Research Networking Technical Working Group Status and Plans

Shawn McKee / University of Michigan, Marian Babik / CERN

Spring 2024 HEPiX Meeting, Paris, France

https://indico.cern.ch/event/1377701/timetable/#20240417

Apr 17, 2024

Research Networking Technical WG

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

Are meetings are available in Indico: https://indico.cern.ch/category/10031/

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

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 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**.

The LHCONE/LHCOPN meeting had presentations on Jumbo frames and BBRv3

- Discussion at the Catania meeting last week was interesting. ESnet high-touch did NOT see many retransmits, indicating that protocols like BBRv3 may not help much
- Jumbo frame results were mixed, depending upon which hosts were involved
- Our goal in this area is to try to use the available bandwidth as effectively as possible.
- We will continue to explore traffic / host tuning and optimization for WLCG and report at future HEPiX and LHCONE/LHCOPN meetings.

Network orchestration

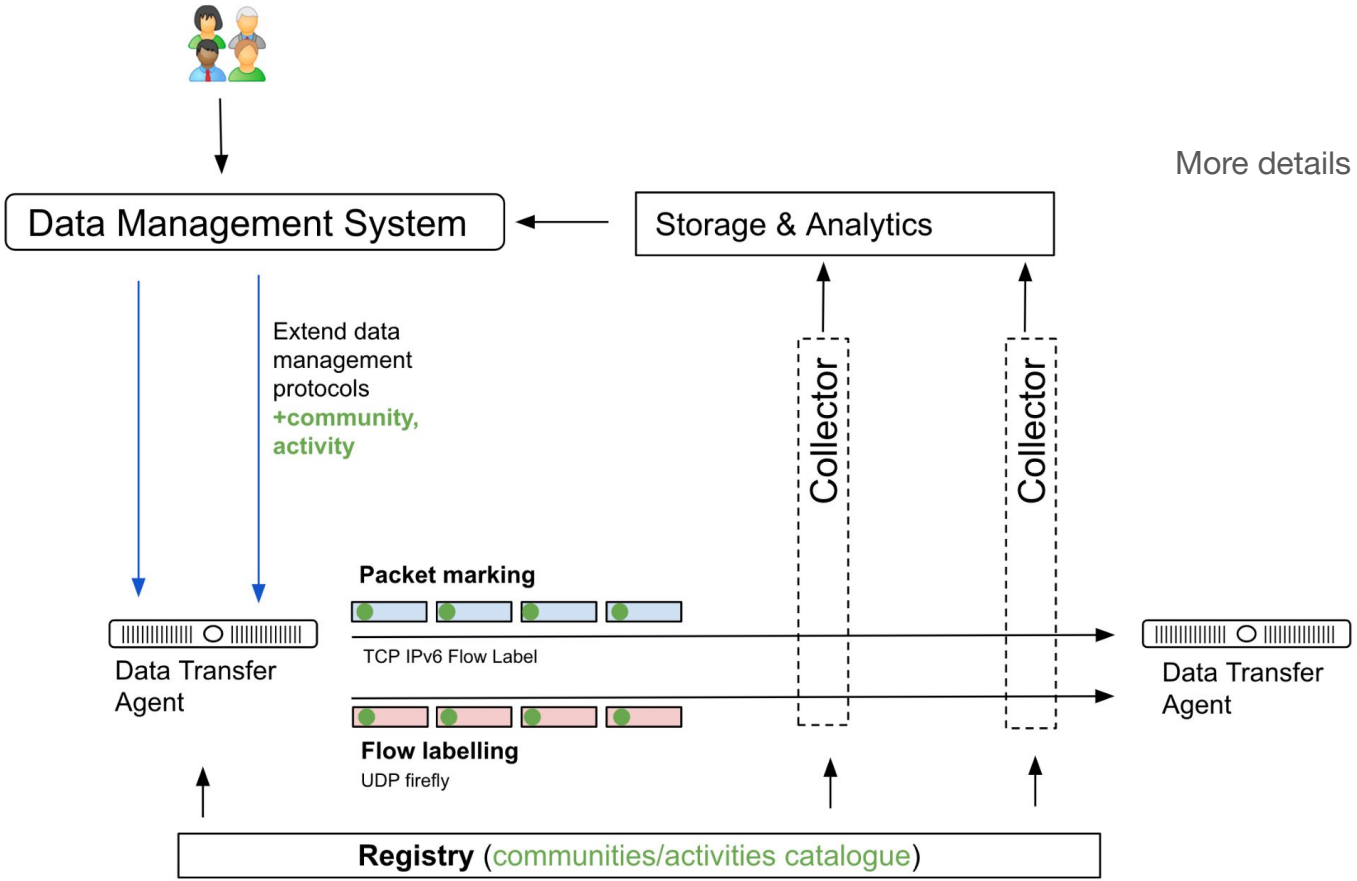
- The ways we organize our computing / storage resources will need to evolve.
- This area is being led by the **GNA-G** (Global Network Advancement Group; https://www.gna-g.net/) and is exploring many options for traffic engineering, resource management and network-application interfaces.
  - The **SENSE** 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.
- It is important that our infrastructure is able to support features the network can provide in the future.
  - To do this takes a significant amount of effort, not the least of which is working with the experiments to ensure their software stacks can benefit from traffic orchestration

Scitags

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

- **Goals**
  - Provide **standardised means of information exchange** on network flows between experiments, sites and network providers.
  - **Improve** experiments' and sites' **visibility** into how network flows perform within network segments.
  - Get insights into how experiments are using the networks and **benefit from additional data from the network providers**.
  - Make **network performance tuning and troubleshooting** easier and more effective by gaining insights into how different network configurations impact performance

How Scitags Work



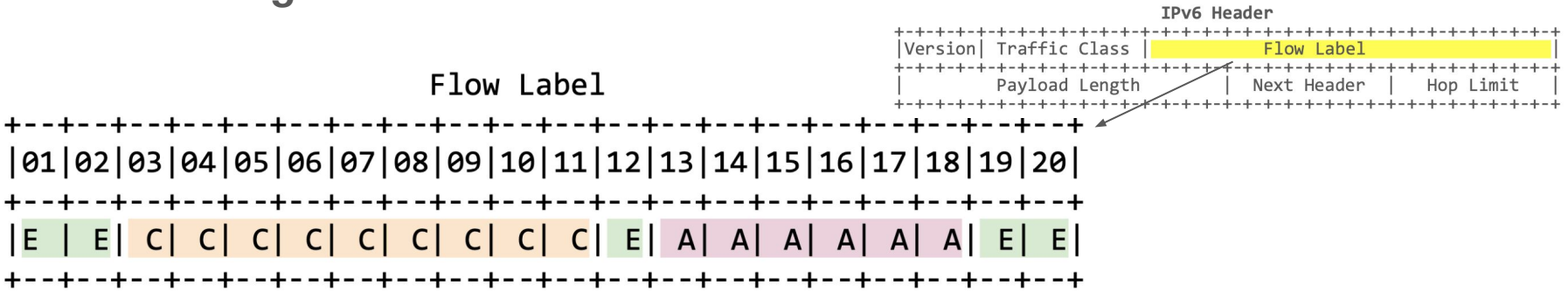
More details in CHEP paper

Scitags Framework Rationale

- Provide an **Open platform** that can be used by **any** data-intensive science community
- **Identify the owner (experiment) & purpose (activity) of the traffic**
- Define **standard(s)** for exchange of information between scientific communities, sites and network operators
  - Packet marking - encoding exp/activity directly in packets
  - Flow labeling - sending a separate UDP packet (firefly) with metadata
- Enable **tracking** and **correlation** with **existing network flow monitoring** and existing monitoring systems deployed by R&E networks
- Quantify global behaviour and analyse trade-offs, **at scale**

Technical Spec for Packet Marking

Packet Marking via the use of the IPv6 Flow Label



- (C) Community identifier: "Who are you affiliated with?"
- (A) Activity identifier: "What are you doing within your community?"
- (E) Entropy bits sprinkled throughout

IETF RFC-Informational Draft is available with more details

Started exploring HbH option as an alternative (eBPF-PDM, eBPF-extHeaders)

Technical Spec for Flow Labeling

- **Flow Labeling** via UDP packets (fireflies):
  - **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.
    - Works for IPv4 and IPv6; content is not limited (as long as it fits in a single frame)
      - Apart from exp/act we now have also usage (bytes sent/rcv) and RTT in fireflies

- The detailed technical specifications are maintained on a Google doc
- The document also covers methods for communicating owner/activity and other services and frameworks that may be needed for implementation.

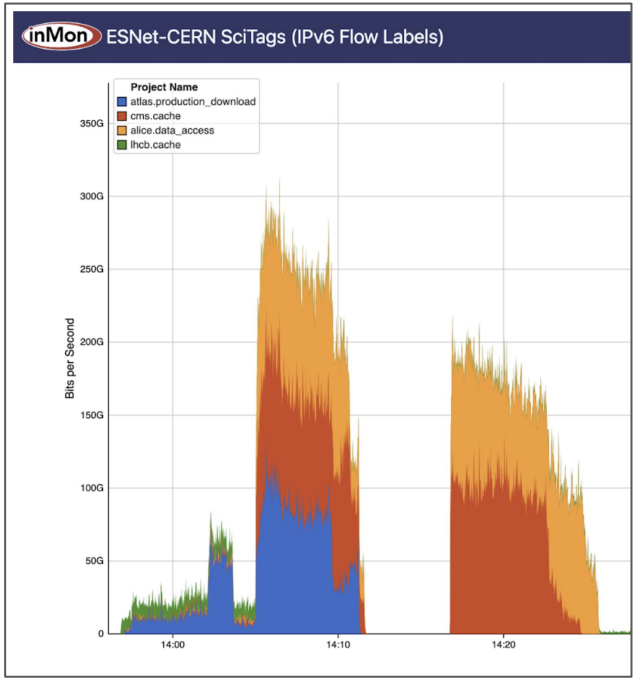
Collectors

- Different types depending on what is being collected
  - **HW/on-the-wire** to collect UDP fireflies and/or IPv6 flow label
  - **SW/network of receivers** - collecting UDP fireflies sent to them
    - Working on update to the current architecture to introduce a message bus - to interconnect different (N)REN collectors and also allow to subscribe
  - **SW/Collectors**
    - Site-collector - forwards fireflies via UDP, optional local storage
    - Regional collector - receives fireflies from sites, stores locally and publishes to message bus
    - Global collector - receives all fireflies (directly or via bus), global store
    - Experiments collector - subscribes to the bus for specific fireflies

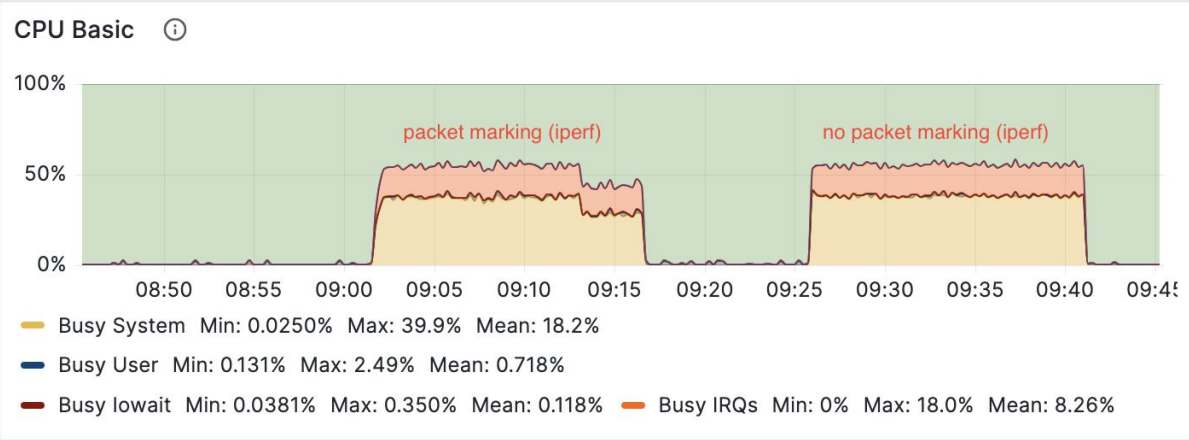
Packet Marking Demo

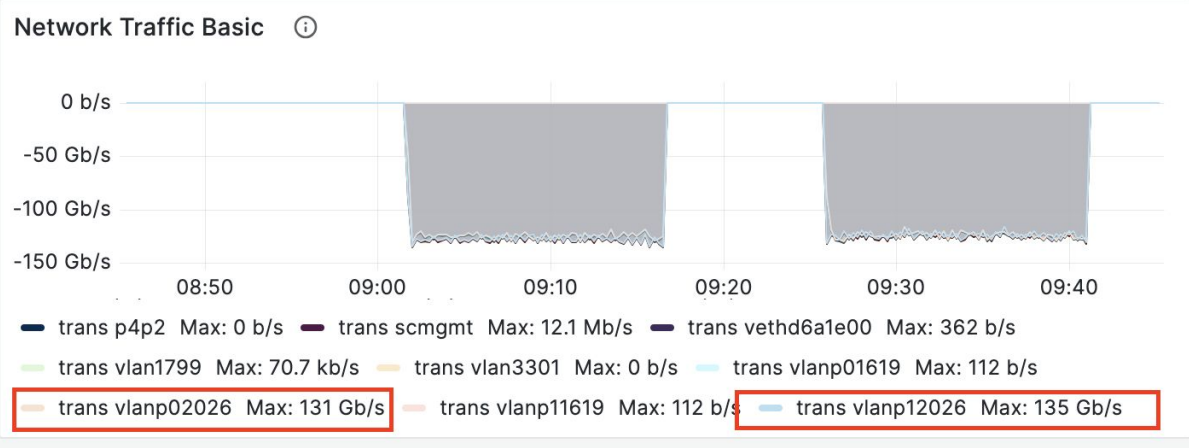
During Supercomputing 23 in Denver, we demonstrated a number of aspects of our packet and flow marking work.

- Showed **packet marking** at **300 Gbps** rates using **xrootd**/iperf3 (with just two nodes; using eBPF).
- Integration with **ESnet's High-Touch Service**
  - Analytics at the packet-level
- In collaboration with inMon, set up packet collectors via sflow and demonstrate **real-time monitoring of flows by community/activity**.
- Demo was run in collaboration with Starlight, ESnet, KIT, University of Victoria, University of Nebraska and CERN



Packet and Flow Marking Demos

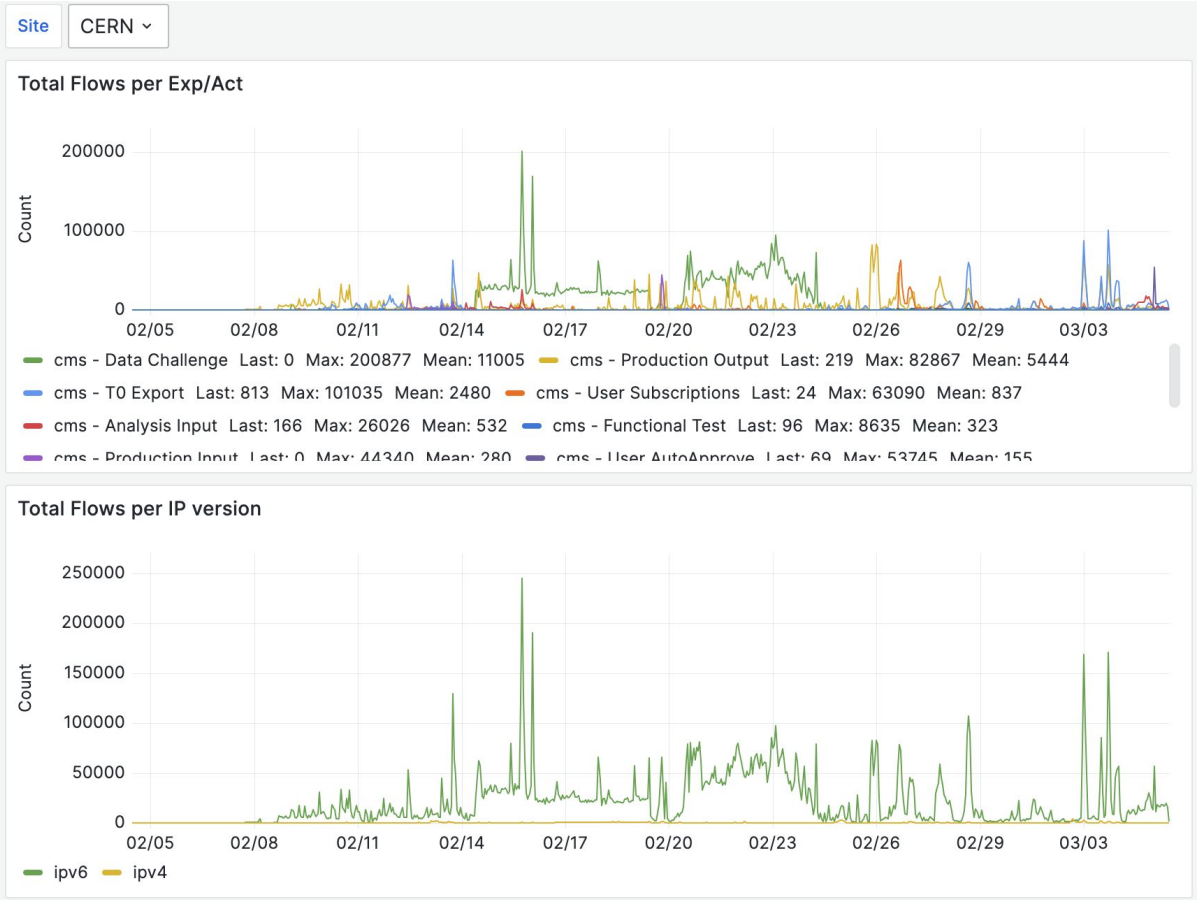




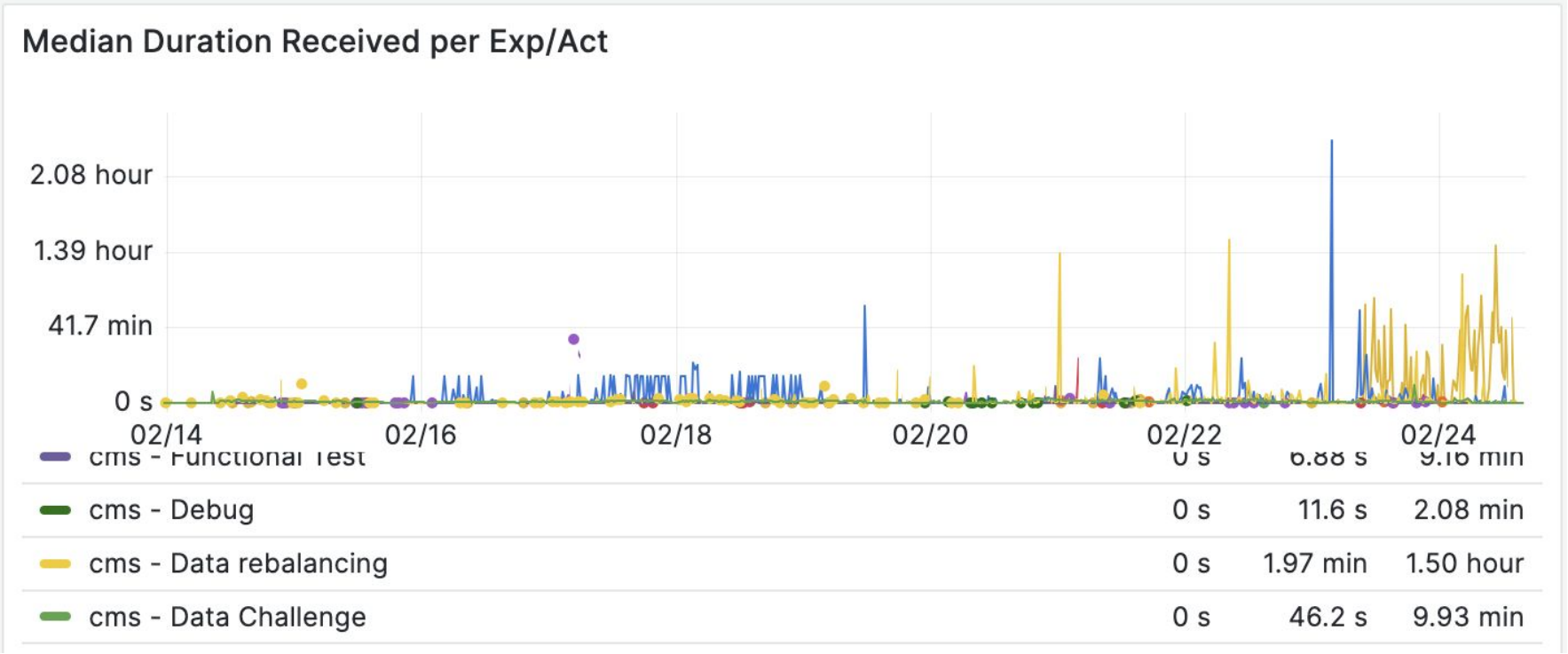
Data Challenge 24

- **Scitags Deployment**
  - 80% of EOS CMS (production), UNL production storage
  - Flow labeling functionality (fireflies)
- **Results**:
  - **Confirmed the capability to propagate Scitags all the way to the storages (for both ATLAS and CMS)**
  - Sending fireflies (from XRootd, EOS storages)
  - **Collection and visualisation at ESnet collector**
    - Results shown in live dashboard
- Issues:
  - We hit an issue with xrootd crashing when receiving scitags http headers
    - This had impact on ATLAS testing and availability of the ATLAS Xrootd storages
  - The issue was fixed quickly but we were unable to rollout (as DC was already running)

CERN EOS CMS plot showing split by experiment/activity and IP versions



CERN EOS CMS
Median duration of flows split by Exp/Activity
Shows duration of DC flows was quite short wrt. production/rebalancing



Current Status

Implementation status:

Propagation:

- **Rucio** supports Scitags from 32.4.0
- **FTS/gfal2** support Scitags from 3.2.10/2.21.0

Storages:

- **XRootD** provides Scitags implementation (from 5.0+)
- **EOS** provides Scitags support from 5.2.19+
  - Working on a project for production rollout at CERN (for WLCG)
- **dCache** prototype exists, roadmap for release pending
- Also working with StoRM and Pelican

Collectors:

- Production deployments at ESnet and Jisc

Summary

- Scitags (flow labelling) ready for production
  - Expecting sites and experiments will gradually enable it during this year
    - **ATLAS, CMS and ALICE ready to enable in production**
    - Sites ramp-up can be quick once it starts
    - Plan to enable fireflies at CERN T0
  - SW/Collector network will need to be ready and scale
    - Network providers are encouraged to deploy a collector to benefit from the initiative
  - Scitags facilitate collaboration with experiments
    - Reporting of issues and follow up becomes easier
- Significant progress in packet marking R&D
    - Will benefit from flow labelling deployment and production

Finding More Information: https://scitags.org

Code

Technical Spec

Mailing List

Presentations

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

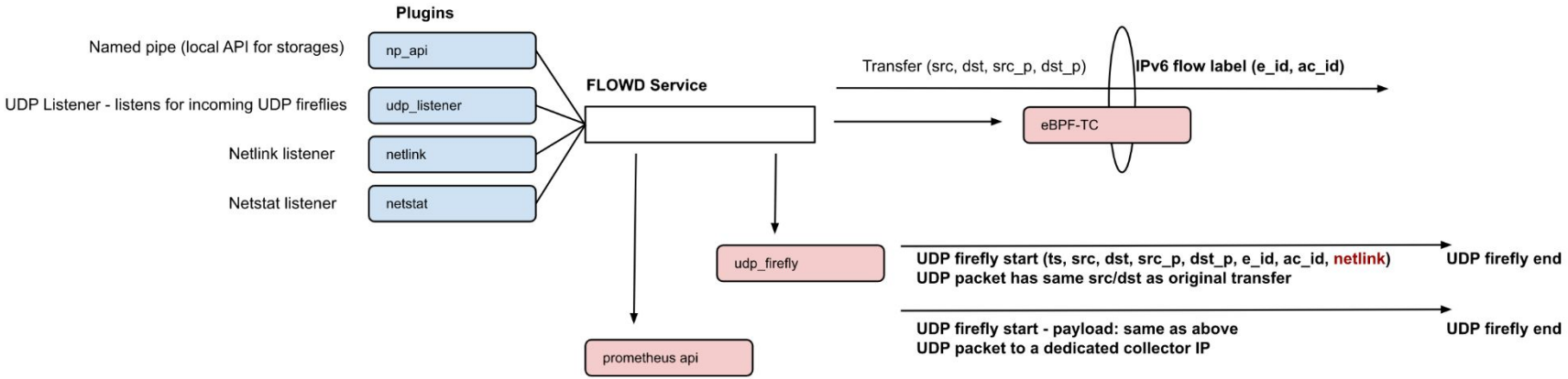
Questions?

Questions, Comments, Suggestions?

Backup slides

Flowd Service

- Flow and Packet Marking service developed in Python



- Plugins provide different ways get connections to mark (or interact with storage)
  - New plugins were added to support netlink readout and UDP firefly consumer
- Backends are used to implement flow and/or packet marking
  - New backends were added to mark packets (via eBPF-TC) and expose monitored connection to Prometheus

FTS & XRootD

FTS and XRootD are key to reaching full potential in programmable networks

XRootD already provides SciTags implementation (from 5.0+)

- Enables using SciTags by R&E networks analytics (ESnet6 High-Touch)
- Currently looking for sites that would configure/test this in production

FTS/gfal2 needed to propagate SciTags to storages

- Extensions proposed for XRoot and HTTP-TPC

FTS as a transfer broker is key component for NOTED

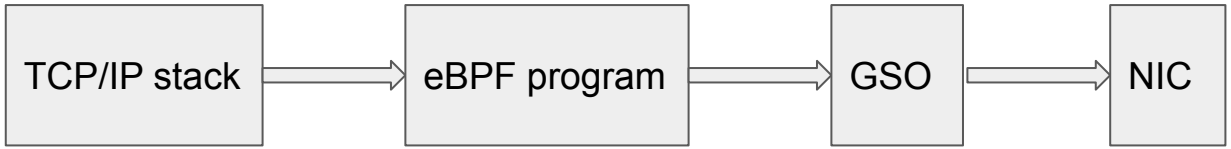
- Understanding where/when on-demand network provisioning is needed
- Combined with analytics to determine duration, capacity, etc.

Programmable networks can be beneficial for FTS and XRootD to get better network performance, flexibility and monitoring

Flowd: Packet Marking via eBPF-TC Backend

- eBPF is a general-purpose RISC instruction set that runs on an in-kernel VM; programs can be written in restricted C and compiled into bytecode that is injected into the kernel (after verification)
- Can sometimes replace kernel modules
- eBPF-TC programs run whenever the kernel receives (ingress) or sends (egress) a packet

Egress path:



- The flowd backend maintains a hash table of flows to mark. The plugin sends the backend (src address, dst address, src port, dst port); this is used as the key in the hash, and the flow label to put on the packets is the value
- Each packet is inspected, and if the attributes match an entry in the hash, the corresponding flow label is put on the packet

NOTE: 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.

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