Experimentation with large scale ICN multimedia services on the Internet made easy

Alina Quereilhac, Daniel Camara, Thierry Turletti and Walid Dabbous
INRIA Sophia Antipolis, France
{alina.quereilhac,daniel.camara,thierry.turletti,walid.dabbous}@inria.fr

1. Introduction

The Information Centric Networking (ICN) paradigm is gaining attention among network researchers and operators as an alternative to share content more efficiently on the Internet. In the ICN paradigm, content naming is independent from physical server location, and since content can be cached at intermediate hops in the network, it can be retrieved from the closest available cache, lowering delay and reducing redundant traffic.

A shift towards an ICN Internet architecture can particularly impact current multimedia traffic patterns, such as video traffic, which represents an increasing fraction of the traffic on the Internet today [1]. Since it is not feasible to envision a clean slate replacement of the current Internet architecture, ICN solutions deployed by Internet operators will coexist with TCP/IP technologies at least in the short to medium term. For this reason, it is crucial to evaluate and understand the behavior of ICN solutions in realistic Internet environments through prior experimentation.

In this paper we present a framework for evaluating ICN solutions in general, and multimedia solutions in particular. This framework simplifies the challenges of conducting large scale experiments on the wild Internet. We leverage on the existing PlanetLab [2] testbed to provide worldwide distributed access to the Internet at minimum cost, and propose the NEPI [3] tool to simplify the design and deployment of experiments. As a means of illustrating the capabilities of the framework, we consider an example experiment in which we evaluate the performance of broadcasting video to over 100 consumers using CCNx [4], against a classical client-server solution.

2. Evaluation framework

The difficulty to realistically simulate the Internet traffic [5] makes simulation environments not sufficient to evaluate the behavior of ICN technologies to be deployed on the Internet. Furthermore, the simplifications linked to simulation environments can hide important issues related to the performance and feasibility of the solution under study. While probably more realistic to reflect performance issues, dedicated or private testbeds can hardly mimic the complexity and diversity of traffic found on the Internet.

For these reasons, one interesting approach is to conduct experiments on the Internet itself. However, for large scale experiments this requires having access to a large number of nodes. The cost of using Amazon EC2 [6] clouds or other paying services might limit the extension to which ICN solutions can be evaluated and affect the transparency of the experiments, since clouds present restricted node management interfaces.

Taking into account these considerations, PlanetLab [2] presents itself as a good alternative for large scale evaluation of ICN technologies to be deployed on the Internet for two main reasons. The first one is that access to the testbed is free for members of PlanetLab partner institutions. The second one is that PlanetLab nodes can be chosen from locations all around the world to best suit the needs of each experiment scenario. However these features have a tradeoff cost: PlanetLab provides a best effort service which means that nodes can be down or irresponsive for long periods of time, and seemly healthy nodes may have broken configurations. These issues add up to the already difficult task of synchronizing large experiment deployments on highly distributed and unpredictable environments such as the Internet.

To alleviate this complexity we developed NEPI [3], network experimentation programming interface. NEPI is an open source tool which provides an experiment description language to design network experiments, describing both topology and applications, and a controller entity to automatically deploy those experiments on target experimentation environments, such as PlanetLab. The controller entity is capable of collecting result files during the experiment execution to a local directory. NEPI also allows to specify node selection filters while designing the experiment, which permits to automatically discover and provision PlanetLab nodes during experiment deployment, without the user having to hand-pick them. NEPI takes into account node health metrics, exposed through the PlanetLab API, to choose the best suited available nodes, and discards unresponsive nodes, blacklisting them to be ignored for future experiments.

For applications, NEPI provides the possibility to upload arbitrary sources, including user modified source code and input files (e.g. video files), to PlanetLab nodes, and gives the possibility to specify custom compilation and installation instructions in the experiment description. It automates installation of package dependencies for application compilation and execution through the yum package manager [7]. This ability to run arbitrary applications, in a very customizable way on PlanetLab nodes, makes it an ideal tool to experiment with ICN technologies on the Internet. NEPI can be used to conduct experiments in two ways: (i) through its graphical user interface, which allows designing experiments by dragging and dropping components, and connecting them on a canvas, or (ii) through a Python script.

3. Example experiment scenario

In this section we exemplify how our framework can be used to conduct large video broadcasting experiments on the Internet, to evaluate ICN technologies. There are currently several active projects working to develop efficient ICN architectures for the future Internet, such as PSIRP, NETInf and CCN [8], to mention a few. For the purpose of this example we chose to evaluate the network performance of the CCNx software, an implementation of CCN [9], against the popular VLC media player [10], when broadcasting a video from a single source to 100 consumers distributed over 12 countries in Europe. VLC supports video broadcasting over the Internet using point-to-point transmissions. We used RTP to stream a 20 minutes long, 320x240, 658kbps, H.264 encoded video to the clients. Figure 1 shows the two-level topology used for this experiment.

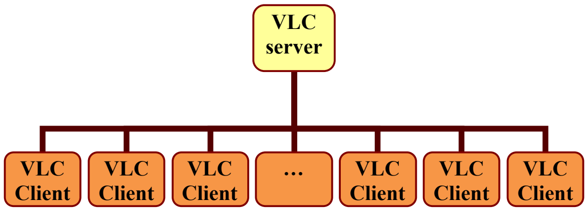

Figure 1. VLC experiment topology

We then used the same video to conduct another experiment with the CCNx software (version 0.7.1). To better exploit the advantages of the content cache, we designed a three-level tree topology where leaf nodes were connected to an intermediate node in the same country, through unicast UDP FIB entries. In turn, intermediate nodes were directly connected to the root node in the same way. All nodes ran a CCNx daemon (*ccnd*), and on the root node the *ccnseqwritter* application was used to publish the video in a local CCNx repository. The 100 leaf nodes retrieved the video using the *ccncat* application. Figure 2 shows the topology chosen for this experiment.

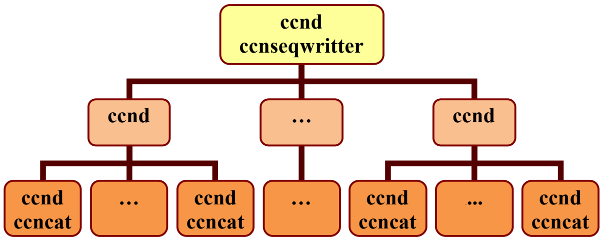

Figure 2. CCNx experiment scenario three-level tree topology on the bottom

In both experiments we made all clients start retrieving the stream simultaneously, and used *tcpdump* to measure the amount of traffic sent from the root node, and received on the leaf nodes. We hand-picked the root and the leaf nodes, and used the same ones for both experiments to ensure comparable results.

Designing and running these experiments took only days with NEPI, while implementing from scratch a script or program to perform the same experiment would have taken several weeks. The scripts used to run the experiments are publicly available on the NEPI source code directory in the “examples/streaming” folder. Instructions on how to download the source code are available at NEPI web page at *http://nepi.inria.fr* .

4. Results

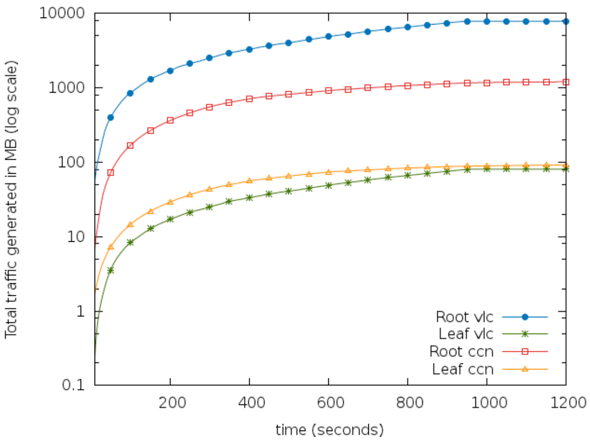

Figure 3. Total traffic generated on root and leaf nodes.

Figure 3 shows the results obtained from this experiment. As expected, we see that the VLC root node generates more traffic (7 times more) than the CCNx root node. The VLC root sends the entire video one time per each client, while the CCNx root node sends the video only one time per intermediate node. However, what strikes as interesting is that the total traffic received in average per leaf node is bigger for CCNx than for VLC, which can be explained by the CCNx protocol overhead.

NEPI was able to retrieve more than 300 results files from the remote nodes automatically. We had to run the NEPI scripts several times in order to get the experiments running, since many times nodes would fail during installation, or SSH connections to the nodes will not respond. However, NEPI managed to detect problems during the deployment face and rapidly finish the experiment providing an error log.

5. Conclusion

In this paper we presented a framework, favorable for the ICN research community, to simplify the task of conducting large ICN experiments on the Internet. The framework is based on two core components, the PlanetLab testbed and the NEPI experiment management tool. We have provided an example showing the usage of the framework to conduct experiments involving over 100 nodes, deploying complex applications and collecting results in a highly customizable way.

References

[1] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian. Internet inter-domain traffic. In *Proceedings of ACM SIGCOMM 2010*.

[2] A. Bavier, M. Bowman, B. Chun, D. Culler, S. Karlin, S. Muir, L. Peterson, T. Roscoe, T. Spalink, and M. Wawrzoniak, Operating System Support for Planetary-Scale Network Services, In *Proceedings of NSDI 2004*.

[3] C. Freire, A. Quereilhac, T. Turletti, and W. Dabbous. Automated deployment and customization of routing overlays on planetlab. In *Testbeds and Research Infrastructure. Development of Networks and Communities*. Spinger vol 44, 2012

[4] CCNx http://www.ccnx.org/

[5] Floyd, Sally, and Vern Paxson. Difficulties in simulating the Internet. *IEEE/ACM Transactions on Networking TON,* volume 9, issue 4, 2001

[6] Amazon EC2. http://aws.amazon.com/ec2/

[7] YUM Package Manager. http://yum.baseurl.org/

[8] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher, and B. Ohlman. A survey of information-centric networking (draft). IEEE Communications Magazine, volume 50, issue 7, 2012

[9] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard. Networking named content. In *Proceedings of CoNEXT 2009*

[10] VLC media player. http://www.videolan.org/

Alina Quereilhac holds a degree in informatics engineering from the University of Buenos Aires, Argentina, received in 2008. She is currently a Ph.D student at the University of Nice and research engineer at the Diana team at INRIA, Sophia Antipolis. The main axes of her research include simulated and live networking evaluation environments, testbed architecture, and networking experiment control.

Dr. Daniel Camara holds a BSc in Computer Science from the Federal University of Paraná, Brazil an MSc and a PhD in Computer Science from the Federal University of Minas Gerais, Brazil. Currently Daniel is working as a research engineer at INRIA Sophia Antipolis. His research interests include wireless networks, distributed systems, artificial intelligence and quality of software.

Dr. Thierry Turletti is a senior research scientist at the Diana team at INRIA Sophia Antipolis. He received the M.S. (1990) and the Ph.D. (1995) degrees in computer science from the University of Nice - Sophia Antipolis, France. He has done his PhD studies in the RODEO group at INRIA where he designed the Inria Videoconferencing System (IVS). During the year 1995-96, he was a postdoctoral fellow in the Telemedia, Networks and Systems group at LCS, MIT and worked in the area of Software Defined Radio (SDR). His current research interests include information centric network architectures, trustable network evaluation platforms and wireless networking.

Dr. Walid Dabbous is a senior researcher at INRIA where he leads the DIANA research team on networking. He received his Doctorat d'Université from the University of Paris XI in 1991. He participated to several FP7 research projects such as: Muse, E-NEXT, OneLab and OpenLab. He is also involved in the French Equipment of Excellence platform project FIT(Future Internet of the Things). His current research interests include: Information Centric Networking Protocols, Networking Experimental Platforms and Simulators and P2P systems performance and privacy.