Application-aware Networking (APN)

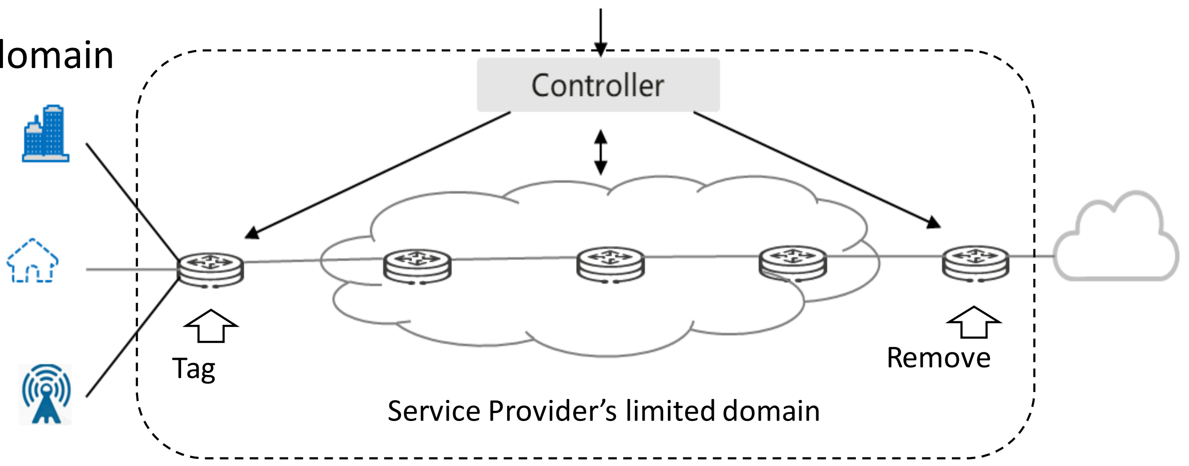
Shuping Peng/Zhenbin Li

Purpose of presenting in this WG

- Our main purpose to present in the Security Area is to collect feedback and suggestions on the security issues which are always challenged by people on the APN work.
- We have studied the potential security and privacy issues that APN might impose. Please find this relevant draft, https://tools.ietf.org/html/draft-peng-apn-security-privacy-consideration-00.
- We have had some discussions in the APN mailing list (https://datatracker.ietf.org/wg/apn/about/). We would like to have more feedback and suggestions from the security experts, to further address the main concerns that were raised by the IESG.
- We would like to know whether using the APN attribute within the controlled operator’s network domain will cause any security issues. If yes, how can we mitigate these issues?

What is APN (Application-aware Networking)?

- APN is focused on developing a framework and set of mechanisms to derive, convey and use an attribute information to allow for the implementation of fine-grain user (group)-, application (group)-, and service-level requirements at the network layer.
- Such information is acquired, constructed, and encapsulated in the packets.
- Such information is treated as an object in the network
  - To it, the network operator applies policies in various nodes/service functions along the path and provides corresponding services.
- APN works within a limited trusted domain.
  - Typically, an APN domain is defined as a service provider's limited domain in which MPLS, VXLAN, SR/SRv6 and other tunnel technologies are adopted to provide services.
  - APN attribute is tagged/removed at the edge of the limited domain



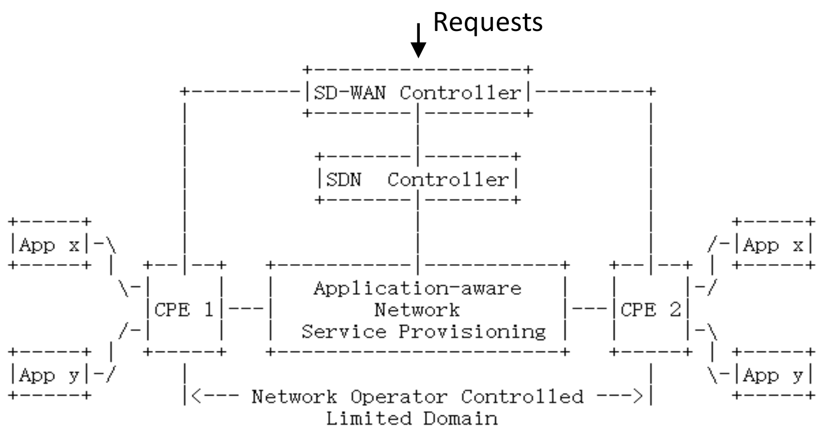
APN: Application-aware Networking

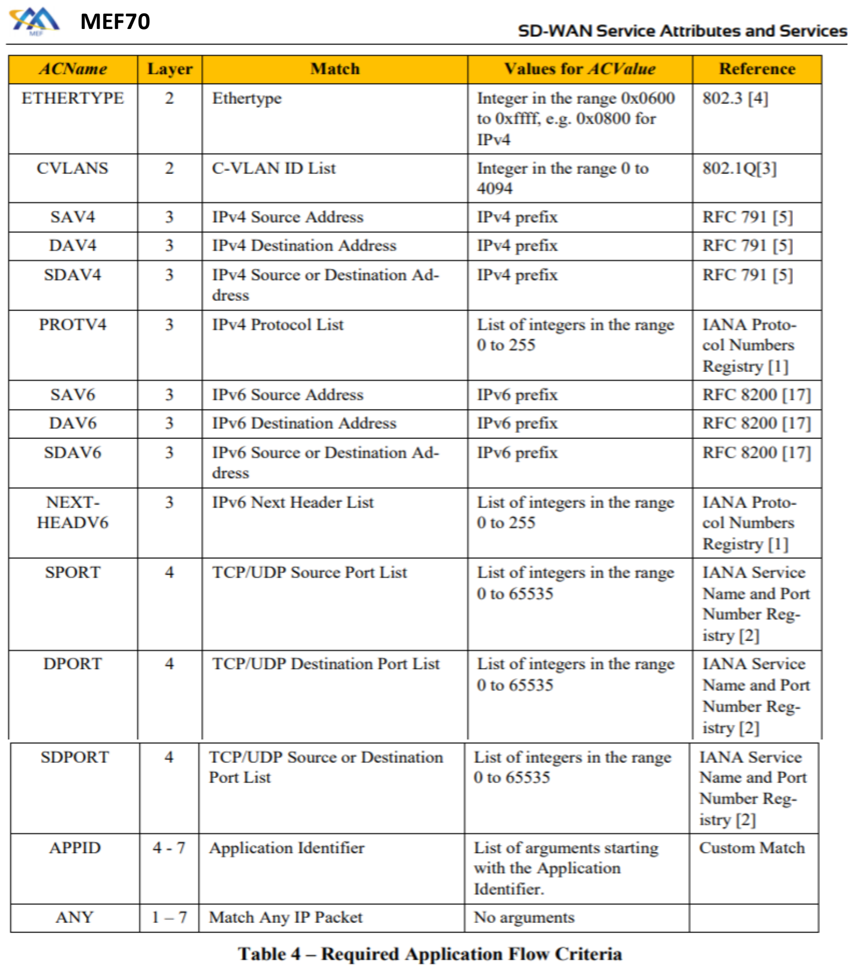
What APN is Not

- APN is not about identifying the application to or within the network
  - The network does not need to know which applications are sending traffic
  - Telling the network which applications are running would break privacy
- So, APN is about telling the network what policies to apply to traffic
  - An application can apply multiple policies to different traffic flows
  - Multiple applications can ask for the same policies
- APN is not PANRG/SPUD/PLUS/Network Tokens
- Conveying the information via the transport/application layer and the network layer are different technologies. APN uses the network layer. – APN Side Meeting @IETF108

Why APN? - The SD-WAN use case

- In the case of SD-WAN, network operators can provide SLA-guaranteed WAN lines to help enterprises to access the clouds.
- When mapping the WAN line into the operator's network, there are usually multiple network paths with different SLA guarantees available between the two endpoints of the tunnel connecting the CPE nodes.
- In MEF70, a list of Match items are specified at the CPE nodes to steer the traffic into corresponding WAN lines across the network operator's network according to users'/applications' requirements.
    - E.g, 5-tuples (i.e. Source/Destination Address, Source/Destination Port, Protocol)
- There is a need to communicate user/application requirements to match to the capabilities of the WAN lines – different from MEF70
- Once the traffic goes into the operator's network, there is a need to apply various policies in different nodes along the network path onto the traffic flow, e.g.,
    - at the headend to steer into corresponding path satisfying SLAs
    - at the midpoint to collect corresponding performance measurement data
    - at the service function to execute particular policies


Requests
SD-WAN Controller
SDN Controller
App x
App y
CPE 1
Application-aware Network Service Provisioning
CPE 2
App x
App y
<--- Network Operator Controlled --->
Limited Domain


MEF70 SD-WAN Service Attributes and Services

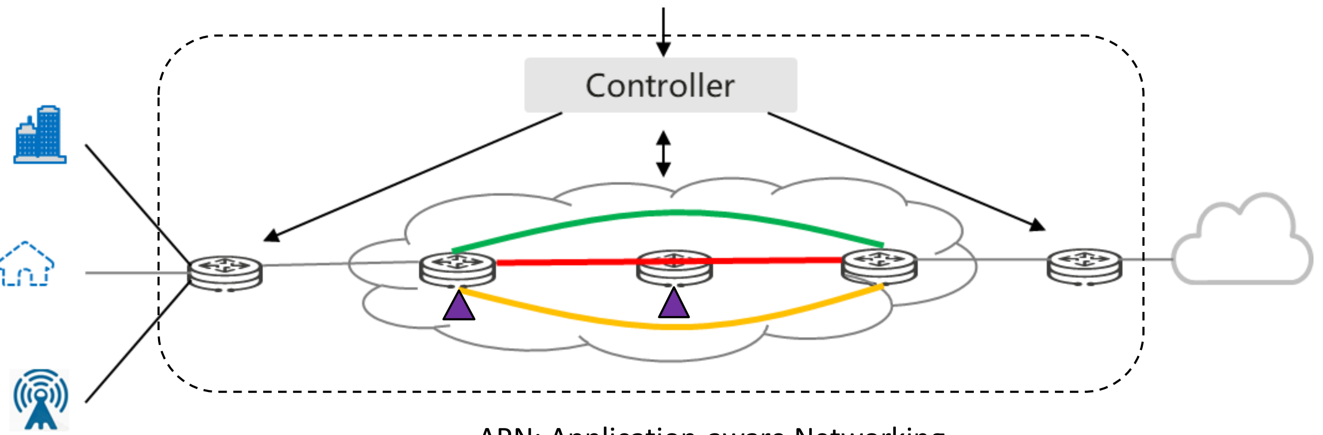
ACName	Layer	Match	Values for *ACValue*	Reference
ETHERTYPE	2	Ethertype	Integer in the range 0x0600 to 0xffff, e.g. 0x0800 for IPv4	802.3 [4]
CVLANS	2	C-VLAN ID List	Integer in the range 0 to 4094	802.1Q[3]
SAV4	3	IPv4 Source Address	IPv4 prefix	RFC 791 [5]
DAV4	3	IPv4 Destination Address	IPv4 prefix	RFC 791 [5]
SDAV4	3	IPv4 Source or Destination Ad-dress	IPv4 prefix	RFC 791 [5]
PROTV4	3	IPv4 Protocol List	List of integers in the range 0 to 255	IANA Proto-col Numbers Registry [1]
SAV6	3	IPv6 Source Address	IPv6 prefix	RFC 8200 [17]
DAV6	3	IPv6 Destination Address	IPv6 prefix	RFC 8200 [17]
SDAV6	3	IPv6 Source or Destination Ad-dress	IPv6 prefix	RFC 8200 [17]
NEXT-HEADV6	3	IPv6 Next Header List	List of integers in the range 0 to 255	IANA Proto-col Numbers Registry [1]
SPORT	4	TCP/UDP Source Port List	List of integers in the range 0 to 65535	IANA Service Name and Port Number Reg-istry [2]
DPORT	4	TCP/UDP Destination Port List	List of integers in the range 0 to 65535	IANA Service Name and Port Number Reg-istry [2]
SDPORT	4	TCP/UDP Source or Destination Port List	List of integers in the range 0 to 65535	IANA Service Name and Port Number Reg-istry [2]
APPID	4 - 7	Application Identifier	List of arguments starting with the Application Identifier.	Custom Match
ANY	1 – 7	Match Any IP Packet	No arguments	

Table 4 – Required Application Flow Criteria

APN: Application-aware Networking

Why APN？ - The issues

- There is currently no way to request policies to be applied to all packets in a traffic flow on various nodes along the network path.
- It may be possible to stack those various policies in a list of TLVs in the header of each packet.
  - This approach would introduce great complexities, damage MTU, and impose big challenges on the hardware processing and forwarding.
- When doing the policy-based routing along the network path, normally ACL via 5 tuples is used, but it is complicated to resolve.
  - With tunnel encapsulation, it is hard to resolve the 5 tuples since the transport layer information is down so deep.
  - With IPSec, it becomes impossible to obtain any transport layer information.
  - In the IPv6 data plane, with the extension headers being added before the upper layer, in some implementations it becomes very difficult and even impossible to obtain transport layer information because that information is so deep in the packet. So there is no 5 tuples anymore, and maybe only 2 tuples are available.



APN: Application-aware Networking

How to solve the issues? – Possible solutions and benefits

- Acquire and construct an attribute at the network edge, and encapsulate it in the packets.
- Such information is treated as an object in the network
  - Network operator applies policies and provides services in various nodes/service functions along the path depending on the information
  - Policy choice is effectively according to user/application group and/or service-level requirement
- Such information will also bring benefits, for example,
  - Improve the forwarding performance since it will only use 1 field in the IP layer instead of resolving 5 tuples, which may also improve the scalability.
  - Very flexible policy enforcement in various nodes and service functions along the network path.
- Furthermore, with such information, more new services could be enabled, for example,
  - The policy execution on the service function can be based only on this value and not based on 5-tuple, which can eliminate the overhead involved by ACL
  - Even more fine-granularity performance measurement could be achieved and the granularity to be monitored and visualized can be controllable, which is able to relieve the processing pressure on the controller when it is facing the massive monitoring data
  - The underlay performance guarantee could be achieved for SD-WAN overlay services, such as explicit traffic engineering path satisfying SLA and selective visualized accurate performance measurement.
- This can be easily done by utilizing this information, which is not possible with any of the current existing mechanisms.

How to solve the issues? – Gap Analysis

- Some mechanisms have been specified in IETF using attribute/identifier to perform traffic steering and service provisioning.

1. DSCP in the IPv4 and IPv6 Headers [RFC2474]
   - The field is not big enough.
2. IPv6 Flow Label [RFC6437] /MPLS Entropy Label [RFC6790]
   - The IPv6 flow label is mainly used for Equal Cost Multipath Routing (ECMP) and Link Aggregation [RFC6438].
   - [RFC6391] adds the Label Stack Entry (LSE) to facilitate the load balancing of the flows within a pseudowire (PW) over the available ECMPs.
3. SFC ServiceID [I-D.ietf-sfc-serviceid-header]
   - Subscriber Identifier and Performance Policy Identifier are carried in the Network Service Header (NSH) [RFC8300] Context Header.
   - The APN attribute can be carried in various data plane encapsulations.
   - The APN attribute is treated as an object in the network, to which the network operator applies policies in various nodes/service functions along the path and provide corresponding services.
4. IOAM Flow ID [I-D.ietf-ippm-ioam-direct-export]
   - Flow ID is used to correlate the exported data of the same flow from multiple nodes and from multiple packets.
   - The APN identifier can serve more various purposes.
5. Binding SID [RFC8402]
   - BSID is bound to an SR Policy, instantiation of which may involve a list of SIDs.
   - The APN identifier is not bound to SRv6 only, and it can be carried in various data plane encapsulations.
6. FlowSpec Label [RFC5575], [I-D.ietf-idr-flowspec-mpls-match], [I-D.ietf-idr-bgp-flowspec-label], [I-D.liang-idr-bgp-flowspec-route]
   - In BGP VPN/MPLS networks, BGP FlowSpec can be extended to identify and change (push/swap/pop) the label(s) for traffic that matches a particular FlowSpec rule. BGP is used to distribute the FlowSpec rule bound with label(s).
   - APN identifier is not bound to MPLS only, and it can be carried in various data plane encapsulations.

APN: Application-aware Networking

How to solve the issues? – Gap Analysis

- The existing solutions are specific to a particular scenario or data plane, not the same as APN and unable to achieve the same effects.
- As driven by ever-emerging new 5G services, fine-granularity service provisioning becomes urgent.
- APN aims to define a generalized attribute used for service provisioning, and can be carried in various data plane encapsulations.

Frequently Asked Questions

- **1. Are there any applications that can benefit from APN?**
  I would like to ask how many of us have experienced the Meetecho issues in this IETF and the virtual ones before. This is one of the applications that can benefit from APN. :)

  In this IETF, we had a hackathon "Application-aware G-SRv6 networking", which shows the improvements that can be achieved with APN, using which the traffic is steered into the SRv6 path [1]. The results are going to be presented and demonstrated in the INFOCOM2021.
  There was also the first demonstration of APN in the INFOCOM2020 [2].

  [1] https://github.com/APN-Community/IETF-110-Hackathon-Demo/blob/master/Application-aware_G-SRv6_networking__Demo_and_Test.pdf
  [2] https://ieeexplore.ieee.org/abstract/document/9162934

  2. How APN can help resolve the QoE issues?
  As shown in the demonstrations, we can see the differences/improvements which APN can make in the network. QoE is complex and has many impacting factors including both the terminal, access and the network. APN aims to provide ways to improve the QoE within the network.

  3. Who is to set the APN attribute?
  It is the network edge device such as CPE (Customer Premises Equipment) not the application.

  4. How to set the APN attribute?
  There are many possible ways that can be used to classify the traffic flow at the network edge, e.g., the N-tuples defined in the MEF70 and the AI technologies. An APN attribute can be derived by using the match items published in MEF70 as well as the access port in the edge device.

  5. How the APN attribute is used in the network?
  The APN attribute is carried in the data packet's header, and it can be used in the various nodes/service functions along the network path to enforce the policies on the differentiated traffic flow, e.g.,
  1) at the headend to steer into corresponding path satisfying SLAs
  2) at the midpoint to collect corresponding performance measurement data
  3) at the service function to execute particular policies

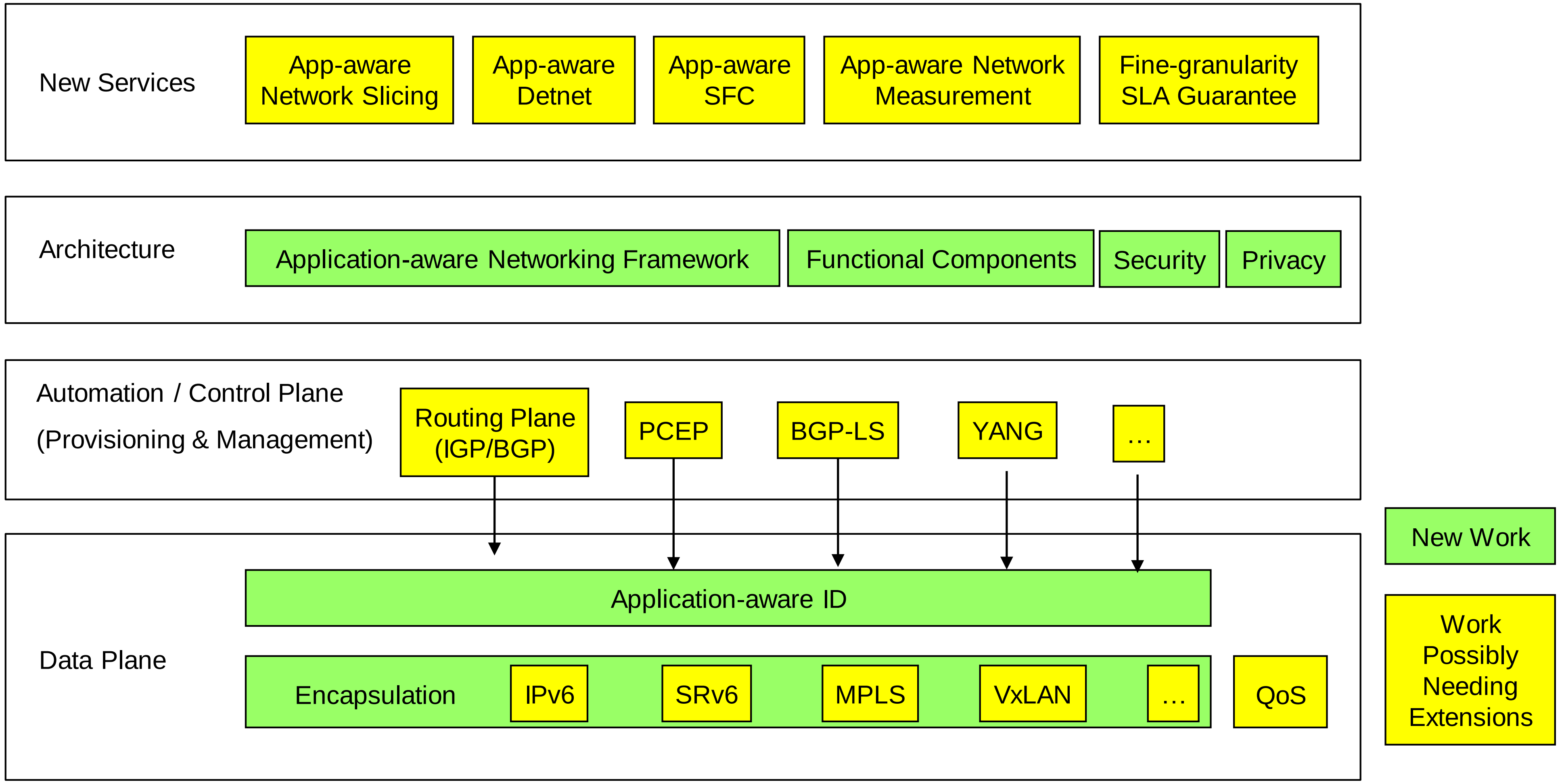
Questions still need to be addressed

- Security Issues
  - What are the security issues when the APN attribute is used within the limited operators' controlled domain?

- Privacy Issues
  - Mitigate the issues by grouping the users and applications
    - ✓ User group
    - ✓ Application group
  - Use an opaque value
  - Focus on describing the classes of policy to be applied to the traffic

Plan to form a working group

- The potential work items as below,


New Services
App-aware Network Slicing
App-aware Detnet
App-aware SFC
App-aware Network Measurement
Fine-granularity SLA Guarantee
Architecture
Application-aware Networking Framework
Functional Components
Security
Privacy
Automation / Control Plane
(Provisioning & Management)
Routing Plane (IGP/BGP)
PCEP
BGP-LS
YANG
…
Data Plane
Application-aware ID
Encapsulation
IPv6
SRv6
MPLS
VxLAN
…
QoS
New Work
Work Possibly Needing Extensions

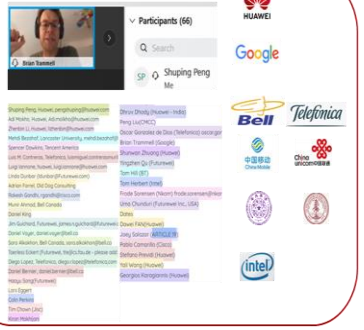
APN: Application-aware Networking

APN Activities

- **Side Meetings** @IETF105 & IETF108
- **Hackathons** @IETF108 & IETF109 & IETF110
- **Demos** @INFOCOM2020 & 2021
- **APN Mailing List** Discussions - apn@ietf.org

IETF105

IETF108

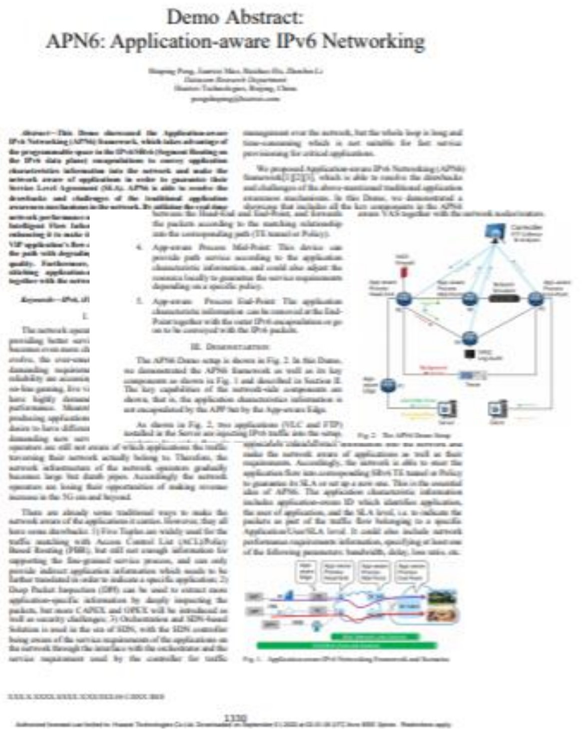


IETF109

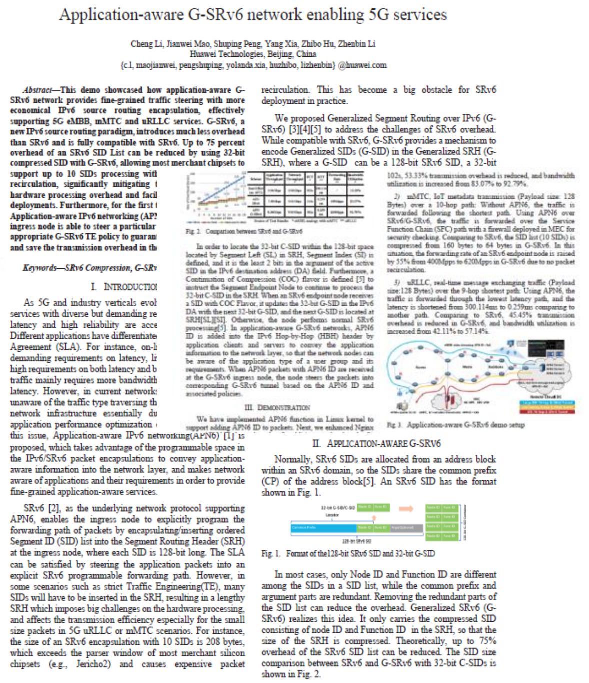
IETF110

https://github.com/APN-Community

https://www.ietf.org/blog/ietf109-bofs/
https://www.ietf.org/blog/ietf110-bofs/



Demo Abstract:
APN6: Application-aware IPv6 Networking



Application-aware G-SRv6 network enabling 5G services

Cheng Li, Jianwei Mao, Shuping Peng, Yang Xia, Zhibo Hu, Zhenbin Li
Huawei Technologies, Beijing, China
{c.l, maojianwei, pengshuping, yolanda.xia, huzhibo, lizhenbin} @huawei.com

Abstract—This demo showcased how application-aware G-SRv6 network provides fine-grained traffic steering with more economical IPv6 source routing encapsulation, effectively supporting 5G eMBB, mMTC and uRLLC services. G-SRv6, a new IPv6 source routing paradigm, introduces much less overhead than SRv6 and is fully compatible with SRv6. Up to 75 percent overhead of an SRv6 SID List can be reduced by using 32-bit compressed SID with G-SRv6, allowing most merchant chipsets to support up to 10 SIDs processing with recirculation, significantly mitigating [illegible]

recirculation. This has become a big obstacle for SRv6 deployment in practice.

We proposed Generalized Segment Routing over IPv6 (G-SRv6) [3][4][5] to address the challenges of SRv6 overhead. While compatible with SRv6, G-SRv6 provides a mechanism to encode Generalized SIDs (G-SID) in the Generalized SRH (G-SRH), where a G-SID can be a 128-bit SRv6 SID, a 32-bit [illegible]

https://ieeexplore.ieee.org/abstract/document/9162934

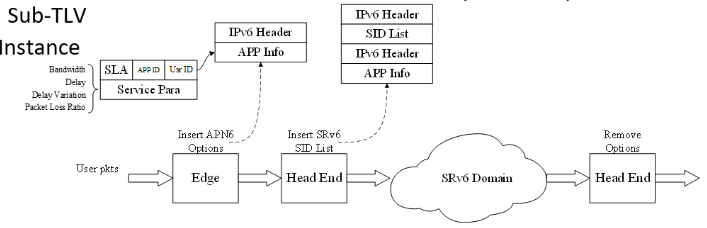
Implemented Functions

- We've implemented the demo based on *P4*, and conducted some simulations based on *BMv2*.
- Functions in Demo
  - APN6:
    1. The encapsulation of APN6 Options and Serice-Para Sub-TLV, support 2 types of APN6 Options and 4 types of Sub-TLV
    2. The encapsulation of the SRv6 SID List according to IPv6 DA and APN6 options
    3. Basic SRv6 END SID processing

Application-aware G-SRv6 network

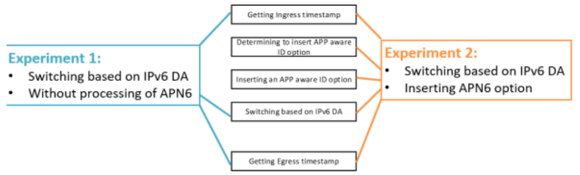
- Champions
  - Jianwei Mao (maojianwei@...)
  - Cheng Li (c.l@...)
  - Shuping Peng (pengshuping@...)
- Projects
  - Develop functions of Generalized SRv6 (G-SRv6
  - Combine G-SRv6 with APN6, to achieve Applica
- Specifications
  - draft-lc-6man-generalized-srh
  - draft-cl-spring-generalized-srv6-np
  - draft-cl-spring-generalized-srv6-for-cmpr
  - draft-li-6man-app-aware-ipv6-network
  - draft-li-apn-framework

- Application-aware traffic control.
  - Make use of the IPv6 extensions header to convey the service requirements, in the form of APN6 options and optional Sub-TLV.
  - Determine the SRv6 SID List based on the encapsulated options and Sub-TLV
- An Instance



Performance Evaluation

- **Processing Latency**


Getting Ingress timestamp
Determining to insert APP aware ID option
Inserting an APP aware ID option
Switching based on IPv6 DA
Getting Egress timestamp

Experiment 1:
- Switching based on IPv6 DA
- Without processing of APN6

Experiment 2:
- Switching based on IPv6 DA
- Inserting APN6 option

- **Result**
  - Send 50,000 packets in each experiment.
  - The interval between 2 packets is 1ms.
  - All results are in nanoseconds

Experiment	*Mean*	*STDEV*	*MAX*	*MIN*	*Range*
1 (IPv6)	364.07436	0.56514087	366	363	3
2 (IPv6 & APN6)	370.63256	0.611774343	373	369	4
DIFF	6.5582	0.046633473	7	6	

https://trac.ietf.org/trac/ietf/meeting/wiki/110hackathon
https://trac.ietf.org/trac/ietf/meeting/wiki/109hackathon
https://trac.ietf.org/trac/ietf/meeting/wiki/108hackathon

References

Please find the APN BoF proposal in the IETF wiki for more information.

- https://trac.tools.ietf.org/bof/trac/wiki/WikiStart

The archived discussions in this APN mailing list can be found here.

- https://mailarchive.ietf.org/arch/browse/apn/

To subscribe the APN Mailing list,

- https://www.ietf.org/mailman/listinfo/apn

Here are some relevant drafts and materials for your reference.

Scope & Gap analysis

- https://tools.ietf.org/html/draft-peng-apn-scope-gap-analysis

Problem statement & Use cases

- https://tools.ietf.org/html/draft-li-apn-problem-statement-usecases
- https://tools.ietf.org/html/draft-liu-apn-edge-usecase
- https://tools.ietf.org/html/draft-zhang-apn-acceleration-usecase
- https://tools.ietf.org/html/draft-yang-apn-sd-wan-usecase

Framework

- https://datatracker.ietf.org/doc/draft-li-apn-framework/

Security & Privacy

- https://datatracker.ietf.org/doc/draft-peng-apn-security-privacy-consideration

APN Community

- https://github.com/APN-Community

Thank you!