

Status and Plans for the Research Networking Technical WG

Shawn McKee / University of Michigan, Marian Babik / CERN

Spring 2023 HEPiX Meeting

<https://indico.cern.ch/event/1222948/contributions/5321009/>

March 29, 2023

Presentation Overview

For High-Energy Physics (HEP), we have identified a need to better understand and optimize our network traffic to ensure we are using the network as effectively (for our science) as possible.

We want to update you on the technical working group, which is focused on addressing some specific areas of interest to HEP that are relevant for the broader R&E community globally.

Reminder: WLCG Network Requirements

- Many WLCG facilities need **network** equipment refresh
 - Routers in many sites are End-Of-Life and moving out of warranty
 - Local area networking often has 10+ year old switches which are no longer suitable
- WLCG planning is including networking to a much greater degree than before
 - HL-LHC computing review: DOMA, [dedicated networking section](#)
 - HL-LHC Computing Conceptual Design Reports, [highlight needs](#)
 - Snowmass CompF4 has [dedicated networking section](#)
 - All include input from HEPiX, LHCONE/LHCOPN and WLCG working groups
- **Requirements Summary**
 - **Capacity:** Run-3 moving to multiple 100G links for big sites, Run-4 targeting Tbps links
 - **Capability:** WLCG needs to understand the impact of new features in networking (SDN/NFV) by [testing](#), [prototyping](#) and [evaluating impact](#). They will need to evolve their applications, facilities and computing models to meet the HL-LHC challenges; *it will take time*.
 - **Visibility:** As the ESnet Blueprinting meetings have shown, our ability to understand our WAN network flows is too limited. We need new methods to mark and monitor our network use
 - **Testing:** We need to be able to develop, prototype and test network features at suitable scale

- HEPiX Network Functions Virtualisation Working Group
 - [Working Group Report](#) was published at the end of 2019 with three chapters
 - Cloud Native DC Networking
 - Programmable Wide Area Networks
 - Proposed Areas of Future Work
- [LHCOPN/LHCONE workshop](#) (January 2020)
 - Requirements on networks from the WLCG experiments
- **Research Networking Technical Working Group**
 - Formed after the workshop in response to the requirements discussion
 - 98 members from ~ 50 organisations have [joined](#)
 - Three main areas of work:
 - **Network traffic visibility**
 - **Network traffic pacing**
 - **Network traffic orchestration**

This working group is focused on some specific, practical network efforts:

1. **Network visibility** via Packet Marking / Flow Labeling
2. **Network usage optimization** via Packet Pacing / Traffic Shaping
3. **Network management** via Network Orchestration / GNA-G DIS / SENSE / NOTED

Charter for the main group is at

<https://zenodo.org/record/6470973#.YmamPNrMJD8>

To undertake the above efforts we have created three subgroups looking into each of the areas above.

- Networks are becoming more programmable and capable with technologies such as P4, SDN, virtualisation, eBPF, etc.
- But with less and less context about the traffic they carry.
 - Cloud deployments, Kubernetes, encryption, tunneling, privacy, etc.
- **Understanding scientific traffic flows in detail is critical for understanding how our complex systems are actually using the network.**
 - Current monitoring/logging tell us where data flows start and end, but is unable to understand the data in flight.
 - Dedicated L3VPNs can be created to track high throughput science domains, but with more domains requiring high throughput this will become expensive, it won't scale, won't work at big sites having to support multiple domains at the same time.
- In general the monitoring we have is experiment specific and very difficult to correlate with what is happening in the network. We suggest this is a general problem for users of the Research and Education Networks (RENs).

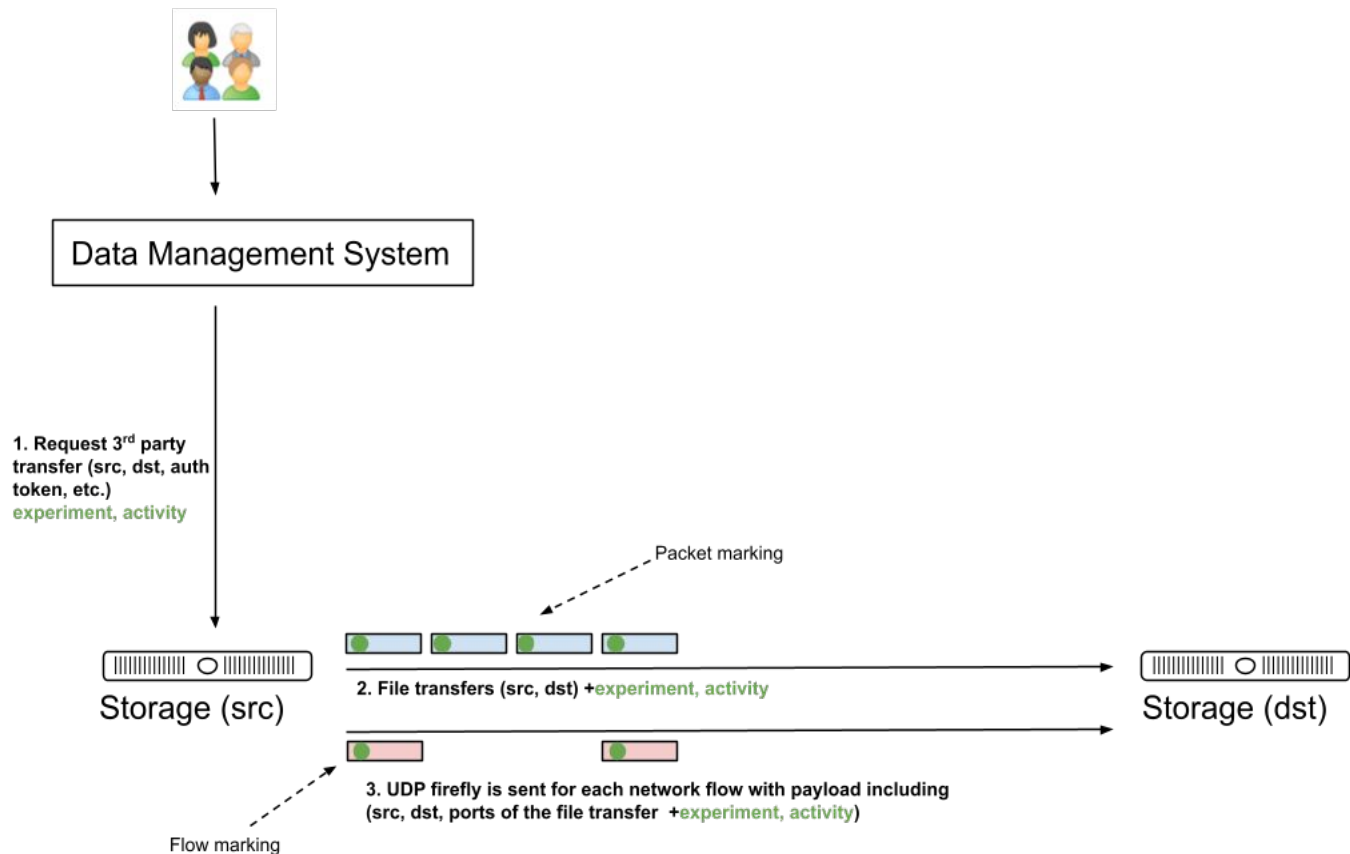
Network Visibility and Scitags

- **Scientific Network Tags** (scitags) is an initiative promoting identification of the science domains and their high-level activities at the network level.

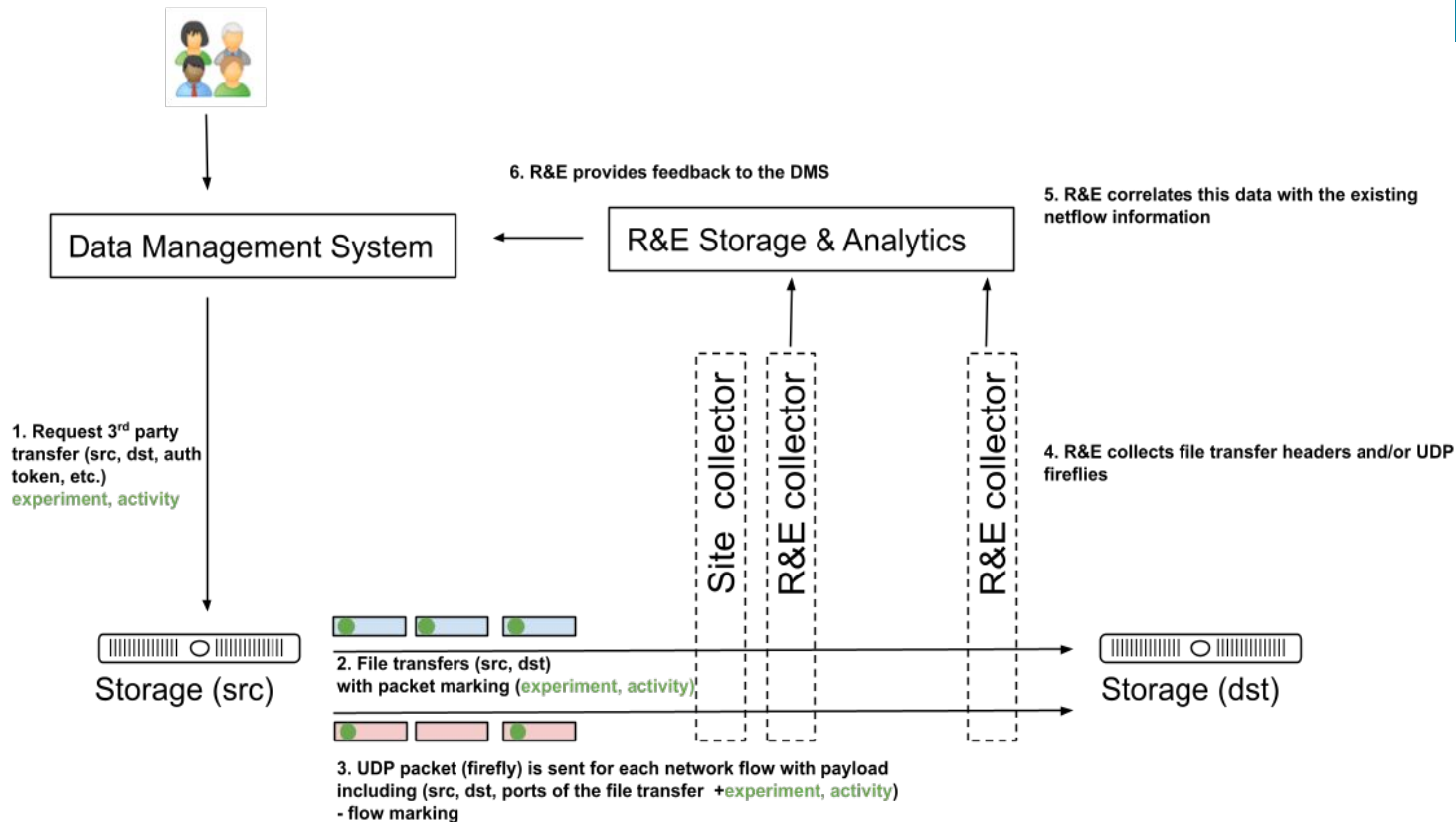


- Enable **tracking** and **correlation** of our transfers with Research and Education Network Providers (R&Es) network flow monitoring
- **Experiments** can better understand how their network flows perform along the path
 - Improve visibility into how network flows perform (per activity) within R&E segments
 - Get insights into how experiment is using the networks, get additional data from R&Es on behaviour of our transfers (traffic, paths, etc.)
- Sites can get visibility into how different network flows perform
 - Network monitoring per flow (with experiment/activity information)
 - E.g. RTT, retransmits, segment size, congestion window, [etc.](#) all per flow

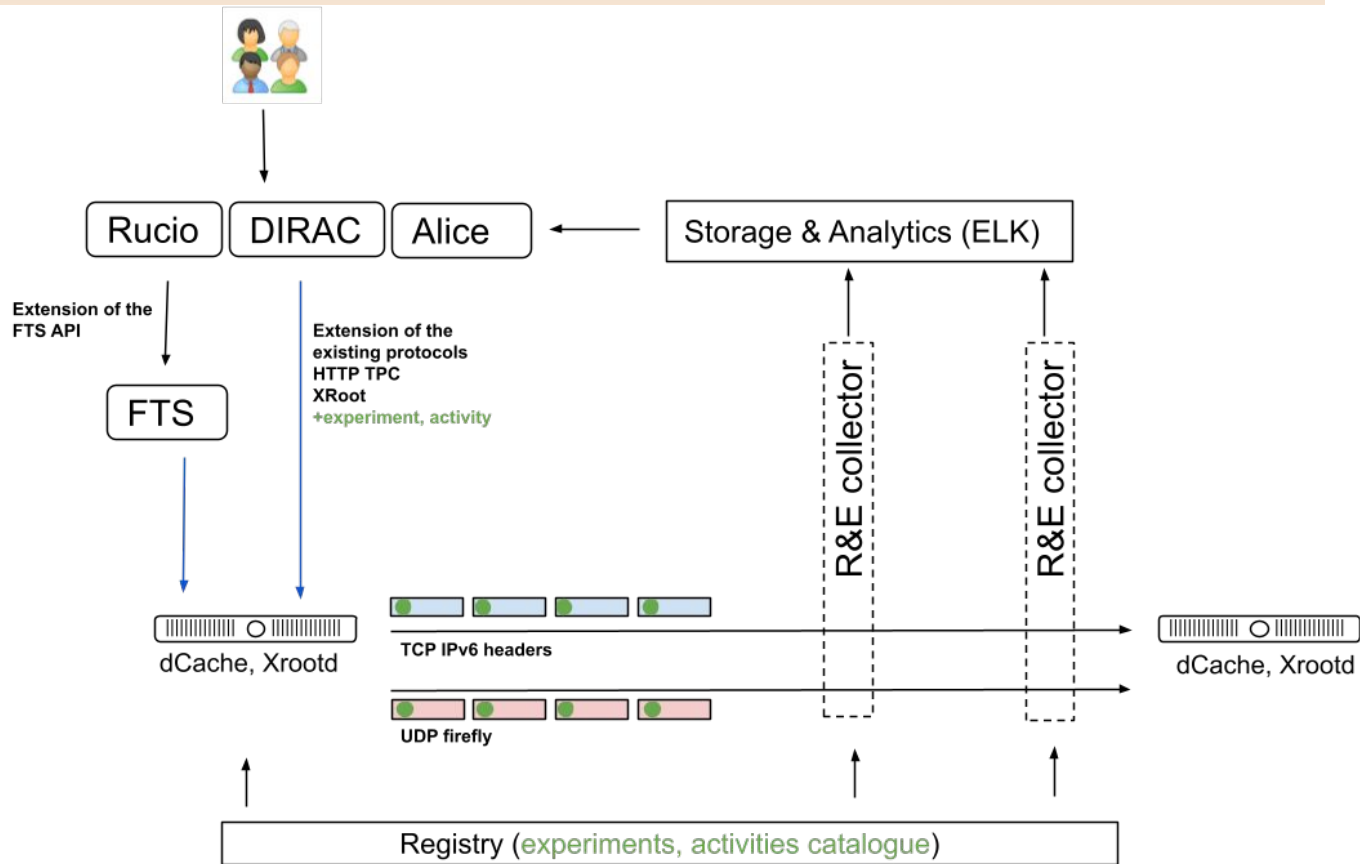
How scitags work



How scitags work



How scitags work



Finding More Information: <https://scitags.org>

Code

scitags.org

Network Flow and Packet Marking for
Global Scientific Computing



Technical Spec

Mailing List

Hosted on GitHub Pages — Theme by orderedlist

Scientific network tags (scitags) is an initiative promoting identification of the science domains and their high-level activities at the network level.

It provides an open system using open source technologies that helps *Research and Education (R&E) providers* in understanding how their networks are being utilised while at the same time providing feedback to the *scientific community* on what network flows and patterns are critical for their computing.

Our approach is based on a network tagging mechanism that marks network packets and/or network flows using the science domain and activity fields. These tags can then be captured by the *R&E providers* and correlated with their existing netflow data to better understand existing network patterns, estimate network usage and track activities.

The initiative offers an **open collaboration on the research and development of the packet and flow marking prototypes** and works in close collaboration with the scientific storage and transfer providers to enable the marking capability. The project is currently in the prototyping phase and is open for participation from any science domain that require or anticipate to require high throughput computing as well as any interested *R&E providers*.

Participants



Upcoming and Past Events

- March 2022: LHCOPN/LHCONE workshop
- November 2021: GridPP Technical Seminar (slides)
- November 2021: ATLAS ADC Technical Coordination Board
- October 2021: LHCOPN/LHCONE workshop (slides)
- September 2021: 2nd Global Research Platform Workshop (slides)

Presentations

Technical Spec for Packet Marking/Flow Labeling

The detailed technical specifications are maintained on a [Google doc](#)

- The spec covers both **Flow Labeling** via **UDP Fireflies** and **Packet Marking** via the use of the **IPv6 Flow Label**.
 - **Fireflies** are UDP packets in Syslog format with a defined, versioned JSON schema.
 - Packets are intended to be sent to the same destination (port 10514) as the flow they are labeling and these packets are intended to be world readable.
 - Packets can also be sent to specific regional or global collectors.
 - Use of syslog format makes it easy to send to Logstash or similar receivers.
 - **Packet marking** is intended to use the 20 bit flow label field in IPv6 packets.
 - To meet the spirit of RFC6437, we use 5 of the bits for entropy, 6 for activity and 9 for owner/experiment.
- The document also covers methods for communicating owner/activity and other services and frameworks that may be needed for implementation.

- **Flow Marking** (UDP firefly) implementations
 - Xrootd 5.4+ supports UDP fireflies
 - https://xrootd.slac.stanford.edu/doc/dev54/xrd_config.htm#_pmark
 - **map2exp** - can be used to map particular path to an experiment
 - **map2act** - can be used to map particular user/role to an activity
 - Flowd - prototype service
 - Issue fireflies from netstat for a given experiment (only for dedicated storages)
- **Collectors**
 - Initial prototype was developed by ESnet (will be available on [scitags github](#) soon)
 - ESnet and Jisc/Janet*
- **Registry**
 - Provides list of experiments and activities supported
 - Exposed via JSON at api.scitags.org
- Simplified deployment was tested during the last DC (& still operating)
 - Flowd + ESnet collector + Registry
 - **AGLT2, BNL, KIT, UNL and Caltech** participated
 - Brunel, Glasgow and QMUL interested to help with further testing
- New **flowd** version will be ready to be deployed shortly (testing now)

Registry

We have standardized the “experiment” and “activity” fields we use for both flow labeling and packet marking.

The scitags.org domain provides an API that can be consulted to get the standard values:

<https://api.scitags.org> or <https://www.scitags.org/api.json>

The underlying source of truth is a set of [Google sheets](#) that are maintained and writeable by a few stewards.

Note: the API provides the defined values **but** how the values are used in packet marking are specified in our [Google sheets](#) (bit location in IPv6 flow label)

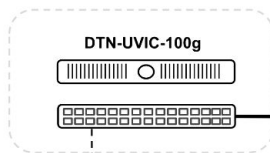
```
{
  - experiments: [
    - {
      expName: "default",
      expId: 1,
      - activities: [
        - {
          activityName: "default",
          activityId: 1
        }
      ]
    },
    - {
      expName: "atlas",
      expId: 2,
      - activities: [
        - {
          activityName: "perfsonar",
          activityId: 2
        },
        - {
          activityName: "cache",
          activityId: 3
        },
        - {
          activityName: "datachallenge",
          activityId: 4
        },
        - {
          activityName: "default",
          activityId: 8
        },
        - {
          activityName: "analysis download",
          activityId: 9
        },
        - {
          activityName: "analysis download direct io",
          activityId: 10
        }
      ]
    }
  ]
}
```



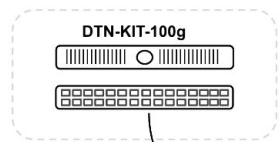
Packet and Flow Marking Demos



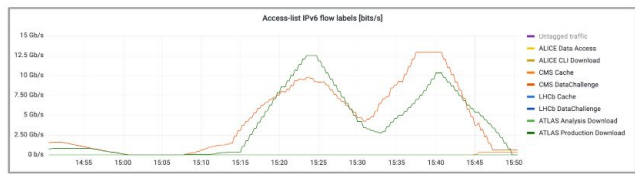
scitags.org
Flow and Packet Marking for Global Scientific Computing



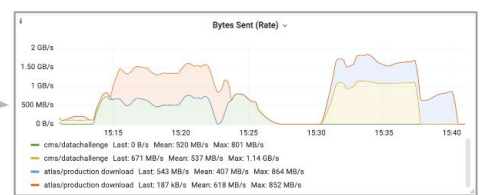
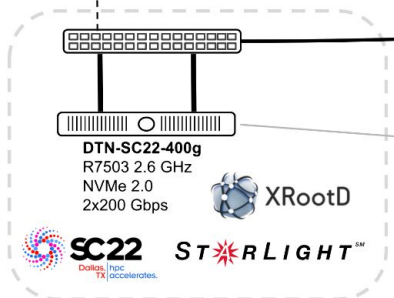
University of Victoria
1. Clients requesting data transfers from/to DTN-SC22-400g while passing science domain and activity fields via transfer protocols.



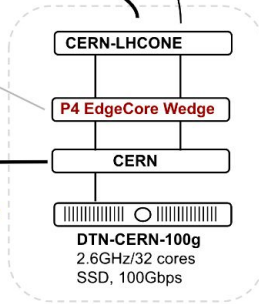
canarie
4. High performance tests using eBPF-TC filters to test encoding of the science domains and activity fields in the IPv6 flow label at scale.



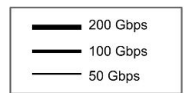
3. P4 programmable switch at CERN collecting the science domains and activity bits encoded in the packets.



2. XRooT D storage responds to the client requests and marks the data transfer packets with the corresponding science domain and activity.



5. Sampling of the low level TCP/IP metrics, which can be used by sites and R&Es to better understand the scientific flows.



During Supercomputing 22 in Dallas, we demonstrated a number of aspects of our packet and flow marking work.

- We showed packet marking at **200 Gbps** rates using flowd with both **xrootd** and **iperf3**.
- Scinet and ESnet set up packet collectors [via sflow](#) and demonstrated real-time monitoring of packets by experiment and activity.
- Demos were also run on LHCONE using equipment in the SC22 booth, KIT, UVic and CERN where packet marking for all transfers was monitored using a P4 programmable switch.



Pacing/Shaping WAN data flows

A challenge for HEP storage endpoints is to utilize the network efficiently and fully.

- An area of interest for the experiments is **traffic shaping/pacing**.
 - Without traffic pacing, network packets are emitted by the network interface in bursts, corresponding to the wire speed of the interface.
 - **Problem:** **microbursts** of packets can cause buffer overflows
 - The impact on TCP throughput, especially for high-bandwidth transfers on long network paths can be **significant**.
- Instead, pacing flows to match expectations [$\min(\text{SRC}, \text{DEST}, \text{NET})$] smooths flows and significantly reduces the microburst problem.
 - An important extra benefit is that these smooth flows are much friendlier to other users of the network by not bursting and causing buffer overflows.
 - Broad implementation of pacing could make it feasible to run networks at much higher occupancy before requiring additional bandwidth

This work has yet to have much effort; we plan to begin work during this summer!

- As we have seen this week, OpenStack and Kubernetes are being leveraged to create very dynamic infrastructures to meet a range of needs.
 - Critical for these technologies is a level of automation for the required networking using both software defined networking and network function virtualization.
 - For HL-LHC, important to find tools, technologies and improved workflows that may help bridge the anticipated gap between the resources we can afford and what will actually be required
- The ways we organize our computing / storage resources will need to evolve.
- This area is being led by the **GNA-G** (Global Network Advancement Group; <https://www.gna-g.net/>) and is exploring many options for traffic engineering, resource management and network-application interfaces.
 - The **SENSE** project is serving as a reference implementation
- The [NOTED project](#) is also an example of a practical way to effectively utilize available paths to better distribute network load.

Network Visibility Plans

- Near-term objectives
 - Continue rollout and testing of Xrootd implementation
 - Detect flow identifiers from storage path/url, activities from user role mapping
 - Finalise development and deploy [network of receivers \(backup slides\)](#)
 - Instrument Rucio/FTS to pass flow identifiers to the storages
 - Good discussions with dCache last week covering SciTags design/implementation
 - Work with the [WLCG Monitoring TF](#) to improve site network monitoring
- Engage other R&Es and explore available technologies for collectors
 - Deploy additional collectors and perform R&D in the packet collectors
 - Improve existing data collection and analytics
- Test and validate ways to propagate flow identifiers
 - Engage experiments and data management systems
 - Validate, test protocol extensions and FTS integration
 - Explore other possibilities for flow identifier propagation, e.g. tokens
- R&D activities
 - **Packet marking** - further testing and validation is required for IPv6 flow label implementation (next meeting or two)

Plans: Network Pacing & Orchestration

The RNTWG has primarily focused on the network visibility area due to limited manpower, but we have two additional areas that are part of the overall group goal: traffic shaping and network orchestration

Traffic Shaping:

- The WLCG experiments would like to explore traffic shaping/packet pacing.
 - Without packet pacing, network packets are emitted by the network interface in bursts, corresponding to the wire speed of the interface.
 - **Problem:** microbursts of packets can cause buffer overflows
 - The impact on TCP throughput, especially for high-bandwidth transfers on long network paths can be **significant**.
- Instead, pacing flows to match expectations $[\min(\text{SRC}, \text{DEST}, \text{NET})]$ smooths flows and significantly reduces the microburst problem.
 - An important extra benefit is that these smooth flows are much friendlier to other users of the network by not bursting and causing buffer overflows.
 - Broad implementation of pacing could make it feasible to run networks at much higher occupancy before requiring additional bandwidth

Network Orchestration:

- This effort is being led by the GNA-g and includes work from the SENSE and FABRIC projects. **Current focus is on integration with the experiments via RUCIO to demonstrate value and capability.**

Discuss SciTag Firefly Implications

One quick heads-up for sites and network providers: we are beginning to send **UDP fireflies** from some of our sites.

UDP fireflies (by default) are sent to the same destination as the data transfer flow. This means UDP packets arriving at storage servers on port 10514.

A site can choose to ignore, block or capture these packets

We are working on an informational RFC (target to publish Fall 2023)

One implication: if packets hit iptables, it may generate noise in the logging that may be a concern (fill /var/log?)

Recommendation is to open port 10514 for incoming UDP packets or explicitly 'drop' them.

Summary

The RNTWG has made significant progress in the identified network priority focus areas for the WLCG community. The current focus is on the network traffic visibility through the work on flow labeling and packet marking.

- There remains a significant amount of work to do, especially regarding enabling packet marking on our storage infrastructure and in the area of collecting, aggregating and making visible the marked traffic.

We have additional near-term work to pursue in traffic shaping:

- While network orchestration has significant activity underway, we need to find new effort interested in developing, prototyping and evaluating traffic shaping

Interested? We are always looking for additional members to join the effort!

Acknowledgements

We would like to thank the **WLCG**, **HEPiX**, **perfSONAR** and **OSG** organizations for their work on the topics presented.

In addition we want to explicitly acknowledge the support of the **National Science Foundation** which supported this work via:

- [OSG: NSF MPS-1148698](#)
- [IRIS-HEP: NSF OAC-1836650](#)

Questions?

HEP*i*X

Questions, Comments, Suggestions?

Useful URLs

[RNTWG Google Folder](#)

[RNTWG Wiki](#)

[RNTWG mailing list signup](#)

HEPiX NFV Final Report [WG Report](#)

RNTWG Meetings and Notes: <https://indico.cern.ch/category/10031/>

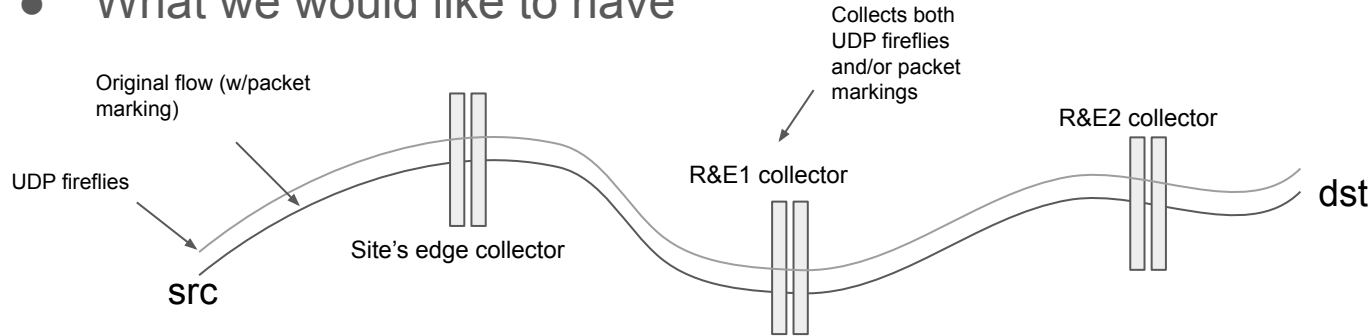
The scitags web page: <https://scitags.github.io>

Code at <https://github.com/scitags/scitags.github.io>

Backup slides

Collectors

- What we would like to have

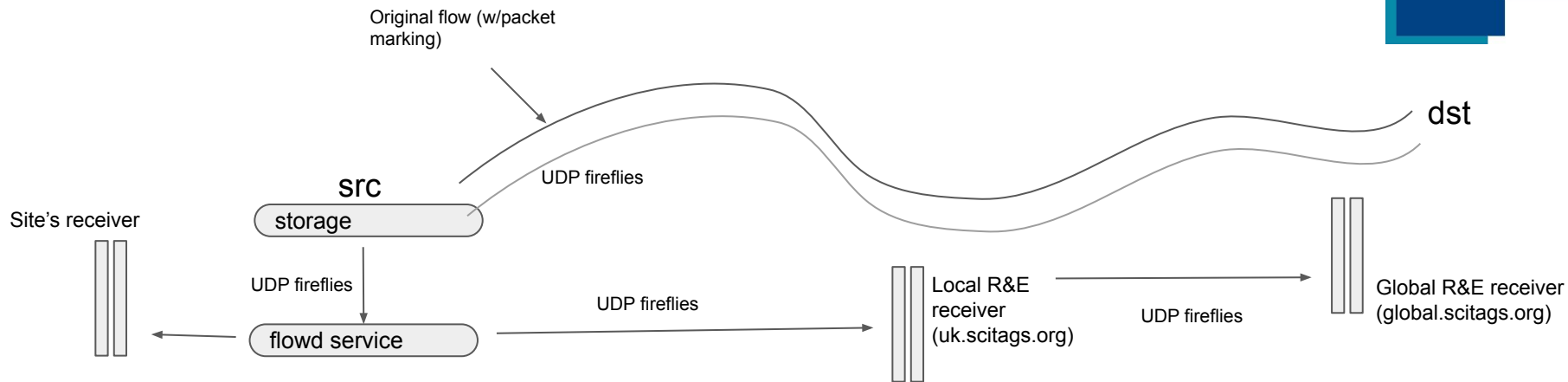


- Enable collection of packet and flow markings along the path
 - In order to extend R&E netflow information with flow identifier (experiment + activity)
 - UDP firefly packets needs to be collected and relayed to ensure they reach all collectors
- Each R&E **can** setup and operate one or more collectors
- Sites have an option to set up their own collector at the edge

Collectors

- Our **recommendation** is to use hardware/in-line collectors where possible
 - Requires port mirroring or other means to capture the fireflies.
 - Easiest to organise and operate as there is no need for a separate collector network.
 - Only way to capture flow markings along the path.
 - However, in-line collectors require the ability to either selectively identify and capture fireflies or the ability to capture IPv6 flow labels from packets
 - Many possible ways to implement.
 - Strategy and technology to implement will depend on the R&E, their topology and hardware.
 - Would be great to get example implementations that can be shared between R&E network operators.
-

Network of receivers



- Storages are configured with predefined DNS aliases (based on region; hosted by scitags.org)
 - Flowd service will expose API for site's local receivers and will also forward UDP fireflies to R&E collector (storage will send fireflies along the path)
 - Local R&E collector can be established (optional) and will need to pass all received fireflies to the global one (can switch to TCP)
- Works with inline/hardware collectors (which can be setup in parallel)
- Easy way to setup local R&E receiver (and correlate with local netflow)
- Lightweight - should be easy to operate, but requires some development in flowd and in the R&E collector
- DNS aliases will give us flexibility to make changes in the future (e.g. move to anycast)

Packet Marking - Storage Elements

The primary challenge here is in two areas:

1. Augmenting the existing storage system to be able to set the appropriate bits in the network packets
2. Communicating the appropriate bits as part of a transfer request
 - a. Likely need some protocol extension to support this
 - b. Other ideas?

Packet Marking - Jobs

As jobs source data onto the network OR pull data into the job, we should try to ensure the corresponding packets are marked appropriately

- Containers and VMs may allow this to be easily put in place
- Still need configuration options that specify the right bits
- Signalling to the “source” about what those bits are also needs to be in place

Packet Marking - IPv6

IPv6 incorporates a “Flow Label” in the header (20 bits)

Fixed header format

Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	<i>Version</i>				<i>Traffic Class</i>				<i>Flow Label</i>																							
4	32	<i>Payload Length</i>																<i>Next Header</i>				<i>Hop Limit</i>											
8	64	<i>Source Address</i>																															
12	96																																
16	128																																
20	160																																
24	192																																
28	224	<i>Destination Address</i>																															
32	256																																
36	288																																

Packet Marking - IPv4

IPv4 incorporates a “Options” in the header (allowing to add more 32 bit words)

IPv4 Header Format

Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Version				IHL				DSCP				ECN				Total Length															
4	32	Identification																Flags				Fragment Offset											
8	64	Time To Live								Protocol								Header Checksum															
12	96	Source IP Address																															
16	128	Destination IP Address																															
20	160	Options (if IHL > 5)																															
24	192																																
28	224																																
32	256																																

Network Functions Virtualisation WG

Mandate: Identify use cases, survey existing approaches and evaluate whether and how Software Defined Networking (SDN) and Network Functions Virtualisation (NFV) should be deployed in HEP.

Team: 60 members including **R&Es** (GEANT, ESNNet, Internet2, AARNet, Canarie, SURFNet, GARR, JISC, RENATER, NORDUnet) and **sites** (ASGC, PIC, BNL, CNAF, CERN, KIAE, FIU, AGLT2, Caltech, DESY, IHEP, Nikhef)

Monthly **meetings** started in Jan 2018 (<https://indico.cern.ch/category/10031/>)

NFV WG produced an interim-report that could serve as one of the inputs for the LHCOPN/LHCONE feedback

Executive summary for NFV Phase 1 report is at

<https://docs.google.com/document/d/1w7XUPxE23DJXn--j-M3KvXlfXHUnYgsVUhBpKFjyUQ/edit#heading=h.flthknqgm3ub>

Report has **3 chapters**:

- Cloud Native DC Networking

- Programmable WAN

- Proposed Areas of Future Work

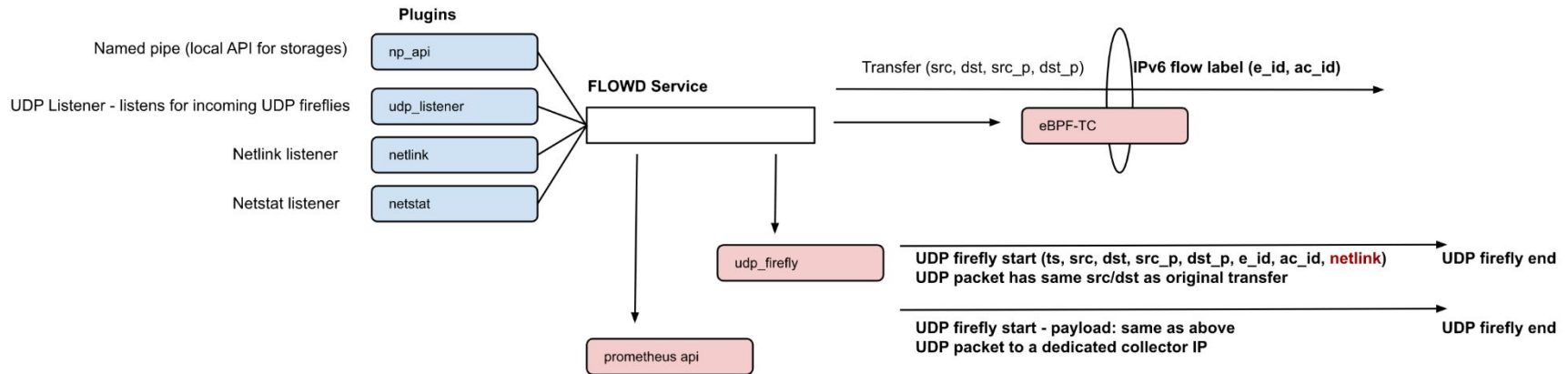
Future (phase 2) is partially the work of this RNT WG, but we may end up separating out a more focused NFV/SDN group.

NFV Report Conclusions

The primary challenge we face is ensuring that WLCG and its constituent collaborations will have the networking capabilities required to most effectively exploit LHC data for the lifetime of the LHC. To deliver on this challenge, automation is a must. The dynamism and agility of our evolving applications, tools, middleware and infrastructure require automation of at least part of our networks, which is a significant challenge in itself. While there are many technology choices that need discussion and exploration, **the most important thing is ensuring the experiments and sites collaborate with the RENs, network engineers and researchers to develop, prototype and implement a useful, agile network infrastructure that is well integrated with the computing and storage frameworks being evolved by the experiments as well as the technology choices being implemented at the sites and RENs.**

Flowd: News

- Flow and Packet Marking service developed in Python
 - Aims to provide a framework to test/evaluate various flow/packet marking ideas



- Plugins provide different ways get connections to mark (or interact with storage)
 - New plugins were added to support netlink readout and UDP firefly consumer

SC22 Packet and Flow Marking Demo

SC22 NRE

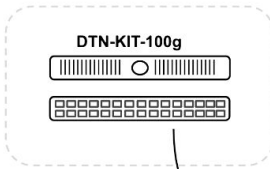
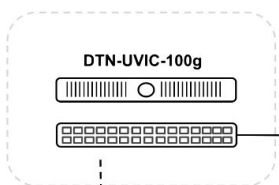
- The SC22 goals of the Packet and Flow Marking NRE demonstration will be to showcase the capabilities of the Scitags architecture and methods for optimizing data intensive science.
- Three demos:
 - IPv6 packet marking with eBPF-TC (100Gbps)
 - XRootD flow and packet marking with flowd + eBPF-TC
 - Accounting of flow label tagged packets using a P4 programmable switch
- Participants:
 - Starlight, CERN, University of Victoria, KIT, ESnet, Canarie

SC22 Demo

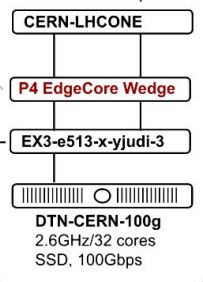
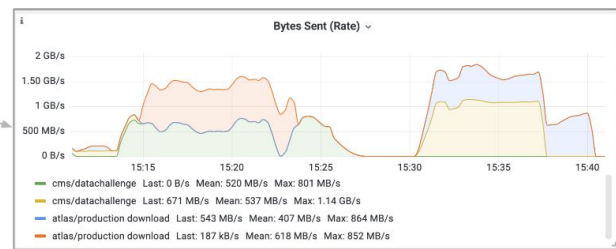
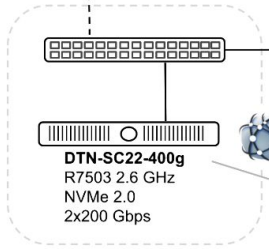
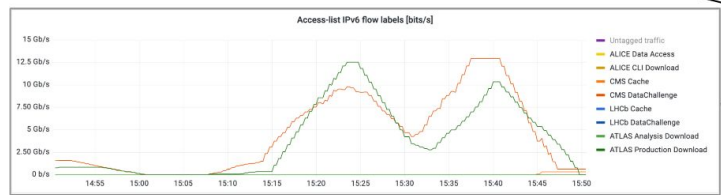
scitags.org

Network Flows and Packet Marking for Global Scientific Computing

HEP*i*X

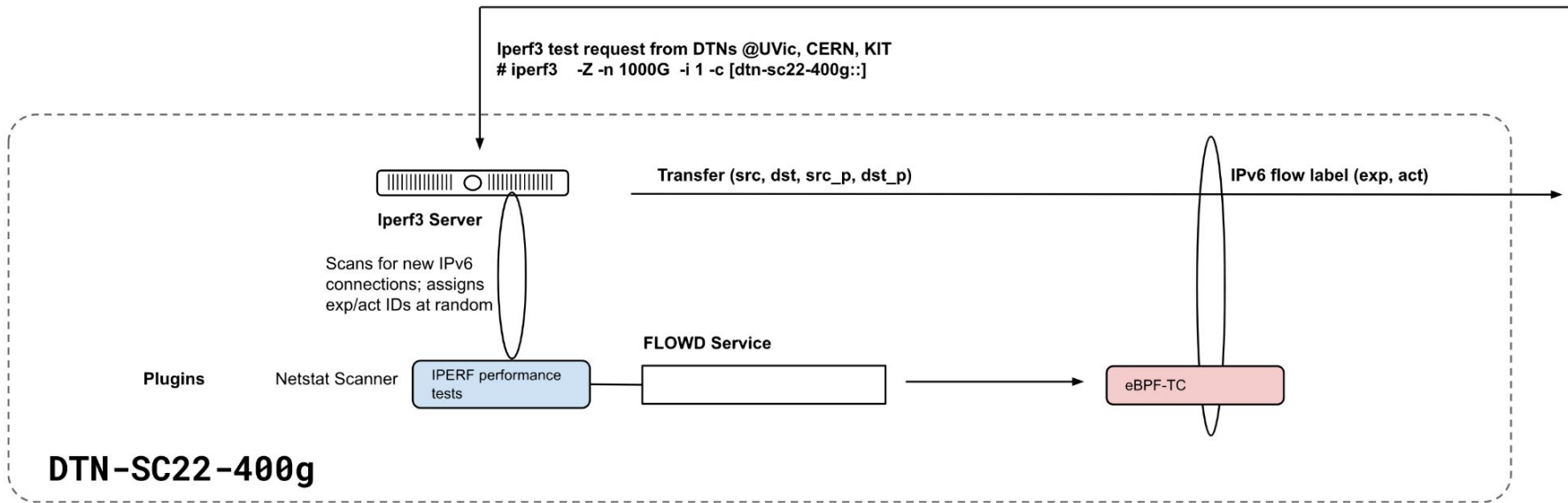


STARLIGHTSM



IPv6 Packet Marking with eBPF-TC and P4

Accounting



Programming Protocol-independent Packet Processors: P4 language

Language for programming the data plane of network devices.

- ❑ Define how packets are processed.
- ❑ P4 program structure: header types, parser/deparsers, match-action tables, user-defined metadata and intrinsic metadata.

Domain-specific language designed to be implementable on a large variety of targets

- ❑ Programmable network interface cards, FPGAs, software switches and hardware ASICs.



EdgeCore Wedge100BF-32QS

100GbE Data Center Switch

- ❑ Bare-Metal Hardware
- ❑ L2/L3 Switching
- ❑ 32xQSFP28 Ports

Data-Plane Programmability

- ❑ Intel Tofino Switch Silicon
- ❑ Barefoot Networks

Quad-Pipe Programmable Packet Processing Pipeline

- ❑ 6.4 Tbps Total Bandwidth

CPU: Intel x86 Xeon 2.0GHz

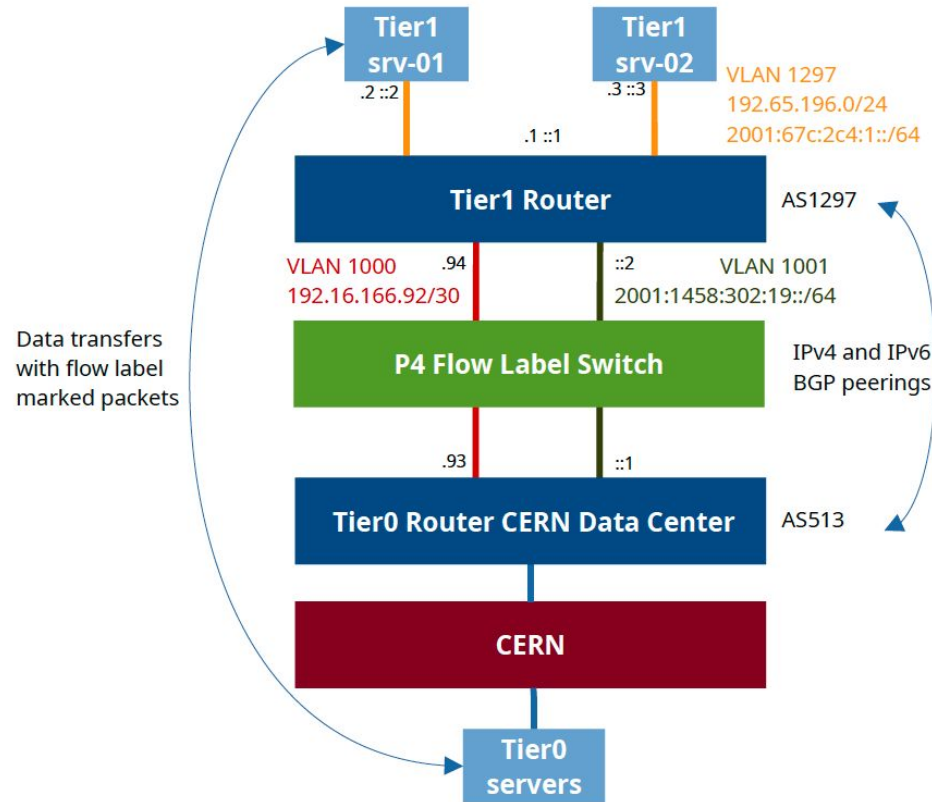
- ❑ 8-core/48GB/2TB SSD



Test: second approach at Ethernet level

Network configuration:

- ❑ Emulates a Tier 1/0 link
- ❑ Tier1/0 routers
 - ❑ IPv4/IPv6 BGP peerings
- ❑ Tier0 router
 - ❑ LHCOPN production border router
- ❑ Pure layer 2 bridges
 - ❑ VLAN 1000: IPv4 traffic
 - ❑ VLAN 1001: IPv6 traffic
- ❑ Tier0 servers
 - ❑ OpenStack product servers



P4 switch network configuration

Pure layer 2 bridges:

```
access-list acl_all_ipv6_flowlabels
  # Match <Experiment> and <ANY Application>
  sequence 10 permit all any all any all flow 131076 & 261884
  sequence 11 permit all any all any all flow 65540 & 261884
  sequence 12 permit all any all any all flow 196612 & 261884
  sequence 13 permit all any all any all flow 32772 & 261884
  # Match <Experiment> and <perfSONAR Application>
  sequence 20 permit all any all any all flow 131072 & 261632
  sequence 21 permit all any all any all flow 65536 & 261632
  sequence 22 permit all any all any all flow 196608 & 261632
  sequence 23 permit all any all any all flow 32768 & 261632
  # Permit the rest of the traffic
  sequence 30 permit all any all any all
exit
```

ATLAS <any>
CMS <any>
LHCb <any>
ALICE <any>

ATLAS <perfSONAR>
CMS <perfSONAR>
LHCb <perfSONAR>
ALICE <perfSONAR>

```
interface sdn1.1000
  description [VLAN ID=1000]
  bridge-group 1
  no shutdown
  no log-link-change
  exit
```

VLAN 1000 belongs to bridge 1

```
interface sdn1.1001
  description [VLAN ID=1001]
  bridge-group 2
  bridge-filter ipv6in acl_all_ipv6_flowlabels
  no shutdown
  no log-link-change
  exit
```

VLAN 1001 belongs to bridge 2
Filter IPv6 traffic at the
input based on the access-list
sentences

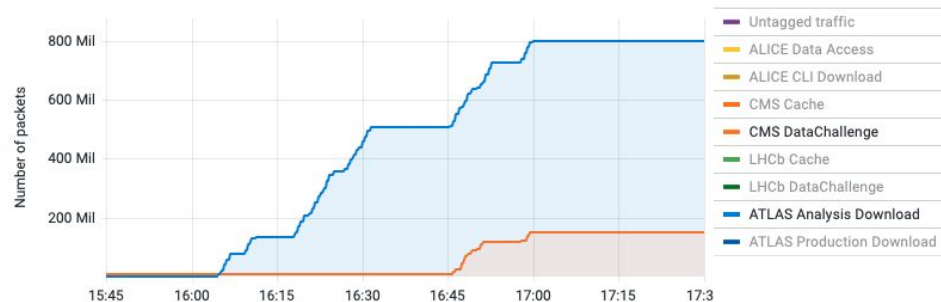
Statistics

The counters of the access-list of the P4 switch are exported to Prometheus DB and Grafana:

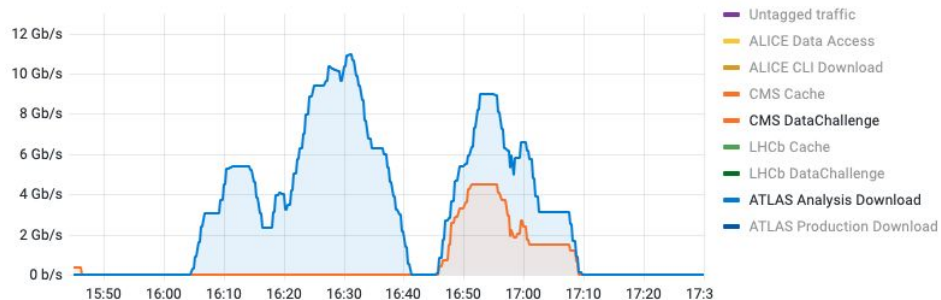
```
E513-E-YECWH-1#show access-list acl_all_ipv6_flowlabels
```

seq	txb	txp	rxb	rxp	last	timeout	cfg	
10	0+0	0+0	0+12374638771	0+8743031	00:03:02	00:00:00	permit all any all any all flow 131076&261884	ATLAS <any>
11	0+0	0+0	0+37019728635	0+24984028	00:02:30	00:00:00	permit all any all any all flow 65540&261884	CMS <any>
12	0+0	0+0	0+23940164205	0+15797973	00:02:00	00:00:00	permit all any all any all flow 196612&261884	LHCb <any>
13	0+0	0+0	0+18150017192	0+12017039	00:02:00	00:00:00	permit all any all any all flow 32772&261884	ALICE <any>
20	0+0	0+0	0+30346726207	0+200005622	00:01:29	00:00:00	permit all any all any all flow 131072&261632	ATLAS <perfSONAR>
21	0+0	0+0	0+25281078379	0+16663278	00:01:29	00:00:00	permit all any all any all flow 65536&261632	CMS <perfSONAR>
22	0+0	0+0	0+28556351375	0+19098806	00:00:58	00:00:00	permit all any all any all flow 196608&261632	LHCb <perfSONAR>
23	0+0	0+0	0+37078713993	0+25770785	00:00:26	00:00:00	permit all any all any all flow 32768&261632	ALICE <perfSONAR>
30	0+0	0+0	0+2715536713	0+1802921	00:00:26	00:00:00	permit all any all any all	

Access-list IPv6 flow labels [number of packets]



Access-list IPv6 flow labels [bits/s]



Interface sdn1 - VLAN 1001: IPv6 traffic [bits/s]



P4 Dashboard

General / P4 dashboard: EdgeCore 100BF-32QS programmable switch interfaces

2022-10-20 11:33:29 to 2022-10-20 16:29:37

interface sdn1

Interface sdn1 - VLAN 1000: IPv4 traffic [bits/s]



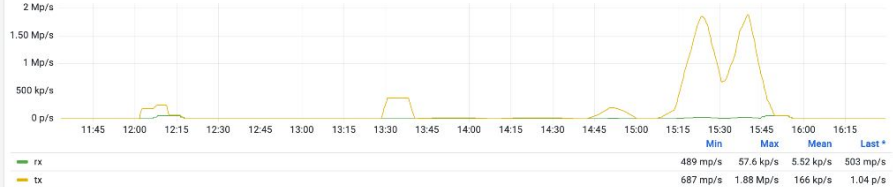
Interface sdn1 - VLAN 1001: IPv6 traffic [bits/s]



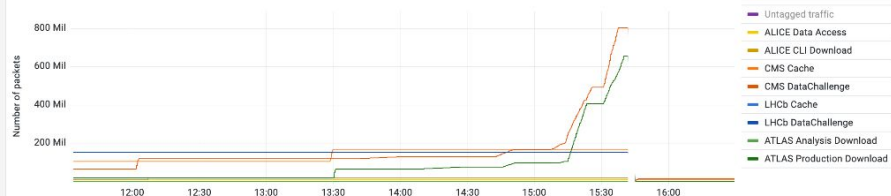
Interface sdn1 - VLAN 1000: IPv4 traffic [packets/s]



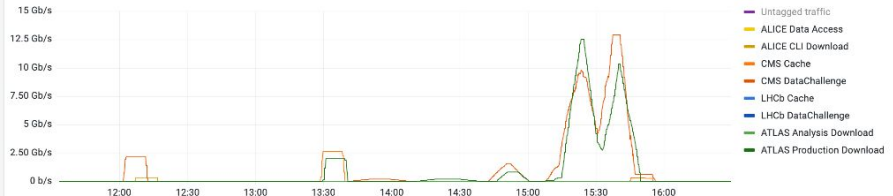
Interface sdn1 - VLAN 1001: IPv6 traffic [packets/s]



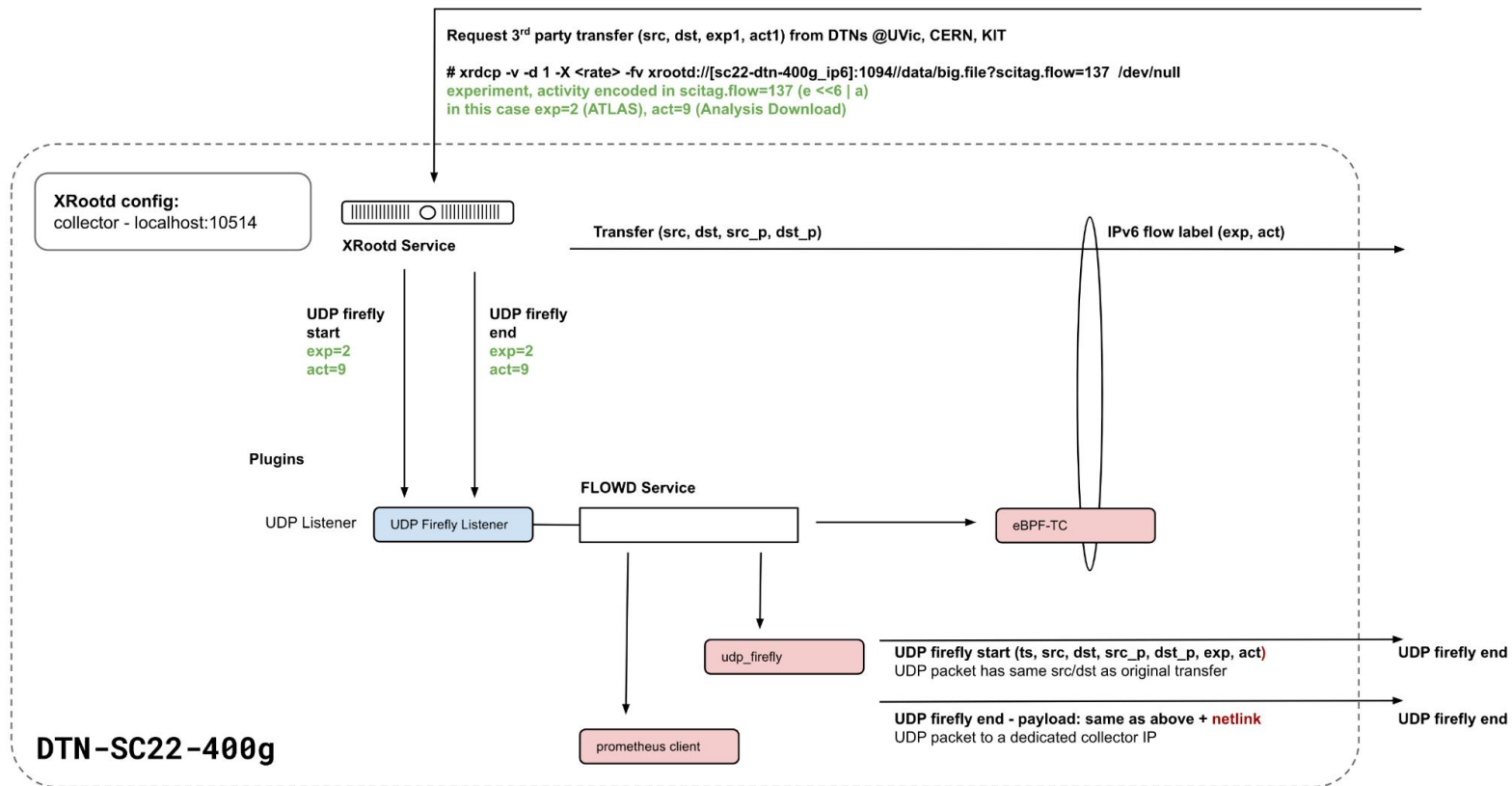
Access-list IPv6 flow labels [number of packets]



Access-list IPv6 flow labels [bits/s]



XRootD Packet and Flow Marking and Netlink



Netlink Dashboard

General / Scientific Network Tags (scitags) - Netlink Dashboard ☆ 🔍

