Problem Statement

• The global Research & Education network ecosystem is comprised of hundreds of international, national, regional and local-scale networks.
Problem Statement

• While these networks all interconnect, each network is owned and operated by separate organizations (called “domains”) with different policies, customers, funding models, hardware, bandwidth and configurations.
Problem Statement

• This complex, heterogeneous set of networks must operate seamlessly from “end to end” to support science and research collaborations that are distributed globally.
Problem Statement

• In practice, performance issues are prevalent and distributed.
• When a network is underperforming or errors occur, it is difficult to identify the source, as problems can happen anywhere, in any domain.
• Local-area network testing is not sufficient, as errors can occur between networks.
Problem Statement: Hard vs. Soft Failures

- “Hard failures” are the kind of problems every organization understands
  - Fiber cut
  - Power failure takes down routers
  - Hardware ceases to function

- Classic monitoring systems are good at alerting hard failures
  - i.e., NOC sees something turn red on their screen
  - Engineers paged by monitoring systems
Problem Statement: Hard vs. Soft Failures

- “Soft failures” are different and often go undetected
  - Basic connectivity (ping, traceroute, web pages, email) works
  - Performance is just poor

- How much should we care about soft failures?
Elephant Flows Place Great Demands on Networks

Physical pipe that leaks water at rate of .0046% by volume.

Network ‘pipe’ that drops packets at rate of .0046%.

Result 99.9954% of water transferred, at “line rate.”
Result 100% of data transferred, slowly, at <<5% optimal speed.

Through careful engineering, we can minimize packet loss.

essentially fixed
determined by speed of light

maximum segment size
\[ \frac{1}{\sqrt{\text{packet-loss rate}}} \]

round-trip time
Soft Failures Cause Packet Loss and Degraded TCP Performance

Throughput vs. Increasing Latency with .0046% Packet Loss

With loss, high performance beyond metro distances is essentially impossible.
public perfSONAR Servers (Sept 2016)

- Over 2000 publicly registered servers
  - Equal number of non-registered servers?
- ESnet: 50
  - mostly 10G, includes a 40G host in Boston
- GEANT: 22
- Internet2: 3
- Some other top deployments:
  - Onenet (24), AMPATH (8), bc.net (10), RNP (8), Canarie (13), kreonet (14), NERO(12), AARnet (19), JGN (17), CENIC (5), KANREN (5)
More perfSONAR Statistics

- Total Hosts: 2001
- Total Domains: 413
- 75% are running latest version (auto-update)
- 7% are running a version < 3.5 (end-of-life, orphan hosts?)
- 27% of the hosts have an IPV6 Address
- 38% are .edu hosts
- 75 total top-level domains, 940 domains
More perfSONAR Stats

• 95% RHEL/CentOS; 5% Debian/Ubuntu
• 10% are VMs
• 40% using jumbo frames
• NIC Speed:
  – 1 Gbps  49.06 %
  – 10 Gbps 44.17 %
  – 40 Gbps  2.88 %
  – 100 Mbps 2.32 %
perfSONAR 4.0

• perfSONAR 4.0 Feature Tour
  – Andy Lake
  – 9:00 AM

• pScheduler Deep Dive
  – Mark Feit
  – 9:30 AM
What comes next?

• perfSONAR Steering Group is engaged in a project to define strategic plan for post-perfSONAR 4.0 efforts
• Four basic themes are being considered:
  – Operations efficiency
  – Automation of configuration, execution, & analysis
  – Performance of the cloud
  – Care and feeding of the open source effort
Operations Efficiency

• Reduce operations effort required of network operators and data intensive science collaborations
  – Easy deployment of ephemeral pS Nodes: perfSONAR containerization (e.g. Docker)
  – Automate log analysis to identify key signatures
  – Improve dashboard visualization
  – Enhance mesh configurations
Automation of Configuration, Execution, & Analysis

- **Configuration**
  - Automatic test configuration
    - Significant progress in perfSONAR 4.0
  - Automatic node discovery / insertion into test messages

- **Execution**
  - Auto-detection of problems along the end-to-end path
  - Automatically run additional tests

- **Analysis**
  - Anomaly detection tools
  - Intelligent alarming (e.g. adaptive via machine learning)
  - Archives for research data
Performance of the Cloud

- Cloud services and cloud based workflows are important going forward.
  - To our researchers
  - To our labs and universities
- Performance monitoring needs to be automated and ‘ephemeral’ till the cloud workflow is active
- Support a targeted popular use case
  - e.g.: perfSONAR Amazon EC2 deployment
  - Showcase what is possible
  - Research how to solve the problem more generally
- Build towards a more general solution in follow-on releases
- Observation: Increasing need / use of perfSONAR on VMs
  - Inform the user of the limitations of what can be concluded
Care and Feeding

- Expand new offerings
  - pScheduler is a new offering (1.0 release) which will prompt a new iteration of community ideas
- Security challenges, vulnerabilities, and inefficiencies
  - e.g. Expand unit testing
  - Review, refine, and reduce inefficient code components
- Support evolving default operating systems
  - e.g. CentOS6 -> Centos7, Debian 7 -> Debian 8 -> Debian 9
- Preferred API interfaces, middleware keep evolving
  - e.g. SOAP/XML -> REST/JSON APIs
  - message passing architectures – e.g. RabbitMQ
- Custom Code Replacement
  - Remove perfSONAR custom code with well-supported packages as they become available
    - ELK (Elastic search, Logstash, Kibana)
    - ESnet’s React Timeseries Charts
    - IU’s Time Series Data Service (TSDS)
- Small Nodes
  - Emerging technologies are rapidly evolving
  - Maintain a current snapshot of best offerings
Community Milestones & Feedback

• Milestones
  – perfSONAR 4.0 – RC1 is out this week
  – perfSONAR 4.0 – RC2 expected in October
  – perfSONAR 4.0 out in November

• Feedback
  – What do you need from perfSONAR post 4.0?