Flexilink

Features that help with security

Background

Traditional networking protocol suites such as TCP/IP route individual packets; all the information required to identify a packet's source and to convey it to its destination is contained in the packet's header. Security (other than detecting data corruption) was not an issue when the system was originally designed, so the format of packet headers does not include information such as authentication of the sender. There is no opportunity for the recipient of a packet to interrogate the sender.

With Flexilink, each packet is part of a "flow" and its header simply contains a "handle" value identifying the flow. An application wishing to communicate with a remote entity begins by setting a flow up, and that process can include as much or as little authentication of the communicating entities (network elements as well as the endpoints) as the application requires. Packets cannot be delivered to an application entity except via a flow for which the entity has participated in the setting-up process.

For details of the procedures see ETSI GS NIN 005.

Network elements

A network consists of **nodes** connected together by **links**. A node can be a single piece of equipment or a group of interconnected devices sharing a single control entity.

Each node can be considered to consist of a control plane, a forwarding plane, and a routing table.

The routing table contains an entry for each flow. The forwarding plane entity (which can be implemented entirely in hardware) identifies the flow to which each incoming packet belongs, and outputs it as defined by the flow's entry in the routing table. The mechanism is different for the two services (basic and guaranteed) but the principle is the same in each case.

Entries in the routing table are written by the control plane entity, as part of the flow set-up and tear-down processes which are implemented by exchanging **signalling** messages with adjacent nodes.

Links

A link carries packets between two nodes; it can be physical (wire or fibre) or wireless. Each node is referred to as the **link partner** of the other. Links that connect more than two nodes are not currently supported; however, similar principles would apply.

When the link is connected, a pair of basic service flows (one in each direction) is set up to carry signalling messages between the two nodes; these are referred to as **signalling flows**. The process of connecting the link can include authentication of the link partner, or in the case of a physical link the two nodes and the cable might be within a secure area. Either way, the amount of trust each node can place in the other will be known.

All packets on a signalling flow can be assumed to have been checked by the link partner, provided it has been identified as trustworthy.

Incoming signalling flows are routed to the node’s control plane, and should be given priority over other flows routed to the control plane (such as network management), to reduce the opportunities for denial of service. Similarly, outgoing signalling packets should be given priority over other basic service flows.

Flow set-up

An application instance (the **originator**) requests a flow to be set up by sending a FindRoute Request message to the link partner on its link to the network. If it has more than one link, it may send the message to more than one partner. The message includes Information Elements (**IEs**) each of which describes some aspect of the flow, such as the destination address or data format.

A remote application instance (such as a server) which responds to a request is called a **responder**; it sends a Response message back to the originator. Request and Response are referred to as message **classes**. Two further classes can be used if required: Confirmation from the originator to the responder and Completion from responder to originator.

A flow is set up in the routing tables along the path followed by a FindRoute message. In the case of a basic service flow this is a message which travels in the direction from destination to sender: on each link the sender of the signalling message allocates a routing table entry to the flow and includes the label value to be used in packets in a BasicAlloc IE. Its link partner uses that value in the “label for next hop” field of its own routing table entry.

There are two options for a bidirectional basic service connection from a client to a server. The simplest is to use two message classes, with the flow from server to client being set up by the Request message. The message can include authentication information, such as username and password hash, in a Password IE (see clause 5.7.8 of GS NIN 005). If the server rejects the call it replies with a Terminate message which results in the routing table entries being freed. Otherwise it sends a FindRoute Response with a BasicAlloc IE for the flow from client to server.

Alternatively, all four message classes can be used, with the Request and Response messages being used for an authentication protocol and the routing tables being written by the Confirmation and Completion messages. Some of the currently reserved IE types could be used for further security-related functionality.

Similar considerations apply to the guaranteed service.