Multicast Troubleshooting with ssmping and asmping

Stig Venaas
UNINETT
Trondheim, NO 7465, Norway
venaas@uninett.no

Tim Chown
School of Electronics and Computer Science,
University of Southampton, SO17 1BJ, UK
tjc@ecs.soton.ac.uk

Keywords: Multicast, SSM, ASM

Abstract
Multicast is generally said to be difficult to understand and manage, in part because there are very few management and troubleshooting tools available for multicast. We aim to address this issue by presenting two new tools that we believe to be helpful in testing and debugging multicast.

The ssmping and asmping tools
The basic idea of the ssmping and asmping tools is to be able to check that one can receive multicast from a host. The tools are used in much the same way as the 'ping' tool that is commonly used for testing that a unicast host is reachable. With ssmping and asmping there are additional replies sent by multicast. If those replies are received, then multicast is working. People who are used to using 'ping' will immediately feel at home with these tools since their output consists of sequence numbers, number of hops, round trip times etc, giving the same type of information as 'ping'.

The tools support both IPv4 and IPv6. They are available for most common UNIX platforms, and also for Windows. The ssmping client can however only work on platforms that have Source Specific Multicast (SSM) support. Documentation and source code for the tools are available [1].

The ssmping tool
The ssmping idea is based on the paper SSM-Ping: A Ping Utility for Source Specific Multicast [2]. While they proposed a network layer solution, this new tool runs purely at the application layer, hence requiring no changes to the underlying network protocols.

An example usage of ssmping would be for a network provider to run a server and announce the server address. Users can then ping the server to check that they have working multicast. There are some public servers that anyone can ping to check that they can receive multicast from the Internet. There is also much more information beyond basic connectivity that can be obtained from using these tools. For example, here is ssmping being used to check whether a host can receive SSM from the public server *ssmping.uninett.no*:

```
$ ssmping ssmping.uninett.no
ssmping joined (S,G) = (2001:700:1:7:211:d8ff:fe8f:1f9b,ff3e::4321:1234)
pinging S from 2001:630:d0:111:250:fcff:fe6a:42b3
unicast from 2001:700:1:7:211:d8ff:fe8f:1f9b, seq=0 dist=20 time=57.106 ms
unicast from 2001:700:1:7:211:d8ff:fe8f:1f9b, seq=1 dist=20 time=56.929 ms
unicast from 2001:700:1:7:211:d8ff:fe8f:1f9b, seq=2 dist=20 time=62.466 ms
multicast from 2001:700:1:7:211:d8ff:fe8f:1f9b, seq=2 dist=12 time=65.706 ms
unicast from 2001:700:1:7:211:d8ff:fe8f:1f9b, seq=3 dist=20 time=57.226 ms
multicast from 2001:700:1:7:211:d8ff:fe8f:1f9b, seq=3 dist=12 time=59.455 ms
unicast from 2001:700:1:7:211:d8ff:fe8f:1f9b, seq=4 dist=20 time=56.090 ms
multicast from 2001:700:1:7:211:d8ff:fe8f:1f9b, seq=4 dist=12 time=58.956 ms

--- 2001:700:1:7:211:d8ff:fe8f:1f9b ssmping statistics ---
5 packets transmitted, time 4744 ms
unicast:
5 packets received, 0% packet loss
rtt min/avg/max/std-dev = 56.090/57.963/62.466/2.296 ms
multicast:
3 packets received, 0% packet loss since first mc packet (seq 2) recvd
rtt min/avg/max/std-dev = 58.956/61.372/65.706/3.077 ms
```

As you can see we received both some unicast and multicast replies. This shows that we are able to receive multicast from the host we pinged. But it tells us more than that. It tells us that it took between one and two seconds for the multicast tree to be established from receiver towards the sender. Right before sending the first request, the ssmping client joins the group, causing construction of the tree to start. It got no replies to the two first requests, but it got one for the request sent approximately two seconds after it joined (seq 2). The two first packets probably got lost because tree was not in place yet.

Further we can see that the unicast replies have travelled 20 router hops, while the multicast ones only 12. This shows that multicast must have travelled a different path than the unicast, and since the number of multicast hops are that much smaller, it is likely that the multicast was tunnelled part of the way.

Finally we measure round trip times. Since what we measure is the time from when we send the unicast request until we receive the unicast and multicast replies, we will be able to compare the one way delay of unicast versus multicast. We can also see that the delay varies more for multicast, although we should measure a large number of packets to draw firm conclusions.

The asmping tool

In principle, asmping is very similar to ssmping. The difference is that it only joins a group, not a source and group. This in itself is not so interesting. The interesting part is that with Any Source Multicast (ASM) there are several possible forwarding paths and, and the path being used may change over time. By using asmping it is possible to some extent to see which path is being used, when changes in the forwarding paths occur.

Like ssmping, asmping tells us how many hops the unicast and multicast packets have travelled. In most cases the different possible forwarding paths will have different number of hops, and in this may help us deduce which path is taken and when. If one knows the topology, one should be able to tell with asmping exactly which path is being used and when. However, even without such intimate knowledge, one should still be able to deduce some information about the paths used.

Also note that some other interesting things might occur in the inter-domain case with Multicast Source Discovery Protocol (MSDP) for IPv4 and Embedded-RP for IPv6. Below is an example of inter-domain IPv4 where MSDP is involved.

```
$ asmping 224.3.4.234 ssmping.uninett.no
ssmping joined (S,G) = (158.38.63.22,224.3.4.234)
pinging S from 152.78.64.13
  unicast from 158.38.63.22, seq=1 dist=23 time=57.261 ms
  unicast from 158.38.63.22, seq=2 dist=23 time=56.032 ms
multicast from 158.38.63.22, seq=2 dist=7 time=207.876 ms
multicast from 158.38.63.22, seq=2 dist=7 time=208.567 ms (DUP!)
  unicast from 158.38.63.22, seq=3 dist=23 time=56.852 ms
multicast from 158.38.63.22, seq=3 dist=21 time=70.352 ms
multicast from 158.38.63.22, seq=4 dist=21 time=57.208 ms
  unicast from 158.38.63.22, seq=4 dist=23 time=57.910 ms
  unicast from 158.38.63.22, seq=5 dist=23 time=56.206 ms
multicast from 158.38.63.22, seq=5 dist=21 time=57.375
```

Note that the first multicast packet received is received twice, has very long delay and travels only 7 hops. All these effects come from the fact that MSDP is involved.

Not much testing has been done with asmping and embedded-RP (RFC3956) at the time of writing, but we feel that the tool offers some interesting possibilities, in that one can both chose the location of the source and the location of the RP; the RP is determined by the group address.

These tools are in fairly wide use today, with several hundred downloads, both for IPv4 and IPv6. This includes use on the M6Bone [3]. The tools are used both for testing on the Internet and on private networks and testbeds.

References

[1] The ssmping and asmping web page, *http://www.venaas.no/multicast/ssmping/*
[2] SSM-Ping: A Ping Utility for Source Specific Multicast, P. Namuburi, K. Sarac and K. Almeroth, the 3rd IASTED International Conference on Communications, Internet, and Information Technology (CIIT), St. Thomas, US Virgin Islands, USA, November 2004.
[3] The M6Bone network, *http://www.m6bone.net*

Vitae

Stig Venaas received his M.Sc. degree in mathematics from University of Trondheim in 1993. He has worked at UNINETT since 2000 and is mainly involved in IPv6 activities. He is active in TF-NGN, and also in the IETF where he is co-chair of the DHC working group.

Tim Chown received a PhD at the University of Southampton in 1991, where he now works as systems manager, researcher and lecturer. Most recently he led the university's involvement in the 6NET and Euro6IX IST projects. He is also active in TF-NGN and the IETF. His specific interests include IPv6, multicast, wireless networking and security.