Application-aware Networking (APN)

Shuping Peng/Zhenbin Li
Purpose of presenting in this WG

• We have been discussing in the APN mailing list regarding the scope of the work [https://datatracker.ietf.org/wg/apn/about/](https://datatracker.ietf.org/wg/apn/about/).

• We want to constrain the scope of APN to only focus on the network-side solution, that is, the APN identifier is derived at the network edge.

• So our purpose is to clarify our scope. To be more specific, we don't believe we need to have a relationship with this WG, but if interested in this work we encourage to join our mailing list.

• We would like to introduce the concepts, clarify the scope, attract people to understand and discuss the topic, and collect feedback and suggestions on this work, to further address the main concerns that were raised by the IESG.
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
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
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.
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.

   • 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.

   • 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.
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.
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

• New applications’ trends and requirements on the network, and how the network can serve these groups/types of applications better, e.g.,
  • The video conferencing type of applications – Low latency
  • Dedicated network for power transmission – Deterministic latency/jitter
  • V2X/IoT – Large number of connections
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**
- Routing Plane (IGP/BGP)
- PCEP
- BGP-LS
- YANG
- ...

**Data Plane**
- Encapsulation
- IPv6
- SRv6
- MPLS
- VxLAN
- ...
- QoS

New Work

Work Possibly Needing Extensions

APN: Application-aware Networking

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

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

[Links to GitHub, IETF blogs, and other resources]

Application-aware G-SRv6 network

- **Champions**
  - Janwei Mao (maojianwei@...)
  - Cheng Li (ccli@...)
  - Shuping Peng (pengshuping@...)
- **Projects**
  - Develop functions of Generalized SRv6 (G-SRv6)
  - Combining G-SRv6 with APN6, to achieve Application-aware traffic control.

- **Specifications**
  - draft-lc-8man-generalized-srh
  - draft-cl-spring-generalized-srv6-8n
  - draft-cl-spring-generalized-srv6-for-cmpr
  - draft-li-8man-app-aware-ipv6-network
  - draft-li-apn-framework

- **An Instance**

Implemented Functions

- We've implemented the demo based on P4, and conducted some simulations based on BmAv2.
- Functions in Demo
  - APN6:
    1. The encapsulation of APN6 Options and SetNext-Para Sub-TLV, support 2 types of APN6 Options and 4 types of Sub-TLV.
    2. The encapsulation of the SID SID List according to IPv6 DA and APN6 options.
    3. Basic SRv6 END SID processing.

Performance Evaluation

- **Processing Latency**

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Mean</th>
<th>STDEV</th>
<th>MAX</th>
<th>MIN</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 (IPv6)</td>
<td>394.29ms</td>
<td>0.00149429</td>
<td>398</td>
<td>393</td>
<td>5</td>
</tr>
<tr>
<td>2 (IPv6 APN6)</td>
<td>395.825ms</td>
<td>0.051774343</td>
<td>397</td>
<td>393</td>
<td>4</td>
</tr>
</tbody>
</table>
References

Please find the APN BoF proposal in the IETF wiki for more information.
• [https://trac.tools.ietf.org/bof/trac/wiki/WikiStart](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/](https://mailarchive.ietf.org/arch/browse/apn/)

To subscribe the APN Mailing list,
• [https://www.ietf.org/mailman/listinfo/apn](https://www.ietf.org/mailman/listinfo/apn)

Here are some relevant drafts and materials for your reference.

Scope & Gap analysis

Problem statement & Use cases

Framework

Security & Privacy

APN Community
• [https://github.com/APN-Community](https://github.com/APN-Community)

APN: Application-aware Networking
Thank you!