Network Working Group
Request for Comments: 3654
Category: Informational

H. Khosravi, Ed.
T. Anderson, Ed.
Intel

November 2003

Requirements for Separation of IP Control and Forwarding

Status of this Memo

Copyright Notice

Abstract

This document introduces the Forwarding and Control Element Separation (ForCES) architecture and defines a set of associated terminology. This document also defines a set of architectural, modeling, and protocol requirements to logically separate the control and data forwarding planes of an IP (IPv4, IPv6, etc.) networking device.

Table of Contents

1. Introduction

An IP network element is composed of numerous logically separate entities that cooperate to provide a given functionality (such as a routing or IP switching) and yet appear as a normal integrated network element to external entities. Two primary types of network element components exist: control-plane components and forwarding-plane components. In general, forwarding-plane components are ASIC, network-processor, or general-purpose processor-based devices that handle all data path operations. Conversely, control-plane components are typically based on general-purpose processors that provide control functionality such as the processing of routing or signaling protocols. A standard set of mechanisms for connecting these components provides increased scalability and allows the control and forwarding planes to evolve independently, thus promoting faster innovation.

For the purpose of illustration, let us consider the architecture of a router to illustrate the concept of separate control and forwarding planes. The architecture of a router is composed of two main parts. These components, while inter-related, perform functions that are largely independent of each other. At the bottom is the forwarding path that operates in the data-forwarding plane and is responsible for per-packet processing and forwarding. Above the forwarding plane is the network operating system that is responsible for operations in the control plane. In the case of a router or switch, the network operating system runs routing, signaling and control protocols (e.g., RIP, OSPF and RSVP) and dictates the forwarding behavior by manipulating forwarding tables, per-flow QoS tables and access control lists. Typically, the architecture of these devices combines all of this functionality into a single functional whole with respect to external entities.

2. Definitions

Addressable Entity (AE) - A physical device that is directly addressable given some interconnect technology. For example, on IP networks, it is a device to which we can communicate using an IP address; and on a switch fabric, it is a device to which we can communicate using a switch fabric port number.

Physical Forwarding Element (PFE) - An AE that includes hardware used to provide per-packet processing and handling. This hardware may consist of (but is not limited to) network processors, ASIC's, line cards with multiple chips or stand alone box with general-purpose processors.

Physical Control Element (PCE) - An AE that includes hardware used to provide control functionality. This hardware typically includes a general-purpose processor.

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/controlled by a CE via the ForCES protocol. FEs may happen to be a single blade(or PFE), a partition of a PFE or multiple PFEs.

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. CEs may consist of PCE partitions or whole PCEs.

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

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 to use. However, this pre-association phase protocol is currently out of scope. 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. Again, this pre-association phase protocol is currently out of scope. 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) what is used within the ForCES protocol (see Section 6, requirement #1). However, the two capability discovery mechanisms may utilize the same FE model (see Section 5). Pre-association phase protocols are not discussed further in this document.

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.

ForCES Protocol Element - A FE or CE.

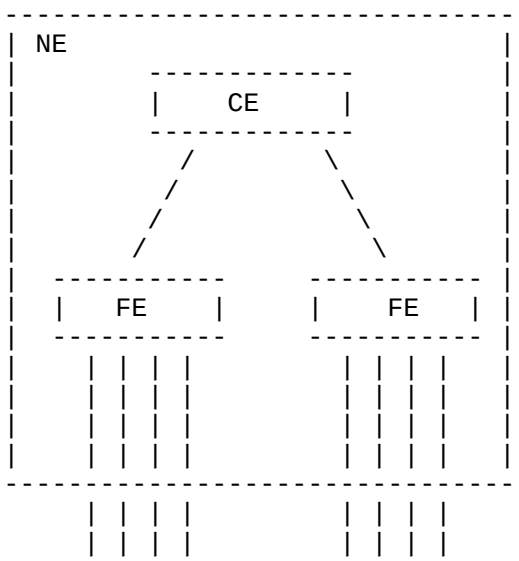
High Touch Capability - This term will be used to apply to the capabilities found in some forwarders to take action on the contents or headers of a packet based on content other than what is found in the IP header. Examples of these capabilities include NAT-PT, firewall, and L7 content recognition.

3. Architecture

The chief components of a NE architecture are the CE, the FE, and the interconnect protocol. The CE is responsible for operations such as signaling and control protocol processing and the implementation of management protocols. Based on the information acquired through control processing, the CE(s) dictates the packet-forwarding behavior of the FE(s) via the interconnect protocol. For example, the CE might control a FE by manipulating its forwarding tables, the state of its interfaces, or by adding or removing a NAT binding.

The FE operates in the forwarding plane and is responsible for per-packet processing and handling. By allowing the control and forwarding planes to evolve independently, different types of FEs can be developed - some general purpose and others more specialized. Some functions that FEs could perform include layer 3 forwarding, metering, shaping, firewall, NAT, encapsulation (e.g., tunneling), decapsulation, encryption, accounting, etc. Nearly all combinations of these functions may be present in practical FEs.

Below is a diagram illustrating an example NE composed of a CE and two FEs. Both FEs and CE require minimal configuration as part of the pre-configuration process and this may be done by FE Manager and CE Manager respectively. Apart from this, there is no defined role for FE Manager and CE Manager. These components are out of scope of the architecture and requirements for the ForCES protocol, which only involves CEs and FEs.



4. Architectural Requirements

The following are the architectural requirements:

1) CEs and FEs MUST be able to connect by a variety of interconnect technologies. Examples of interconnect technologies used in current architectures include Ethernet, bus backplanes, and ATM (cell) fabrics. FEs MAY be connected to each other via a different technology than that used for CE/FE communication.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.