

CPR: Campus-Wide Network Performance Monitoring and Recovery

Warren Matthews, Russ Clark, Christopher Hood and Chris Kelly.

The Georgia Institute of Technology, Atlanta GA, USA
Contact: warren.matthews@oit.gatech.edu

Abstract. One of the biggest challenges that network managers face is the need to quickly identify problems and determine their true cause. In many cases, the true problem is not with the network itself but with applications, end-hosts or their configuration. Tracking down these problems can be extremely time consuming in a large, complex network. The Campus-wide network Performance monitoring and Recovery (CPR) project is an active measurement system designed to address these diagnostic problems. CPR provides quantitative data about network availability and network application performance from more than fifty monitoring stations in a large campus network. It is successfully being used to resolve problems on a production campus network used by thousands of faculty, staff and students. The data gathered by this project and the ongoing results are being made available to others to facilitate further research in distributed network measurement¹.

1 Introduction

As networks have grown larger and more complex it is increasingly difficult to perform the necessary troubleshooting and diagnostic functions that are required of the network administrator. Further, in many cases, the symptom that a user attributes to the network may in fact be a problem of host or application software that is out of the reach of the network administrator to assess or properly correct, yet the burden of proof is with the network administrator to prove it is not a network problem. In these cases, the end user experiences a great deal of frustration while the parties involved assign blame and perhaps all too often, never solve the actual problem.

The Campus-wide network Performance monitoring and Recovery (CPR) project is designed to operate in such an environment and to improve on the ability to correctly troubleshoot network and network-related problems. It is designed for a large (50,000 plus) node campus network with a diverse group of users, and perhaps more importantly, a diverse group of network and system administrators, none of whom have complete responsibility or control over the whole.

¹ The authors wish to be considered for the dataset award.

The system is deployed among buildings linked by the campus backbone network and maintained by the central network services team at the Georgia Institute of Technology (GT).

CPR provides a platform for deploying and using the best available active measurement tools and for integrating and reporting results from those tools. A principal design goal is to establish a baseline from which performance problems can be identified. The resulting analysis is available to a range of consumers including network administrators, system and application support personnel, and the end users. An important theme of the project is to empower the users to help diagnose problems, real or merely perceived, for themselves.²

The CPR project is consistent with much of the ongoing work in Internet performance measurement and is seen as a necessary extension of this work to the edges. The objective is to bring measurement tools and data that are compatible with the emerging international monitoring infrastructure [1] out into the campus environment.

Another important part of the CPR effort is the opening up of the data being gathered to other administrators and researchers. To this end, the centralized data architecture has been designed to facilitate analysis work both for researchers at GT and beyond.

2 Architecture

The CPR project was born of the simple, logistical problem of needing a way for campus backbone network administrators to gain local perspective on the networks in the buildings for which they are responsible. Historically, this was done one network at a time as problems arise by asking the local building system administrator to create an account or run a set of tests. The problem has been further compounded over recent years with the widespread deployment of firewall devices on the campus that further isolate the administrators from system on the networks that they administer.

Instead of relying on the availability of a local administrator to grant occasional access, the idea was to deploy a permanent set of network administrator test machines. This is similar to the Abilene Measurement Infrastructure (AMI) [1] or PlanetLab [2] but within the campus network.

Active tests run on the CPR hosts so administrators have data already available when a problem occurs and are able to recognize a problem before the users complain. One of the original design goals was to have a standard software distributions on the CPR nodes and to automate the configuration and data analysis from a central device. Currently, the CPR nodes run a standard collection of active measurements including pings, traceroutes and network service measurements.

² This is consistent with the theme of several other projects on the GT network focused on self-service. For instance, being able to ask the question “Which firewall rules are affecting me right now?” or “What are the bandwidth limitations of my connection?”

Figure 1 shows the set up of the CPR system. Numerous tools make measurements and store them locally on the CPR host. Periodically the data is copied to a staging area and uploaded to the central server. The data is copied into a database. The data is then available for graphing, analysis, or access by a 3rd party via web services.

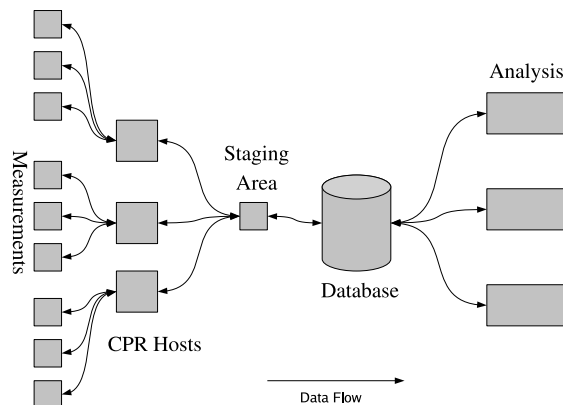


Fig. 1. The Software Architecture. Numerous measurements are collected in a central database and available for graphing, analysis or can be accessed via web services.

2.1 Distributed Monitoring with Centralized Analysis

While the individual CPR nodes provide deep access to the network performance from very near the customer's perspective, each individual node can only see a limited view of the overall performance picture. Because of this limited view, it was decided that the CPR architecture would also include a centralized analysis server that would gather measurement data from the distributed nodes and analyze it together. This gives the obvious advantage that widespread problems can be easily identified and distinguished from problems that only effect a single subnetwork.

2.2 Current Deployment

There are 54 CPR hosts currently deployed on the Georgia Tech main campus. In addition CPR hosts have been deployed at satellite campuses in Savannah, Georgia, and Metz, France.

2.3 Measurements

The smokeping [3] and nagios [4] tools are installed on all CPR hosts. Smokeping measures and creates graphs of the round trip delay between hosts. Smokeping runs every 5 minutes to measure the round trip delay between each pair of hosts. This high granularity leads to rapid indication of problems and their scope. Nagios is used to track the TCP response times to central services such as WWW and email. In addition, nagios is tailored for each CPR host to monitor local services such as file servers and DHCP offers.

Traceroute is also run between all CPR hosts. Within campus, it is unlikely that the route will change. However, tracking it provides a sanity check and should be an early indication of serious problems on the backbone if anything does appear to change.

The Pathload tool [5] has been tested on some CPR hosts. It provides information on the available bandwidth. It may be a real-time, low-impact indication of high bandwidth utilization, such as with a denial of service attack.

The Internet2 e2epi tools; BWCTL, OWAMP and NDT [1] are also installed on some hosts. However, because of problems with the accuracy of NTP and technical issues with the patched kernel, these tools are still being evaluated.

The CPR hosts can easily drive 100 Mbps of network throughput across campus. However, the CPU usage required for this impacts ping measurements. This impact looks like high jitter on the network, which does not exist. The Internet2 end-to-end group experienced similar problems running their BWCTL and OWAMP tools on the same hardware. Their solution is to deploy the tools on separate hardware which is not a viable option for CPR: Instead, we plan to develop a meta-scheduler to ensure that tests run at different times.

From the initial concept, it was planned that CPR would be a platform for other tools and projects. The security tools nessus and nmap are installed on all CPR hosts. The information security team can monitor activity within the subnet, behind the firewall. Also their monitoring can be distributed and not have to run for very long periods of time, impacting the network.

2.4 Datasets

Data collected by the CPR hosts is available using web services. The CPR web service interface [6] is based on the interface created for Abilene [1] and AMP [9] data by one of the authors of this paper. The services implement the schemas defined by the Network Monitoring Working Group (NM-WG) within the Global Grid Forum (GGF) [7]. Authorized users may access the database using the SOAP or XMLRPC protocols. Limited ability to request additional measurements is available. One of the design goals was to make CPR useful to the researcher as well as the different groups of administrators and end users on campus.

2.5 Management

As the number of CPR hosts increase, the challenge of maintaining them becomes harder. Each time a new host is added, or even a minor configuration change has to be made, all the hosts have to be touched. Often several files will be involved. The solution is to create a tool capable of distributing changes from a central server to all the deploy hosts. Existing tools developed within GT were considered but none of them were flexible enough to meet our needs. In addition, the maintenance of the infrastructure is a research project in itself, and it was decided to experiment with the emerging standards and develop a service oriented architecture.

2.6 Similar Projects

There are numerous measurement tools and monitoring infrastructures. Well known projects such as PingER [8] and AMP [9] have gathered a tremendous amount of data on links around the world. However, in terms of a campus network, the authors believe CPR is the most comprehensive deployment.

3 Analysis and Visualization

The visualization tool can be accessed by authorized users through any web browser. It uses PHP to dynamically create all of its pages, and therefore all of the work is done on the server side and will be independent of the user accessing it. The tool currently only graphs ping times obtained from Smokeping between a specific host machine and target machine, however it was designed to handle any type of data set and development continues. The index page gathers all the node names from the database and allows the user to select any number of host and target machine pairs they desire. Once they submit their request, the tool will create a database query and use the results from that query to create a graph.

The graph is rendered using the latest version of PHPlot [10], which had to be modified to address certain functionality requirements. PHPlot uses the GD graphics library to create a PNG image, which is then placed into a web page along with buttons that allow the user to scroll or zoom within that graph. The scrolling and zooming functions were added to help the user focus on any particular time frame they desired, which allows viewers to have a visual representation of how much and how often ping times were changing. These graphs and statistics, along with live data, will be used to generate a "Top N Problems" list for the network technicians to use on a daily basis, catching problems that may otherwise go unnoticed until a user submits a trouble ticket.

Figure 2 shows real data gathered on July 17th, 2005 from 4:45 AM to 5:30 AM. It shows that the CPR machine in 490 Tenth Street (490) was experiencing some delays while accessing the CPR machine in the Ajax building (Ajax) but was consistent while accessing the CPR machine in the French building (French).

It also shows that French was able to access Ajax very consistently. This graph would quickly show a network technician that the problem is related to the link between 490 and Ajax.

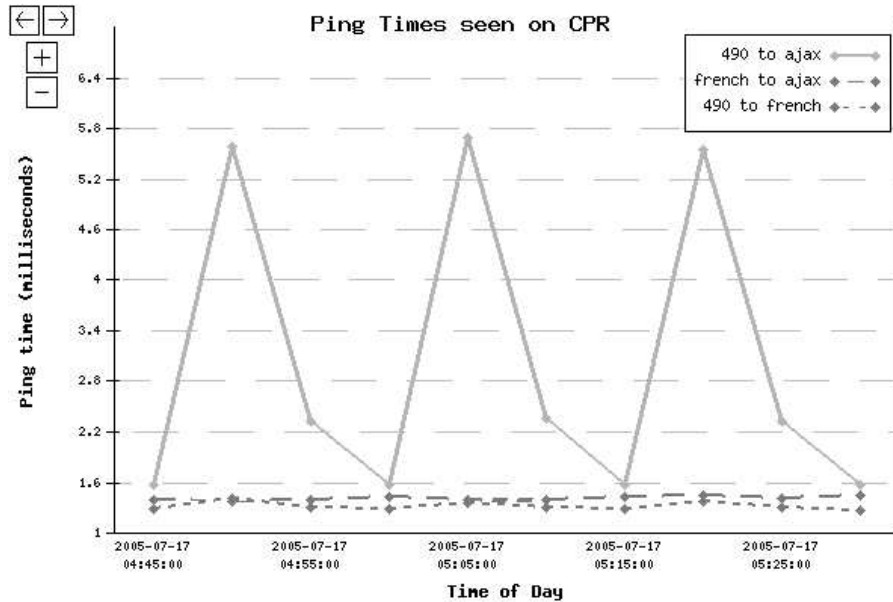


Fig. 2. A graph of real data gathered between CPR hosts. The ability to compare performance from different views allows the network administrator to quickly determine the common point of failure. Note the control buttons in the top left corner. The user can zoom in or out, or pan forward or backward without creating a new graph.

Further goals for the visualization tool include adding the ability to graph multiple data sets and the ability to combine multiple graphs into one graph or one PNG image. One use of combining multiple graphs into one would be having the ability to instantly view the changes in ping times along with changes in traceroutes, if they exist, which would provide important information needed for troubleshooting a problem.

Hundreds of megabytes of performance data are collected daily and stored in a central database. This data can be queried directly by the reporting tools, but due to the massive amount of data it is often too computationally expensive to do complex calculations on-demand. To speed up analysis and spotting trends over time, a computing thread regularly runs to compute statistics about the data in daily, quarter day, and cumulative intervals. Once on a granularity finer than 6 hours, the reporting tools will need to access the raw data and make the calculations itself.

The focus of trend reporting so far has been with Smokeying data. Currently CPR computes 9 different statistics. Straightforward statistics such as minimum, maximum, and mean can be used for finding general peaks and trends in the data. However, to allow for easier identification of problems, CPR calculates statistics like the standard deviation and the interquartile range and mean. The standard deviation indicates how stable the performance the link between a pair of hosts is. A change in the standard deviation over time can indicate a change in performance that might otherwise go unnoticed because of outliers. The interquartile mean can be compared to the mean to determine if outliers are weighted in a specific direction and (when compared with the maximum) how many of them there are. A change in the difference between the mean and interquartile mean independent of the outliers during a given time period is usually indicative of a problem.

4 Experiences

CPR has quickly become a tool used by the members of the network group responsible for trouble-shooting and fixing problems. CPR has helped solve several issues.

In one case, users reported they could not exchange email with collaborators at another University. Another group had investigated the issue before giving up and passing the ticket to the network group. A CPR host was already deployed on the GT subnet. Typically the network group would not have direct access to the subnet and this scenario would lead to finger pointing, particularly at the firewall. However, using the CPR node, the issue was pinpointed within 20 minutes. An ACL in another network between the Universities was blocking traffic from the specific subnet where the mail server was connected. However, it took several weeks to identify the right contacts at the network to have the ACL removed. In this case, there probably isn't a standard test that could be regularly performed to spot this problem in the future. The value was having a CPR box in the right place. A centralized system would not have provided quantitative data to help resolve the issue. In fact, it would not have shown any problem at all.

In another case, users in a particularly building reported losing connectivity to their home directories housed on a server in another building. A CPR host was deployed. Immediately the CPR host provided evidence that the server had been compromised and was running unauthorized software. It was removed from service, cleaned and returned to service. The disconnections ceased. In this case, regular port scanning would have revealed the server was compromised. The CPR team is working with the Information Security group to run regular scans.

In another example, one particular user complained about poor throughput between two buildings. CPR hosts were deployed on both subnets. CPR confirmed poor throughput, eliminating the possibility of mis-configuration on the users host. Coincidentally, errors were observed on the switchport leading to the diagnosis of a duplex mismatch. In this case, with only one user complaining,

the assumption would have been a misconfigured host or application. Regular scanning for duplex mismatches would have revealed the problem sooner. The CPR team is exploring the use of passive tools and the NDT tool to check for duplex mismatches.

5 Further Work

CPR was designed both as a production quality trouble-shooting tool and as a platform for research and development in network monitoring, analysis and reporting. In this role there are several ongoing efforts to extend the system and address open issues.

5.1 Further Deployment

At the time of this writing, the Georgia Tech campus network consists of 176 network buildings with over 1600 switches and 50,000 ports. The 54 CPR hosts currently deployed represent approximately 30% of the network buildings and less than 10% of subnets. The goal is to deploy a CPR host in every building and in some cases, every network stack. In addition to the current stream of donated hardware, the team is investigating the use smaller footprint devices like the Mac mini computer and embedded, diskless devices like a Linksys AP running Linux. The current plan is to specify a CPR node as part of the installation of any new network equipment. The team is also working on the logistics of deploying CPR hosts at global partner campuses in Singapore, Shanghai and India.

5.2 Development

There are still many opportunities to be explored with the CPR project and ongoing development is limited primarily by available time and resources. Some specific priority areas that the team will be exploring are:

- **Passive monitoring.** Passive monitoring tools such as Netflow and NTOP can provide important additional information regarding the actual experiences of network users. The current CPR devices are not capable of supporting both full-time packet capture and perform accurate active measurements. We are exploring ways to incorporate passive data gathered from other devices or Netflow data from the switches themselves into the analysis tools for CPR.
- **Personalized Interfaces.** The tools are already being developed to present specific monitoring data to users based on a login and user profile. We are also interested in developing a user-centric interface that demonstrates the personalized set of performance reports that are currently effecting a user. For instance, the "myCPR" report would identify bottlenecks or give the all clear to a user based on the current subnetwork, current applications they are using and the hosts with which they are currently exchanging data.

- **User Supplied Data.** While the CPR nodes themselves are able to bring data from close to the user’s perspective, they are not actually monitoring network services from the client systems. Further, the number of client systems is hundreds of times greater than what could ever be deployed solely for monitoring. For these reasons it is useful to enable simple, portable collection tools that run directly on client systems and can be used to provide valuable user input to the system. Several ongoing projects at GT, including the NETI@home project [11], are looking at gathering monitoring data from clients. The CPR team is working on API mechanisms to bring that data into the CPR data services as part of a community support initiative.
- **Data Management.** The large amount of data gathered from the CPR monitors presents difficult challenges in how to store and process it efficiently. It is necessary to archive some portions of the data but keep it accessible to support researchers doing long term studies. Continued work is necessary in data storage and representation techniques to improve this part of the architecture.

Acknowledgements

The authors would like to thank Constantinos Dovrolis and Matt Sanders for their advice and guidance in writing this paper.

A special thank to the former developers who helped get CPR off the drawing board; Mark Sponsler, Steven Scott, Sam Yi, Jay Bentley, Joe Yeager, and Robert Vineyard.

References

1. The Internet2 End-to-end Performance Initiative (e2epi), <http://e2epi.internet2.edu>
2. The Planet Lab project, <http://www.planet-lab.org>
3. The Smokeping package, <http://people.ee.ethz.ch/~oetiker/webtools/smokeping/>
4. The Nagios system, <http://www.nagios.org>
5. Manish Jain and Constantinos Dovrolis, Pathload: a measurement tool for end-to-end available bandwidth, Proceedings of the 3rd Passive & Active Measurement Workshop Proceedings, PAM 2002, Fort Collins, Colorado, USA, March 2002.
6. The CPR Web Services Interface. To be published.
7. The Network Monitoring Working Group (NMWG) of the GLOBAL Grid Forum (GGF), <http://nmwg.internet2.edu>
8. The ping end-to-end reporting (PingER) project, <http://www-iepm.slac.stanford.edu/pinger>
9. The Active Measurement Project (AMP), <http://amp.nlanr.net>
10. The PHPLOT tool, <http://www.phplot.com/>
11. Charles Robert Simpson, Jr. and George F. Riley, NETI@home: A Distributed Approach to Collecting End-to-End Network Performance Measurements, Proceedings of the 5th annual Passive & Active Measurement Workshop, PAM 2004 Antibes Juan-les-Pins, France, April 2004.