Ground segment protocol

Communications department

November 8, 2016

1 Protocol model/suit

In order to find the protocols that will rule the ground communications of the network, it has been studied 4 protocol models and suits. It has to be found the advantages and disadvantages of each one and then, assess which suits better to the porpouse. The models and suits are:

- OSI
- TCP/IP
- NetBEUI
- IPX/SPX

OSI and TCP/IP have a complex structure which it is ideal for large networks. Althoght, this complexity makes the protocols inefficient in small networks. On the other hand NetBEUI and IPX/SPX protocols are simpler and optimum in simple comunications. They cannot work in a large network unless adding extensions.

In the begining of the network workint, the ground segment would not be so large as could be the space segment. Since there wont be lots of Ground Stations (3 at the begining, but it cuould increase) the big part of nodes will be the clients. If it is want to be versatile and adapt to the demand it has to be implemented protocols of the TCP/IP suit or based in the OSI model. Although these 2 will be more dificult to implement and configuate, it will be the better option to ensure a good coverege for the demand and a friendly use to the costumers.

The main diference between OSI and TCP/IP is that the first is a model and the second is a suit of potocols. OSI is structured in 7 layers, and it descives how these layers should work. It is a theorical guide for build a protocol but nobody had never implemented a complet OSI protocol. TCP/IP is structured in 4 layers and it is formed by a family of protocols which can be used in this layers. In the practice almost every network is ruled by TCP/IP protocols.

The decicison between OSI model and TCP/IP suit could be resumed as making a protocol or use existing ones. The work of the ground segment is not really diferent of many existing systems, so the better option is to use the existing TCP/IP protocols. It has to be found which fits better to the system in every layer and adapt it if it is need.

2 TCP/IP

The TCP/IP protocol suite provides end-to-end data communication specifying how data should be packetized, addressed, transmitted, routed and received. This functionality is organized into four abstraction layers which are used to sort all related protocols according to the scope of networking involved. From lowest to highest:

1. Link layer
2. Internet layer
3. Transport layer
4. Application layer

2.1 Link layer

The link layer defines the networking methods within the scope of the local network link on which hosts communicate without intervening routers. This layer includes the protocols used to describe the local network topology and the interfaces needed to effect transmission of Internet layer datagrams to next-neighbor hosts.

The most used prtocols in this layer are stuied and presented in order to find the better option for the ground segment communications.

Ethernet

Over the years, Ethernet, which is technically IEEE 802.3 CSMA/CD LANs, has become the most commonly used standard for enterprise networks. These networks carry voice, graphics, and video traffic. Today's Ethernet networks run considerably faster than the original Ethernet. The most common top speed, for example, is a full thousand times faster than the original Ethernet — 10 gigabits per second (Gbps) versus 10 megabits per second (Mbps). With so much data being carried, the potential increases for more and more packet collisions in the collision domain. When a collision occurs, both data frames must be re-sent and this cuts down drastically

Systems communicating over Ethernet divide a stream of data into shorter pieces called frames. Each frame contains source and destination addresses, and error-checking data so that damaged frames can be detected and discarded; most often, higher-layer protocols trigger retransmission of lost frames. As per the OSI model, Ethernet provides services up to and including the data link layer.

IEEE 802.11

IEEE 802.11 is a set of media access control (MAC) and physical layer (PHY) specifications for implementing wireless local area network (WLAN) computer communication in the 900 MHz and 2.4, 3.6, 5, and 60 GHz frequency bands.

With this connection the client would have a range of arround 60 m and a data rate that can arriive to 5 Gbit/s.

Frame Relay

Frame Relay is a standardized wide area network technology that specifies the physical and data link layers of digital telecommunications channels using a packet switching methodology. Originally designed for transport across Integrated Services Digital Network (ISDN) infrastructure, it may be used today in the context of many other network interfaces.

This system requires an economical hardware, but it provides to the user a data rate in the order of 500 Mbit/s.

ATM

Asynchronous Transfer Mode (ATM) is a telecommunications concept for carriage of a complete range of user traffic, including voice, data, and video signals. It was designed for a network that must handle both traditional high-throughput data traffic (e.g., file transfers), and real-time, low-latency content such as voice and video.

The reference model for ATM approximately maps to the three lowest layers of the ISO-OSI reference model: network layer, data link layer, and physical layer.

ATM provides functionality that is similar to both circuit switching and packet switching networks: ATM uses asynchronous time-division multiplexing, and encodes data into small, fixed-sized packets (ISO-OSI frames) called cells. This differs from approaches such as the Internet Protocol or Ethernet that use variable sized packets and frames. ATM uses a connection-oriented model in which a virtual circuit must be established between two endpoints before the actual data exchange begins.

The ATM system can work i data ates between 1 and 50 Mbit/s

Conclusion

The better option for the Astrea system will be that the clients could enter the wide with their own internet local network. It will be easy and frendly for the client since it can access the service with any especial hardware. He/she would only need a computer and its own connection to internet. The client will be free for using a LAN (Ethernet) a WAN (IEE 802.11), or other variants. This systems are able to work at 25 Mbit/s, which is the minimum data rate that it has to be ensured. Despite this, for avoiding conflicts, it has to be informed to the client that is required a local connection of at least 25 Mbit/s.

For the Ground Station nodes the better option will be to use Ethernet system beacouse it is the sistem which allows a higher performance in terms of data rate.

2.2 Internet Layer

The internet layer is a group of internetworking methods, protocols, and specifications in the Internet protocol suite that are used to transport datagrams (packets) from the originating host across network boundaries, if necessary, to the destination host specified by a network address (IP address) which is defined for this purpose by the Internet Protocol (IP). The internet layer derives its name from its function of forming an internet (uncapitalized), or facilitating

internetworking, which is the concept of connecting multiple networks with each other through gateways.

The internet layer has three basic functions:

- For outgoing packets, select the next-hop host (gateway) and transmit the packet to this host by passing it to the appropriate link layer implementation.
- For incoming packets, capture packets and pass the packet payload up to the appropriate transport layer protocol, if appropriate.
- Provide error detection and diagnostic capability.

IPv4

IPv4 is the most widely deployed Internet protocol used to connect devices to the Internet. IPv4 uses a 32-bit address scheme allowing for a total of 2^{32} addresses (just over 4 billion addresses). With the growth of the Internet it is expected that the number of unused IPv4 addresses will eventually run out because every device -including computers, smartphones and game consoles- that connects to the Internet requires an address.

IPv6

IPv6 is the successor to Internet Protocol Version 4 (IPv4). It was designed as an evolutionary upgrade to the Internet Protocol and will, in fact, coexist with the older IPv4 for some time. IPv6 is designed to allow the Internet to grow steadily, both in terms of the number of hosts connected and the total amount of data traffic transmitted.

While increasing the pool of addresses is one of the most often-talked about benefit of IPv6, there are other important technological changes in IPv6 that will improve the IP protocol:

- Auto-configuration
- No more private address collisions
- Better multicast routing
- Simpler header format
- Simplified, more efficient routing
- True quality of service (QoS), also called "flow labeling"
- Built-in authentication and privacy support
- Flexible options and extensions
- Easier administration

For this layer will be used IPv6. With the same porpoue it is clear that IPv6 is more efficient than IPv4. Astrea system would be operational for years, and it does not take sense to use a protoccol which would be obsolete in the future. Even so, it could be added some extension protocols that will make the system more robust.

IPsec

Internet Protocol Security (IPsec) is a protocol suite for secure Internet Protocol (IP) communications that works by authenticating and encrypting each IP packet of a communication session. IPsec includes protocols for establishing mutual authentication between agents at the beginning of the session and negotiation of cryptographic keys to be used during the session. IPsec can be used in protecting data flows between a pair of hosts (host-to-host), between a pair of security gateways (network-to-network), or between a security gateway and a host (network-to-host). Internet Protocol security (IPsec) uses cryptographic security services to protect communications over Internet Protocol (IP) networks.

ICMPv6

ICMPv6 is an integral part of IPv6 and performs error reporting and diagnostic functions and has a framework for extensions to implement future changes. ICMPv6 messages may be classified into two categories: error messages and information messages. ICMPv6 messages are transported by IPv6 packets in which the IPv6 Next Header value for ICMPv6 is set to 58.

NDP

The Neighbor Discovery Protocol (NDP) is a protocol in the Internet protocol suite used with Internet Protocol Version 6 (IPv6). It operates in the Link Layer of the Internet model, and is responsible for address autoconfiguration of nodes, discovery of other nodes on the link, determining the addresses of other nodes, duplicate address detection, finding available routers and Domain Name System (DNS) servers, address prefix discovery, and maintaining reachability information of other active neighbor nodes.

SEND

The Secure Neighbor Discovery (SEND) protocol is a security extension of the Neighbor Discovery Protocol (NDP) in IPv6. NDP is insecure and susceptible to malicious interference. It is the intent of SEND to provide an alternate mechanism for securing NDP with a cryptographic method that is independent of IPsec, the original and inherent method of securing IPv6 communications.

MLD

Multicast Listener Discovery (MLD) protocol enables IPv6 routers to discover multicast listeners, the nodes that are configured to receive multicast data packets, on its directly attached interfaces. The protocol specifically discovers which multicast addresses are of interest to its neighboring nodes and provides this information to the active multicast routing protocol that makes decisions on the flow of multicast data packets.

Conclusion

ICMPv6 will be imlemented in order to ensure that the data arrives where it has to. For the adecuate work of the sistem it has to be also included NDP, in order to make easy the connection of the client to the network. It is descarted to use MLD. It does not report any significant benefit to the system and it would be useless.

The security is a very imortant criteria for designing the protocols. It has to be considered that the ground segment is the most susceptible to be attacked. The privacy of the information that it will be managed has to be ensured. For this reason IPsec and SEND will be impemented.

In summary, for the inernet layer it will be implemented a IPv6 extended with IPsec, ICMPv6 and NDP protected with SEND.

2.3 Transport Layer

In computer networking, the transport layer is a conceptual division of methods in the layered architecture of protocols in the network stack in the Internet Protocol Suite. The protocols of the layer provide host-to-host communication services for applications. It provides services such as:

- Connection-oriented communication
- Same order delivery
- Reliability
- Flow control
- Congestion avoidance
- Multiplexing

There were analized the following existing protocols.

TCP

The Transmission Control Protocol (TCP) is one of the main protocols of the Internet protocol suite. It originated in the initial network implementation in which it complemented the Internet Protocol (IP). TCP provides reliable, ordered, and error-checked delivery of a stream of octets between applications running on hosts communicating by an IP network. Major Internet applications such as the World Wide Web, email, remote administration, and file transfer rely on TCP.

At the lower levels of the protocol stack, due to network congestion, traffic load balancing, or other unpredictable network behaviour, IP packets may be lost, duplicated, or delivered out of order. TCP detects these problems, requests re-transmission of lost data, rearranges out-of-order data and even helps minimise network congestion to reduce the occurrence of the other problems. If the data still remains undelivered, its source is notified of this failure. Once the TCP receiver has reassembled the sequence of octets originally transmitted, it passes them to the receiving application. Thus, TCP abstracts the application's communication from the underlying networking details.

TCP is a reliable stream delivery service which guarantees that all bytes received will be identical with bytes sent and in the correct order. Since packet transfer by many networks is not reliable, a technique known as 'positive acknowledgement with re-transmission' is used to guarantee reliability of packet transfers. This fundamental technique requires the receiver to respond with an acknowledgement message as it receives the data. The sender keeps a record of each packet it sends and maintains a timer from when the packet was sent. The sender re-transmits a packet if the timer expires before the message has been acknowledged. The timer is needed in case a packet gets lost or corrupted.

TCP is optimised for accurate delivery rather than timely delivery. Therefore, TCP sometimes incurs relatively long delays (on the order of seconds) while waiting for out-of-order messages or re-transmissions of lost messages. It is not particularly suitable for real-time applications such as Voice over IP. For such applications, protocols like the Real-time Transport Protocol (RTP) operating over the User Datagram Protocol (UDP) are usually recommended instead.

UDP

UDP uses a simple connectionless transmission model with a minimum of protocol mechanism. UDP provides checksums for data integrity, and port numbers for addressing different functions at the source and destination of the datagram. It has no handshaking dialogues, and thus exposes the user's program to any unreliability of the underlying network and so there is no guarantee of delivery, ordering, or duplicate protection. If error correction facilities are needed at the network interface level, an application may use the Transmission Control Protocol (TCP) or Stream Control Transmission Protocol (SCTP) which are designed for this purpose.

UDP is suitable for purposes where error checking and correction is either not necessary or is performed in the application, avoiding the overhead of such processing at the network interface level. Time-sensitive applications often use UDP because dropping packets is preferable to waiting for delayed packets, which may not be an option in a real-time system.

SCTP

Stream Control Transmission Protocol (SCTP) is a transport-layer protocol, serving in a similar role to the popular protocols TCP and UDP. SCTP provides some of the same service features of both: it is message-oriented like UDP and ensures reliable, in-sequence transport of messages with congestion control like TCP; it differs from these in providing multi-homing and redundant paths to increase resilience and reliability.

Features of SCTP include:

- Multihoming support in which one or both endpoints of a connection can consist of more than one IP address, enabling transparent fail-over between redundant network paths.
- Delivery of chunks within independent streams eliminate unnecessary head-of-line blocking, as opposed to TCP byte-stream delivery.
- Path selection and monitoring to select a primary data transmission path and test the connectivity of the transmission path.
- Validation and acknowledgment mechanisms protect against flooding attacks and provide notification of duplicated or missing data chunks.

DCCP

The Datagram Congestion Control Protocol (DCCP) is a message-oriented transport layer protocol. DCCP implements reliable connection setup, teardown, Explicit Congestion Notification (ECN), congestion control, and feature negotiation. DCCP provides a way to gain access to congestion control mechanisms without having to implement them at the application layer. It allows for flow-based semantics like in Transmission Control Protocol (TCP), but does not provide reliable in-order delivery. Sequenced delivery within multiple streams as in the Stream Control Transmission Protocol (SCTP) is not available in DCCP.

DCCP is useful for applications with timing constraints on the delivery of data. Such applications include streaming media, multiplayer online games and Internet telephony. The primary feature of these applications is that old messages quickly become stale so that getting new messages is preferred to resending lost messages. Currently such applications have often either settled for TCP or used User Datagram Protocol (UDP) and implemented their own congestion control mechanisms, or have no congestion control at all.

Conclusion

UDP and DCCP are protocols which priorize the time for transmiting over the quality of the data recived. That involves a signifiant lost of infomation. It has to be ensured the integrity of the messages sent. Taking this restriction, it is assumed a increasing of the latency, which does not mean that it could not be ensured an adecuate one.

The performance of TCP and SCTP are simlar and comparable. In [reference] it is seen an experimental comparation between these 2 protocols. The results shows that SCTP ensures a lower latency and a better throughput from same conditions. Also, it has to be taken in acount that TCP is more vulnerable agaist Denial of Service attacks than SCTP. For this reasons, SCTP is the better option for the transport layer of the system.

2.4 Application layer

An application layer is an abstraction layer that specifies the shared protocols and interface methods used by hosts in a communications network. In TCP/IP, the application layer contains the communications protocols and interface methods used in process-to-process communications across an Internet Protocol (IP) computer network. The application layer only standardizes communication and depends upon the underlying transport layer protocols to establish host-to-host data transfer channels and manage the data exchange in a client-server or peer-to-peer networking model.

There were analized the following existing protocols.

HTTP

The Hypertext Transfer Protocol (HTTP) is an application protocol for distributed, collaborative, hypermedia information systems. HTTP is the foundation of data communication for the World Wide Web.

HTTP functions as a request–response protocol in the client–server computing model. A web browser, for example, may be the client and an application running on a computer hosting a website may be the server. The client submits an HTTP request message to the server. The server, which provides resources such as HTML files and other content, or performs other functions on behalf of the client, returns a response message to the client. The response contains completion status information about the request and may also contain requested content in its message body.

FTP

The File Transfer Protocol (FTP) is a standard network protocol used to transfer computer files between a client and server on a computer network.

FTP is built on a client-server model architecture and uses separate control and data connections between the client and the server.[1] FTP users may authenticate themselves with a clear-text sign-in protocol, normally in the form of a username and password, but can connect anonymously if the server is configured to allow it. For secure transmission that protects the username and password, and encrypts the content, FTP is often secured with SSL/TLS (FTPS). SSH File Transfer Protocol (SFTP) is sometimes also used instead, but is technologically different.

Setting up an FTP control connection is quite slow due to the round-trip delays of sending all of the required commands and awaiting responses, so it is customary to bring up a control connection and hold it open for multiple file transfers rather than drop and re-establish the session afresh each time.

SMTP

Simple Mail Transfer Protocol (SMTP) is an Internet standard for electronic mail (email) transmission. Email is submitted by a mail client (mail user agent, MUA) to a mail server (mail submission agent, MSA). The MSA delivers the mail to its mail transfer agent (mail transfer agent, MTA). Often, these two agents are instances of the same software launched with different options on the same machine. Local processing can be done either on a single machine, or split among multiple machines; mail agent processes on one machine can share files, but if processing is on multiple machines, they transfer messages between each other using SMTP, where each machine is configured to use the next machine as a smart host. Each process is an MTA (an SMTP server) in its own right.

SMTP is a connection-oriented, text-based protocol in which a mail sender communicates with a mail receiver by issuing command strings and supplying necessary data over a reliable ordered data stream channel, typically a Transmission Control Protocol (TCP) connection. An SMTP session consists of commands originated by an SMTP client (the initiating agent, sender, or transmitter) and corresponding responses from the SMTP server (the listening agent, or receiver) so that the session is opened, and session parameters are exchanged. A session may include zero or more SMTP transactions. An SMTP transaction consists of three command/reply sequences:

- MAIL command, to establish the return address.

- RCPT command, to establish a recipient of the message. This command can be issued multiple times, one for each recipient. These addresses are also part of the envelope.
- DATA to signal the beginning of the message text; the content of the message, as opposed to its envelope. It consists of a message header and a message body separated by an empty line. DATA is actually a group of commands, and the server replies twice: once to the DATA command itself, to acknowledge that it is ready to receive the text, and the second time after the end-of-data sequence, to either accept or reject the entire message.

SNMP

Simple Network Management Protocol (SNMP) is an Internet-standard protocol for collecting and organizing information about managed devices on IP networks and for modifying that information to change device behavior. Devices that typically support SNMP include routers, switches, servers, workstations, printers, modem racks and more.

In typical uses of SNMP one or more administrative computers, called managers, have the task of monitoring or managing a group of hosts or devices on a computer network. Each managed system executes, at all times, a software component called an agent which reports information via SNMP to the manager.

- An SNMP-managed network consists of three key components:
- Managed device
- Agent — software which runs on managed devices Network management station (NMS) — software which runs on the manager

The SNMP agent receives requests on UDP port 161. The manager may send requests from any available source port to port 161 in the agent. SNMP agents expose management data on the managed systems as variables. The protocol also permits active management tasks, such as modifying and applying a new configuration through remote modification of these variables. The variables accessible via SNMP are organized in hierarchies. These hierarchies, and other metadata (such as type and description of the variable), are described by Management Information Bases (MIBs).

TLS

Transport Layer Security (TLS)is a cryptographic protocol that provides communications security over a computer network.Several versions of the protocol find widespread use in applications such as web browsing, email, Internet faxing, instant messaging, and voice-over-IP (VoIP). Major websites use TLS to secure all communications between their servers and web browsers.

The Transport Layer Security protocol aims primarily to provide privacy and data integrity between two communicating computer applications. When secured by TLS, connections between a client and a server have one or more of the following properties:

- The connection is private (or secure) because symmetric cryptography is used to encrypt the data transmitted. The keys for this symmetric encryption are generated uniquely for each connection and are based on a shared secret negotiated at the start of the

session. The server and client negotiate the details of which encryption algorithm and cryptographic keys to use before the first byte of data is transmitted . The negotiation of a shared secret is both secure (the negotiated secret is unavailable to eavesdroppers and cannot be obtained, even by an attacker who places themselves in the middle of the connection) and reliable (no attacker can modify the communications during the negotiation without being detected).

- The identity of the communicating parties can be authenticated using public-key cryptography. This authentication can be made optional, but is generally required for at least one of the parties (typically the server).
- The connection ensures integrity because each message transmitted includes a message integrity check using a message authentication code to prevent undetected loss or alteration of the data during transmission.

HTTPS

HTTPS is a protocol for secure communication over a computer network which is widely used on the Internet. HTTPS consists of communication over Hypertext Transfer Protocol (HTTP) within a connection encrypted by Transport Layer Security. The main motivation for HTTPS is authentication of the visited website and protection of the privacy and integrity of the exchanged data.

Web browsers know how to trust HTTPS websites based on certificate authorities that come pre-installed in their software. Certificate authorities are in this way being trusted by web browser creators to provide valid certificates. Therefore, a user should trust an HTTPS connection to a website if and only if all of the following are true:

- The user trusts that the browser software correctly implements HTTPS with correctly pre-installed certificate authorities.
- The user trusts the certificate authority to vouch only for legitimate websites.
- The website provides a valid certificate, which means it was signed by a trusted authority.
- The certificate correctly identifies the website (e.g., when the browser visits "https://example.com", the received certificate is properly for "example.com" and not some other entity).
- The user trusts that the protocol's encryption layer (SSL/TLS) is sufficiently secure against eavesdroppers.