# An implementation of Classless Interdomain Routing for ns-2

Rob Ballantyne

February 3, 2002

## 1 Introduction

The motivation for the work described in this paper was the desire to accurately simulate a link state routing protocol in ns-2, specifically the interest was in IS-IS [Cal90] but it could have easily been OSPF [Moy98] instead.

The objective was to accurately simulate route aggregation and to investigate ways to significantly speed up the convergence time of link state algorithms. This was an investigation aimed at proving that modified standard routing algorithms could be used to implement fast failover and recovery of networks, making the use of MPLS techniques unnecessary.

Initial investigations of ns-2 as the simulation platform uncovered that it lacked the detailed forwarding and addressing scheme of IP version 4. ns-2 supports both a flat addressing scheme and a fixed hierarchical addressing scheme but not the variable scheme specified in the current set of RFCs that describe the IP version 4 forwarding scheme — Classless Inter-domain Routing (CIDR).

This paper describes the implementation of the CIDR forwarding in ns-2.

## 2 Internet Protocol version 4 (IPv4) and CIDR

Familiarity with the IP addressing scheme is expected of the reader. A good book on the material, for those that need a review, is [Ste94]. We follow the usual conventions for writing IP addresses and netmasks as "dotted quads."

### 2.1 Classfull Forwarding

IP addresses are 32 bit quantities uniquely identifying interfaces within the Internet. These addresses are considered to have two parts: a network number and a host number.

The network portion of the address is determined by the so-called *class* of the address. There are five classes definied: class A, B, C, D, and E. Classes A, B, and C are the unicast address spaces. Class D is the multicast space and class E is for experimental purposes. The class of a particular address is

| Class | High-order byte | Addrress Range | | Bit Positions | |
|---|---|---|---|---|---|
| | | Low | High | Network | Host |
| A | 0xxx xxxx | 0.0.0.0 | 127.255.255.255 | 31–24 | 23–0 |
| B | 10xx xxxx | 128.0.0.0 | 191.255.255.255 | 31–16 | 15–0 |
| C | 110x xxxx | 192.0.0.0 | 223.255.255.255 | 31–8 | 7–0 |
| D | 1110 xxxx | 224.0.0.0 | 239.255.255.255 | — | — |
| E | 1111 xxxx | 240.0.0.0 | 255.255.255.255 | — | — |

Table 1: Network classes, address ranges and sizes.

determined by it's high-order bits. Table 1 lists the identifying features of each of the classes of address space, and Figure 1 displays the actual bit layout of addresses of the various classes. Note that the network portion of the address includes the class distinguishing high-order bits.

In this classfull scheme the forwarding information is contained, as usual, in a routing table. Entries in this table take the form:

$$\langle network, host\rangle \rightarrow next\_hop\_addr$$

Each such route is additionally marked as either a host specific route or a network route. There is also a default route. Note that the "next hop addr" must be the address of an adjacent node. The division of the network and host parts of a route are simply for emphasis; a single 32 bit quantity is used to store both.

When a packet arrives it is first examined to check if its destination is any of the addresses of this node. If it is the packet is delivered locally to this node. If the address is not one of the node's addresses the routing table is examined. First the host specific routes are examined to see if there is a route that matches both the network and host portions of the packet if so the packet is forwarded to the indicated directly attached host indicated in that routing entry. If no host specific routes match then the network routes are examined to see if the network portions only match. In which case the packet is forwarded to the indicated node. Finally, if the packet has matched no routes and the table contains a default route the packet is forwarded to that address.

Clearly this scheme has it limitations. No network actually has $2^{24}$ hosts, in the case of a class A network. The scheme was extended in [JM85] to permit network administrators to sub-divide their networks into a collection of smaller sub-networks. The IP address was divided into three parts: the network, the sub-network, and the host part. The network part remained as before, defined by the class of the network. It was the host part that was sub-divided into the two other parts. A *netmask* was used to indicate which part of the address was the sub-net. The netmask is 32 bit quantity which has binary ones in positions corresponding to positions in the IP address that are in the network or sub-network parts of the address and binary zeros in the new sub-network host part. Figure 3 is an example of a Class B network with a subnet mask further partitions the address space into 16 sub-networks.

```
Let p denote the arriving packet, p.dst the destination address
of the packet, and p.src the source address.  Let local[k] be an
array of the addresses of this packet.  The routing table route[k]
is an array of routing table entries.  Let route[k].addr be the
network and host address of the particular entry and let
route[k].nh be the next hop address for destinations matching
route[k].addr.

1.  for each of the local node addresses local[i] do
        if p.dst = local[i] then deliver p locally
    end

2.  for each of the host specific routes route[i] do
        if p.dst = route[i].addr then queue p for delivery
           to route[i].nh
    end

3.  for each of the network routes route[i] do
        if (p.dst && netmask) = (route[i].addr && netmask) then
           queue p for delivery to route[i].nh
    end

4.  if p hasn't matched any of the previous steps and there is
        a default route defined then queue p for delivery to
        that host .
```

Figure 1: Algorithm for Classfull (without subnetworks) routing

| | | | | |
|---|---|---|---|---|
| Class A | **0nnnnnnn** | **hhhhhhhh** | **hhhhhhhh** | **hhhhhhhh** |
| Class B | **10nnnnnn** | **nnnnnnnn** | **hhhhhhhh** | **hhhhhhhh** |
| Class C | **110nnnnn** | **nnnnnnnn** | **nnnnnnnn** | **hhhhhhhh** |
| | 31 24 | 23 16 | 15 8 | 7 0 |

Figure 2: Address format of the classes. `n` indicates the bit is part of the network portion of the address and `h` is similarly a part of the host portion.

|  |  |  |  |  |
|---|---|---|---|---|
| **Mask** | 11111111 | 11111111 | 11110000 | 00000000 |
| **Address** | 10nnnnnn | nnnnnnnn | ssssshhhh | hhhhhhhh |
|  | 31 24 | 23 16 | 15 8 | 7 0 |

Figure 3: A subnet mask example. `n`,`h` indicate as in the last figure. `s` is the subnet portion.

In the classfull and subnetted scheme routing table entries also carry the subnetwork mask:

$$\langle netowork, host, mask\rangle \rightarrow next\_hot\_addr$$

The forwarding algorithm is slightly changed. When a packet arrives it is again examined for local delivery and then for host specific routes as before. When the packet is compared against network routes the matching condition is now that portion of the destination address in the packet selected by the non-zero portion of the network mask be equal to the same portion of the route entry destination address. As before if no matches occur then the packet is forwarded to the default router if a default route is in the routing table.

There are two important things to recognize about this classfull sub-networked scheme: that addresses are allocated in a classfull network at a time and that only routers within the classfull address range make use of sub-network information. An implication of these two facts is that considerable address space is wasted. If a user of address space requires slightly more than a class C worth of address space they must be given a class B address. Another consequence is the size of the routing table within the core routers of the internet. Since the core routers (routers with out default routes) must be able to forward packets to all networks that have been allocated and since there is no hierarchy to the networks in this classful sub-networked scheme there must be a route, concivably, for each class A, B, and C network. This amounts to $2^7 + 2^{14} + 2^{21} = 2113664$ routes. If you consider that core routers today may handle millions to billions*** of packets per second even efficent methods of search such large tables would prove inadequate to deliver the packets in a timely fashion.

## 2.2 Classless Interdomain Forwarding

Classless Interdomain Routing and associated address allocation policies were introduced in [War92], [Li93], and [War93] so that more there would be more efficent use of the available address space and to reduce the number of routes required in the core routers of the Internet.

With the introduction of CIDR network masks started to be specified as *prefix lengths*. The prefix length was simply the number of contigeous(sp?) 1s in the netmask. Addresses with their netmask are now more usually written like: 142.58.0.0/16. Where left of the "/" is the network address and to the right is the prefix length, in this case corresponding to 255.255.0.0.

Until the introduction of CIDR network masks were always used to further subdivide the class A, B, or C networks. With the introduction of CIDR network masks also could aggregate networks together. Two class C networks, say 199.60.2.0/24 and 199.60.3.0/24, each with the default 24 bit prefix (255.255.255.0) could be aggregated into a single network 199.60.2.0/23 (netmask 255.255.254.0). Notice that the initial 23 bits of both 199.60.2.0 and 199.60.3.0 are the same. Not all consecutive networks are aggregatable in this way, for example 199.60.1.0/24 and 199.60.2.0/24 are not.

The forwarding algorthim changes little with CIDR. The routing table entries remain the same as with the subnetted case above. In this case netmask prefix lengths are arbitrary. All of the old class distinctions are removed. One important change is that the routing table may have several entries that may apply to a particular destination address — each with a different length netmask prefix length. For example: entries $\langle$142.58.128.0/17$\rangle$ (which represents all of the address space from 142.58.128.0 to 142.58.255.255) and $\langle$142.58.200.0/24$\rangle$ (which represents the piece from 142.58.200.0 to 142.58.200.255) may occur in the routing table.

Since there are potentially conflicting entries in the routing table the procedure to determine the next hop address is changed slightly. Again if the arriving packet is destined for one of the nodes addresses the packet is delivered locally. The packet is then compared to the routing table entries. The routing table is searched for entries where the masked portions of the packet's destination address and the table entrie's destination address match. If there are several potential matches the one with longest network prefix length is chosen[1].

# 3 ns-2 Internals

- tcl C++ dual nature
- classifiers
- routing structure

# 4 Additions to ns-2 required to support IPv4 & CIDR simulations

- new classifier
- new routing alg

---

[1] There may still be multiple matches of the same length. In this case further information is required to indicate what to do. Some vendors specify *administrative preferences* other use this to indicate that traffic to that destination be forwarded over both links in a load balancing way

## 5 Testing

- cases

## 6 Conclusions

- forwarding

## 7 Future work

- routing algorithms
- speed ups

## 8 References

- Host Requirement RFC
- Router Requirement RFC (Gateway)
- Interconnection, Radia Pearlman
- TCP/IP Illustrated, Richard Stevens

▯

## References

[Cal90] R. Callon. Use of osi is-is for routing tcp/ip and dual environments, 1990. RFC 1195, ISO D99999.

[JM85] J. Postel J. Mogul. Internet standard subnetting procedure, 1985. RFC 950.

[Li93] Y. Rekhter T. Li. An architecture for ip address allocation with cidr, 1993. RFC 1518.

[Moy98] J. Moy. Ospf version 2, 1998. RFC 2328.

[Ora90] D. Oran. Osi is-is intra-domain routing protocol, 1990. RFC 1142.

[Pea99] Radia Pearlman. *Interconnections, 2nd Ed.* Addison Wesley, 1999.

[Ste94] W. Richard Stevens. *TCP/IP Illustrated, Volume 1.* Addison Wesley, 1994.

[War92] V. Fuller T. Li J. Yu K. Wardhan. Supernetting: an address assignment and aggregation strategy, 1992. RFC 1338.

[War93] V. Fuller T. Li J. Yu K. Waradhan. Classless inter-domain routing (cidr): an address assignment and aggregation strategy, 1993. RFC 1519.