Security of Computer Networks

Jan Verschuren

TNO-EIB, P.O. Box 5013,
2600 GA Delft, The Netherlands
verschuren@tpd.tno.nl

Abstract. A few decades ago, most computers were stand-alone machines: they were able to process information using their own resources. Later computer systems were connected to each other enabling a computer system to use resources of an other computer as well.

With the coupling of computer systems, security items emerge: in general the link between computers is not protected by thorough physical means. That enables an attacker to tap or modify information on the link.

In this article, a model of a distributed system is given. The provided model can also form a basis for security evaluations. Threats to the process of information exchange between computer systems are indicated. In order to counter the threats envisaged, so called security services can be implemented: two communicating computer systems can use security services in order to secure their communication link. In this article, two principal implementations of security services are addressed. The qualities of the two implementations of security services are compared with each other.

1 Introduction

Formerly, computer systems were mainly stand-alone: a computer was able to process programs (also referred to as application programs (APs)) using its resources. Later computers were coupled by means of a network. The possibility to exchange data between computers was considered to be an advantage. The distributed configuration offered for example the possibility to store certain data at only one location. Besides it became possible to run jobs on other computer systems, if one's own computer was not able to do so.

In this article we will address security items which are specific for the distributed configuration. In order to do so we start with modelling a network of computing entities (section 2). In the framework of network security a security policy will be needed, stating security requirements which need to be fulfilled by the network. This topic is dealt with in section 3. In section 4, attention is paid to the communication process. In this way, a background is given for two different implementations of security services which aim at a secure information exchange between computers. Implications of the use of both implementations are given. Section 5 concludes the article.

B. Preneel, V. Rijmen (Eds.): COSIC'97 Course, LNCS 1528, pp. 163-185, 1998.

2 Model of a Computer Network

In the world, there are a lot of entities (human beings, computer systems). These can influence each other.

First we will model the way how human beings can influence each other. A human being can notice changes in his environment or - otherwise stated - a human being can notice events. In this way he can retrieve information from the environment. Thus the human being can get input. A person can also perform actions which can be seen as outputs. Both input and output can be modelled by means of bitstreams.

A human being will act according to a certain way, in other words, a relation will exist between input and outputs of a human being or - speaking in terms of our model - a relation will exist between bitstreams representing inputs and bitstreams representing outputs. This can be represented as follows: at a certain moment person A has got a set of inputs $Si_A = \{i_{1,A}, i_{2,A}, \ldots, i_{ni(A),A}\}$. In general a human being will be able to generate more than one output, so a *set* of outputs $So_A = \{o_{1,A}, o_{2,A}, \ldots, o_{no(A),A}\}$ is possible at a certain moment. Outputs of a person can be varying from sending a mail to starting to buy shares. The input-output relation of a human being at a certain moment of time t_i, can be given by a pair consisting of:

- a set of inputs Si_A which have been received up to time t_i.
- a set of outputs So_A which can be generated at time t_i.

During the time a human being may continue to receive inputs. So at moments of time, a person A will dispose of different sets of inputs $Si_{A,1}$, $Si_{A,2}$, …, $Si_{A,m(A)}$. With each of these sets of inputs, a set of outputs $So_{A,1}$, $So_{A,2}$, …, $So_{A,m(A)}$ is defined where $So_{A,1}$ goes with $Si_{A,1}$, $So_{A,2}$ goes with $Si_{A,2}$ and so on. If the human being has for instance received the inputs of $Si_{A,1}$, then he can provide an output which is one of the outputs of $So_{A,1}$.

The behaviour of a person A can be controlled, if during the life time of the human being no other sets of inputs arise than $\{Si_{A,1}, Si_{A,2}, \ldots, Si_{A,m(A)}\}$.

It is also possible that a person uses a computer system: the outputs available from a computer system can serve as input for a human being. On the other side, a person can use his outputs as commands for a computer system. A computer system also has an input-output relation: the output of a computer system will be dependent on the received information. This relation can be given by means of a table which has the following form.

Table 1: input-output table of a computing entity, describing the input-output relationship of a computing entity.

Input	Old state	New state	Output
input 1	state 1	state 5	output 1
input 1	state 2	state 1	output 1
input 2	state 2	state 1	output 3

This has to be read as follows. If the computer system is in the old state indicated in column 2 (e.g. state 2) and if it receives an input as indicated in the first column (e.g. input 2), then it will switch to the new state indicated in the third column (state 1) and generate the output in the fourth column (output 3).

Here all possible inputs which an entity can handle are indicated in the input part of the table. It is possible that an identical input may lead to different outputs dependent on the state of the computing entity. The computer system - acting according to his input-output table - can be controlled by limiting inputs to the computer system.

Using the input-output relationships of human beings and also using the tables which describe the input-output relation of computing entities, it is possible to follow the information flows. More specifically, human beings can start the information flow by sending one of their outputs to another entity. By means of the input-output relationship of each entity, it is possible to foresee the outputs of each entity concerned. Figure 1 gives an illustration.

In figure 1, the behaviour of each entity is specified: the behaviour of human beings is given by indicating pairs of sets of inputs and outputs (each pair consists of a set of inputs and a corresponding set of outputs); the behaviour of the computer entities is given by a references to tables like Table 1 above.

In addition to this, the state of the network is given by indicating for each entity its present state. The present state for human beings is indicated by giving the received set of inputs; the state of a computing entity is given by mentioning the (old) state of its input-output table.

The state of the network is further defined by indicating if the computing entities are waiting for input (wfi) or if they are aiming to transmit their output; in other words if they are ready for output: rfo. This is also indicated for human beings.

Figures like figure 1 can be used to find out what information flows may occur and what actions (outputs) can be performed by entities. In this way it is possible to find out if outputs can be foreseen which are not desired from security point of view. The security policy will define if and what outputs are acceptable. This will be further dealt with in the next section.

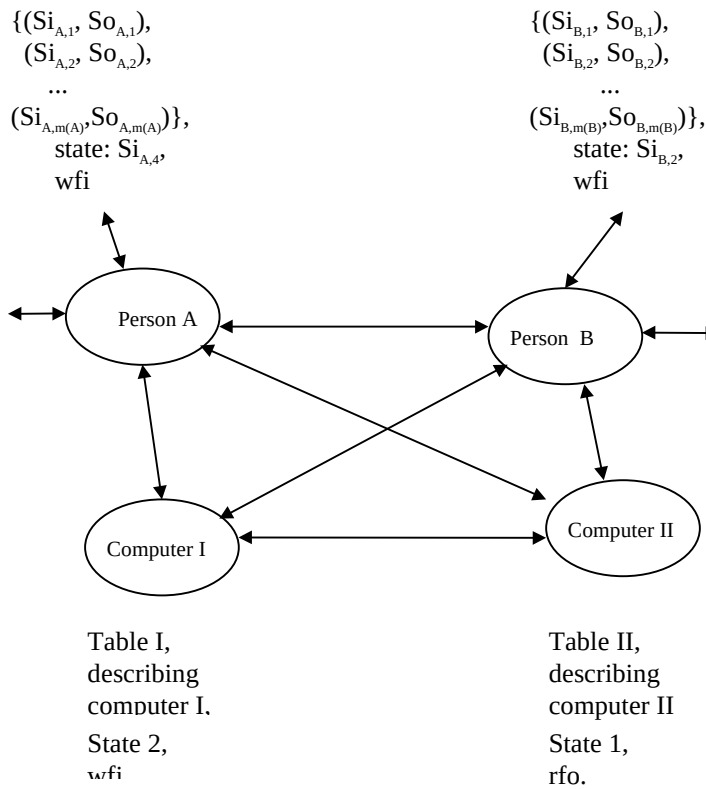

Fig. 1. Exchange of information between entities (human beings and computer systems)

3 Security Policy

3.1 Definition

Initially, the security policy will refer to human beings. ITSEC [1] defines security policy as "the set of laws, rules and practices that regulate how assets including sensitive information are managed, protected and distributed within a user organisation". Referring to the model of figure 1, the security policy will indicate:

- which outputs a human being may generate and
- the conditions when a human being may generate the outputs.

More information and examples of security policies can be found in [2], [3], [4], [5].

3.2 Verification of a Security Policy

The security policy states for each human being when he may generate a specific output. The human being generates an output dependent on the characteristics of the inputs he receives. From this it follows that two items need to be fulfilled for a security evaluation:

1. verification if the behaviour of the human being is as claimed; if we do not know when a human being generates certain outputs, then it is difficult to control the human being.
2. If the behaviour of the human being is known, then the human being can be controlled by means of limiting the information which is sent to the human being. This statement only holds if the human being correctly assesses the received inputs. An example can illustrate this. Assume that a human being has the intention to treat data of confidentiality class 3 in another way than data of confidentiality class 4. Then the human being can only do this correctly if he assesses the confidentiality class of the received data correctly. So it is necessary from security viewpoint to be sure about the characteristics of the received input becoming available to the human being. That is the second item of evaluation: it needs to be verified if the characteristics of the inputs which arrive at the human being are reliable so that the human being can react correctly on the received input.

Different kinds of inputs can arrive at a human being. In the rest of this article we will concentrate on *information* which is sent from a computer system to the human being. From security viewpoint, actually two characteristics can be assigned to that information: confidentiality and integrity [6].

Confidentiality
If the received information is characterised as “information of a certain confidentiality class” then this means that the information has not been disclosed to a defined set of entities.

Integrity
If the received information is characterised as “information of a certain integrity class”, then this means that the information has not been modified by a defined set of entities.

As we do not concentrate on the mentioned first item of a security evaluation, we want to verify if the confidentiality and integrity characteristics of the information received by the human being, are correct.
We will clarify this by means of an example which is illustrated in figure 2.

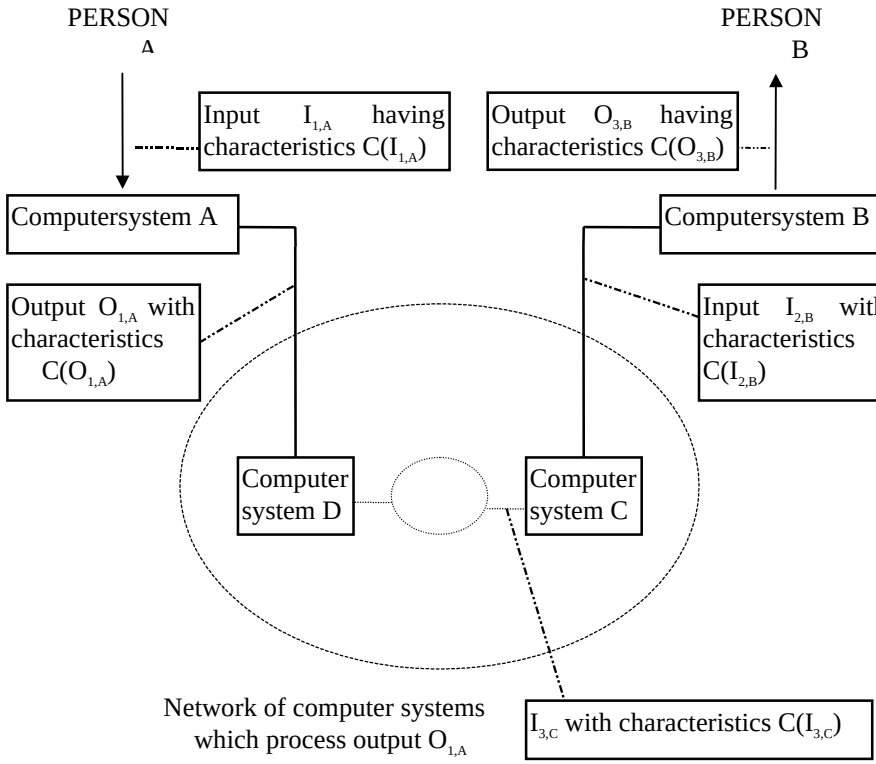

Fig. 2. Illustration of information exchange between two persons via a network of computing entities**.**

In figure 2, we illustrated the situation where person A sends information $I_{1,A}$ which has characteristics $C(I_{1,A})$. $I_{1,A}$ is sent into the network, resulting into an input $I_{2,B}$ to computer system B. After analysing the received input $I_{2,B}$, computersystem B supposes that the characteristics of input $I_{2,B}$ are given by $C(I_{2,B})$. In the framework of a security evaluation, it is necessary to verify if the confidentiality and integrity characteristics of $I_{2,B}$ as given in $C(I_{2,B})$ are correct.

For a complete security evaluation, it is necessary that these checks are done for all possible inputs to computer system B, for all states of the network and for all possible outputs from computer system A.

Several starting points can be chosen. In figure 2, computer system A is the starting point and computer system B the end-point. If computer system D would have been chosen as starting point and computer system C as end-point, then only a qualification of a subnetwork would result. By means of evaluation of several subnetworks, it is possible to come to an evaluation of a larger network consisting of

the subnetworks. In the rest of this article, we will concentrate on the realisation of a secure subnetwork.

4 Realisation

Up to now we briefly talked about definition and evaluation of network security. Now we want to address realisation aspects. In order to realise a secure network despite attackers which tap or modify the information transmitted via the communication link between two computer systems, it is necessary to protect the link. First we will briefly describe the mechanism of communication between two computer systems. Then we will concentrate on adding security measures which can prevent or detect attacks on the communication link.

4.1 Communication Between Computer Systems

A computer system can be schematically represented by figure 3.

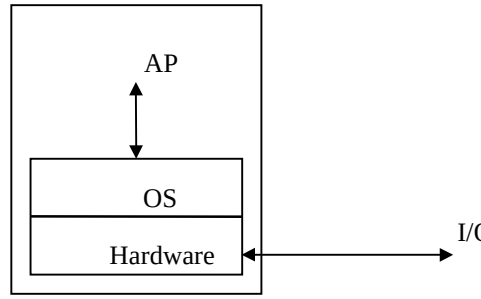

Fig. 3. Schematic representation of a computer system.

An application program AP is executed by means of the operating system OS which is built on a hardware platform. Via an I/O-channel, the Application Program can retrieve and transmit information.

Via I/O channels, the computer system can be connected to a medium. Thus it is possible that an AP communicates with an AP on another computer system. The medium may be a simple link, it can also be a network to which more computer systems are connected. In order to achieve that the information block which is sent by an AP (AP_s) arrives correctly at the addressee (AP_r), additional information has to be appended to the information block of the sending AP. In general a separate computing entity is used for this purpose. Such a so called communication subsystem

appends control information to the information block which is sent by AP_s; thus the communication subsystem controls the information exchange between two APs on two computer systems. This is illustrated in figure 4.

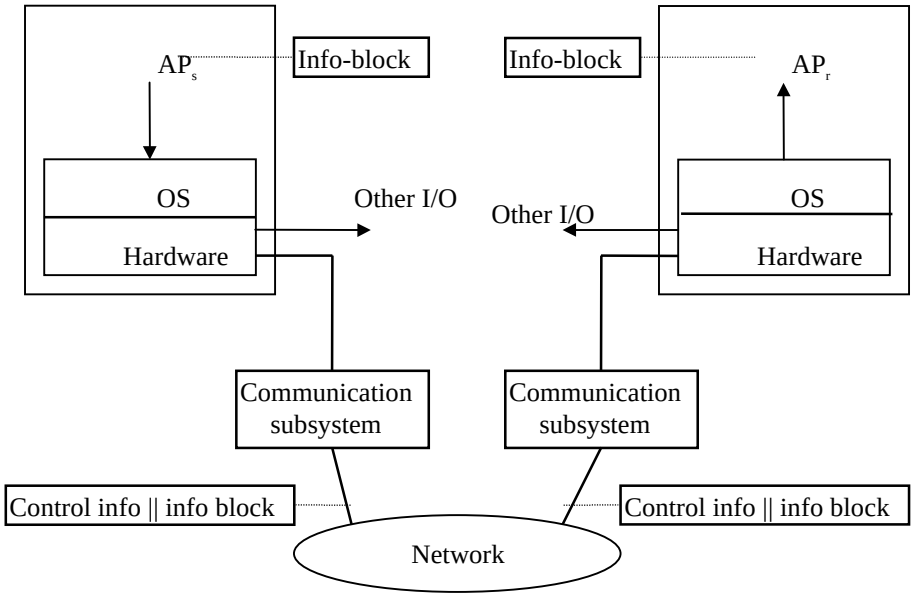

Fig. 4. AP - AP - communication via a network.

Several communication subsystems exist. In this article we concentrate on the communication subsystem according to the ISO-Reference Model for Open Systems Interconnection.

4.2 The ISO Reference Model for Open Systems Interconnection

The ISO Reference Model for Open Systems Interconnection consists of 7 layers which form together the communication subsystem (fig. 5) , [7].

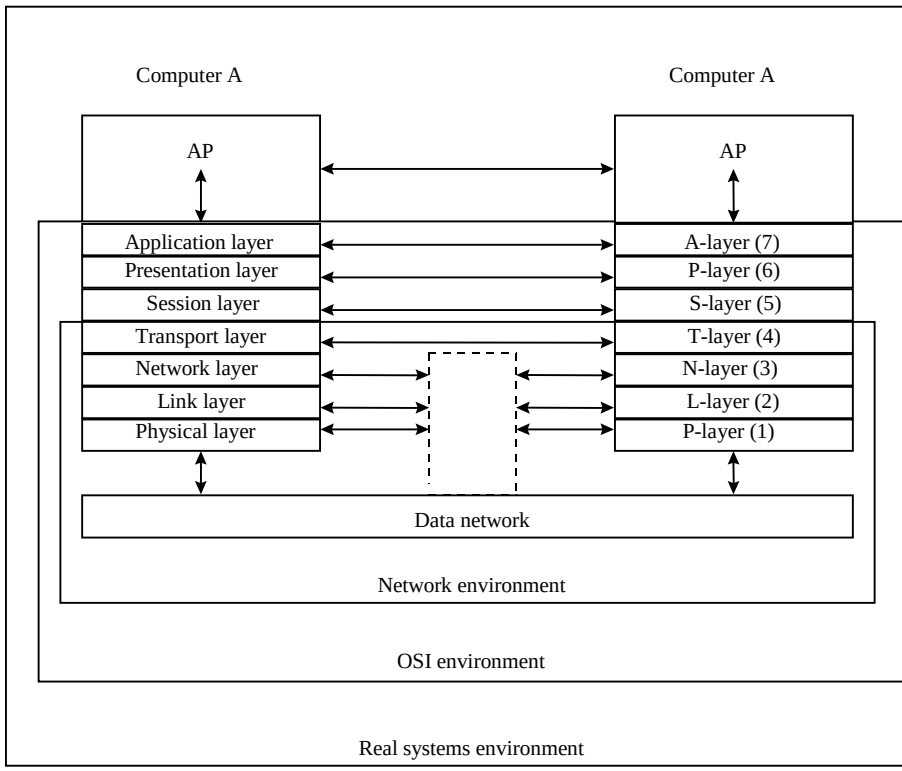

Fig. 5. Overall structure of the ISO-Reference Model

The layers can be indicated by means of their names or by means of a number N (N = 1,2,…,7). Each layer performs a well defined function in the context of the communication between two APs. It operates according to a defined protocol. (Protocols are sets of rules that govern the exchange of information between two entities.) Performing this protocol results in the exchange of Protocol Data Units (PDUs) between two (protocol) entities that belong to the same layer. PDUs which are exchanged between two protocol entities of layer N are referred to as N-PDUs. These are transported by invoking services provided by the lower layer. More specifically, the N-PDU (which is also named (N-1)-SDU: N-1 Service Data Unit) is a parameter of a service request which is issued by the N-layer to the (N-1)-layer. Subsequently the (N-1)-protocol entity will feel the need to exchange an (N-1)-PDU with a peer entity of layer (N-1). The (N-1)-PDU will be equal to the N-PDU concatenated with a so called PCI-block (protocol control information block) which will contain at least the source and the destination addresses of the service access points of the two communicating protocol entities. This PCI-block governs the PDU-exchange between the protocol entities as it helps the receiving protocol entity to correctly interpret the received PDU. At an (N-1)-service access point ((N-1)-SAP) a

protocol entity of layer N can invoke a service provided by a protocol entity of layer (N-1). Figures 6 and 7 illustrate the exchange of PDUs between protocol entities.

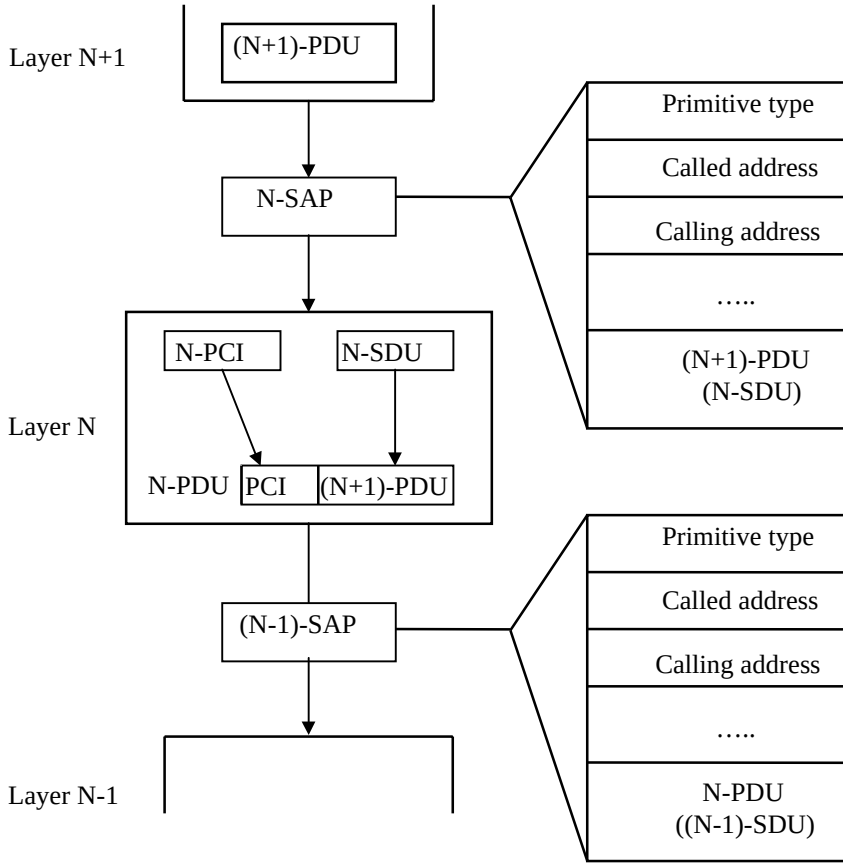

Fig. 6. Interactions between protocol entities in the same system

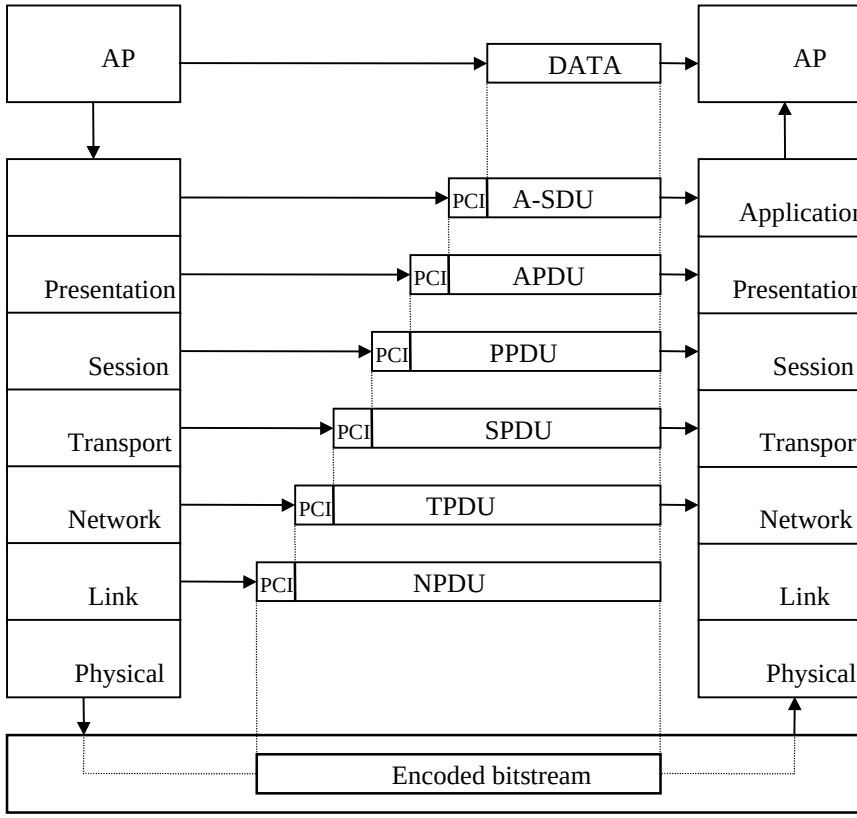

Fig. 7. Interactions between protocol entities in different systems

The functions of the different layers can be described as follows.

- the physical layer transforms the information to be sent (represented in bits) to (physical) signals which can be transported by the transmission medium.
- The link layer provides the network layer with a reliable information transfer facility. It is thus responsible for such functions as error detection and, in the event of transmission errors, the retransmission of messages.
- The network layer is responsible for the establishment and clearing of a network wide connection between two transport layer protocol entities. It includes such facilities as network routing (addressing).
- The transport layer provides the session layer with a reliable data transfer facility which is *independent* of the type of network which is being used to transfer the data.

- The session layer is responsible for establishing and synchronising the dialogue between APs. Synchronising a dialogue means that the dialogue can be resumed from specific synchronisation points in case of errors.
- The presentation layer performs the conversion from an abstract syntax (e.g. type character) to a concrete syntax (e.g. ASCII) and vice versa. Figure 8 illustrates this.

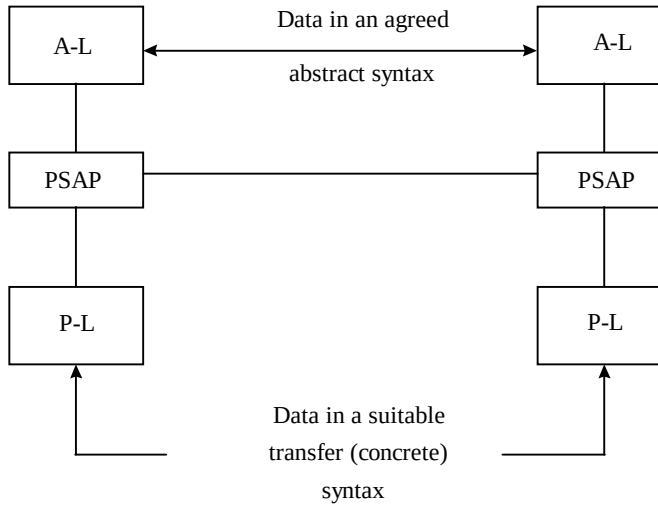

Fig. 8. Function of the presentation layer

- The application layer enables APs to get access to a range of network wide distributed information services. For instance an AP can get access to a remote file server AP which enables the requesting AP to get access to files managed by the file server AP.

As mentioned before, an N-protocol entity has to transmit PDUs (N-SDUs) - which it gets as a parameter in a service request - to a peer protocol entity. Two possible ways exist to do this.

1. The entity first establishes a connection with the peer entity. If this is settled, then the N-SDU in question is transmitted via the connection. This is referred to as a “connection oriented” service.

2. The N-SDU is transmitted without first establishing a connection. This is denoted by the term “connectionless” service.

Out of the foregoing it turned out that the network layer is responsible for routing PDUs between computer systems. A schematic representation of a computer system is given in figure 9.

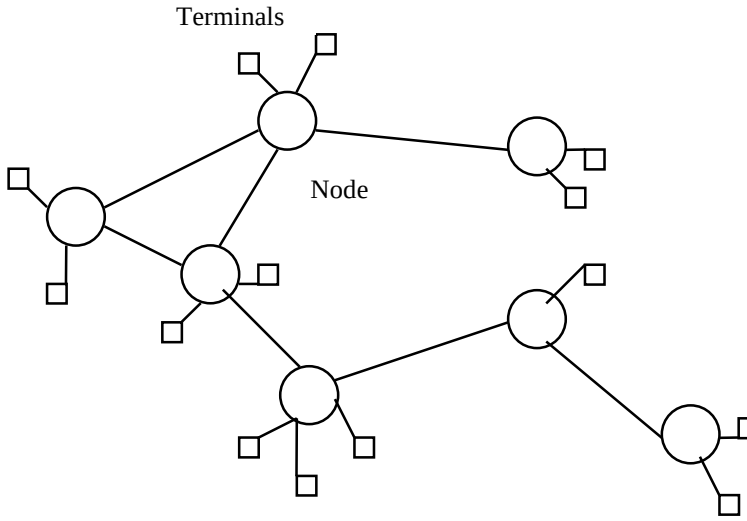

Fig. 9. Computernetwork

The nodes must be able to transmit an arriving PDU in the right direction. Therefore a node must be equipped with protocol entities belonging to the lower three layers of the OSI-RM. Figure 10 gives a schematic illustration of a node and the function it performs.

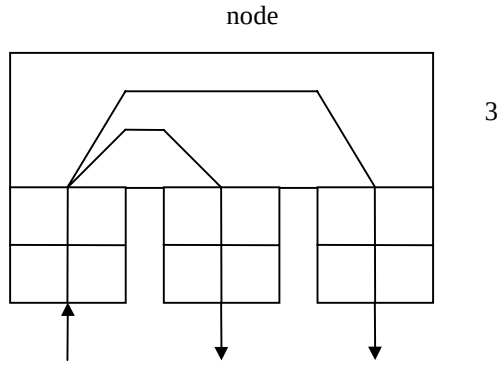

Fig. 10. Routing function of a node

The total address of an AP does not only consist of the physical network-wide address of the computer system it is running on. It is built up out of SAPs in the following way.

APaddress = PSAP + TSAP + NSAP

Here PSAP, TSAP and NSAP denote the addresses of the service access points between the application layer protocol entity to which the AP is connected and the presentation layer, between the session layer and the transport layer and between the transport and the network layer. The NSAP also contains the physical network wide address of the system in which the AP is resident.

4.3 Threats

The fact that information transferred between two computers is transported via a public medium (e.g. a telephone network), makes it extra vulnerable in comparison with information exchanged between APs both running on the same stand-alone system. More specifically, data in stand-alone systems is not easily available whereas data transported via public networks is: in general public media are not protected by a physical barrier. Therefore everyone can easily tap information or even modify it. Thus two forms of attacks can be discerned.

- passive attacks resulting in unauthorised disclosure of information.
- active attacks resulting in modification of the transmitted information.

From section 4.2 it turned out that the information which is transferred via the data communication network is twofold:

- user data
- data governing the communication between two entities (protocol control information)

So passive attacks can result in the disclosure of user data or of information concerning the communication (e.g. the addresses of the sending and receiving computersystem).

Active attacks intend to modify user data or PCI-blocks. Changing the address of the sending computersystem is an example of modifying the PCI-block. That will make the receiver think that the data came from another system.

The threats just mentioned refer to the situation where two APs are already communicating. Prior to this phase, the communication must be started: an AP attempts to set up a connection with another AP. Then it must be decided if the two APs are allowed to communicate with each other. More specifically, it must be prevented that an AP uses resources (e.g. data belonging to another AP) in an unauthorised way.

After the data has been transmitted from sender to receiver, other threats are present:

- the receiver states it did not get the data
- the sender states it did not send the data

So the threats to be envisaged, can be summarised as follows:

- unauthorised access to resources
- disclosure or modification of transmitted data
- false statements of entities which sent or received data

4.4 Realisation of Security Services

Protection needs to be supplied to prevent the threats as mentioned in section 4.3 to work out successfully. This protection can be realised by means of security services.

While addressing the realisation of security services, we can discern two principal locations where security services can be realised.

1. In the communication subsystem

   By means of a communication subsystem an AP can exchange information with another AP. So the communication subsystem supports an AP: it can provide an AP with the functionality of a communication link. Implementing security services into the communication subsystem, leads to a

situation where the communication subsystem can provide the AP with a *secure* communication link. The option where security services are implemented in a communication subsystem according to the OSI-RM is addressed in section 4.4.1.

2. At the level of the Application Program

   The communication subsystem provides the AP with a communication link; initially, the communication subsystem does not provide a *secure* communication link. Incorporating security services in the AP itself, leads to a situation where the AP itself builds a *secure* link on the offered insecure communication link. This option is further dealt with in section 4.4.2.

Security Services in the OSI-RM In section 4.3, the threats to user data and to the protocol control information were mentioned. Looking at figure 7, it can be derived what information can be protected at what layer. It follows that at the level of the AP only user data can be protected. Security services within the protocol entities of the OSI-RM can protect protocol control information as well. The security services - integrated in the protocol entities - can have different forms. That is mainly dependent on the way the communication is set up between two entities. (More specifically, that can be done by means of a connectionless or a connection-oriented service.)

Now the definition of the security services (with respect to the OSI-environment) will be given [8].

- Authentication.

  This service can take two forms: peer entity authentication and data origin authentication. The first one refers to a connection-oriented mode of transmitting PDUs, the second one assumes a connectionless mode.

  Peer entity authentication

  This service is provided at the establishment of, or at times during the data transfer phase of, a connection to confirm the identities of one or more of the entities connected. This service provides confidence that an entity is not attempting a masquerade or an unauthorised replay of a previous connection.

  Data origin authentication

  The service, when provided by the (N)-layer, provides corroboration to the (N+1)-layer that the source of the data is the claimed (N+1)-peer entity. The data origin authentication service is the corroboration of the source of a single connectionless data unit. The service cannot protect against duplication of a data unit.

- Access Control

  This service provides protection against unauthorised use of resources accessible via OSI. These may be OSI or non-OSI resources accessed via OSI-protocols. This protection service may be applied to various types of access to a resource (e.g. the use of a communications resource; the reading, writing or the deletion of an information resource; the execution of a processing resource) or to all accesses to a resource.

- Data confidentiality.

  The following forms of this service are defined.

  Connection confidentiality

  This service provides for the confidentiality of all (N)-user-data on an (N)-connection.

  Connectionless confidentiality

  This service provides for the confidentiality of all (N)-user-data in a single connectionless (N)-SDU

  Selective field confidentiality

  This service provides for the confidentiality of selected fields within the (N)-user-data on an (N)-connection or in a single connectionless (N)-SDU.

  Traffic flow confidentiality

  This service provides for the protection of the information which might be derived from observation of traffic flows.

- Data integrity

  The following forms of this service are defined.

  Connection integrity with recovery

  This service provides for the integrity of all (N)-user-data on an (N)-connection and detects any modification, insertion, deletion or replay of any data within an entire SDU sequence (with recovery attempted; i.e. after detecting that the integrity of the user data is not fulfilled, subsequent attempts are carried out to get the user data of which the integrity is preserved).

  Connection integrity without recovery

  The same as the previous one but with no recovery attempted.

  Selective field connection integrity

  This service provides for the integrity of selected fields within the (N)-user data of an (N)-SDU transferred over a connection and takes the form of

determination of whether the selected fields have been modified, inserted, deleted or replayed.

Connectionless integrity
This service provides for the integrity of a single connectionless SDU and may take the form of determination of whether a received SDU has been modified. Additionally, a limited form of detection of insertion or replay may be provided.

Selective field connectionless integrity
This service provides for the integrity of selected fields within a single connectionless SDU and takes the form of determination of whether the selected fields have been modified.

- Non-repudiation
  This security service can take two forms

  Proof of origin
  The recipient of data is provided with proof of the origin of data which will protect against any attempt by the sender to falsely deny having sent the data.

  Proof of receipt
  The sender of data is provided with proof of receipt of data which will protect against any attempt by the recipient to falsely deny having received the data or its contents.

Part 2 of ISO-standard 7498 [8] gives guidelines at which layer(s) the security services can be provided (figure 11).

Making a choice out of the possibilities denoted by fig. 11, depends on the security policy of the two APs. As is seen in chapter 3, a security policy specifies how sensitive information has to be protected. Each AP tries to follow a certain security policy when communicating with another AP. If both policies do not agree initially, then negotiation is necessary. If this is successful, then the agreed security policy will result in a set of invoked security services which will be applicable to the communication. More specifically, not all security services available, will be used for each information exchange. Consider for example the following situation: an AP wants to extract address-information out of a file system on a remote end-system. In that case the requesting AP is concerned about the integrity of the received address, not about its confidentiality. So the integrity service needs to be applied whereas the confidentiality service is not necessary.

So, dependent on the agreed security policy a set of security services needs to be invoked.

For realising the security services, so called security mechanisms have to be applied. E. g. the confidentiality service can be realised by the encipherment

	Layer						
Service	1	2	3	4	5	6	7(*)
Peer Entity authentication	●	●	Y	Y	●	●	Y
Data Origin Authentication	●	●	Y	Y	●	●	Y
Access Control Service	●	●	Y	Y	●	●	Y
Connection Confidentiality	Y	Y	Y	Y	●	●	Y
Connectionless Confidentiality	●	Y	Y	Y	●	●	Y
Selective Field Confidentiality	●	●	●	●	●	●	Y
Traffic Flow Confidentiality	Y	●	Y	●	●	●	Y
Connection Integrity with recovery	●	●	●	Y	●	●	Y
Connection Integrity without recovery	●	●	Y	Y	●	●	Y
Selective Field Connection Integrity	●	●	●	●	●	●	Y
Connectionless Integrity	●	●	Y	Y	●	●	Y
Selective Field Connectionless Integrity	●	●	●	●	●	●	Y
Non-repudiation, Origin	●	●	●	●	●	●	Y
Non-repudiation, Receipt	●	●	●	●	●	●	Y

Key:

Y: Yes, service should be incorporated in the standards for the layer as a provider option

● Not provided

* It should be noted, with respect to layer 7, that the application process may, itself, provide security services

Fig. 11. Illustration of the relationship of security services and layers

mechanism. Figure 12 gives the relationship between services and mechanisms. The description of all mechanisms is given in [8]. To prevent misunderstanding of the table given in figure 12, the following remark may be helpful. Although encipherment forms only one column of the table, cryptographic techniques may be employed as part of other mechanisms such as digital signature, data-integrity, authentication. An example of a security mechanism which needs not to use

Service \ Mechanism	Encipherment	Digital Signature	Access Control	Data Integrity	Authentication Exchange	Traffic Padding	Routing Control	Notarization
Peer Entity Authentication	Y	Y	•	•	Y	•	•	•
Data Origin Authentication	Y	Y	•	•	•	•	•	•
Access Control Service	•	•	Y	•	•	•	•	•
Connection Confidentiality	Y	•	•	•	•	•	Y	•
Connectionless Confidentiality	Y	•	•	•	•	•	Y	•
Selective Field Confidentiality	Y	•	•	•	•	•	•	•
Traffic Flow Confidentiality	Y	•	•	•	•	Y	Y	•
Connection Integrity with Recovery	Y	•	•	Y	•	•	•	•
Connection Integrity without Recovery	Y	•	•	Y	•	•	•	•
Selective Field Connection Integrity	Y	•	•	Y	•	•	•	•
Connectionless Integrity	Y	Y	•	Y	•	•	•	•
Selective Field Connectionless Integrity	Y	Y	•	Y	•	•	•	•
Non-repudiation, Origin	•	Y	•	Y	•	•	•	Y
Non-repudiation, Delivery	•	Y	•	Y	•	•	•	Y

Y: Yes, the mechanism is considered to be appropriate, either on its own or in combination with other mechanisms

● The mechanism is considered not to be appropriate

Fig. 12. Relationship of security services and mechanisms

cryptographic techniques is routing. The routing mechanism causes information to be transmitted via a special communication path, e.g. via one or more secure subnetworks.

To give an idea of the effect of placing a security service at different layers examples with respect to three security services will be given.

Confidentiality

Encipherment at two different layers is considered.

- application layer
- physical layer

As was described in section 4.2, the nodes of the network are equipped with protocol entities belonging to the three lower layers of the OSI-RM. This implies that encipherment of the information at the application layer does not cause problems: the nodes can read and interpret the address information in the PCI-block added by the network-layer as this is not enciphered. (The same counts for encipherment at the level of the transport-layer.) Encipherment applied at the transport or application layer is referred to as end-to-end-encryption as during transport no deciphering of the PDUs takes place; they are deciphered for the first time when they arrive at their destination. Encipherment at the physical-layer necessitates decipherment of the PDU at the nodes to enable the node to determine the direction in which it has to send the PDU. This implies that at the node the whole PDU (including the user data sent by the AP) is in the clear. This may be unacceptable for the communicating APs. Encipherment at the physical layer has an advantage however. It will imply that an enemy who taps the line will see nothing of the structure of the information. He will therefore be unaware of the sources and destinations of messages and may even be unaware whether messages are passing at all. This provides 'traffic-flow confidentiality' which means that not only the information, but also the knowledge of where the information is flowing and how much is flowing is concealed from the enemy. By implementing encipherment at the lowest level, traffic-flow confidentiality can be obtained.

Access Control and Authentication

In section 4.2 the address structure was seen. APaddress = PSAP + TSAP + NSAP. As a result of this, it can be said that the access control service at the A-layer can deny or approve access to a specific AP as at that level the AP is fully specified. At the N-layer (or T-layer) however, only access can be given or denied to a group of APs. Analogously, the data-origin and peer-entity authentication services realised in the N-layer or T-layer can only assure that the data origin respectively peer-entity is a member of a group of entities. Realisation of the mentioned security services in the A-layer gives assurance with respect to a *specific* entity.

Security Services Realised at the Level of the Application Program When security services are implemented at the level of the Application Program (AP), then

it follows from figure 7, that no Protocol Control Information (PCI) can be protected. Only the confidentiality and integrity of the information block which is sent by the AP can be protected. So a realisation of security services in the communication subsystem can lead to a richer security functionality than a realisation of security services at the level of the AP.

Realising the security services at the level of the AP implies that the receiving AP must be able to interpret the secured messages from the sending AP. In other words, standardising at the level of APs has to take place. This means that at *two* locations standardisation has to take place: at the level of APs and at the level of communication subsystems.

5 Conclusions

Network security concentrates on protecting the link between two computing entities. More specifically, information which is sent from one computer system to another computer system needs to be protected.

Use of so called security services can lead to a secure link. In order to realise network security, two principal implementations can be discerned:

- Implementation of security services in a communication subsystem (e.g. OSI-RM).
- Implementation of security services at AP-level.

Comparing these two possibilities, we can say that realising security services in a communication subsystem, can lead to a more extensive security level.

Realising security services at the level of an AP, leads to a less open environment: APs can only communicate securely with each other if the security services adopted by both APs match with each other. Realisation of security services in an OSI communication subsystem can lead to an open environment which still can be secure.

6 Literature

[1] Information Technology Security Evaluation Criteria (ITSEC), Version 1.2, June 1991.

[2] Landwehr, Carl E., ”Formal Models for Computer Security”, *Computing Surveys*,Vol. 13, No. 3, September 1981, pp. 247 - 278

[3] Bell, D. Elliot and LaPadula, Leonard J., Secure Computer Systems: Unified Exposition and Multics Interpretation, MTR 2997 rev. 1, The MITRE Corporation, March 1976.

[4] Biba, K.J., Integrity Considerations for Secure Computer Systems, MTR-3153, The MITRE Corporation, June 1975; ESD-TR-76-372, April 1977.

[5] Clarck, D.D., Wilson, D.R.. “A Comparison of Commercial and Military Computer Security Policies”, *Proceedings of the 1987 IEEE Symposium on Security and Privacy*, IEEE Computer Society Press, April 1987.

[6] Rueppel, R.A., A Formal Approach to Security Architectures, Advances in Cryptology - EUROCRYPT ’91, Lecture Notes in Computer Science 547, D.W. Davies (Editor), Springer-Verlag.

[7] Open Systems Interconnection Reference Model, Part 1: Basic Reference Model, ISO 7498-1 (CCITT X.200). Melbourne 1988.

[8] Open Systems Interconnection Reference Model, Part 2: Security Architecture, ISO DIS 7498-2, July 19, 1988.