

2014-10-28 - BoF notes - SDNTrace

[Notes by Joe Breen]

Guys,

Thanks for a very productive brainstorm discussion tonight regarding a sdn traceroute/ path route utility!

I have pasted the raw notes I took below. Please feel free to add/delete/improve.

I have also taken a stab at starting the definition, scope and requirements document that might come out of our notes. Please iterate.

Dan, do we have an appropriate working group wiki space or Box account where we put the working document for ease of sharing?

#####

Potential basic approach: SDN Working Group community defines the specifications for this tool in a succinct format with a tight focus. Work with Deniz Gurkan to scope problem into a student project(s). Have 1-2 community mentors volunteer to meet regularly in a virtual manner to help guide students and provide the years of production experience required. Students mock up in mininet and on GENI. Students provide to community mentors for iteration. Students provide to community to test in local environments, on AL2S, etc. Iterate. Students present posters at GEC and write papers with community mentors (if applicable).

#####

Basic problem definition:

The community needs a tool which will give the traditional "traceroute" capabilities along an OpenFlow defined path, regardless of whether the path is a layer 2 IP, a layer 2 non-IP, or a layer 3 path.

Scope:

Create an ethernet frame based tool.

Should not be an IP only tool

Can be a CLI command and/or an Application definition that will run on top of a generic controller

Requirements:

- * Must support multiple upper level protocols, not just IP

- * Must support the capability to inject frames from this tool into the SDN path, without being on an end host that might not support the tool yet, i.e. a router. (application on controller?)

- * Frame definitions must track at least the following:

- ** timestamp

- ** DPID/switch name

- ** Local Organizational Identifier: i.e. AS number, unique ID

- ** ingress interface

- ** egress interface

- ** next hop DPID, if known from LLDP or some other mechanism

- *** must be able to identify gaps

- * Must be able to provide portion of 12-tuple header upon which control plane will execute action

- * Frames must be scalable to allow more definitions of things to track and provided feedback, i.e. topological feedback, etc.

- * Tool must provide raw text feedback.

- * Tool may provide JSON feedback.

- * Each controller must respect defined path route frame, process appropriately and pass it onwards for more processing

#####

Begin notes [2014-10-2014](#) SDN Working Group Raw Notes

#

#

Attendees

Anita Nikolich, Mike Van Norman, Dan Schmiedt, Chris Konger, Dale Carder, Kevin Mayeshiro, Joe Breen

SDN Working group - lead with projects

Potential project:

how to troubleshoot cross-domain

pre-defined Loopbacks, insert test points at key locations; known locations for perfSONAR -- can build up circuits

Use GENI as testpoints? -- how to use RSpec for testing;

Show what is possible with staffing as a working group -- off-the-shelf doesn't exist today;

- * Need more experimentation - test; put up work
- * documenting the switches, controllers
- * issue with people getting burnt out on early
- * Traceroute issue -- tight focus
- ** how to dynamically query topology - from layer2
- ** how to determine who is controller is? Who to go query edge of Layer2/Layer3
- ** layer2 traceroute -- how to decrement, different ethertype?; Controller could add information; made up a special mac address? issue with special frame; layer 2 traceroute that tags off MAC and/or IP -- minimal layer 2 rule for layer2 traceroute; experimental ID # - lowest common layer 2 or layer3;

Multiprotocol traceroute

want packet to go through path - when hits controller, controller adds info to reply (where it came in and where destined) sends response -- doesn't need to be sent back; multiple probes -- write an opendaylight plugin to handle the packet; define it as a reference for other controllers; Response from each controller has an ID number; instead of updating TTL, should send something and hit next controller; controller can get a lot of info about the topology and put in the packet - optional fields with JSON object with whatever info;

- * Summarize description:
- * Iterate on problem
- * Start in mininet?
- * Define top 5 info pieces -- utilization on interface? , interface type, start time/timestamp,(issue with controllers), DPID/name, AS number (or some local ID info), interface incoming, interface outgoing, if know far end DPID info so know if gap in data set (useful for cross-domain),
- ** minimal info with a url and magic key for more info? issue with too much maintenance -- want a simple tool - this idea is probably too much
- ** what sending to the controller? need a cookie for each controller - payload is list of controller cookies?
- *** query - hypothetical header - info
- *** anycast to controller?
- *** minimal - have an app running on controller that can inject packet/frame/

<<Dan>> give Deniz this problem for students to solve; Working group to spec out.

Other important notes - layer 2 only and layer 3 only switches

willys@clem

mvn@ucla.edu

kmayeshiro@ucdavis.edu

ckonger@clemson.edu

dwcarter@wisc.edu