Xavid Pretzer
October 30, 2007
NSF GRFP Application
Proposed Research

Mondo

Bringing Distributed File Systems to the People

Keywords: distributed file systems, scalability, peer-to-peer, untrusted, decentralized systems

Hypothesis

A peer-to-peer distributed file system can be built that allows individuals to easily and efficiently share mutable files with each other without requiring centralized servers or a shared trust model; this system can be made to have performance comparable to other file distribution systems. This system can have high file availability even in the face of failed and malicious peers using incentives.

Motivation

Distributed file systems such as the Andrew File System [3] have long been a core part of many academic and corporate network environments due to the convenience of file access and sharing they enable. However, outside of these environments, the need for dedicated servers and skilled administration of these systems has prevented their widespread use. While some systems such as FARSITE [1] have tried to address some of these issues, dependence on centralized trust and lack of malicious clients have prevented these systems from being used outside of limited-access environments. Conversely, peer-to-peer file transfer systems (e.g., [2]) have received widespread use, but do not support the modification operation that is key to a usable filesystem. Thus, Mondo seeks to learn from both these areas and create a system that has all the features of a distributed file system while achieving the low investment and high usability of peer-to-peer file sharing. In addition, Mondo seeks to achieve high replication even with limited or occasional network connectivity to improve its ability to share data in third-world countries without reliable network connections.

Proposed Design Overview

Mondo will be implemented in two layers. The file location layer will be implemented by a distributed hash table as in [4], associating each file's random unique identifier, with a list of peers who have the file. In the storage layer, each client keeps copies of each file it "owns", as designated by the user, and also cached copies of recently used files. File read permissions are handled using encryption and file write permissions are handled using cryptographic signatures. Block checksums ensure file integrity. Directories contain file ids, owners and necessary file keys to verify signatures. There can be multiple root directories, specified by a directory ID and the public key of its owner.

Clients increase the replication of files they own using a tit-for-tat strategy, agreeing to store a copy of another peer's file if that peer stores a file of similar size it owns. A peer periodically ensures the other client is keeping its part of the bargain by checking remote file existence using salted hash codes.

Research Plan

I will implement a prototype of the Mondo system according to the design outlined above, focusing on implementation speed and correctness over efficiency. I will then evaluate the system's performance against benchmarks including the Andrew benchmark [3] and various usage simulations. I will compare the results on these benchmarks to various control filesystems, including the Andrew File System [3] and Frangipani [5]. Based on any performance bottlenecks or reliability problems discovered during this evaluation and associated profiling, I will revise the design and reevaluate as necessary. Once the design has no major outstanding issues, I will do a low-level re-implementation focusing on efficiency and clean implementation, and compare this implementation's performance against the control filesystems on the benchmarks. Based on this data, I will evaluate my hypothesis and determine the need for additional research.

Anticipated Results

I anticipate that the final evaluation will show Mondo to have performance comparable to, but likely not superior to, other distributed filesystems. Mondo's distributed trust model introduces a slight overhead, but it should not be significant in most usecases. I expect that Mondo's use of incentives will give it high availability even in the presence of peer falures or reasonably proportions of malicious peers.

I certify that this is an original research proposal containing only my own work except as cited.

References

[1] Atul Adya, William J. Bolosky, Miguel Castro, Gerald Cermak, Ronnie Chaiken, John R. Douceur, Jon Howell, Jacob R. Lorch, Marvin Theimer and Roger P. Wattenhofer. "FARSITE: Federated, Available, and Reliable Storage for an Incompletely Trusted Environment". (2002)

[2] Bram Cohen. "Incentives Build Robustness in BitTorrent". (2003)

[3] John H. Howard, Michael L. Kazar, Sherri G. Menees, David A. Nichols, M. Satyanarayanan, Robert N. Sidebotham and Michael J. West. "Scale and Performance in a Distributed File System". (1988)

[4] Ion Stoica, Robert Morris, David Karger, M. Frans Kaashoek and Hari Balakrishnan. "Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications". (2001)

[5] Chandramohan A. Thekkath, Timothy Mann and Edward K. Lee. "Frangipani: A Scalable Distributed File System".