SSM-Based Receiver-Controlled Communication in the Internet

Kamil Sarac
Department of Computer Science
University of Texas at Dallas

In this abstract, we present a snapshot of our ongoing work on developing a mechanism for a receiver-based communication service in the Internet. Our main goal is to give an end system the ability to control who to communicate with and when to communicate explicitly, and therefore, avoid potential attacks coming from other nodes in the network. In the case of a server providing a service over the network (such as a web server), attacks can be in the form of sending excessive packets to keep the server or the communication links around it busy. And, in the case of two end systems communicating with each other, attacks can be in the form of interrupting the communication between the two sites in several different ways. In our model, when two end systems agree on communicating with each other, they will be able to do so without experiencing interruptions from the others in the network.

In our approach, we use the Source Specific Multicast (SSM) service for providing a controlled communication service described above. SSM is designed to support one-to-many multicast applications in the Internet. The key characteristics of the SSM service model that we exploit for our purposes are (1) only the source S can send to an SSM channel (S,G), (2) data delivery to a receiver depends on receiver's joining the SSM channel, and (3) RPF based forwarding rule protects the communication against address spoofing attacks. Based on these characteristics, we formulate a controlled point-to-point communication service as follows.

First, we require the existence of SSM-based multicast support in the network. Then, we reserve a portion of SSM multicast addresses to support our communication service (e.g. 232.240/12, approximately one million addresses). Finally, when two remote end systems want to communicate with each other, each one creates an SSM channel and joins the SSM channel of the other. This way, they carry their communication over SSM channels. This procedure guarantees that sites receive packets only from each other and other end systems cannot easily interfere with the communication.

The second context for this model is controlling access to an application (or a web) server over the Internet. This control may be desirable in a number of application domains such as military or emergency preparedness applications where we may want to be able to access a critical information server anywhere and anytime. In addition, we may want to control/limit the usage of the server by others and avoid others from preventing us from accessing the server over the Internet. To achieve this, we can configure the network around the server to rate limit the unicast traffic coming to the server and configure our server to communicate via multicast as well as unicast. In this setting, the privileged users can first contact the server via unicast to identify an SSM channel pair to communicate as described above. As a result, due to the rate limit on the unicast traffic coming into the server site, the server will not be overwhelmed by unicast-only clients and will be able to communicate with the privileged clients via SSM. Finally, by announcing the server's network as multicast-reachable-only through BGP path announcements, we can completely eliminate unicast accesses to the server site. In this case, the server will only be reachable via multicast. Therefore, in order to send to the server, the remote sites will need the server to explicitly join their SSM channels. However, the server will still have unicast access to outside world.

The above mechanism protects end systems from receiving unwanted packets from remote systems in the network. This service can be very helpful in achieving a successful communication between two end points on the face of potential attacks that specifically attempt to block or deteriorate the communication between the two sites. However, one potential attack scenario that our approach may not be that effective is man in the middle attacks where the attacking site is on the direct path between the two sites and can easily interfere with the communication by inserting spoofed packets into the multicast forwarding tree on the correct RPF interfaces. This attack requires the adversary either to cause significant changes in the underlying unicast forwarding mechanism to re-route the packets through itself or requires him to have access to the network path between the two end systems.

We believe that achieving the first case is quite difficult and preventing the intruder from causing a service disruption in the second case is also quite difficult. In addition, our approach described so far does not provide data protection. That is, an adversary can join the SSM channel (S,G) of the source S and receive data from the network. Even though data confidentiality is not our primary concern in this work, we can use a mechanism to hide the channel information from others and make it difficult for them to join the channel. For this, we can use a mechanism to dynamically and deterministically change the channel address that is used during the communication and therefore make it more difficult for the intruder to keep track of the communication. In this regard, we believe that we can borrow available techniques that have been used in wireless communication such as frequency hopping techniques, etc. Finally, our approach can significantly increase the amount of multicast forwarding state in the network. In order to control the memory requirement by these states, one potential approach would be to aggregate multicast forwarding states in the routers.

1 E-mail: ksarac@utdallas.edu