Anuket Weekly Technical Discussions - 2021.08.25

Participants

Please add your name to the list

- Gergely Csatari
- Beth Cohen
- Georg Kunz
- Ulrich Kleber
- Sridhar Rao
- Luc Provoost
- Karine Sevilla
- Ildiko Vancsa
- Cedric Ollivier

Antitrust policies

- Linux Foundation Anti-Trust Policy
- GSMA Anti-Trust Policy Notice
- Recorded Policies:
    - https://zoom.us/rec/play/uMAuIuyoqG43EtWS4QSDAf4oW9XsKP2s1CQW-fYNzxu3VSIGN1L3Z7YSNOX-H7MYUyogBuINABoVavXq?autoplay=true&startTime=1589371628000
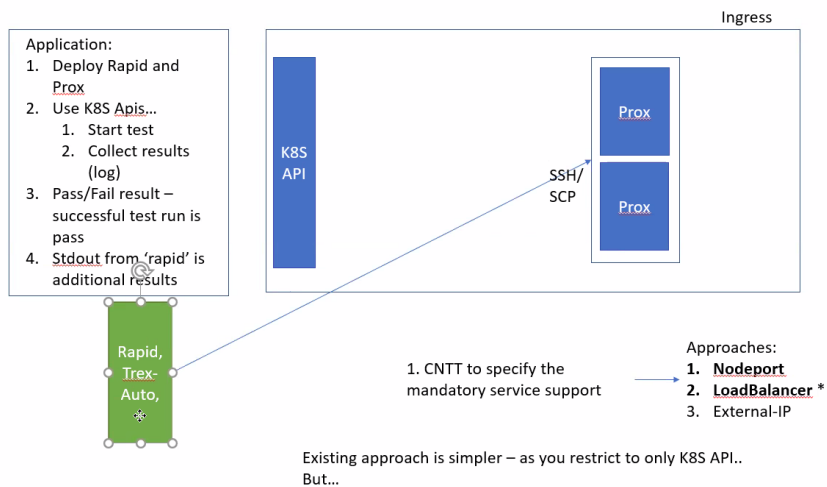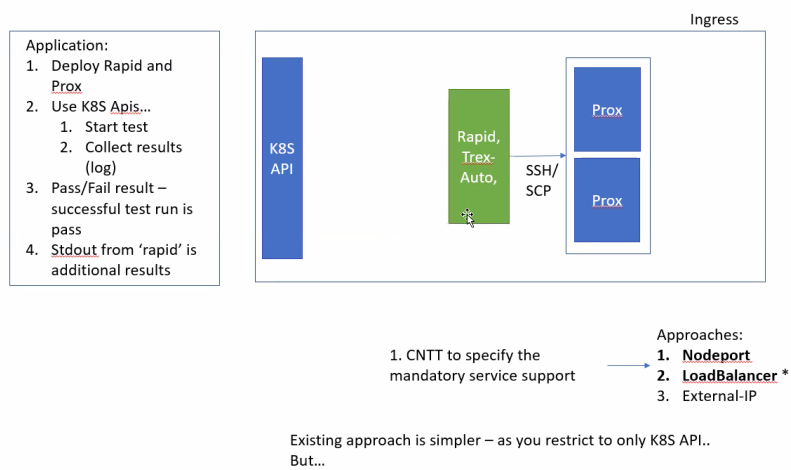
Action item register

Organisation topics

- N/A

Technical topics

Containerization of traffic generators with Xtesting - Sridhar Rao

- Initial architecture: Integration with Xtesting - Lakelse
- Discussion of traffic generators. Where do they need to be installed?
- They need to have additional tools installed either inside or outside the environment to test.
- Need to deploy the tools (Rapid and Prox), then use K8S APIs to run the tests and collect results.
- For this work the test results are pass/fail. Stdout from "rapid" is additional results.
- So where do you put the control function (Rapid) to start and stop the traffic generator (Prox),
- Currently it HAS to be inside the environment, better to move It to outside.
- One approach
    - There is a need for an application to LCM the test framework pods (Rapid, Prox), collect the results and calculate the test results
    - If this application is outside of the cluster and uses the Kubernetes API-s
    - For this the integration between Rapid and Xtesting needs to be modified
    - Presentation of the output needs to be also changed
- Maybe there are alternate approaches
    - If the complete Xtesting runs within the cluster the integration will be more easy, but in that case collecting the results is ugly and difficult
- In RA2 there are no guidelines about what are the mandatory service types
    - E.g.: Nodeport, Loadbalancer, External IP
    - There is an RA2 issue about this: https://github.com/cntt-n/CNTT/issues/2453
    - The approach for SSH connection is not defined without this
        - Best candidate would be Loadbalancer, for this RA2 should add Loadbalancer as a mandatory service
    - There are tests about Ingress in the Kubernetes conformance test suite
    - We are not sure if KubeRef has Loadbalancer or External IP installed
    - This is a pre requisite for successful testing
- Desired architecture:

Application:
1. Deploy Rapid and Prox
2. Use K8S Apis…
   1. Start test
   2. Collect results (log)
3. Pass/Fail result – successful test run is pass
4. Stdout from 'rapid' is additional results

K8S API

SSH/SCP

Prox

Prox

Ingress

Rapid, Trex-Auto,

1. CNTT to specify the mandatory service support

Approaches:
1. **Nodeport**
2. **LoadBalancer** *
3. External-IP

Existing approach is simpler – as you restrict to only K8S API.. But…

-
- Possible workaround:

Application:
1. Deploy Rapid and Prox
2. Use K8S Apis…
   1. Start test
   2. Collect results (log)
3. Pass/Fail result – successful test run is pass
4. Stdout from 'rapid' is additional results

K8S API

Rapid, Trex-Auto,

SSH/SCP

Prox

Prox

Ingress

1. CNTT to specify the mandatory service support

Approaches:
1. **Nodeport**
2. **LoadBalancer** *
3. External-IP

Existing approach is simpler – as you restrict to only K8S API.. But…

-
- Next steps
  - Nodeport is not a nice solution, but that supported by every Kubernetes deployments
    - For this the exact node and the port should be figured out
    - Nodeport is disliked by several telcos
    - It is good as a first iteration, let's start with this
  - Desired solution would be Loabalancer, but that needs an discussion in RA2 what cold take some time

AoB

- None