Global Information Assurance Certification Paper

IPv4 Multicast Security: A Network Perspective

By Tom Bachert

Abstract:

Multicast holds great promise in reducing the network bandwidth required for simultaneous communication between multiple hosts. Documented routing protocols and distribution methods are now enabling multicast implementations to move out of the LAN arena and into the larger world of the internet. Multicast's methods of operation pose new and extended demands on security models developed primarily for unicast data transmission.

This paper examines the security implications of multicast communications as they relate to network management. It begins with a general description of multicast communications and then progresses to discussing multicast methods of operation within the Internet Protocol (IPv4) framework while contrasting them against the more familiar unicast operations. Security issues specific to multicast communications are identified and discussed. Possible solutions including the extension of IPsec to MIPsec are examined.

What is Multicast? :

In the most general terms multicast communications is the transfer of information in a single transmission to multiple receivers. A group of co-workers chatting around the office coffee- pot is simple example of multicast communications. The person speaking makes a single transmission, which is heard by all the members of the group. If the speaker were restricted to a unicast environment, he/she would be forced to seek out each individual member of the group to speak his/her message and thus would need to repeat the message several times. If we equate this speaker's words to bits and bytes we see that multicast communication can save considerable bandwidth in an electronic communication arena.

This simple example also illustrates other principles applying to multicast communication systems. Obviously a group of chatting co-workers is not likely to have a single talker (transmitter) but rather, most if not all of the listeners (receivers) will also want to participate in the conversation and thus become both transmitters and receivers. It should also be noted that membership in the group is not necessarily static. Perhaps one co-worker sees his/her boss approaching and thus decides to make a speedy retreat back to the sanctity of his/her cubicle while another co-worker joins the group while getting a morning caffeine fix. Indeed the group's membership might not even be known to all the participants as there may be someone lurking in a nearby cubicle listening to every word being spoken and perhaps even interjecting his/her own words in a disguised voice. Thus multicast communication can be used to save resources (words, bandwidth) but it also possesses some characteristics unique from unicast communications. These unique characteristics may undermine many assumptions on which standard communication security models are built.

IPv4 Multicast Operation:

Multicast's ability to simultaneously deliver a single stream of information to many recipients makes its use on data networks and the Internet quite attractive. It is most often associated with the transmission of multimedia information such as video and audio teleconferences or steaming applications delivering music or video to interested clients. However, other applications such as "one to many" file transfers, white board applications, and even some routing protocols such as OSPF have found multicast's ability to conserve bandwidth useful.

Multicast communications relies heavily on the idea of a "group". Within the IP framework a multicast group is a set of receivers interested in receiving a particular steam of data. The group is composed of an arbitrary number of members and may span large geographic areas as well as multiple areas of administrative control. Group members may join and or leave the group at will. The data stream received by the group members may originate from one or multiple sources and the sources may or may not be members of the group. [7; 13]

Communications within an IP network depend on source and destination IP addresses. IP addresses are used to communicate outside the local area network arena while MAC addresses, sometimes referred to as hardware addresses, are used to address packets within a local area network. Routers, which are used to interconnect multiple networks, map a IP address to an associated MAC address before forwarding the packet out their appropriate interface. (This translation of IP to MAC address will become important as we discuss some of the security implications later in this paper.) In the unicast scenario, receivers read messages from the network only if the message has a destination address matching the receivers own internal address. However, with multicast , all receivers within a given multicast group must read packets addressed to the group's address as well as packets addressed to the receiver's individual unicast address. When a host wants to join a group it modifies its IP stack so that it reads packets addressed to the group's address as well as packets addressed to its unicast address. A host, which wants to transmit to the entire group, uses the group's address as the destination address for the data it wants to send.

IPv4 addresses are 32 bits in length. In order to make them more humanly readable they are generally broken into 4 chunks of eight bits (an octet) and each octet is expressed as the decimal equivalent of its binary value. The octets are then separated from each other with period. This results in a range of possible addresses from 0.0.0.0 to 255.255.255.255. This range of values has been segregated into different classes to be used for different purposes. Class "D", which has the range of 224.0.0.0 to 239.255.255.255, is reserved for use as multicast group addresses. The Internet Assigned Number Authority (IANA) has further segregated this range of multicast addresses for specific purposes. [7; 9; 13] Figure 1 lists these reserved ranges and their intended uses.

Table 1. [7; 9]

Multicast Address Assignments

Name	Range	Use Description
Multicast Traffic	224.0.0.0-239.255.255.255	All multicasts
Reserved Link Local	224.0.0.0-224.0.0.255	Local network protocols
Globally Scoped	224.0.1.0-238.255.255.255	Internet spanning mcast groups
Source Specific	232.0.0.0-232.255.255.255	One source to multiple receivers
GLOP	233.0.0.0-233.255.255.255	Organizations with assigned AS#
Limited Scope	239.0.0.0-239.255.255.255	Used only within local domain

Hosts, which wish to participate in a multicast group, may reside on different network segments or possibly even across the Internet and on networks administered by different organizations. Group members need a method of informing the routers and switches separating them from other group members of their need to receive the group's data. The Internet Group Management Protocol (IGMP) is used by hosts to inform routers of their need to receive data addressed to a specific multicast group. IGMP is a local network protocol and is not forwarded past the local router. When a host wishes to join a multicast group it sends a "join" IGMP message to the network's router. The router then uses a multicast routing protocol to inform other routers of its desire to receive packets destined for the multicast group. The router may also use IGMP to query the attached network segments for specific group members. Currently IGMP has progressed through three versions. Version 1 is currently all but obsolete. Version 2 is the most widely used at the time of this writing and provides a method for hosts to communicate their desire to leave a specific group. Version 3 adds the ability for hosts to selectively receive a groups packets based on the source address of the packets. It is hoped this ability can be used to increase security especially by "source specific" multicast groups. [14] Version 3 IGMP is currently supported by the latest versions of the Windows, Macintosh and UNIX operating systems. [7]

It should be emphasized that IGMP is used by hosts to communicate group membership information to routers, which are IP or layer 3 devices. Most modern network topographies often have layer 2 switches between hosts and their routers. Since IGMP is a layer 3 protocol these switches are not aware of the "join", "leave" and "query" messages being exchanged between the hosts and routers. These switches will be unaware that a host attached to one of its ports desires to receive packets addressed to a specific group MAC address . By default a switch will flood frames addressed to unknown MAC addresses out all of its ports. Thus if a switch has 48 active ports and one port is attached to a host which has joined a specific multicast group, the packets addressed to that group will be forwarded out all 48 ports despite the fact that the other 47 hosts are not members of the group. This behavior is synonymous to our previous example where the conversation being held around the coffeepot is overheard by non-group members sitting in nearby cubicles. Obviously this is not desirable behavior from a bandwidth or security point of view.

Once a router receives an IGMP join message from a host, the router needs a method to inform all the routers upstream (towards the source(s) of the group) that it desires to have the traffic forwarded to it. Routers must also determine which of its interfaces should forward a given multicast packet. Multicast poses a unique problem to routers. Unicast routing is based on the unique destination address in an IP packet.

The router looks at this destination address and uses its unicast routing algorithm to decide which interface leads to the destination. Multicast packets have the non-unique group address in the destination field. Multicast packets may need to be forwarded out multiple interfaces to reach all interested hosts. However, a router must take care not to forward the packet back in the direction of the source or a routing loop would result.

Multicast routing protocols build distribution trees to meet these challenges. Two generic styles of distribution trees are used; source trees and shared trees. Source trees have their root at the source of the multicast data. Data packets flow from the source through the network routers, dividing into separate "branches" at the routers which have interested downstream receivers on separate interfaces. The branches terminate onto "leaf" network segments on which the individual receiving hosts reside. Multicast groups, which have multiple data sources, would be made up of multiple distribution trees with each tree representing the shortest path between the source and its receivers. Shared trees use a single root for all of the group's data sources. The base of a shred tree is referred to as the group's rendezvous point (RP). Branches extend from this RP through the network routers, finally terminating at the receiving hosts located on the leaf network segments. Since source based trees connect each source to it receivers via a "shortest path" they have the advantage of minimizing latency in the delivery of their information. However shared trees place higher memory demands on the router since they require the storage of tree information for every source in a group. Shared trees reduce this memory demand by storing only one tree that is shared by all the sources in a group. However shared trees do not guarantee that the packets from a given source will take the "shortest path" to a given receiver and they also create a possible bandwidth bottleneck at the rendezvous point. Both types of trees must "grow" new branches and "prune" old branches as group member join or leave the group [7].

There are several multicast routing protocols available for use today including Distance Vector Multicast Routing Protocol (DVMRP), Multicast Open Shortest Path First (MOSPF), Multicast Border Gateway Protocol (MBGP) and Protocol-Independent Multicast (PIM). These routing protocols differ in their use of shared or source rooted trees and the methods employed to graft or prune branches to and from their distribution trees [7]. The original Multicast Backbone project (MBONE) made use of DVMRP which relied on tunnels through non-multicast enabled networks to link together multicast networks. This approach has difficulty scaling when large numbers of administratively diverse networks need to be interconnected. MOSPF and MBGP are extensions to existing unicast routing protocols with MBGP becoming quite popular for use in routing multicast traffic between different administrative domains. PIM is being developed by the IETF's Inter-Domain Multicast Routing working group as a standards track protocol to provide scalable multicast routing within the Internet [15]. PIM is really two protocols, PIM sparse mode and PIM dense mode. PIM sparse mode requires an explicit join before a new branch is grafted onto the distribution tree whereas routers using PIM dense mode forward a groups traffic out all interfaces and then prunes branches from the tree if there are no listeners on a given interface [13].

Multicast Security Issues:

The differences between the operation of multicast from unicast in an IP environment create some unique security related issues. Common issues such as message authentication and privacy become more complex in a multicast environment. The multicast environment also introduces other concerns including group access control, group center trust and router trust. The dynamic nature of group membership further complicates these issues [8].

Unicast communications by definition are point to point. The sender addresses the message to the intended receiver's known and hopefully unique address and relies on internal mechanisms of the network to deliver it only to the correct recipient. These mechanisms provide a very rudimentary level of security. Duplicated addresses are to some extent detected, routing tables keep track of where the recipient is located and layer-2 switches maintain tables recording the port to which the intended recipient is connected. Unicast also can make use of the well-defined, connection oriented, reliable, Transmission Control Protocol (TCP) with its session management, and data sequencing capabilities. Although all these structures can be circumvented, they do contribute to the depth of security possible in unicast communications. They also provide a framework on which more robust security tools can be built.

Multicast, in its most basic form, does not provide any assurance that data packets are read only by the intended recipient(s). More often than not, multicast transmitters don't even know who the recipients are or where they reside geographically. They simply send packets addressed to the group destination address and spew them onto the attached network. Any other host reachable via this network can receive the packets by simply instructing its IP stack to read packets addressed to the group (i.e. join the group). Since the group address (the destination address of the packet) may have legitimate receivers on multiple router interfaces, multicast routers must base their forwarding decisions on the source address rather than the destination address. This means that the packet might be forwarded out any or all of the router's interfaces with the exception of the interface leading back to the source [10].

If the distribution of multicast data at the Network layer is rather unrestrained, the situation at the Data Link (layer 2) level is a virtual free for all. Part of the problem comes about from the need to map Network layer (IP) packet addresses to Data Layer (ethernet) frame addresses. An IP packet has 32-bit source and destination address fields. Since all multicast packets use class D addresses, the four high order bits of these address fields are fixed as 1-1-1-0. This means that 28 bits are left in the field to designate specific multicast group addresses. IEEE 802 LANs such as ethernet use frames with 48-bit address fields. However, the first 24 bits of these frames are used to encode an Organizationally Unique Identifier as well as to designate the frame as a multicast or unicast frame. One additional bit of the remaining 24 bits is reserved, thus only 23 bits are available to use in creating group addresses at the Data Link layer. When a layer 3 IP multicast packet gets mapped to a layer 2 frame, its 28 bits of address information needs to be squeezed into the 23 bits available in the frame. This results in a loss of 5 bits of address information and thus a single ethernet (layer 2) multicast group address maps to 32 IP (layer 3) multicast groups [13]. This loss of address granularity means that a host, which has joined a single IP multicast group, may actually receive data from 31 other groups in addition to the data from the group it

has joined. The receiving host's upper layer processes are left to recognize and appropriately handle this unwanted data. In other words, the people having the discussion around the coffee pot are not only hearing their discussion but are also able to overhear what is being said in the adjoining conference room and nearby offices.

Multicast data may further find its way to unintended receivers when it reaches an ethernet switch. Switches learn which layer 2 addresses are associated with a given port by looking at the source addresses of the frames received on the port. It stores this information in a table. When a new frame is received the frame's destination address is looked up in this table to determine which port should receive the frame. If the address of the destination is not in this table, the switch by default floods the frame out of all of its ports. Multicast frames have the group address for a destination. Since this group address never is used as a source address, the switch's table never has the group address associated with a specific port and thus multicast frames are flooded out all the switch's ports [1]. Such flooding of data not only results in data reaching unintended hosts but also provides a gateway to launch a DoS attack against all the hosts attached to the switch.

The preceding discussion illustrates the difficulties encountered when attempting to restrict which machines on a network receive a specific, possibly private, multicast group's data. The source of the data transmitted to the group can also be suspect, as the source address of multicast data can be 'spoofed" just as it can be in a unicast environment. Privacy, authentication, integrity and non-repudiation can not be guaranteed in either a unicast or multicast environment by simply relying on the source and destination addresses utilized in the layer 2 or layer 3 network protocols. Within the unicast world, other tools such as connection oriented Transport layer protocols (TCP), encryption and Public Key Infrastructure (PKI) have been developed to help provide these aspects of security. However it is often quite difficult to adapt these same tools to work properly in the multicast environment.

For instance, multicast's characteristic of transmitting one copy of a message to multiple receivers make it very difficult to develop a reliable, connection oriented transport protocol similar to TCP. TCP and other connection oriented transports rely on specific 'hand-shaking" sequences between the transmitter and receiver to establish and tear down point-to-point virtual connections A multicast transmitter is not sending data to a single destination but rather to a group of destinations (perhaps numbering in the thousands) A single shared virtual connection between the transmitter and all group members will not suffice since if connectivity to any single group member were lost the entire connection would need to be flagged as down. The alternative, establishing virtual connections between the source and each group member, essentially reverts the communications back to unicast scenario.

TCP uses the interaction of packet sequence numbers and acknowledgement messages to insure data integrity between two virtually connected unicast hosts. This mechanism does not work properly within a multicast environment. Unicast hosts communicating via TCP use a 'sliding window' to insure transmitted packets are received correctly at the other end. If data is lost or corrupted in transit, the transmitter becomes aware of the problem when it fails to receive an acknowledgment for the data from the intended receiver. However, a multicast data source is attempting to send data to multiple receivers sharing a common address (the group address). Some of the

receivers might receive the transmission error free while other intended receivers may not see the transmitted data at all. Furthermore, even if all the intended receivers did receive the transmission error correctly, they would then all send acknowledgments back to the original source. If the multicast group consists of a large number of members, these acknowledgments could overwhelm the source host or the network to which it is attached. This would essentially be a DoS attack by design.

The dynamic and diverse nature of many multicast groups also makes it difficult to implement standard encryption and authentication infrastructures [8]. Multicast groups can be large or small. Some groups, such as streaming audio or video presentations, may consist of one transmitting source and many listening members. Other groups might be used for an audio or videoconference where all members are both sources and receivers. Some groups may be more or less permanent while other may be created and torn back down within an hour. Some multicast groups may have hosts joining and leaving the group at will while others may have rather static memberships. Encryption and authentication schemes which involve sharing a key can work reasonably well in groups of limited size with fairly static membership but do not scale very well to large dynamic groups. Shared key approaches also do not provide a means of authenticating the source of a multicast packet since all group members know the key and thus any one of them could have been the source. Multicast security models which make heavy use of Public Key Infrastructure tend to require a high level of computational overhead which can be a serious problem in large groups with members consisting of hosts with varying degrees of computing power.

Multicast Security Solutions:

Limiting where within the network environment a given multicast group's data is permitted to flow is an important first step in improving multicast security. The data comprising a teleconference between a companies top executives should certainly not exit the company's internal network and should probably not even be accessible to all of the company's internal subnets. Internal subnets supporting essential, high bandwidth communications between computational nodes should not be burdened with carrying unneeded multicast traffic. There are several mechanisms available to limit the scope of where multicast traffic traverses. The two primary methods of scoping multicast traffic at the IP level (layer 3) are TTL scoping and Administrative scoping [1].

TTL scoping was the original method used to limit where multicast traffic was allowed to flow within a network and is still widely deployed today. It makes use of the 8 bit Time To Live (TTL) field within the IP packet's header. Routers examine this field when they receive a packet on an interface and forward the packet if the value stored in the TTL field is at least 1. If a packet is to be forwarded, the router decrements the TTL value by 1 before forwarding it out the outbound interface. The router discards packets that have TTL values of zero. This process limits how many routers a packet may pass through before it is dropped from the network. Applications sourcing multicast packets can be configured to set specific initial values in the TTL field. Multicast enabled routers allow TTL thresholds to be assigned to specific interfaces and will only forward a multicast packet across this interface if the TTL field is greater than the configured value [13]. Thus multicast applications can use their ability of setting initial TTL values to regulate how many router interfaces their traffic will be allowed to pass through. In order for TTL scoping to function predictably, coordination of the application assigned TTL values and the thresholds configured on the router interfaces must exist. TTL values and their conventional scoping boundaries are listed in Table 2.

Table 2. Conventional TTL Values and Scoping Limits [13]

TTL Value	Scoping Limit
0	Restricted to local host
1	Restricted to same subnet
15	Restricted to same site
63	Restricted to same region
127	Worldwide
191	Worldwide with limited bandwidth
255	Unrestricted

Scoping multicast traffic by using the TTL field has several shortcomings and limitations. TTL-based scoping can not easily handle overlapping regions. Router interfaces which have TTL thresholds configured, apply that threshold limit to all

packets regardless of which multicast group the packet is destined for. TTL-based scoping also requires coordination between the intranet manager, who configures the TTL thresholds on the router interfaces, and the people controlling the applications which set packet's initial TTL value. This method of multicast range limiting requires the source to be aware of the TTL limits imposed by the network and to act in accordance with them. Because of these limitations, TTL-based scoping does not scale well to large, complex networks and should only be used as a 'safety net' device at the outside edge of such networks [13]. Perhaps more importantly, TTL scoping causes serious problems with the pruning of branches from the distribution trees created by certain multicast protocols which use a broadcast-and-prune forwarding algorithm such as DVMRP [1].

Administrative scoping limits the flow of multicast traffic based on the packet's IP destination address. (I.e. the group address) It provides for the establishment of well-defined boundaries, which may overlap, and allows the same multicast group addresses to be used by multiple administrative domains at the same time. The multicast address range 239.0.0.0 to 239.255.255.255 (i.e. 239.0.0.0 /8) has been reserved for administratively scoped multicast groups. The IETF's RFC 1884 suggests that this address range be further divided as illustrated in Table 3.

Table 3. Administratively Scoped Address Ranges [13]

Address Range	Scope Usage
239.192.0.0 to 239.195.255.255	Local to Organization
239.253.0.0 to 239.253.255.255	Local to Site
239.255.0.0 to 239.255.255.255	Local-can not pass any administratively scoped boundary

The network manager establishes the boundaries for these scope regions by defining boundary definitions on specific router interfaces. He/she may also further subdivide these regions into smaller ranges in order to provide for specific organizational requirements. For example, a school might decide to reserve the address range 239.253.0.0 to 239.253.7.255 for videoconferences between staff members. The appropriate router interfaces could then be configured to not allow data packets addressed within this range to pass through the interface. Router interfaces can be configured as the boundary for multiple scoped areas and given LAN segments can be included in multiple scoped areas.

Administrative and TTL scoping provides a means to administrate the flow of multicast traffic within defined IP (layer 3) boundaries but it does not provide a method of limiting this traffic at the ethernet or layer 2 level. As stated previously, layer 2 switches decide which ports to forward packets out of based on tables of layer 2 addresses. These tables are built by listening to the traffic received on each port and learning which source addresses are being heard on each given port. When a new frame arrives, the switch looks at the destination address, consults its internal table of

ports/associated layer 2 addresses, and forwards the packet out the port which has an associated layer 2 address matching the frame's destination address. If an incoming frame's destination address does not match any address in this table, the frame is forwarded out ALL of the switch's ports. Since multicast frames are addressed to group addresses and group addresses are never seen in the source address field, the switch's port/layer 2 address table will never associate a multicast frame's destination to a given port. This results in all multicast traffic being forwarded out all the ports on the switch.

One could avoid this problem of flooding multicast data to all ports of a switch by manually configuring the addresses of joined groups into the switch's port/layer-2 address table. However this is labor intensive and becomes quite unmanageable as the number of multicast hosts and groups increases. Cisco, a leading manufacturer of switches, developed the Cisco Group Management Protocol (CGMP) to help solve this problem. CGMP must run on both the switch and the multicast router to which it connects. The router uses CGMP to inform the switch what multicast groups have received IGMP join messages from the switch's own ports [1].

A non-proprietary and more direct solution is to give the switch the ability to directly listen in on the IGMP messages, which are being exchanged between the switch and the multicast router. This is referred to as "IGMP Snooping" [1]. IGMP is the protocol used by layer 3 routers to learn which interfaces connect to hosts interested in receiving traffic for specific multicast groups. When a host joins a multicast group it sends an IGMP "join" message to its nearest multicast router. The router then begins forwarding traffic addressed to this group out the interface, which received the join message. Most modern layer-2 switches can be configured to use IGMP snooping to learn which ports are interested in receiving data from specific multicast groups. Switch ports with attached hosts interested in participating in the multicast environment should have IGMP snooping enabled.

Even after constraining the flow of multicast data through the use of scoping and IGMP snooping, multicast still generally relies the unreliable, connectionless UDP mechanism for its transport [2]. Unreliable transport mechanisms, by definition, do not guarantee the integrity or delivery of transmitted data. Obviously from a security point of view, it is desirable to use a reliable transport mechanism if data integrity is an important consideration. In recent years much effort has been directed towards developing reliable multicast protocols with some concept of a session or connection to answer these concerns. One of the better known of these is Scalable Reliable Multicast (SRM). SRM recognizes that a multicast sender transmitting to a large number of receivers would be overburdened if it were solely responsible for guaranteeing delivery to all receivers. SRM requires the receivers to actively participate in the reception and repair of the data. [1] When a receiver detects that it has lost a packet, it generates an SRM "repair request" addressed to the group. Nearby group members, hearing this repair request, retransmit the "repair data" to the group. If several group members notice they are missing data, the repair request first issued causes other members missing data to suspend the issuing of repair requests of their own. This protects the network from being flooded with multiple repair requests if a large number of members fail to receive the same piece of transmitted data. Since the reply to the repair request is transmitted to the group address, all the members receive

the reply, regardless of whether of not they have issued their own repair request. This allows them to obtain the missing data without ever having requested its retransmission. The network is protected from a possible flood of repair replies with a similar mechanism. The first repair reply issued to the group causes other members which have the requested data, to suppress the sending of their replies. The fact that SRM uses a negative acknowledgement mechanism to insure data delivery coupled with the fact that group members actively participate in repairing each other's data, prevents the sender from being overrun with acknowledgements and/or repair requests. SRM's procedure of sending repair requests to the entire multicast group does create some problems. If a several members are located on an unstable segment of the network, they will generate a disproportionate number of repair requests, which will be heard by all the group members. Later versions of SRM reduce this problem by implementing the concept of a "local recovery group" which limits the scope of the repair requests/replies [5].

Several other protocols have been developed in order to provide a reliable transport within a multicast environment. The decision as to which reliable transport mechanism to be used by an application should consider the application's needs as well as the services offered by the various transport protocols. Reliable Multicast Transport Protocol (RMTP) developed by AT&T Bell Laboratories is similar to SRM but uses the concept of "designated receivers" to scope repair traffic. It also can use either multicast or unicast to make repairs depending on the number of receivers in need of repair [4]. Multicast Transport Protocol (MTP and MTP-2) synchronizes the ordering of received messages by designating group member roles of master, producer or consumer [3]. Reliable Adaptive Multicast Protocol (RAMP) has become increasing popular for use by collaborative applications because of its ability to be both sender and receiver reliable. When RAMP is used a sender knows the unicast address of all of its receivers [13]. The above listed protocols exist on the Transport layer (layer-4) of the ISO Layered Protocol model. One protocol commonly used to transmit audio and video data, which does not firmly reside at this layer, is the Realtime Transport Protocol (RTP). RTP is an extension used with application layer protocols to provide real-time transport of data across either a unicast or multicast network [13]. RTP is not a reliable transport because it does not provide a mechanism to guarantee the successful delivery of the data. However, it does provide many useful transport services such as identification of group members, data sequence numbering, identification of data source and tools for monitoring how well the transmitted data is being received. Applications are free to use these services to create their own mechanisms for providing reliability to the transport.

Multicast data, even after being properly scoped and provided with an appropriate transport method, is still vulnerable to the same threats faced by unicast data. Indeed, because of its variability and wide dispersion, it is probably more at risk. The fact that multicast data will quite likely transverse multiple administrative domains with differing security policies makes it essential that the security protocols used to protect it be flexible [2]. They should be able to use various authentication methods and cryptographic algorithms. IPsec is a standardized framework developed to provide security mechanisms within the IP protocol suite. One of IPsec's design goals is to define security protocols which can operate independently from the cryptographic,

authentication, and key exchange mechanisms employed in securing specific applications, so its use seems logical in a multicast environment [2]. However, IPsec depends on the establishment of Security Associations (SAs) between individual hosts participating in secure communications. A SA is the mechanism used to agree which specific cryptographic, authentication algorithms and security features a host will use when securely communicating with another host. Unicast SAs are established between a single sender and receiver and are identified by a "triple" consisting of the destination address, the security parameter index (SPI) and its protocol (ESP or AH). The SPI and its associated protocol are selected by the receiver who then negotiates with the sender other parameters such as the encryption algorithm [6]. However, RFC 2401 states, "IPsec SA management mechanisms currently are defined only for unicast SAs" [6].

The unicast SA model does not fit well with multicast where there are usually multiple receivers listening to each sender. The Internet Research Task Force's (IRTF) Secure Multicast Group (SMuG) has proposed an extension of the IPsec framework to use in the multicast environment, which is appropriately named "MIPsec". MIPsec uses a Group Security Association (GSA) in much the same way that IPsec uses a SA. The GSA, like the SA contains source and destination addresses, security parameter indexes (SPIs) and the cryptographic policies and keys to be used for the communication. However, since a GSA specifies the parameters to be used in the communications between multiple hosts, it is actually an aggregation of the multiple SAs needed for the communications between the various hosts [11]. SMuG's Group Key Management Building Block (GKMbb) proposes that a conceptual entity identified as the Group Control and Key Server (GCKS) be used to manage and distribute the GSAs utilized by a secure multicast group. The GCKS acts as a control point for the initialization, authentication and security management of the multicast group. The GSA generated by the GCKS would be an aggregation of at least three SAs, which are classified into three standard types as described in Table 4 [11].

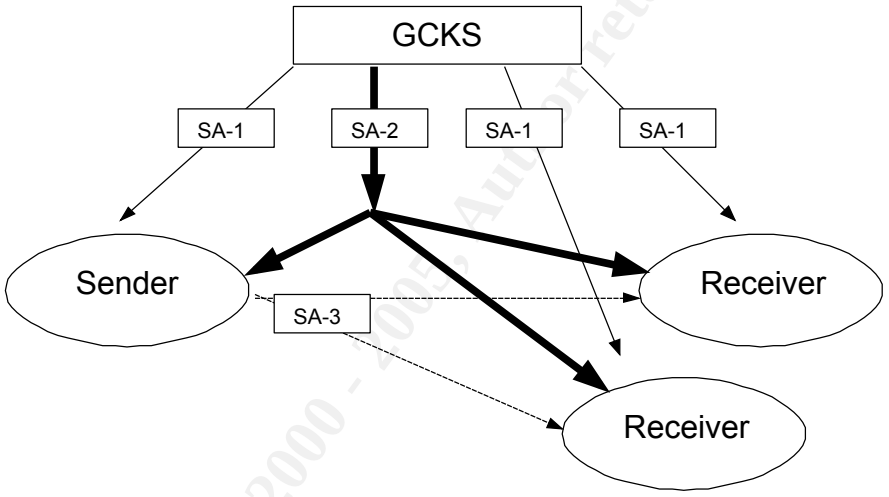
Table 4. Security Association Types and Usage

TYPE	USAGE
SA-1	Established between members and GCKS when group joined; unicast
SA-2	Established between GCKS and members for re-keying and key management
SA-3	Established between members of group for group traffic

When a host application joins or initiates a secure multicast group it establishes a SA-1 with the GCKS via unicast. During this process the GCKS authenticates the host according to established policy procedures and exchanges the keying and security parameters to be used by the group. The GCKS creates the GSA for the group, which contains at least three SAs corresponding to the types enumerated in Table 4. As already mentioned, the type SA-1 is used to secure the information exchanged between the GCKS and the individual sender/receiver hosts. This SA may exist throughout the lifetime of the group or it may expire after the initial exchange occurring between the GCKS and the host. The SA-2 exists to support key management

activities initiated by the GCKS and pushed out to the group members. A group's policy may specify that the cryptographic keys change whenever a member leaves the group (forward re-key) or whenever a member joins the group (reverse re-key). The SA-2 is used to secure the information needed to achieve such re-keying activities. At least one SA-2 must exist in every GSA (multiple may exist depending on the application). The actual data exchanged between the group members is protected by one or more SA-3s (one SA-3 if all group members share a group key, multiple SA-3s if each sender uses a different key). The GCKS establishes and distributes the SA-3(s) used but does not use it for its management functions. Diagram 1 illustrates the relationship between the SA types and the various components of a secure multicast group [11].

Diagram 1 SA Type Relationships in A Multicast Group



The unicast, IPsec framework operates in either a tunneled or a transport mode. Tunnel mode must be used whenever one or both endpoints of a SA are a security gateway such as a firewall or router providing security services. In tunnel mode an application's data is placed into an IP datagram containing the normal source and destination addresses of the sender and receiver. The security gateway, upon receiving this packet, encapsulates it into another IP packet containing the source and destination addresses of the two communicating gateways. The original datagram, including the addresses of the original source and destination, are thus protected by the security protocol (ESP or AH) being used. Transport mode IPsec, establishes a SA between the original end point hosts and thus the source and destination addresses are not protected by the security protocol being employed [6]. Transport mode is used when the two endpoint hosts are not operating as security gateways. Since multicast group members are end hosts and not security gateways, the group security associations (GSAs) they create operate in transport mode [12]. Since transport mode does not hide the addresses of the original and final endpoints, it is vulnerable to traffic analysis techniques. If this is a serious concern multicast traffic can be further protected by routing it through an unicast IPsec tunnel established between two security gateways [12].

Sources Cited:

1. Cisco Corporation. “Multicast in a Campus Network: CGMP and IGMP Snooping.” URL:http://www.Cisco.com/warp/public/473/22.html (30 Aug. 2002).
2. Kruus, Peter. A Survey of Multicast Security Issues and Architectures. Washington DC: National Research Laboratory. 1998.
3. Armstrong, S; Freier, A; Marzullo, K. “Multicast Transport Protocol.” RFC-1301. Feb. 1992. URL: ftp://src.doc.ic.ac.uk/rfc1301.txt (8 Aug. 2002).
4. Bell Laboratories. “RMTP: A Reliable Multicast Transport Protocol.” 12 Aug. 1997. URL:http://www.bell-labs.com/project/rmtp/rmtp.html (21 Aug. 2002).
5. Ching-Gung, Liu; Estrin, Deborah, ShenKer, Scott; Zhang, Lixia. “Local Error Recovery in SRM: Comparison of Two Approaches.” 1997. URL: http://www.usc.edu/dept/cs/Technical_reports.html (27 Aug. 2002).
6. Kent, S; Atkinson, R. “Security Architecture for the Internet Protocol.” RFC-2401. Nov. 1998. URL: http://www.faqs.org/rfcs/rfc2401.html (22 Aug. 2002).
7. Cisco Corporation. “IP Multicast Technology Overview.” 18 April 2002. URL: http://www.cisco.com/warp/public/cc/pd/iosw/tech/ipmu_ov.htm (28 Aug. 2002).
8. Canetti/Lambda, Ran; Garayy, Juan; Itkisv, Gene; Micciancіox, Daniele; Naor, Moni; Pinkask, Benny. “Multicast Security: A Taxonomy and Some Efficient Constructions.” URL: http://www.research.ibm.com/security/multicast.ps (2 May 2002).
9. Internet Assigned Number Authority. “Internet Multicast Addresses.” 2 May 2002. URL: http://www.iana.org/assignments/multicast-addresses (13 May 2002).
10. Cisco Corporation. “Internet Protocol Multicast.” URL: http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/ipmulti.htm#xtocid2 (6 May 2002).
11. Harney, Hugh; Baugher, Mark; HardJono, Thomas. “GKM Building Block: Group Security Association (GSA) Definition.” Sept. 2000. URL: http://www.securemulticast.org/draft-irtf-smug-gkmbb-gsadef-01.txt (3 May 2002).
12. Hardjona, Thomas. “Group Security Association (GSA) Definition for IP Multicast.” 25 Feb. 1999. URL: http://www.securemulticast.org/draft-irtf-smug-gsadef-00.txt (3 May 2002).
13. Maufer, Thomas. Deploying IP Multicast In The Enterprise. Upper Saddle River, NJ: Prentice Hall Inc, 1997.
14. Cain, Brad; Deering, Steve; Fenner, Bill; Kouvelas, Isidor; Ericsson, Ajit. “ Internet Group Management Protocol Version 3.” April 2002. URL: http://search.ietf.org/internet-drafts/draft-ietf-idmr-igmp-v3-10.txt (15 May 2002).
15. PIM Working Group. “Protocol Independent Multicast (pim).” 17 April 2002. URL: http://www.ietf.org/html.charters/pim-charter.html (28 June 2002).