to as ExIP header in the rest of this document.

B. Neither Internet core / backbone routers, nor private network gateways will handle this addition to the IP header, since their original designs do not recognize this option for transporting ExIP information.

2. System Architecture

A. Referring to Figure 1, instead of assigning each entity a public IPv4 address as today, Semi-public routers will be inserted in between an edge router and its connections to premises for utilizing the third octet of 192.168.nnn/24 to serve multiple premises. Each of these could be either a LAN or a HAN.

```
                                             +----------+
                                   +---------+ Internet +----------+
                                   |         +----+-----+          |
                                   |              |                |
                                +-----+   ...  +--+--+   ...    +-----+
                          +-----+  E1 |        |  W1 |    +-----+  E2 |
                          |     +-----+        +-----+    |     +-----+
                          |69.41.190.140   69.41.190.143  |69.41.190.145
                          |                               |
                       +-----+                         +-----+
                 +-----+  S1 |                   +-----+  S2 |
                 |     +-----+                   |     +-----+
                 |192.168.2.0                    |192.168.5.0
                 |                               |
              +-----+                         +-----+
        +-----+ RG1 +-----+             +-----+ RG2 +-----+
        |     +-----+     |             |     +-----+     |
        |192.168.2.9      |192.168.2.3  |192.168.5.10     |192.168.5.40
        |                 |             |                 |
     +-----+    ...    +-----+       +-----+    ...    +-----+
     | T1a |           | T1x |       | T2a |           | T2x |
     +-----+           +-----+       +-----+           +-----+

            Premises 1                       Premises 2
```

E1, E2: Internet Edge Router
RG1, RG2: deployed Residential Gateway
S1, S2: proposed Semi-public Router
T1a … T2x: Internet of Things / private network host
W1: Web Server

Figure 1 ExIP System Architecture

B. On each LAN / HAN, the fourth octet of 192.168.nnn.mmm/32 continues to be used by deployed LAN Gateways and HAN RGs to identify the IoTs on a private basis. The 256 combination from this octet is 38 times more than the projected average need of 6.58 IoTs per person. (See Discussion below for savvy users who may have more IoTs.)

C. Since the current physical connections to subscriber’s RGs do appear at the edge router, it is natural to have the semi-public routers be collocated with the edge router. It follows that the simple routing function provided by the new semi-public router may be absorbed into the edge router through a straightforward operational software enhancement.

D. Consequently, the public / private demarcation line will remain at the RG where utility services enter a subscriber's remises.

E. Since each customer premises is now identified by the third octet "nnn" of the 192.168.nnn/24 in additional to a conventional IPv4 public address, the "nnn" part needs be appended to the end of an IPv4 address to completely identify a premises.

Note: To simplify the presentation, it is assumed in this paper that the TCP Port number [4] assigned to an IoT by a RG's NAT equals to the fourth octet of that IoT's private IP address assigned by the RG's DHCP which is unique within each respective private network. For improved security in actual operation as well as allowing each IoT to have multiple simultaneous sessions, this direct correlation shall be avoided.

F. For example, the customer "Premises 2" in the diagram may use 69.41.190.145-5 to identify itself to other Internet users. The following are proposed short hand notations for representing an IoT. Note that by appending the conventional TCP Port number ":10", the unique identification of T2a is complete.

```
ExIP Address notation (IPv4 + Semi-Public + TCP Port Number)
   RG1: 69.41.190.140-2       RG2:  69.41.190.145-5
   T1a: 69.41.190.140-2:9     T2a:  69.41.190.145-5:10
   T1x: 69.41.190.140-2:3     T2x:  69.41.190.145-5:40
```

This notation suggests an efficient format for transporting this "Extension Address" information in the IP header as described below.

3. Header Format

To enhance an existing system, not only the current operations should not be disturbed, but also the added overhead must be minimized. The suggested ExIP address notation enables us to efficiently utilize the option mechanism already defined in IP header to achieve these goals.

It is discovered that an organization called EnhanceIP.Org [2] has been, for some years independently developing a very similar approach. Their recent draft proposal to IETF called Enhanced IP (EnIP) [3] details a method of transporting a full secondary IPv4 address (based on the 10/8 block) for identifying a host to the destination across the Internet. To transport the resultant 64-bit address, it makes use of some overhead added to an IPv4 header following its predefined option format. According to the referenced draft document, EnIP traffic has been successfully sent through Internet during actual experiments. This provides an excellent baseline for implementing the proposed ExIP.

Since core routers throughout Internet are not programmed to recognize the additional information for the new option, an EnIP packet may travel through Internet as if it had only the traditional 32-bit IPv4 address, until it reaches an enhanced edge router. Such edge router is only needed where this EnIP service is offered to subscribers by an ISP, case by case, until this becomes a common practice.

To facilitate the discussion, Figures 2 & 3 below is a copy of the EnIP header layout. It is an IPv4 header utilizing the option mechanism.

****

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  1 |Version|  IHL  |Type of Service|          Total Length         |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  2 |         Identification        |Flags|     Fragment Offset     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  3 | Time to Live  |    Protocol   |         Header Checksum       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  4 |                       Source Host Number                      |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  5 |                    Destination Host Number                    |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  6 |    EnIP ID    |    EnIP ID    |E|E|                           |
    |     (0X9A)    | Option Length |S|D|         (Reserved)        |
    |               |               |P|P|                           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  7 |                    Enhanced Source Address                    |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  8 |                  Enhanced Destination Address                 |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2 EnIP Header

We will list the fields that are unique to EnIP before proceeding to adapting this design for implementing ExIP:

Field	Bits	Description
IHL	4	Internet Header Length is identical to IPv4's and is set to 8 words
Total Length	16	The Total Length field is identical to that of IPv4 and is set to 32 octets
		EnIP specific
EnIP ID	8	The EnIP ID field equals to 0x9A or 1001 1010. It's meaning is given below
EnIP Option Length	8	The EnIP Option Length field is 12 octets
ESP	1	This flag indicates if an Enhanced Source Address is present
EDP	1	This flag indicates if an EnIP Destination Address is present
Reserved	14	Reserved for future use
Enhanced Source Address	32	This is the source host address used by EnIP.
Enhanced Destination Address	32	This is the destination host address used by EnIP.

The meaning of the EnIP ID field,

1001 1010 (0x9A hexadecimal)

is as follows:

If an EnIP packet traverses a router and must be fragmented because of a link with a smaller MTU, the copy bit ensures that fragments include the 12-byte IP option header in all fragmented packets. The rest of the bits and their meaning are as follows.

Meaning of 1 00 11010 the EnIP ID		
1	00	11010 (or 26 base 10)
copy bit set	Class is Control	Option is a new value

Figure 3 EnIP Id Field Interpretation

Note that 26 is the IP option value used in the EnIP experiments.

****

Briefly, EnIP proposes to lengthen the basic IP header from five 32 bit words to eight by setting the IHL (Internet header Length) value accordingly. The additional three words consist of one leading word identifying the EnIP option plus the remaining two carrying the Enhanced Source and Destination full IPv4 addresses, respectively.

A. Since the Extended Address in ExIP has only one octet (8 bits), we can carry both Extended Source and Destination Addresses by using only half of Word 7. This means that two octets in this word are not needed. As will be analyzed latter, this is the motivation for compacting this overhead for a more efficient ExIP header.

B. Since Word 8 is not needed, the IHL in ExIP header is changed to 7. The following layout depicts the preliminary format of an ExIP header.

Making use of EnIP Header Format

```
      0                   1                   2                   3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   1 |Version|IHL(7) |Type of Service|       Total Length (28)       |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   2 |         Identification        |Flags|     Fragment Offset     |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   3 | Time to Live  |    Protocol   |        Header Checksum        |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   4 |                       Source Host Number                      |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   5 |                    Destination Host Number                    |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   6 |    ExIP ID    |     ExIP      |E|E|                           |
     |     (0X9B)    | Option Length |S|D|        (Reserved)         |
     |               |      (8)      |P|P|                           |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   7 |    Extended Source Address    |  Extended Destination Address |
     +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4 ExIP Header (Preliminary)

Fields that content has been modified:

Field	Bits	Description
IHL	4	Internet Header Length is set to that of ExIP header which is 7 words
Total Length	16	The Total Length field is 28 octets
ExIP ID	8	The ExIP ID field is set to 0x9B or 1001 1011 to distinguish from EnIP
ExIP Option Length	8	The ExIP Option Length field is 8 octets
ESP	1	This flag indicates if an ExIP Source Address is present
EDP	1	This flag indicates if an ExIP Destination Address is Present
Extended Source Address	16	This is the source host address used by ExIP

Extended Destination Address 16 This is the destination host address used by ExIP

Briefly, this preliminary ExIP format uses 33% less overhead than EnIP.

Compacting ExIP Header:

Examining the bit usage in the preliminary ExIP header above, it is determined that only 16 bits of word 7 are being used, while there are 14 reserved bits in word 6. Also, 8 bits in it are allocated for the ExIP Option Length which is an overkill for the actual length of no more than 8 octets. It follows that we can make the ExIP header more compact by not having word 7. Figure 5 is the more compact ExIP header layout.

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  1 |Version|IHL(6) |Type of Service|      Total Length (24)        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  2 |        Identification         |Flags|     Fragment Offset     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  3 | Time to Live  |    Protocol   |        Header Checksum        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  4 |                       Source Host Number                      |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  5 |                  Destination Host Number                      |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  6 |    ExIP ID    |    ExIP   |E|E|   Extended    |   Extended    |
    |     (0X9B)    |Opt. Leng. |S|D|    Source     |  Destination  |
    |               |    (4)    |P|P|    Address    |   Address     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5 ExIP Header (Compacted)

Note that ESP and EDP bits may not be necessary either, if we set up a convention that when ESA (Extended Source Address) or EDA(Extended Destination Address) field consists of all “1”, it means that the corresponding address does not exist. Then, these 2 bits may not be needed and could be reserved as spares as well. Additionally, since the ExIP overhead is now only 4 octets, 4 bits are more than adequate to represent the length. In all, 4 bits may be reserved as spares. Figure 6 shows the optimized ExIP header format.

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  1 |Version|IHL (6)|Type of Service|      Total Length (24)       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  2 |         Identification        |Flags|     Fragment Offset     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  3 | Time to Live  |    Protocol   |         Header Checksum       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  4 |                       Source Host Number                      |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  5 |                  Destination Host Number                      |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  6 |    ExIP ID    | ExIP  |       |   Extended    |   Extended    |
    |     (0X9B)    |Opt. L.| Spare |    Source     |  Destination  |
    |               |  (4)  |       |    Address    |    Address    |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 6 ExIP Header (Optimized)

In summary, ExIP header as showing in Figure 5 or 6 requires only one extra word relative to the basic IP header of five words. This is just 20% overhead as compared to EnIP header which requires three new words or 60% overhead.

4. Implementation

The above derivation defines a format of carrying the proposed “Extension Address” in the modified IP header that we name as the ExIP Header. Referring to Figure 1, the following are a couple examples of Extended TCP/IP headers, and their transitions to and from basic TCP/IP headers as packets traverse through several routing devices.

A. T1a (IoT) initiates a session towards W1 (Web Server):

a. Tia sends a packet towards RG1 with basic TCP/IP header: Note that to focus on our discussion, only the first word of TCP header is included as word 6.

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  1 |Version|IHL (5)|Type of Service|       Total Length (20)       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  2 |        Identification         |Flags|     Fragment Offset     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  3 | Time to Live  |    Protocol   |        Header Checksum        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  4 |               Source Host Number (192.168.2.9)                |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  5 |            Destination Host Number (69.41.190.143)            |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  6 |           Source Port (9)     |   Destination Port (all 1's)  |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 7 Basic TCP/IP Header: Tia to RG1

b. RG1 relays the packet to S1 still using basic TCP/IP Header):

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  1 |Version|IHL (5)|Type of Service|       Total Length (20)       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  2 |        Identification         |Flags|     Fragment Offset     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  3 | Time to Live  |    Protocol   |        Header Checksum        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  4 |               Source Host Number (192.168.2.0)                |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  5 |            Destination Host Number (69.41.190.143)            |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  6 |           Source Port (9)     |   Destination Port (all 1's)  |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 8 Basic TCP/IP Header: RG1 to S1

c. S1 relays the packet to E1 to be sent to W1. The IP header is now reconstructed with Ex TCP/IP format:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  1 |Version|IHL (6)|Type of Service|       Total Length (24)       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  2 |        Identification         |Flags|     Fragment Offset     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  3 | Time to Live  |    Protocol   |         Header Checksum       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  4 |              Source Host Number (69.41.190.140)               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  5 |           Destination Host Number (69.41.190.143)             |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    ExIP ID    | ExIP  |       |    Extended   |   Extended    |
  6 |     (0X9B)    |Opt. L.| Spare |Source Address |Destination Add|
    |               |  (4)  |       |      (2)      |  (all 1's)    |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  7 |          Source Port (9)      |   Destination Port (all 1's)  |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 9 Ex TCP/IP Header: S1 to W1

This set of TCP/IP headers demonstrates that neither an IoT, not an RG needs to know about ExIP scheme. Such information is encoded by the Semi-public router. Even so, the routing from S1 to W1 is still purely based upon public IPv4 addresses. That is, the Extended portion of the IP addresses are ignored by Internet core routers, just like the TCP Port numbers.

B. Reply from W1 (ExIP capable):

a. W1 through E1 to S1 with Ex TCP/IP format

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 1 |Version|IHL (6)|Type of Service|       Total Length (24)       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 2 |         Identification        |Flags|     Fragment Offset     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 3 | Time to Live  |    Protocol   |         Header Checksum       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 4 |               Source Host Number (69.41.190.143)              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 5 |            Destination Host Number (69.41.190.140)            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |    ExIP ID    | ExIP  |       |    Extended    |   Extended   |
 6 |     (0X9B)    |Opt. L.| Spare |Source Address |Destination Add|
   |               |  (4)  |       |    (all 1's)   |     (2)      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 7 |      Source Port (all 1's)    |      Destination Port (9)     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 10 Ex TCP/IP Header: W1 to S1

b. S1 relays packet to RG1 with basic TCP/IP Header:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 1 |Version|IHL (5)|Type of Service|       Total Length (20)       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 2 |         Identification        |Flags|     Fragment Offset     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 3 | Time to Live  |    Protocol   |         Header Checksum       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 4 |               Source Host Number(69.41.190.143))              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 5 |               Destination Host Number (192.168.2.0)           |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 6 |       Source Port (all 1's)   |   Destination Port (9)        |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 11 Basic TCP/IP Header: S1 to RG1

c. RG1 relays packet to T1a with Basic ICP/IP Header

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  1 |Version|IHL (5)|Type of Service|       Total Length (20)       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  2 |         Identification        |Flags|     Fragment Offset     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  3 | Time to Live  |    Protocol   |         Header Checksum       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  4 |             Source Host Number (69.41.190.143)                |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  5 |            Destination Host Number (192.168.2.9)              |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  6 |        Source Port (all 1's)  |      Destination Port (9)     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 12 Basic TCP/IP Header RG1 to T1a

In this set of headers for the reply packet, W1 has to be aware of the ExIP format to construct the full Extended TCP/IP header, so that the packet may reach T1a. As in the other direction, Internet core routers only route on basic IPv4 addresses. As well, RG1 and T1a do not need to know anything about ExIP to receive reply packets from W1.

C. A packet being sent toward T1a from S2 on behalf of T2x who knows the full identification of T1a & T2x.

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  1 |Version|IHL (6)|Type of Service|       Total Length (24)       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  2 |        Identification         |Flags|     Fragment Offset     |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  3 | Time to Live  |    Protocol   |        Header Checksum        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  4 |              Source Host Number (69.41.190.145)               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  5 |           Destination Host Number (69.41.190.140)             |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    ExIP ID    | ExIP  |       |    Extended    |   Extended    |
  6 |     (0X9B)    |Opt. L.| Spare |Source Address |  Destination  |
    |               |  (4)  |       |      (5)       |  Address (2)  |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  7 |       Source Port (40)        |    Destination Port (9)       |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 13 Ex TCP/IP Header: Between Two ExIP-Capable IoTs

This is an Extended TCP/IP header between two ExIP capable IoTs for direct end-to-end communication. The fields for fully identify of both ends are completely populated.

5. Discussion

A. Although very similar in basic concepts, there is a practical consideration that distinguishes ExIP from EnIP. Rather than making a full length private institution address (10/8) publicly routable resulting in a 64 bit public address system and essentially bypassing the LAN Gateway, ExIP’s basic intention is to make only the third octet of the private network address (192.168/16) publicly routable resulting in a 40 bit public address system allowing the deployed RGs continue their current private network routing services but reduces to only routing the fourth octet of this address block.

B. The above description outlines how to make use of the reserved private network address block, 192.168/16 to mitigate IPv4 public pool shortage. Namely, the third octet is utilized to multiply assignable public address by 256, with the private network address pool restricted to only the fourth octet. For savvy Internet users, the larger reserved private institution network blocks, 10/8 or 172.16/12 may be utilized to implement ExIP. After the available leading 8 bits are assigned for semi-private routing, there still will be 16 or 12 bits, respectively remaining for private network routing. These result in 64K or 4K possible private hosts per site, respectively.

C. Of course, to indicate which one of the three private network address block is being used without explicitly mentioning its leading bits in the ExIP header, each must be deployed under a distinguishable group of basic public IPv4 blocks. Since 1/16th of the current IPv4 addresses with ExIP is more than 8.6 time of the expected population by Year 2020, it is feasible to allocate two additional separate groups of the basic IPv4 address blocks, one for each of these two larger address blocks out of the spare 15/16th IPv4 pool. By recognizing the basic IPv4 address group in an ExIP header, an edge router will be able to prefix appropriate missing leading private network identification digits to the compact “Extension Address” (one octet), so that the full secondary address is reconstructed for the semi-public routing process.

D. Figure 14 is a summary of the number of possible assignable addresses per basic IPv4 public address under different configurations. Basically, EnIP approach completely transforms the three private network address blocks to be publicly routable, thus bypassing the LAN gateway. ExIP approach transforms only partially (one octet worth of) these address blocks to become publicly routable, retaining the remainder as privately routable addresses. Consequently, not only the public IPv4 address pool exhaustion issue is mitigated, but also the familiar LAN gateway and HAN RG private network routing functionality is preserved.

E. Although routers other than the edge routers do not need be enhanced to handle ExIP packets, the terminal devices, such as Web Servers and IoTs do need to. Being aware of ExIP format, these devices will be able to properly construct a full ExIP header for the associated packet to take advantage of the extended routing service.

F. To roll out the ExIP configuration outlined in Figure 14, we will need to enhance web servers to be aware of ExIP. Next, ISPs may begin deploying semi-public routers to multiply the number of assignable public IPv4 addresses. Once these are ready, ExIP-aware IoTs may begin direct communications with one another.

	10/8	172.16/12	192.168/16
Basic IPv4 (IHL = 5, Overhead = 0% - Reference)			
Public	0	0	0
Private	16M	1M	64K
Extended IPv4 (IHL = 6, Overhead = 20%)			
Semi-Public	256	256	256
Private	64K	4K	256
Enhanced IPv4 (IHL = 8, Overhead = 60%)			
Public	16M	1M	64K
Private	0	0	0

Figure 14 Allocating IPv4 Public vs. Private Addresses

6. Summary

To resolve the IPv4 public address pool exhaustion issue, a technique called Extended IPv4 (ExIP) is proposed. It utilizes one octet worth of the reserved private network address block to multiply the assignable public address pool by 256 fold. ExIP is developed independent of EnIP. The former focuses on residential application while the latter is driven by business need. Nevertheless, the EnIP header format enables ExIP be quickly implemented. The two approaches are complimentary to each other. In fact, both may coexist with the basic IPv4 giving users a wide range of flexibilities.

ExIP header overhead is more efficient, because a specific block of reserved address is used, thus avoiding carry the known leading address bits. EnIP requires 60% overhead by adding three words to the five 32-bit words of the basic IPv4 header. In comparison, ExIP needs only 20% overhead because it uses one new word instead. This allows the ExIP header to be more efficient in bit usage. The tradeoff is that the edge router will have to be capable of striping off the leading bits in the “Extension Address” in constructing an ExIP header as well as overlaying the appropriate template to the incoming packet for reconstructing the full address to deliver the IP packet to specific IoT.

7. References

7.1. Informative References

[1] https://nishithsblog.files.wordpress.com/2014/04/internet-of-things-market-forecast.jpg

[2] http://www.enhancedip.org/docs/enip.pdf

[3] https://tools.ietf.org/html/draft-chimiak-enhanced-ipv4-00

[4] http://www.potaroo.net/ispcol/2004-07/2004-07-isp.htm

8. Acknowledgments

As referenced throughout this writing, the implementation of ExIP is greatly facilitated by the EnIP work led by William Chimiak of Laboratory for Telecommunication Sciences, 8080 Greenmead Drive, College Park, MD 20740.

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Address

Abraham Y. Chen
Avinta Communications, Inc.
142 N. Milpitas Blvd., #148
Milpitas, CA 95035-4401 USA

Phone: +1 (408) 942-1485
Email: AYChen@Avinta.com