An Information Model for Service Chaining Policy
draft-hares-dunbar-i2rs-sfc-policy-im-01

Abstract

   This draft describes an I2RS information model for managing the
   service chain steering policy rules to a router via the I2RS
   interface (SFC-Policy IM).

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

   This draft describes an I2RS information model for managing the
   Service Chain via the I2RS interface.

2. Definition of terms

   NFV: Network Function Virtualization

      [NFV-Terminology].

SF: Service Function

   [I-D.ietf-sfc-problem-statement].

SFF: Service Function Forwarder

Service Chain

   [I-D.bitar-i2rs-service-chaining] defines a service chain as an
   ordered set of services applied to a packet of flow. An example
   of this is a sequence of service function such as Chain#1 {s1, s4,
   s6} or Chain#2{s4, s7} at functional level. Also see the
   definition of Service Function Chain in
   [I-D.bitar-i2rs-service-chaining]

Service Chain Instance Path

   The actual Service Function Instance Components selected for a
   service chain.

SFFN: Service Function Forwarder Node

   [I-D.bitar-i2rs-service-chaining]states service function can run:
   a) natively within a router (or routing system), b) on a virtual
   machine on a server or service engine, or in a dedicated
   standalone hardware appliance.

VNF: Virtualized Network Function

   [NFV-Terminology]

STOPO

   Service topology is a topology of Service nodes (SFF).

SFFaddr: Service Node Address

   [I-D.ietf-sfc-problem-statement] states this address should be IP
   Address, or tuple of (SFFaddr, host system IP address) or tuple of
   (host system IP address, system internal ID for service engine).

Service Type

   [I-D.ietf-sfc-problem-statement].

3. Service Chaining Background

```
                                |1  -----    |n          |21   ----  |2m
             +---+---+    +---+---+    +-+---+    +--+-----+
             | SF#1  |    |SF#n   |    |SF#i1|    |SF#im   |
             |       |    |       |    |     |    |        |
             +--+---+     +--+---+     +-+--+     +--+-+--+
                :            :           :          :  :
                :            :           :          :  :
                 \          /             \         /
      +-------------+   +--------+              +---------+
 -- >| Chain        |   | SFF    |   ------     | SFF     | --->
     |classifier    |   |Node-1  |              | Node-i  |
     +-------------+    +----+---+              +----+--+-+
            \              |                        /
             \            | SFC Encapsulation  /
              \           |                   /
 ,. .............................................._
 ,-'                                            `-.
 /                                                `.
 |                    Network                      |
 `.                                               /
 `.__.............................. _,-'
```
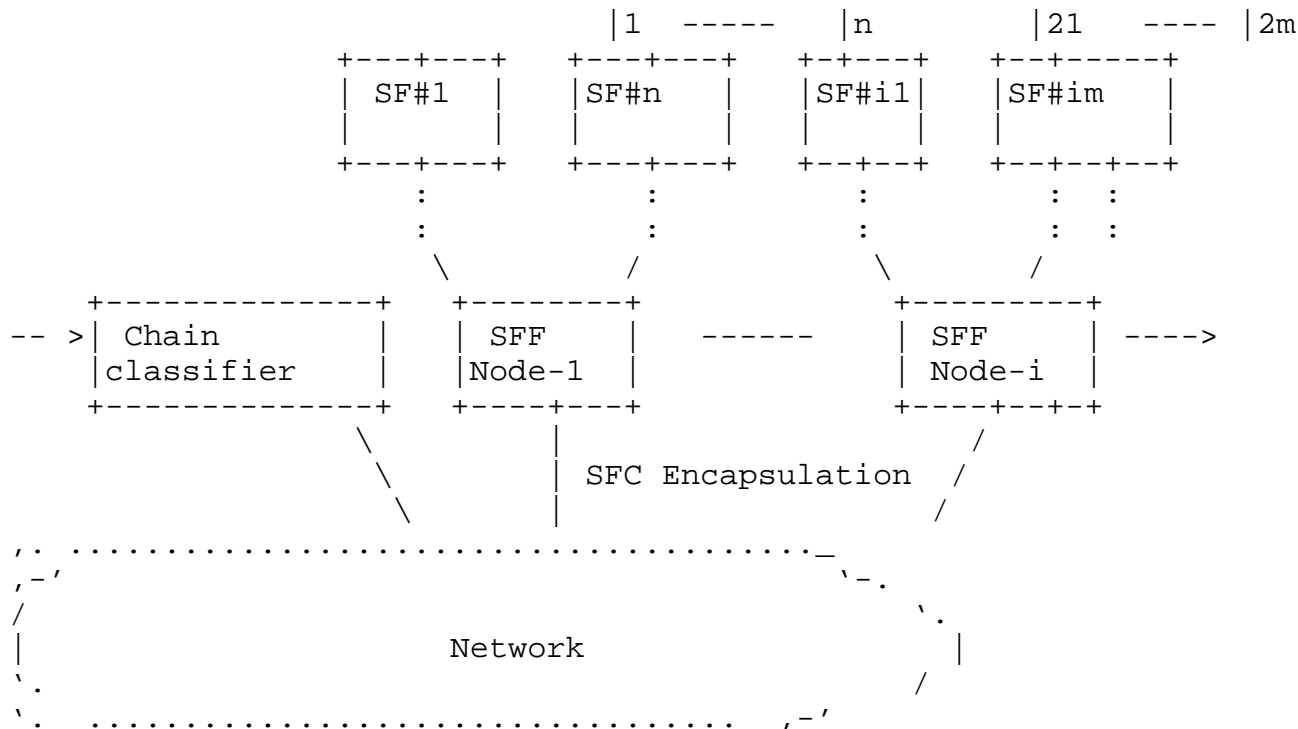
        Figure 1   Framework of Service Chain

4. Overview of information model for Service Chain

   There are three major categories of information models for Service
   Chain management:

      1) Service function instances discovery;

      2) Traffic flow steering rules on a router for specific service
      chain.

      3) Service Chain path computatino

   This document focuses on the first (service function instance
   discovery), and the second (the traffic flow steering rules) as
   expressed in I2RS policies. The third category, the Service Chain
   path computation, is out of scope for this document. An I2RS
   information model for Service Topology with its Traffic Engineering
   Databased (TED) and associated inventory can be found in
   [I-D.hares-i2rs-info-model-service-topo]. Additional I2RS modes on
   basic network policy (BNP IM) and Policy based Routing (PBR IM) is
   contained in [I-D.hares-i2rs-info-model-policy].

5. Requirements for Service Function Forwarder Node (SFFN) Resources
    SFC Flow Filtering

    This section reviews the requirements for SFC Flow Filtering Policies
    for an existing service for SFFNs within the SFC domain.

    Inherent in the [I-D.ietf-sfc-problem-statement] is the need for
    policies that establish and filter data flow on the Service chain
    pathways. This document defines an I2RS model to interface to the
    SFC's Service Function Forwarder (SFF) to install or change the the
    Policy controlling data flow to their designated and service
    functions and subsequent SFFN.

    The SFC use case [I-D.bitar-i2rs-service-chaining] suggests SFF
    resources that must be on each SFF Node (SFFN). The SFFN resources
    include the following elements that the I2RS Client-I2RS Agent
    protocol can utilize:

    SFC-Use-REQ01:Address (R)

        has the following address requirements:

        * IP address

        * service-node tuple (service node IP address, Host system
           address)

        * host-node tuple (hosting system IP-address, system internal
           identifier)

    SFC-Use-REQ02:Supported Service Types (R/W) SHOULD include:

        NAT, IP Firewall, Load balancer, DPI, and others. Note that the
        current SFC WG suggest that the SFF does not need to know the SFFN
        type to steer the data to their designated service function.
        Therefore, this parameter has been given "a forwarding" function
        value.

    SFC-Use-REQ03:Virtual contexts (R/W)SHOULD include:

        * Maximum Number of virtual contexts supported

        * Current number of virtual contexts in use

        * Number of virtual contexts available

   * Supported Context (VRF)

   SFC-Use-REQ04: Customers currently on node (R)

   SFC-Use-REQ05: Customer Support Table (per customer ID) (R)

      with the following contents per entry:

      * Customer-id

      * List of supported Virtual Contexts

   SFC-Use-REQ06: Service Resource Table (R/W)

      which includes:

      * index: Comprised of service node, virtual context, service type

      * service bandwidth capacity

      * supported packet rate (packets/second)

      * supported bandwidth (kps)

      * IP Forwarding support: specified as routing-instance(s), RIBs,
         Address-families supported

      * Maximum RIB-size

      * Maximum Forward Data Base size

      * Maximum Number of 64 bit statistics counters for policy
         accounting

      * Maximum number of supported flows for services

   SFC-Use-REQ07: Virtual Network Topology (VNT) (R)

      which includes:

      * number of access points to which service topology applies

      * topology of access points

6. Service Forwarder Node RBNF

```
        <SFF_node> ::= <SFFN_address>   /*SFC-Use-REQ01 */
                [<Attached_Service_node>]   /*SFC-Use-REQ02 */
                [<SFFN_virtual_contexts>]   /*SFC-Use-REQ03 */
                [<SFFN_customer_cnt>]    /*SFC-Use-REQ04 */
                [<SFFN_Customer_support_table>] /*SFC-Use-REQ05 */
                [<SFFN_Service_Resource_table>] /*SFC-Use-REQ06 */
                [<SFFN_VNTopo>]             /*SFC-Use-07*/

        <SFFN_address> ::== <ip_address>

        <Attached_Service_node> ::=
                | [ (<service-node-ip_address>
                     <host-system-ip_address>)]
                | [ (<hosting-system-ip_address>
                     <system-internal_ID>)]

           <service-node-ip_address> ::= <ip_address>
           <host-system-ip_address> ::= <ip_address>
           <hosting-system-ip_address> ::= <ip_address>
           <system-internal_ID> ::= INTEGER-64;

        /* SFC-Use-02 */
        <SFFN_supported_types> ::= <Attached_Service_node_types>



        /* These are the types specified by the SFC-REQ-02]
            <SF_Types> ::= [<SF_TYPE_FW>]
                                [<SF_TYPE_LB>]
                                [<SF_TYPE_DPI>]
                                [<SF_TYPE_NAT>]
        /* SFC-Use-03 */                    ...
        <SFFN_virtual_contexts> ::== <VContext_max>
                    <VContext_current_inuse>
                    <VContext_current_avail>
                    <SFFN_Types>

        /*SFC-Use-04 */
        <SFFN_customer_cur_cnt> ::= INTEGER;

        /* SFC-Use-05: Customer Support Table per Customer ID */
        <SFFN_customer_table> ::= [<SFFN_customer< ...]

        <SFFN_customer> ::= <SFFN_customer_Name>
                        <SFFN_customer_ID>
                        <SFFN_customers_contexts>
```

```
        <SFFN_customers_contexts> ::= <SFFN_Types>

        /*SFC-Use-REQ06 */
        <SFFN_Service_Resource_table> ::=  <SFF_Service_resource_index>
                    <SFFN-SR_service_BW_capacity>
                    <SFFN-SR_packet_rate_max>
                    <SFFN-SR_BW>
                    <SFFN-SR_IP_fwd_instance_list>
                    <SFFN-SR_MAX_RIB>
                    <SFFN-SR_MAX_FIB>
                    <SFFN-SR_MAX_COUNTER64>
                    <SFFN-SR_MAX_Flows>

            <SFF_Service_resource_index> := <SFFN_Address>
                    <VContext_ID>
                    <Service_types>

        /*SFC-Use-REQ07
         * SFC_topology is defined by
         * ietf-hares-i2rs-service-topology
         * which includes node code
         */
        <SFF_VNT> ::= <SFC_Topology>
```

7. Information Model for Service Chain Function Instance Discovery

   A Service function instance can be either attached to a router via a
   physical interface or instantiated on a virtual machine that is
   attached to a router. Following are our assumptions:

      1) The Service Chain Manager can get all the IP addresses or IP
      prefix of the service function instances needed from a database or
      provisioning system, and passed to the relevant routers. The
      Routers can send ARP/ND broadcast/multicast messages to all the
      attached nodes.

      2) Service function instances will respond to ARP (IPv4)/ND (IPv6)
      requests from its L2/L3 boundary router.

```
                    Service Chain Manager/Controller
                                  ^   |
                                  |   |  A: Set filter for
              B:                  |   |     the interested service
                Router reports the|   |     function instances
             Directly attached    |   |
             Service Function      |   |
             Instances             |   V
                    +------+---+------------+
                    |         Router        |
                    ++-----+----------+------+
                     /      |          |      \
                    /       |          |       \
              +-+-+      +-+-+      +-+-+      +-+-+
              |   |...|  |   |      |   |  ...|   |  |
              +---+      +---+      +---+      +---+ Server racks
              |   |...|  |   |      |   |  ...|   |  | for hosting
              +---+      +---+      +---+      +---+ Service
              |   |...|  |   |      |   |  ...|   |  | Function
              +---+      +---+      +---+      +---+ Instances
```
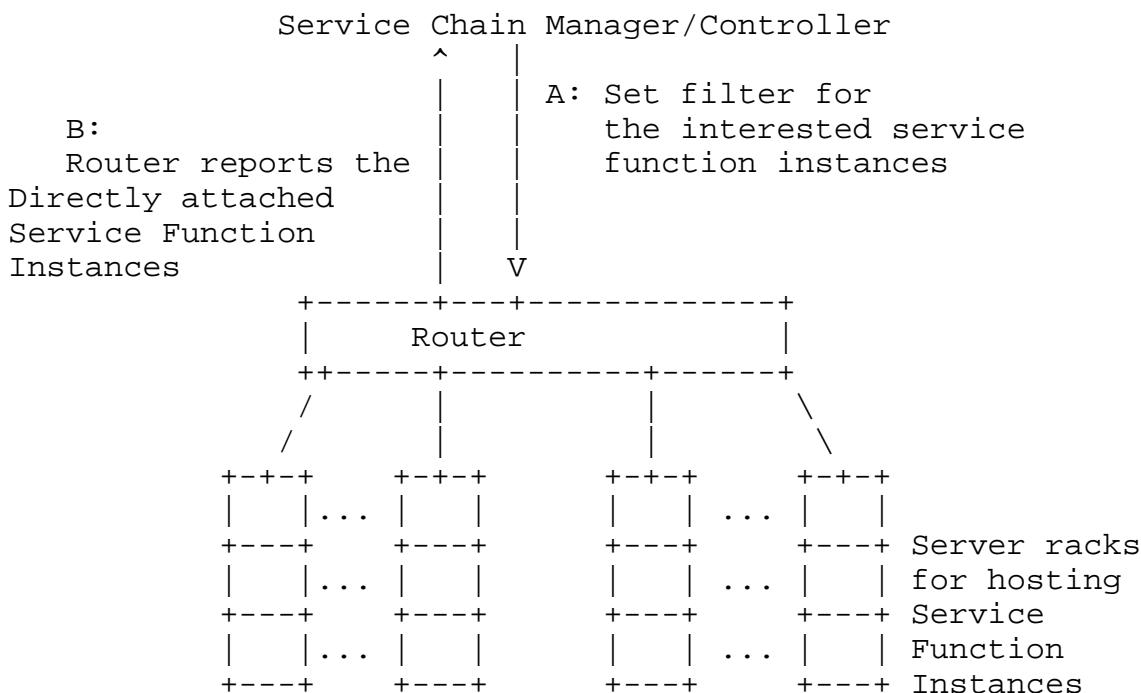
Figure 1: Service Function Instances

8. Information Model for Interested Service Function Instances

   Service Function Instances placement can be managed by entities that
   are not integrated with Service Chain Manager. Therefore, it is
   necessary for the Service Chain Manager to discover all the Service
   Function Instances that might be needed for a specific service chain.
   Service Chain Manager can send down the filter periodically or on-
   demand (i.e. when there is a request for building a specific service
   chain for a client).

   Some service function instances are attached to router via tunnels,
   e.g. VxLAN. Service Function Instances might be partitioned by
   clients, which are differentiated by different network ID (e.g.
   VNID, VPN ID, etc). Some filter will carry the network ID (tenant
   ID, or VPN ID) to get specific service functions.

   The I2RS Client can operate as the service chain manager/controller
   communicating with the I2RS Agents operating in the router or I2RS
   Agents operating on the service function instances in the server
   racks to discover and control specific service function instances.

   The I2RS Client-Agent must be able to discover the I2RS Agent
   associated with a specific Service Function instance by querying for:
   SFFN Address,SFFN type, or SFFN virtual context or SFFN Customer;

9. SFFN Instances Addresses

        <interested-SF-filter> ::= <SF-FILTER-NAME>
                         [<ipv4-address-list>|<ipv6-address-list>]
                         [<client-identifier>]

        <ipv4-address-list> ::= ((<ipv4-address>
                          |<ipv4-prefix>) ...)
        <ipv4-prefix> ::= <IPV4_ADDRESS><IPV4_PREFIX_LENGTH>
        <ipv6-address-list> = ((<ipv6-address>
                          |<ipv6-prefix>) ...)
        <ipv6-prefix> ::= <IPV6_ADDRESS><IPV6_PREFIX_LENGTH>

        <client-identifier> ::= <client-identifier-type>
                         <client-identifier >
        <client-identifier-type> ::= <GRE>
                                | <VxLAN>
                                | <NVGRE>

        <client-identifier > ::= (<VXLAN> <VXLAN_IDENTIFIER>)
                             | (<NVGRE> <VIRTUAL_SUBNET_ID>)
                             | (<GRE> <GRE_KEY>)

10. Information Model for Reporting Directly Attached Instances

   When a router receives the filter of the interested Service Function
   Instances, it can scan through all its interfaces to check if any of
   the addresses in the filter list are attached to the interfaces. For
   the Service Function Instances attached via Layer 2, the router can
   send ARP/ND to get the matching instances to respond. For the
   Service Function Instances attached via Layer 3, the router can use
   Ping to check if the addresses in the filter are attached.

   The response should be grouped by <SF-FILTER-NAME >

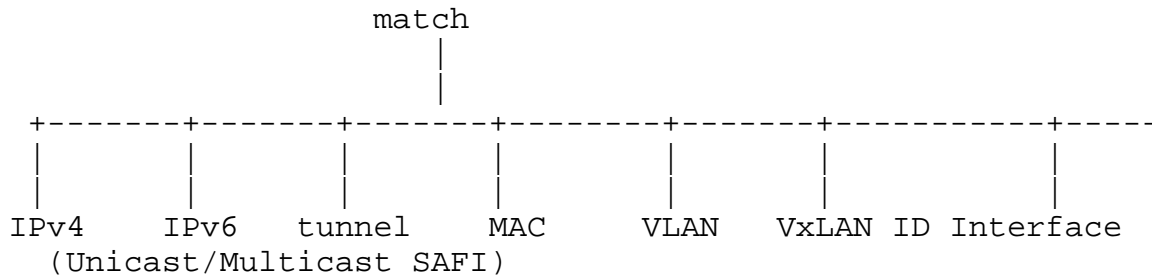11. RBNF for Reporting Directly Attached Instances

        <sf-instance-list> ::= <INSTANCE-LIST-NAME>
                 < SF-FILTER-NAME>
                 [<INTERFACE_IDENTIFIER>
                 |<ipv4-address-list>
                 |<ipv6-address-list>]]

12. Information Model for Traffic steering rules

   The semantics of traffic steering rules is "Match" and "Action",
   similar to the "route" described in [I-D.ietf-i2rs-rib-info-model].
   However, there are more matching criteria for traffic steering rules.

```
                          match
                            |
                            |
        +-------+-------+-------+--------+-------+-----------+-----
        |       |       |       |        |       |           |
        |       |       |       |        |       |           |
       IPv4    IPv6   tunnel    MAC      VLAN    VxLAN ID  Interface
        (Unicast/Multicast SAFI)
```

   The steering rules include matches on combinations of:

   o Addresses: IP addresses (IPv4/IPv6), Multicast IP addresses, MAC
      Addresses;

   o Label fields: MPLS labels, VLAN-IDs, GRE-Keys

   o interfaces

   o Layer 4 fields

   o packet sizes

13. Traffic Steering Rules RBNF

```
        <SFC-Policy> ::= <steering-rules>

        <steering-rules> ::= <match> <action>
          <match> ::= <address>
                      |<label>
                      |<interface>
                      |<l4-field>
                      |<packet-size>
          <address> ::= <ip-address>
                      |<multicast-address>
                      |<mac-address>]
          <ip-address> ::= <IPV4_ADDRESS>
                      |<ipv4-prefix >
                      |<IPV6_ADDRESS>
                      |<ipv6-prefix >]
          <label> ::= [MPLS_LABEL] | <VXLAN-ID> | <GRE-KEY>
          <l4-field>::= <TCP_PORT> | <UDP-PORT>

          <action> ::= <nexthop-list>
                      |<drop-policy>
                      |<metadata>
          <drop-policy> ::= <PASS>|<DROP>
          <metadata> ::= [<ATTACH> <object>] | <detach>
```

     The I2RS interface of a router should allow "write" and "read" of ab
ove objects.

14. Security Considerations

   The SC use cases described in this document assumes use of I2RS
   programmatic interfaces described in the I2RS framework mentioned in
   [I-D.ietf-i2rs-architecture]. This document does not change the
   underlying security issues inherent in the existing in
   [I-D.ietf-i2rs-architecture].

15. IANA Considerations

   This draft includes no request to IANA.

16. Acknowledgements

   We'd like to thank Qin Wu for his comments on this document relating
   to the service topologies.

17. References

17.1. Normative References

   [I-D.ietf-i2rs-architecture]
              Atlas, A., Halpern, J., Hares, S., Ward, D., and T.
              Nadeau, "An Architecture for the Interface to the Routing
              System", draft-ietf-i2rs-architecture-04 (work in
              progress), June 2014.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119, March 1997.

17.2. Informative References

   [I-D.bitar-i2rs-service-chaining]
              Bitar, N., Heron, G., Fang, L., ramki, r., Leymann, N.,
              Shah, H., and W. Haddad, "Interface to the Routing System
              (I2RS) for Service Chaining: Use Cases and Requirements",
              draft-bitar-i2rs-service-chaining-01 (work in progress),
              February 2014.

   [I-D.hares-i2rs-info-model-policy]
              Hares, S. and W. Wu, "An Information Model for Network
              policy", draft-hares-i2rs-info-model-policy-02 (work in
              progress), March 2014.

   [I-D.hares-i2rs-info-model-service-topo]
              Hares, S., Wu, W., and X. Guan, "An Information model for
              service topology", draft-hares-i2rs-info-model-service-
              topo-00 (work in progress), February 2014.

   [I-D.ietf-i2rs-rib-info-model]
              Bahadur, N., Folkes, R., Kini, S., and J. Medved, "Routing
              Information Base Info Model", draft-ietf-i2rs-rib-info-
              model-03 (work in progress), May 2014.

   [I-D.ietf-sfc-problem-statement]
              Quinn, P. and T. Nadeau, "Service Function Chaining
              Problem Statement", draft-ietf-sfc-problem-statement-07
              (work in progress), June 2014.

   [I-D.medved-i2rs-topology-requirements]
              Medved, J., Previdi, S., Gredler, H., Nadeau, T., and S.
              Amante, "Topology API Requirements", draft-medved-i2rs-
              topology-requirements-00 (work in progress), February
              2013.

   [I-D.white-i2rs-use-case]
             White, R., Hares, S., and A. Retana, "Protocol Independent
             Use Cases for an Interface to the Routing System", draft-
             white-i2rs-use-case-05 (work in progress), June 2014.

   [NFV-Terminology]
             "Network Functions Virtualization (NFV); Terminology for
             Main Concepts in NFV",
             <http://www.etsi.org/deliver/etsi_gs/
             NFV/001_099/003/01.01.01_60/gs_NFV003v010101p.pdf>.

Authors' Addresses

   Susan Hares
   Huawei
   7453 Hickory Hill
   Saline, MI 48176
   USA

   Email: shares@ndzh.com

   Linda Dunbar
   Huawei
   6340 Legacy Drive, Suite 175
   Plano, TX 75024
   USA

   Phone: +1-469-277-5840
   Email: ldunbar@huawei.com