Internet Draft
Expiration: March 2003
File: draft-ietf-forces-framework-01.txt
Working Group: ForCES

L. Yang
Intel Labs
R. Dantu
Netrake Corp.
T. Anderson
Intel Labs

Sept 2002

ForCES Architectural Framework

draft-ietf-forces-framework-01.txt

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as ``work in progress.''

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC-2119].

1. Abstract

This document defines the architectural framework for ForCES network elements (NE), and identifies the associated entities and the interaction among them. This framework is intended to satisfy the requirements specified in the ForCES requirements draft [FORCES-REQ].

2. Definitions

A set of terminology associated with the ForCES requirements is defined in [FORCES-REQ] and we only include the ones that are most relevant to this document here.

Forwarding Element (FE) - A logical entity that implements the ForCES protocol. FEs use the underlying hardware to provide per-packet processing and handling as directed by a CE via the ForCES protocol.

Control Element (CE) - A logical entity that implements the ForCES protocol and uses it to instruct one or more FEs how to process packets. CEs handle functionality such as the execution of control and signaling protocols.

ForCES Network Element (NE) - An entity composed of one or more CEs and one or more FEs. To entities outside a NE, the NE represents a single point of management. Similarly, a NE usually hides its internal organization from external entities.

Pre-association Phase - The period of time during which a FE Manager (see below) and a CE Manager (see below) are determining which FE and CE should be part of the same network element. Any partitioning of PFEs and PCEs occurs during this phase.

Post-association Phase - The period of time during which a FE does know which CE is to control it and vice versa, including the time during which the CE and FE are establishing communication with one another.

ForCES Protocol - While there may be multiple protocols used within the overall ForCES architecture, the term "ForCES protocol" refers only to the ForCES post-association phase protocol (see below).

ForCES Post-Association Phase Protocol - The protocol used for post-association phase communication between CEs and FEs. This protocol does not apply to CE-to-CE communication, FE-to-FE communication, or to communication between FE and CE managers. The ForCES protocol is a master-slave protocol in which FEs are slaves and CEs are masters. This protocol includes both the management of the communication channel (e.g., connection establishment, heartbeats) and the control messages themselves. This protocol could be a single protocol or could consist of multiple protocols working together.

FE Manager - A logical entity that operates in the pre-association

phase and is responsible for determining to which CE(s) a FE should communicate. This process is called CE discovery and may involve the FE manager learning the capabilities of available CEs. A FE manager may use anything from a static configuration to a pre-association phase protocol (see below) to determine which CE(s) to use. Being a logical entity, a FE manager might be physically

combined with any of the other logical entities mentioned in this section.

CE Manager - A logical entity that operates in the pre-association phase and is responsible for determining to which FE(s) a CE should communicate. This process is called FE discovery and may involve the CE manager learning the capabilities of available FEs. A CE manager may use anything from a static configuration to a pre-association phase protocol (see below) to determine which FE to use. Being a logical entity, a CE manager might be physically combined with any of the other logical entities mentioned in this section.

Pre-association Phase Protocol - A protocol between FE managers and CE managers that is used to determine which CEs or FEs to use. A pre-association phase protocol may include a CE and/or FE capability discovery mechanism. Note that this capability discovery process is wholly separate from (and does not replace) that used within the ForCES protocol. However, the two capability discovery mechanisms may utilize the same FE model.

FE Model - A model that describes the logical processing functions of a FE.

ForCES Protocol Element - A FE or CE.

3. Introduction to Forwarding and Control Element Separation (ForCES)

An IP network element (NE) appears to external entities as a monolithic piece of network equipment, e.g., a router, NAT, firewall, or load balancer. Internally, however, an IP network element (NE) (such as a router or switch) is composed of numerous logically separated entities that cooperate to provide a given functionality (such as routing or IP switching). Two types of network element components exist: control element (CE) in control plane and forwarding element (FE) in forwarding plane (or data plane). Forwarding elements typically are ASIC, network-processor, or general-purpose processor-based devices that handle data path operations for each packet. Control elements are typically based on general-purpose processors that provide control functionality like routing and signaling protocols.

ForCES aims to define a framework and associated protocol(s) to standardize the exchange of information between the control plane and the forwarding plane. Having standard mechanisms between the CEs and FEs allow these components to be physically separated. This physical separation accrues several benefits to the ForCES architecture. Separate components would allow component vendors to specialize in one component without having to become experts in all

components. It also allows CEs and FEs from different component vendors to interoperate with each other and hence it becomes possible for system vendors to integrate together CEs and FEs from different component vendors. This translates into a lot more design choices and flexibility to the system vendors. Overall, ForCES will

enable rapid innovation in both the control and forwarding planes while maintaining interoperability. Scalability is also easily provided by this architecture in that additional forwarding or control capacity can be added to existing network elements without the need for forklift upgrades.

One example of such physical separation is at the blade level. Figure 1 shows an example configuration of a router, with two control blades and multiple router (forwarding) blades, all interconnected into a switch fabric backplane. In such chassis configuration, the control blades are the CEs while the router blades are FEs, and the switch fabric backplane provides the physical interconnect for all the blades. Routers today with this kind of configuration use proprietary interface for messaging between CEs and FEs. The goal of ForCES is to replace such proprietary interface with a standard protocol. With a standard protocol like ForCES implemented on all blades, it becomes possible for control blades from vendor X and routing blades from vendor Y to work seamlessly together in one chassis.

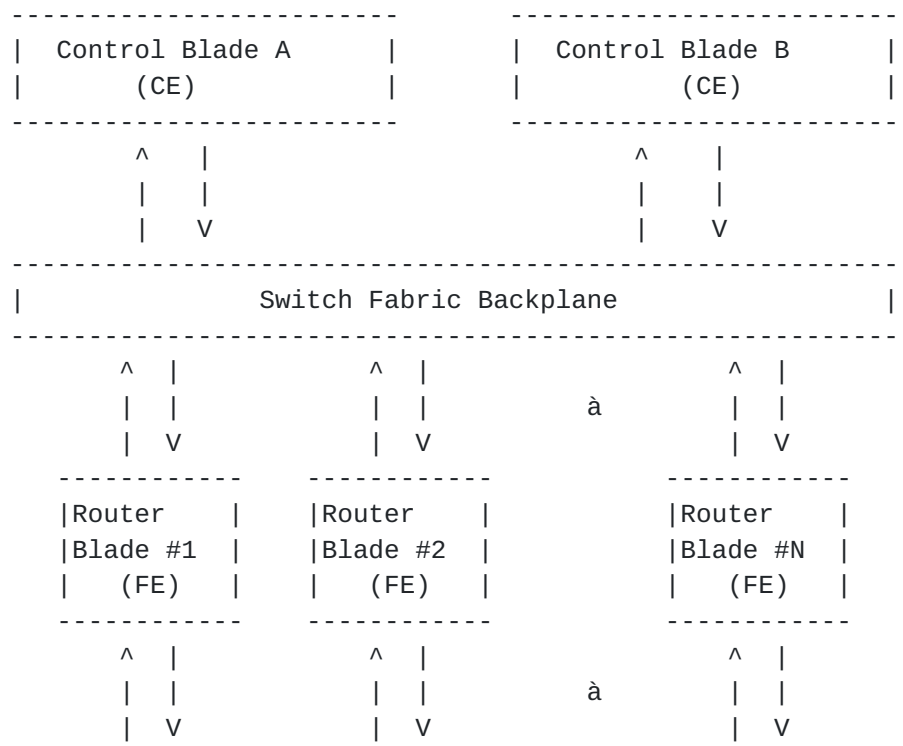


Figure 1. A router configuration example with separate blades.

Another level of physical separation between the CEs and FEs can be at the box level. In such configuration, all the CEs and FEs are physically separated boxes, interconnected with some kind of high

speed LAN connection (like Gigabit Ethernet). These separated CEs and FEs are only one hop away from each other within a local area network. The CEs and FEs communicate to each other by running ForCES, and the collection of these CEs and FEs together become one routing unit to the external world. Figure 2 shows such an example.

In this example, the same physical interconnect (Ethernet) is shared for both CE-to-FE and FE-to-FE communication. However, that does not have to be the case. One reason to use different interconnect might be that CE-to-FE interconnect does not have to be as fast as the FE-to-FE interconnect, so the more expensive fast ports can be saved for FE-to-FE. The separate interconnects may also provide reliability and redundancy benefits for the NE.

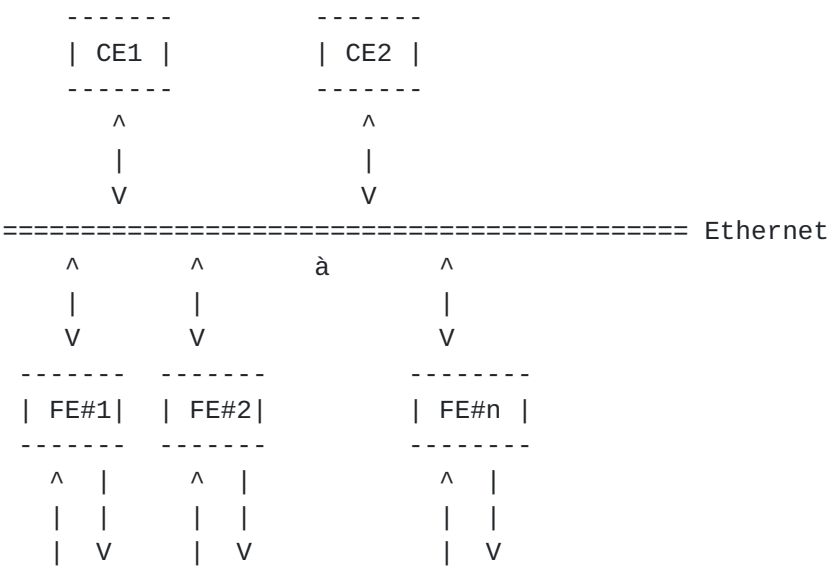


Figure 2. A router configuration example with separate boxes.

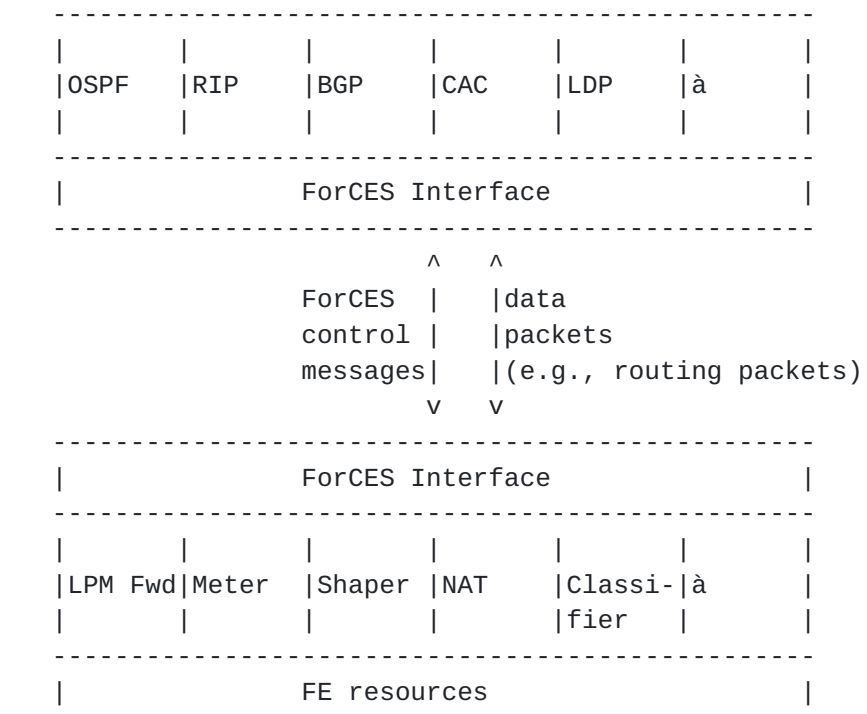


Figure 3. Examples of CE and FE functions

Some examples of control functions that can be implemented in the CE include routing protocols like RIP, OSPF and BGP, control and signaling protocols like CAC (Call Admission Control), LDP (Label Distribution Protocol) for MPLS, etc. Examples of forwarding functions in FE include LPM (longest prefix match) forwarder, classifiers, traffic shaper, meter, NAT, etc. Figure 3 shows a diagram with examples in both CE and FE. Any given NE may contain one or many of these CE and FE functions in it. The diagram also shows that ForCES protocol is used to transport both the control messages for ForCES itself and the data packets that are originated/destined from/to the control functions in CE (e.g., routing packets). Section 5.2.4 provides more detail on this.

A set of requirements for control and forwarding separation is identified in [FORCES-REQ]. This document describes a ForCES architecture that satisfies the architectural requirements of that document and defines a framework for ForCES network elements and associated entities to facilitate protocol definition.

4. Architecture

This section defines the ForCES architectural framework and the associated logical components. This ForCES framework defines components of ForCES NEs including several ancillary components. These components may be connected in different kinds of topologies for flexible packet processing.

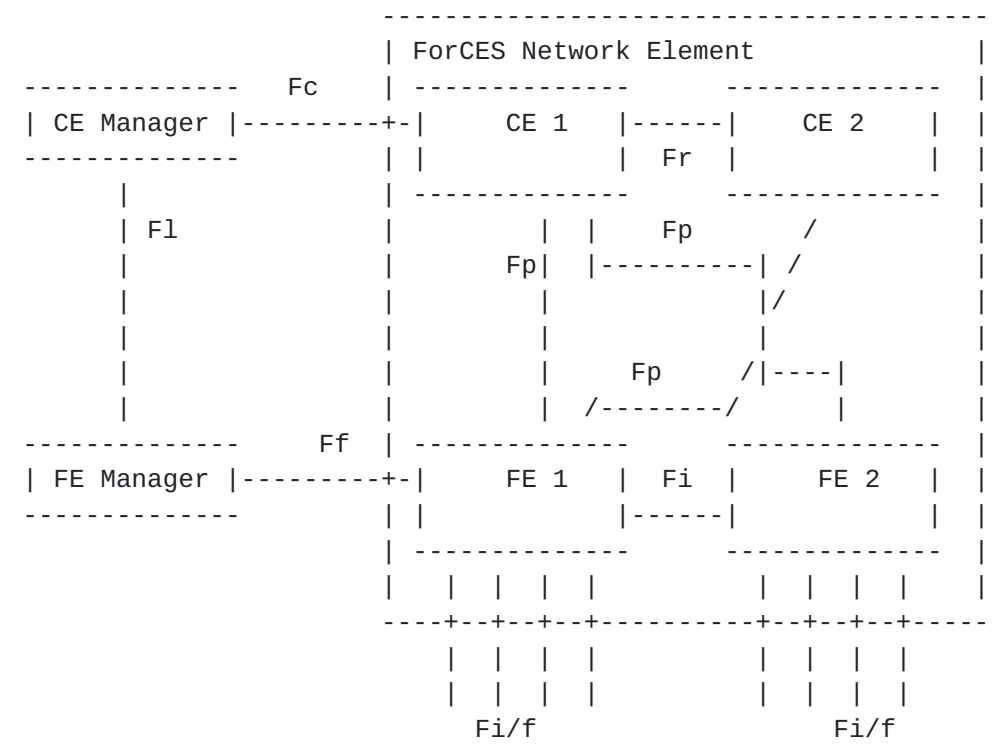


Figure 4. ForCES Architectural Diagram

The diagram in Figure 4 shows the logical components of the ForCES architecture and their relationships. There are two kinds of

components inside a ForCES network element: control element (CE) and forwarding element (FE). The framework allows multiple instances of CE and FE inside one NE. Each FE contains one or more physical media interfaces for receiving and transmitting packets from/to the external world. The aggregation of these FE interfaces becomes the NEÆs external interfaces. In addition to the external interfaces, there must also exist some kind of interconnect within the NE so that the CE and FE can communicate with each other, and one FE can forward packets to another FE. The diagram also shows two entities outside of the ForCES NE: CE Manager and FE Manager. These two entities provide configuration to the corresponding CE or FE in the pre-association phase (see Section 5.1). There is no defined role for FE Manager and CE Manager in post-association phase, thus these logical components are not considered part of the ForCES NE.

For convenience, the logical interactions between these components are labeled by reference points Fp, Fc, Ff, Fr, Fl, and Fi, as shown in Figure 4. The FE external interfaces are labeled as Fi/f. More detail is provided in Section 4 and 5 for each of these reference points. All these reference points are important in understanding the ForCES architecture, however, the ForCES protocol is only defined over one reference point -- Fp.

The interface between two ForCES NEs is identical to the interface between two conventional routers and these two NEs exchange the protocol packets through the external interfaces at Fi/f. ForCES NEs connect to existing routers transparently.

4.1. Control Elements and Fr Reference Point

It is not necessary to define any protocols across the Fr reference point to enable control and forwarding separation for simple configurations like single CE and multiple FEs. However, this architecture permits multiple CEs to be present in a network element. In cases where an implementation uses multiple CEs, it is expected the invariant that the CEs and FEs together appear as a single NE MUST be maintained.

Multiple CEs may be used for redundancy, load sharing, distributed control, or other purposes. Redundancy is the case where one or more CEs are prepared to take over should an active CE fail. Load sharing is the case where two or more CEs are concurrently active and where any request that can be serviced by one of the CEs can also be serviced by any of the other CEs. In both redundancy and load sharing, the CEs involved are equivalently capable. The only difference between these two cases is in terms of how many active CEs there are. Distributed control is the case where two or more CEs are concurrently active but where certain requests can only be

serviced by certain CEs.

When multiple CEs are employed in a ForCES NE, their internal organization is considered an implementation issue that is beyond the scope of ForCES. CEs are wholly responsible for coordinating

amongst themselves via the Fr reference point to provide consistency and synchronization. However, ForCES does not define the implementation or protocols used between CEs, nor does it define how to distribute functionality among CEs. Nevertheless, ForCES will support mechanisms for CE redundancy or fail over, and it is expected that vendors will provide redundancy or fail over solutions within this framework.

4.2. Forwarding Elements and Fi reference point

FEs perform per-packet processing and handling as directed by CEs. FEs have no initiative of their own. Instead, FEs are slaves and only do as they are told. FEs may communicate with one or more CEs concurrently across reference point Fp. FEs have no notion of CE redundancy, load sharing, or distributed control. Instead, FEs accept commands from any CE authorized to control them, and it is up to the CEs to coordinate among themselves to achieve redundancy, load sharing or distributed control. The idea is to keep FEs as simple and dumb as possible so that FEs can focus its resource on the packet processing functions.

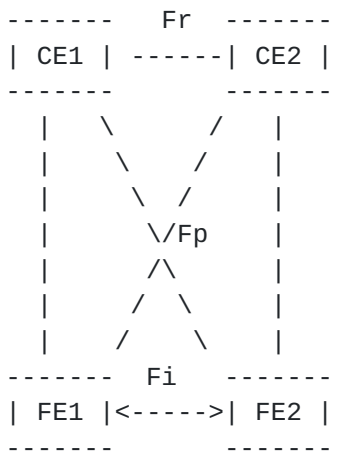


Figure 5. CE redundancy example.

For example, in Figure 5, FE1 and FE2 can be configured to accept commands from both the primary CE (CE1) and the backup CE (CE2). At the beginning, CE1 issues commands to FEs while CE2 silently remains in sync with CE1 via CE to CE protocol over Fr reference point. When CE1 fails, CE2 detects it and starts to take over. Before CE2 starts issuing commands to the FEs, it might need to recheck the FEs' state and instruct FEs whether or not it is ok to preserve their current state.

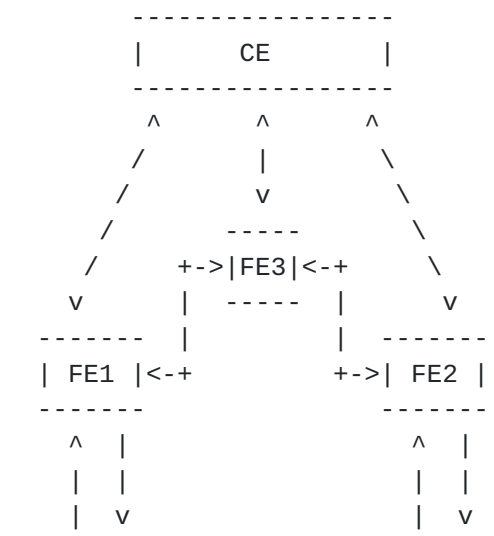
Distributed control can be achieved in the similar fashion, without

much intelligence on the part of FEs. For example, FEs can be configured to detect RSVP and BGP protocol packets, and forward RSVP packets to one CE and BGP packets to another CE. Hence, FEs may need to do packet filtering for forwarding packets to specific CEs.

This architecture permits multiple FEs to be present in a NE. [FORCES-REQ] dictates that the ForCES protocol MUST be able to scale to at least hundreds of FEs (see [FORCES-REQ] Section 5, requirement #11). Each of these FEs may potentially have a different set of packet processing functions, with different media interfaces. FEs are responsible for basic maintenance of layer-2 connectivity with other FEs and with external entities. Many layer-2 media include sophisticated control protocols. The FORCES protocol (over the Fp reference point) will be able to carry messages for such protools so that, in keeping with the "dumb FE model" the CE can provide appropriate intelligence and control over these media.

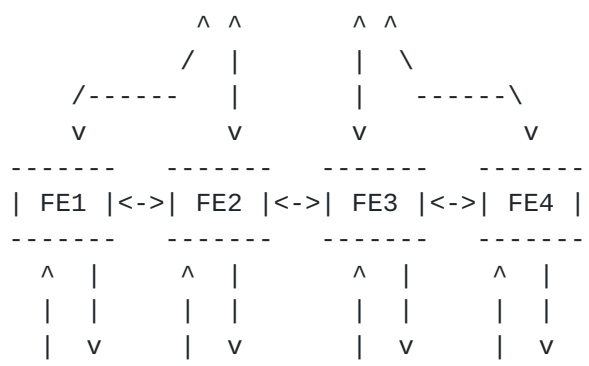
When multiple FEs are present, ForCES requires that packets MUST be able to arrive at the NE by one FE and leave the NE via a different FE (See [FORCES-REQ], Section 5, Requirement #3). Packets that enter the NE via one FE and leave the NE via a different FE are transferred between FEs across the Fi reference point. The Fi reference point is a separate protocol from the Fp reference point and is not currently defined by the ForCES architecture.

FEs could be connected in different kinds of topologies and packet processing may spread across several FEs in the topology. Hence, logical packet flow may be different from physical FE topology. Figure 6 provides some topology examples. When it is necessary to forward packets between FEs, CE needs to understand the FE topology. The FE topology can be queried from FEs by CEs. If the most common FE topology is full mesh among FEs, ForCES can assume it as the default topology for FEs and hence no query is needed for such default cases.

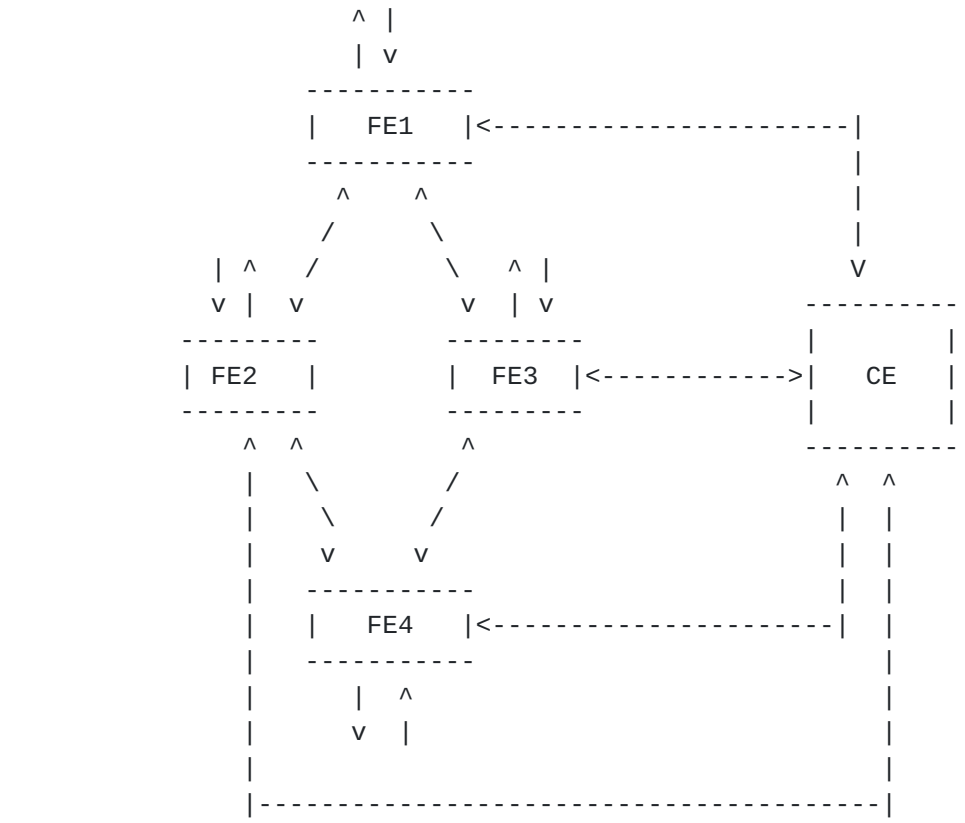


(a) Full mesh among FE1, FE2 and FE3.


CE



(b) Multiple FEs in a daisy chain



(c) Multiple FEs connected by a ring

Figure 6. Some examples of FE topology.

4.3. CE Managers

CE managers are responsible for determining which FEs a CE should control. It is legitimate for CE managers to be hard-coded with the knowledge of with which FEs its CEs should communicate. A CE manager may also be physically embedded into a CE and be implemented as a

simple keypad or other direct configuration mechanism on the CE. Finally, CE managers may be physically and logically separate entities that configure the CE with FE information via such mechanisms as COPS-PR [RFC3084] or SNMP [RFC1157].

4.4. FE Managers

FE managers are responsible for determining to which CE any particular FE should initially communicate. Like CE managers, no restrictions are placed on how a FE manager decides to which CEs its FEs should communicate, nor are restrictions placed on how FE managers are implemented.

5. Operational Phases

Both FEs and CEs require some configuration in place before they can start information exchange and function as a coherent network element. Two operational phases are identified in this framework -- pre-association and post-association.

5.1.Pre-association Phase

Pre-association phase is the period of time during which a FE Manager and a CE Manager are determining which FE and CE should be part of the same network element. The protocols used during this phase may include all or some of the message exchange over Fl, Ff and Fc reference points. However, all these may be optional and none of this is within the scope of ForCES protocol.

5.1.1. Fl Reference Point

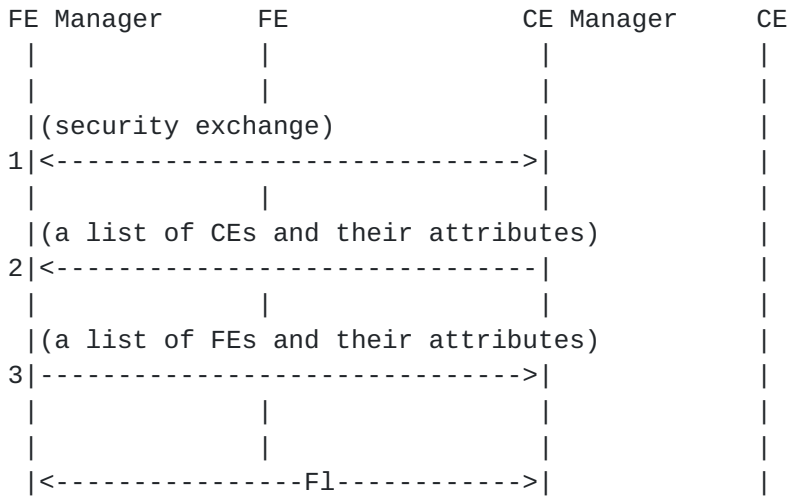


Figure 7. An example of message exchange over Fl reference point

CE managers and FE managers may communicate across the Fl reference point in the pre-association phase in order to determine which CEs and FEs should communicate with each other. Communication across the Fl reference point is optional in this architecture. No requirements are placed on this reference point.

CE managers and FE managers may be operated by different entities. The operator of the CE manager may not want to divulge, except to specified FE managers, any characteristics of the CEs it manages.

Similarly, the operator of the FE manager may not want to divulge FE characteristics, except to authorized entities. As such, CE managers and FE managers may need to authenticate one another. Subsequent communication between CE managers and FE managers may require other security functions such as privacy, non-repudiation, freshness, and integrity.

Once the necessary security functions have been performed, the CE and FE managers communicate to determine which CEs and FEs should communicate with each other. At the very minimum, the CE and FE managers need to learn of the existence of available FEs and CEs respectively. This discovery process may or may not entail one or both managers learning the capabilities of the discovered ForCES protocol elements. Figure 7 shows an example of possible message exchange between CE manager and FE manager over Fl reference point.

5.1.2. Ff Reference Point

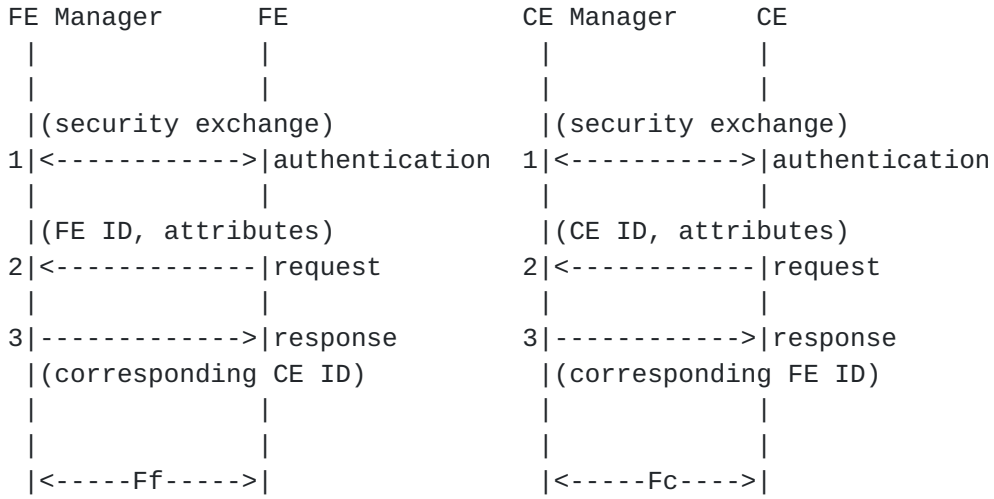


Figure 8. Examples of message exchange over Ff and Fc reference points.

The Ff reference point is used to inform forwarding elements of the association decisions made by FE managers in pre-association phase. Only authorized entities may instruct a FE with respect to which CE should control it. Therefore, privacy, integrity, freshness, and authentication are necessary between FE manager and FEs when the FE manager is remote to the FE. Once the appropriate security has been established, FE manager instructs FEs across this reference point to join a new NE or to disconnect from an existing NE. Figure 8 shows example of message exchange over Ff reference point.

Note that when the FE manager function may be co-located with the FE (such as by manual keypad entry of the CE IP address), in which case

this reference point is reduced to a built-in function.

5.1.3. Fc Reference Point

The Fc reference point is used to inform control elements of the association decisions made by CE managers in pre-association phase. When the CE manager is remote, only authorized entities may instruct a CE to control certain FEs. Privacy, integrity, freshness and authentication are also required across this reference point in such a configuration. Once appropriate security has been established, the CE manager instructs CEs as to which FEs they should control and how they should control them. Figure 7 shows example of message exchange over Fc reference point.

As with the FE manager and FEs, configurations are possible where the CE manager and CE are co-located and no protocol is used for this function.

5.2. Post-association Phase and Fp reference point

Post-association phase is the period of time during which a FE and CE have been configured with information necessary to contact each other and includes both communication establishment and steady-state communication. The communication between CE and FE is performed across the Fp ("p" meaning protocol) reference point. ForCES protocol is exclusively used for all communication across the Fp reference point.

5.2.1. Proximity and Interconnect between CEs and FEs

The ForCES Working Group has made a conscious decision that the first version of ForCES will not be designed to support configurations where the CE and FE are located arbitrarily in the network. In particular, ForCES is intended for "very close" CE/FE localities in IP networks, as defined by ForCES Applicability Statement ([FORCES-APP]). Very Close localities consist of control and forwarding elements that either are components in the same physical box, or are separated at most by one local network hop.

CEs and FEs can be connected by a variety of interconnect technologies, including Ethernet connections, backplanes, ATM (cell) fabrics, etc. ForCES should be able to support each of these interconnects (see [FORCES-REQ] Section 5, requirement #1). ForCES will make use of an existing RFC2914 compliant L4 protocol with adequate reliability, security and congestion control (e.g. TCP, SCTP) for transport purposes.

5.2.2. Association Establishment

As an example, figure 9 shows some of the message exchange that need to happen before the association between CE and FE is fully established. Typically, FE would need to inform the CE of its own capability and its topology in relation to other FEs. The capability

of FE is represented by FE model, described in another separate document. The model would allow FE to describe what kind of packet processing functions it contains, in what order these processing happen, what kind of configurable parameters it allows, what

statistics it collects and what events it might throw, etc. Once such information is available to CE, CE sends all the necessary configuration to FE so that FE can start receiving and processing packets. For example, CE might need to send a snapshot of the current routing table to FE so that FE can start routing packets correctly. Once FE starts accepting packets for processing, we say the association of this FE with its CE is now established. From then on, CE and FE enter steady-state communication as described in 5.2.2.

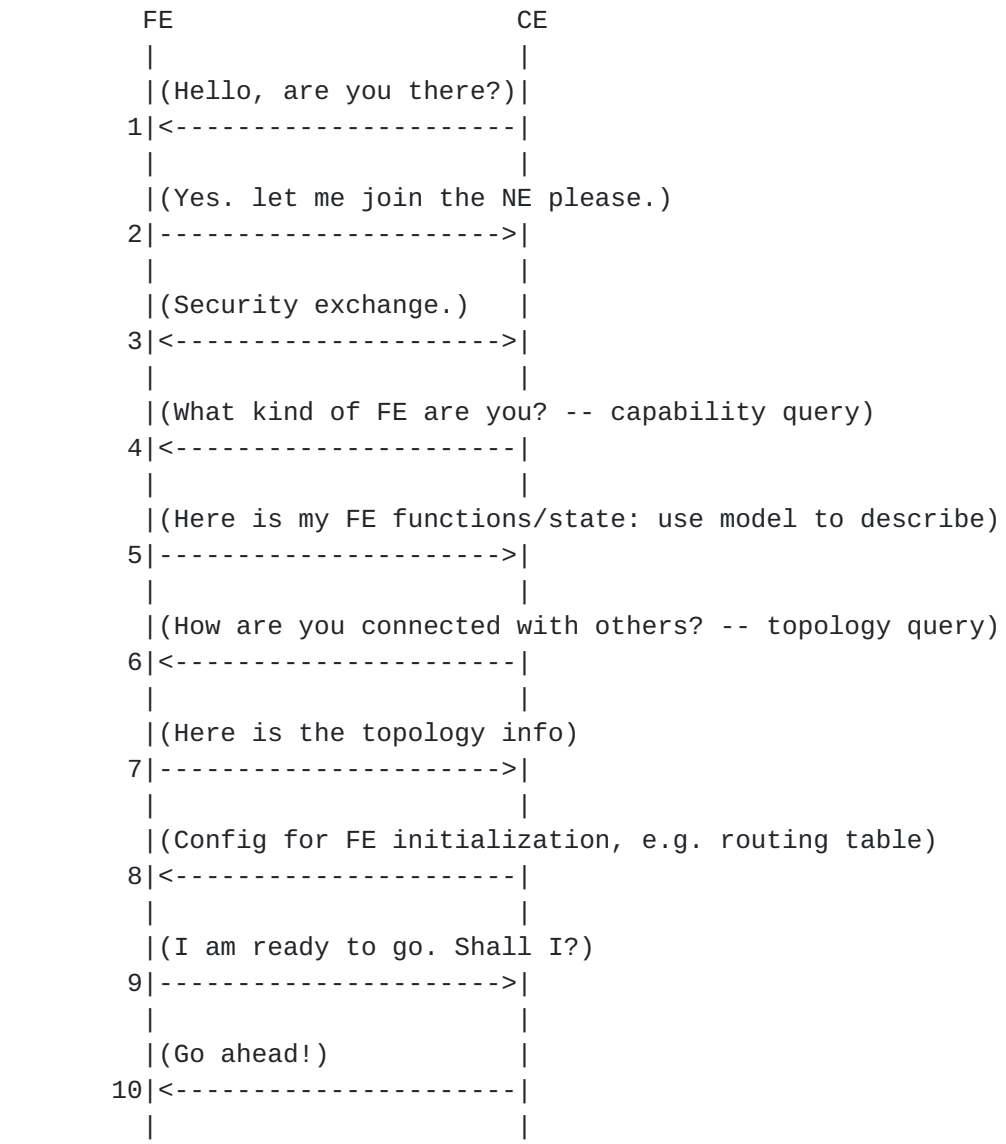


Figure 9. Example of message exchange between CE and FE over Fp to establish NE association

5.2.3. Steady-state Communication

Once an association is established between the CE and the FE, the ForCES protocol is used by the CE and the FE over Fp reference point to exchange information to facilitate packet processing.

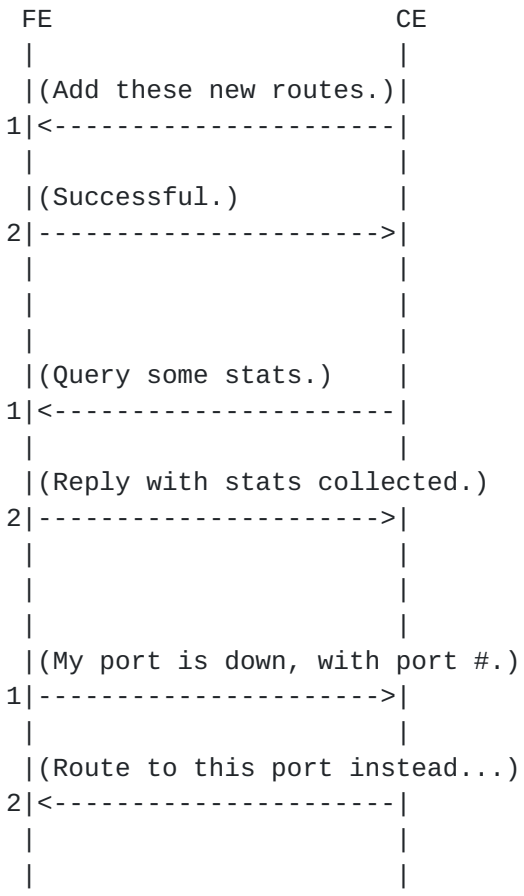


Figure 10. Examples of message exchange between CE and FE over Fp during steady-state communication

Based on the information acquired through CEs' control processing, CEs will frequently need to manipulate the packet-forwarding behaviors of their FE(s) by sending instructions to FEs. For example, Figure 10 shows one such message exchange in which CE sends new routes to FE so that FE can add them to its routing table. CE can also query FE for statistics collected by FE and FE can also notify CE of some important events, like interface up and down, etc. Figure 8 also shows such examples.

5.2.4. Data Packets across Fp reference point

Control packets (such as RIP, OSPF messages) addressed to any of NE's interfaces are typically redirected by the receiving FE to its CE, and CE may originate packets and have its FE deliver them to other NEs. Therefore, the communication across the Fp reference

point includes not only the control messages from CEs to FEs and the status or statistics report from FEs to CEs, but also the data packets that are redirected between them. Moreover, one FE may be controlled by multiple CEs. In this configuration, the control protocols supported by the FORCES NEs may spread across multiple

CEs. For example, one CE may support OSPF and another supports BGP. FEs are configured to recognize these protocol packets and forward them to the corresponding CE.

Figure 11 shows one example of how the OSPF packets originated by router A are passed to router B. In this example, ForCES protocol is used to transport the packets from CE to FE inside router A, and then from FE to CE inside router B. In light of the fact that ForCES protocol is responsible to transport both the control messages and the data packets between CE and FE over Fp reference point, it is possible to use either a single protocol or multiple protocols to achieve that.

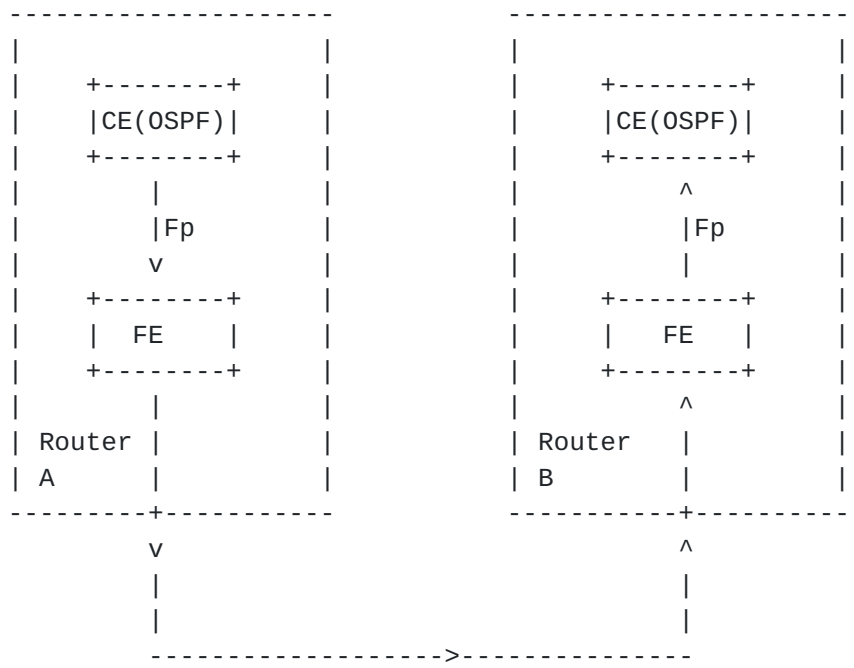


Figure 11. Example to show data packet flow between two NEs.

5.2.5. Proxy FE

In the case where a physical FE cannot implement (e.g., due to the lack of a general purpose CPU) the ForCES protocol directly, a proxy FE can be used in the middle of Fp reference point. This allows the CE communicate to the physical FE via the proxy by using ForCES, while the proxy manipulates the physical FE using some intermediary form of communication (e.g., a non-ForCES protocol or DMA). In such an implementation, the combination of the proxy and the physical FE becomes one logical FE entity.

5.3. Association Re-establishment

FEs and CEs may join and leave NEs dynamically (see [FORCES-REQ] Section 5, requirements #12 and #13). When a FE or CE leaves the NE, the association with the NE is broken. If the leaving party rejoins

a NE later, to re-establish the association, it may or may not need to re-enter the pre-association phase. Loss of association can also happen unexpectedly due to loss of connection between the CE and the FE. Therefore, the framework allows the bi-directional transition between these two phases, but the ForCES protocol is only applicable for the post-association phase. However, the protocol should provide mechanisms to support association re-establishment (see [FORCES-REQ] Section 5, requirement #7).

Let's use the example in Figure 5 to see what happens when the association is broken and later re-established again. Section 4.2 already explains what happens if CE1 fails and how CE2 can take over. Note that if no CE redundancy is provided, FEs need to be told at the association establishment time what to do in the case of CE failure. FEs may be told to stop packet processing all together if its CE fails. Or, FEs may be told to continue forwarding packets even in the face of CE failure. No matter what, it needs to be part of the configuration when the association is established.

Let's now look at the case when FE1 leaves the NE temporarily, assuming CE1 is the working CE for the moment. FE1 may voluntarily decides to leave the association. Or, it is more likely that FE1 stops functioning simply due to unexpected failure. In former case, CE1 receives a "leave-association request" from FE1. In the latter, CE1 detects the failure of FE1 by some other mean. In both cases, CE1 would keep a note of such event for FE1 while continue commanding FE2. When FE1 decides to rejoin again, or when it is back up again from the failure, FE1 would need to re-discover its master (CE). This can be achieved by several means. It may re-enter the pre-association phase and get that information from its FE manager. It may retrieve the previous CE information from its cache, if it decides that the information is still valid. Or, that information can be simply hard-coded or pre-configured into it. Once it discovers its CE, it starts message exchange with CE to re-establish the association just as outlined in Figure 9, with the possible exception that it might be able to bypass the transport of the complete initialization information. Suppose that FE1 still have its routing table and other state information from the last association, instead of sending all the information again from scratch, it can choose to use more efficient mechanism to re-sync up the state with its CE. For example, a checksum of the state might give a quick indication of whether or not the state is in-sync with its CE. By comparing its state with CE first, it sends information update only if it is needed.

6. Applicability to RFC1812

[FORCES-REQ] Section 5, requirement #9 dictates that "All proposed ForCES architecture MUST explain how that architecture may be applied to support all of a router's functions as defined in [RFC1812]." RFC1812 discusses many important requirements for IPv4 routers from the link layer to the application layer. This section

addresses the relevant requirements for implementing IPv4 routers based on ForCES architecture and how ForCES satisfies these requirements.

6.1. General Router Requirements

Routers have at least two or more logical interfaces. When CEs and FEs are separated by ForCES within a single NE, some additional interfaces are needed for intra-NE communications. Figure 12 shows an example to illustrate that. This NE contains one CE and two FEs. Each FE has four interfaces; two of them are used for receiving and transmitting packets to the external world, while the other two are for intra-NE connections. CE has two logical interfaces #9 and #10, connected to interfaces #3 and #6 from FE1 and FE2, respectively. Interface #4 and #5 are connected for FE1-FE2 communication. So this router NE provides four external interfaces (#1, 2, 7 and 8).

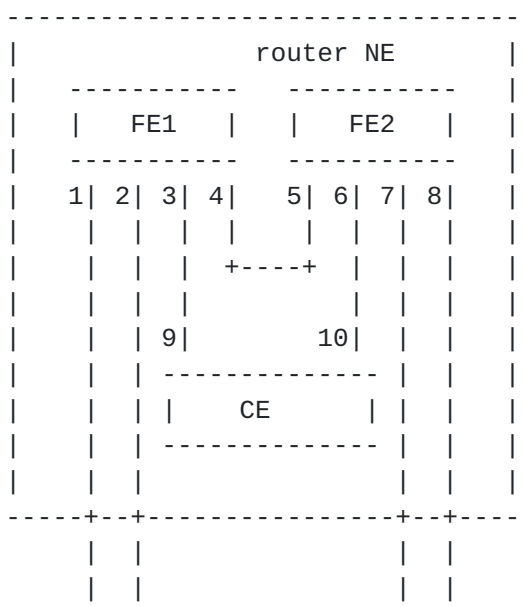


Figure 12. A router NE example with four interfaces.

IPv4 routers must implement IP to support its packet forwarding function, which is driven by its FIB (Forwarding Information Base). This Internet layer forwarding (see [RFC1812] Section 5) functionality naturally belongs to FEs in the ForCES architecture.

A router may implement transport layer protocols (like TCP and UDP) that are required to support application layer protocols (see [RFC1812] Section 6). One important class of application protocols is routing protocols (see [RFC1812] Section 7). In ForCES architecture, routing protocols are naturally implemented by CEs. Routing protocols require routers communicate with each other. This

communication between CEs in different routers is supported in ForCES by the FEs' ability to redirect data packets addressed to routers (i.e., NEs) and CEs' ability to originate packets and have them delivered by their FEs. This communication

occurs across Fp reference point inside each router and between neighboring routers' interfaces, as illustrated in Figure 11.

6.2.Link Layer

Since FEs own all the external interfaces for the router, FEs need to conform to the link layer requirements in RFC1812. Theoretically, ARP support may be implemented in either CEs or FEs. As we will see later, a number of behaviors that RFC1812 mandates fall into this category -- they may be performed by the FE and may be performed by the CE. A general guideline is needed to ensure interoperability between separated control and forwarding planes. The guideline we offer here is that CEs are required to be capable of these operations while FEs may or may not choose to implement them. FE model should indicate its capabilities in this regard.

Interface parameters, including MTU, IP address, etc., must be configurable by CEs via ForCES. CEs must be able to determine whether a physical interface in an FE is available to send packets or not. FEs must also inform CEs the status change of the interfaces (like link up/down) via ForCES.

6.3.Internet Layer Protocols

Both FEs and CEs must implement IP protocol and all mandatory extensions as RFC1812 specified. CEs should implement IP options like source route and record route while FEs may choose to implement those as well. Timestamp option should be implemented by FEs to insert the timestamp most accurately. FE must interpret the IP options that it understands and preserve the rest unchanged for use by CEs. Both FEs and CEs might choose to silently discard packets without sending ICMP errors, but such events should be logged and counted. FEs can report statistics for such events to CEs via ForCES.

When multiple FEs are involved to process packets, the appearance of single NE must be strictly maintained. For example, Time-To-Live (TTL) must be decremented only once within a single NE. For example, it can be always decremented by the last FE with egress function.

FEs must receive and process normally any packets with a broadcast destination address or a multicast destination address that the router has asked to receive. When IP multicast is supported in routers, IGMP is implemented in CEs. CEs are also required of ICMP support, while it is optional for FEs to support ICMP. Such an option can be communicated to CEs as part of the FE model. Therefore, FEs can always rely upon CEs to send out ICMP error messages, but FEs also have the option to generate ICMP error

messages themselves.

6.4.Internet Layer Forwarding

IP forwarding is implemented by FEs. After routing protocol update its routing tables at CEs, ForCES is used to send the new routing table entries from CEs to FEs. Each FE has its own routing table and uses this table to direct packets to the next hop interface.

Upon receiving IP packets, FE verifies the IP header and process most of the IP options. Some options can't be processed until the routing decision has been made. Routing decision is made after examining the destination IP address. If the destination address belongs to the router itself, the packets are forwarded to CE. Otherwise, FE determines the next hop IP address by looking up in its routing table. FE also determines the network interface it uses to send the packets. Sometimes FE may need to forward the packets to another FE before packets can be forwarded out to the next hop. Right before packets are forwarded out to the next hop, FE decrements TTL by 1 and processes any IP options that cannot be processed before. FE performs any IP fragmentation if necessary, determines link layer address (e.g., by ARP), and encapsulates the IP datagram (or each of the fragments thereof) in an appropriate link layer frame and queues it for output on the interface selected.

Other options mentioned in RFC1812 for IP forwarding may also be implemented at FEs, for example, packet filtering.

FEs typically forward packets destined locally to CEs. FEs may also forward exceptional packets (packets that FEs don't know how to handle) to CEs. CEs are required to handle packets forwarded by FEs for whatever different reasons. It might be necessary for ForCES to attach some meta-data with the packets to indicate the reasons of forwarding from FEs to CEs. Upon receiving packets with meta-data from FEs, CEs can decide to either process the packets themselves, or pass the packets to the upper layer protocols including routing and management protocols. If CEs are to process the packets by themselves, CEs may choose to discard the packets, or modify and re-send the packets. CEs may also originate new packets and deliver them to FEs for further forwarding.

Any state change during router operation must also be handled correctly according to RFC1812. For example, when an FE ceases forwarding, the entire NE may continue forwarding packets, but it needs to stop advertising routes that are affected by the failed FE.

6.5.Transport Layer

Transport layer is typically implemented at CEs to support higher layer application protocols like routing protocols. In practice, this means that most CEs implement both the Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP).

Both CEs and FEs need to implement ForCES protocol. If some layer-4 transport is used to support ForCES, then both CEs and FEs need to implement the L4 transport and ForCES protocols. It is possible that all FEs inside an NE implements only one such protocol entity.

6.6. Application Layer -- Routing Protocols

Both interior routing protocols and exterior routing protocols are implemented on CEs. The routing packets originated by CEs are forwarded to FEs for delivery. The results of such protocols (like routing table update) are communicated to FEs via ForCES.

6.7. Application Layer -- Network Management Protocol

RFC1812 also dictates "Routers MUST be manageable by SNMP." (see [RFC1157] Section 8) In general, for post-association phase, most external management tasks (including SNMP) SHOULD be done through interaction with the CE in order to support the appearance of a single functional device. Therefore, it is recommended that SNMP management agent be implemented by CEs and the SNMP messages sent to FEs be redirected to their CEs. This also requires that ForCES In certain conditions (e.g. CE/FE disconnection), it may be useful to allow SNMP to be used to diagnose and repair problems. However, care should be taken when exercising such mechanisms and guidelines are provided in [FORCES-REQ], Section 5, requirement #4.

7. Summary

This document defines an architectural framework for ForCES. It identifies the relevant components for an ForCES network element, including (one or more) FEs, (one or more) CEs, FE manager (optional), and CE manager (optional). It also identifies the interaction among these components and discusses all the major reference points. It is important to point out that, among all the reference points, only the interface between CEs and FEs is within the scope of ForCES. ForCES alone may not be enough to support all different NE configurations. However, we believe ForCES is the most important element in realizing the physical separation and interoperability of CEs and FEs, and hence the first interface that ought to be standardized. Simple and useful configurations can still be implemented with only CE-FE interface being standardized, e.g., single CE with full-meshed FEs and static configuration without the need for CE/FE managers.

8. Security Considerations

The security necessary across each reference point except Fp is discussed throughout the document. In general, the physical separation of two entities usually requires much stricter security measurement in place. For example, we pointed out in Section 5.1 that authentication becomes necessary between CE manager and FE manager, between CE and CE manager, between FE and FE manager in some configuration. The physical separation of CE and FE also

imposes serious security requirement for ForCES protocol. The security requirements for reference point Fp (i.e., ForCES protocol) are discussed in detail in [FORCES-REQ] Section 8.

9. Intellectual Property Right

The authors are not aware of any intellectual property right issues pertaining to this document.

10. Normative References

[RFC2914] S. Floyd, "Congestion Control Principles", RFC2914, September 2000.

[RFC1157] J. Case, et. al., "A Simple Network Management Protocol (SNMP)", RFC1157, May 1990.

[RFC3084] K. Chan, et. al., "COPS Usage for Policy Provisioning (COPS-PR)", RFC3084, March 2001.

[RFC1812] F. Baker, "Requirements for IP Version 4 Routers", RFC1812, June 1995.

11. Informative References

[FORCES-REQ] T. Anderson, et. al., "Requirements for Separation of IP Control and Forwarding", work in progress, February 2002, <draft-ietf-forces-requirements-02.txt>.

[FORCES-APP] A. Crouch, et. al., "ForCES Applicability Statement", work in progress, February 2002, <draft-ietf-forces-applicability-00.txt>.

12. Acknowledgments

Joel M. Halpern gave us many insightful comments and suggestions and pointed out several major issues. Many of our colleagues and people in the ForCES mailing list also provided valuable feedback.

13. Authors' Addresses

Lily L. Yang
Intel Labs
2111 NE 25th Avenue
Hillsboro, OR 97124 USA
Phone: +1 503 264 8813
Email: lily.l.yang@intel.com

Ram Dantu
Netrake Corporation
3000 Technology Drive
Plano, Texas 75074
Phone: +1 214 291 1111

Email: ramd@netrake.com

Todd A. Anderson

Intel Labs
2111 NE 25th Avenue
Hillsboro, OR 97124 USA
Phone: +1 503 712 1760
Email: todd.a.anderson@intel.com