Power-Proxying on the NIC: A Case Study with the Gnutella File-Sharing Protocol*

Pradeep Purushothaman, Mukund Navada, Rajagopal Subramaniyan,
Casey Reardon, Alan D. George
High-Performance Computing & Simulation (HCS) Research Laboratory
Department of Electrical and Computer Engineering, University of Florida
Gainesville, Florida 32611-6200
{purushothaman, navada, subraman, reardon, george}@hcs.ufl.edu

Abstract

Edge devices such as desktop and laptop computers constitute a majority of the devices connected to the Internet today. Peer-to-Peer (P2P) file-sharing applications generally require edge devices to maintain network presence whenever possible to enhance the robustness of the file-sharing network, which in turn can lead to considerable wastage of energy. We show that energy can be saved by permitting edge devices to enter into standby state and still maintain network connectivity by proxying protocols in the Network Interface Card (NIC).

1. Introduction

A Peer-to-Peer (P2P) network is a distributed system that breaks the traditional client-server computing paradigm as nodes can act as both a client as well as a server. One of the popular applications of P2P networks is file-sharing. Gnutella, Bit Torrent, FastTrack, SoulSeek, etc. are a few examples of popular P2P file-sharing protocols. The topology of a P2P file-sharing network is largely dynamic with nodes joining or leaving the network randomly. P2P file-sharing is attractive with client/server-based, file-sharing due to the fault tolerance property of P2P networks. Since multiple nodes on a P2P file-sharing network usually maintain copies of a single file, file-sharing is not necessarily interrupted as a result of nodes leaving the network. However, P2P file-sharing protocols require nodes to maintain network presence whenever possible to provide this robustness.

Power consumption of edge devices connected to the Internet will be major cause of concern in the near future [1]. It has been estimated that a desktop is generally left idle for 75% of the time after it is powered on. To conserve power, operating systems (OS) usually incorporate a low-power standby state. Edge devices enter this state after a period of idle activity defined by the user. Transitioning into the standby state usually disables the Network Interface Card (NIC) resulting in disruption of network connectivity. Such interruptions could affect network applications such as P2P file-sharing. Consequently, system users are forced to disable standby settings.

This paper proposes a new method called "power-proxying" that refers to selectively proxying a subset of protocol semantics of a network protocol through a separate hardware controller on the NIC. Proxying helps increase the amount of time an idle edge device spends in the standby state without compromising network connectivity. This NIC with the hardware controller for proxying is called a "smart" NIC as it intelligently wakes up the host when it receives a packet that cannot be proxied. The idea is validated by considering Gnutella as a case study and obtaining an estimate of power savings.

2. The Gnutella Protocol

The current version of this protocol, Gnutella V0.6, consists of the following descriptors: PING, PONG, QUERY, QUERYHIT, and PUSH. The PING descriptor is used to actively discover hosts on the network. A servant (Gnutella node) receiving a PING descriptor is expected to respond with one or more

* This material is based on work supported by the National Science Foundation under Grant No. 0519951.

1-4244-0419-3/06/$20.00 ©2006 IEEE

PONG descriptors. QUERY is the primary mechanism for searching for files in the P2P network. A servant receiving a QUERY descriptor will respond with a QUERYHIT if it shares the requested file.

3. Experimental Setup

The experimental setup to determine the energy savings obtained by proxying Gnutella consists of two steps: trace collection followed by simulation of the operation of a power-proxy in a Gnutella network. Trace collection involves collecting statistics on the received Gnutella packets at an edge device. The collected trace is then used as input to a simulator, designed and built to study and estimate the amount of power savings that could be obtained. The Gnutella network bandwidth as seen from the edge device can also be varied to study the influence of node bandwidth and standby time on obtained power savings.

The simulator consists of a peer-network entity that abstracts a Gnutella network and an edge device with a "smart" NIC connected to the peer network. The peer network issues Gnutella packets to the edge device based upon the input trace. When the edge device is in standby state, all the Gnutella packets except the file upload requests are handled by the proxy.

4. Results and Inference

Traces show that average inter-arrival time for 90% of PING, PONG, and QUERY packets is within 90 sec, 4 sec, and 30 sec, respectively. These observations show that Gnutella network is not really idle even on a passive node (a node that does not initiate queries).

Simulation of the P2P network was performed using the simulator described in Section 3. The standby time for the host was varied from 150 sec to 600 sec in steps of 150 sec, and the bandwidth of the edge device was varied from 100 Kbps to 100,000 Kbps. Figure 1 shows that an edge device in the Gnutella network is in the standby state for more than 85% of time as a result of proxying, irrespective of network bandwidth. For high-bandwidth nodes, increasing standby time settings only reduces the amount of time spent in standby state, as the node waits a longer time to enter standby state.

Figure 2 shows the inter-arrival time for file-upload requests for an edge device in the Gnutella network. As shown in the figure, for file upload requests at a peer with 100 Mbps bandwidth and standby time of 300s, 60% of the inter-arrival times are one hour or longer. Thus, proxying Gnutella can result in the edge device spending more time in the standby state, as only file-upload requests need to be handled by the host.

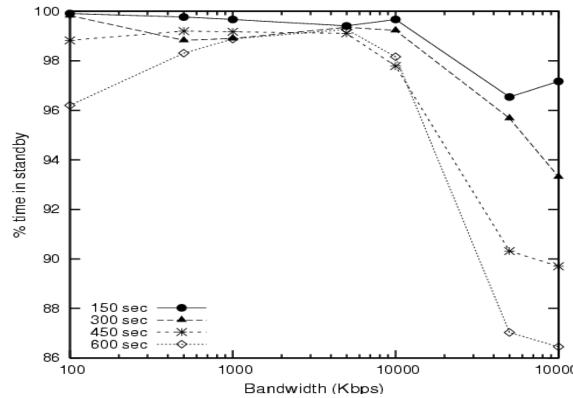

Figure 1: Analysis of Gnutella Proxying

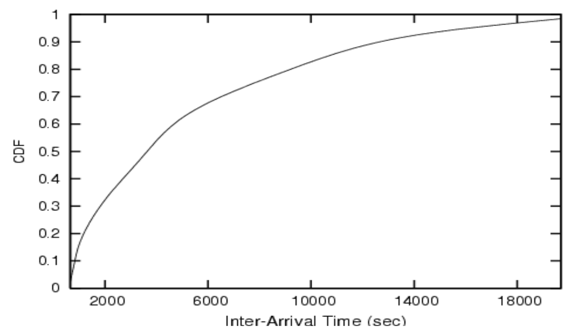

Figure 2: File Upload Request

5. Conclusions

Proxying is a suitable technique to reduce power consumption in networked edge devices. Analysis of the collected traces justifies the need for proxying and simulation results show that a substantial amount of power can be saved by proxying Gnutella. Implementing the power-proxy on a NIC can definitely achieve significant energy savings as it allows the rest of the system to remain in low-power standby mode. Future work would focus on various architectural design strategies and tradeoffs for realizing the proxy in hardware. A prototype NIC with proxy functionality will also be developed to quantify the power savings.

6. Reference

[1] K. Christensen, P. Gunaratne, B. Nordman, and A.George "The next frontier for communications networks: power management," *Computer Communications*, Vol. 27, No. 18, pp. 1758-1770, December 2004.