INTRODUCTION

Due to the Internet, there has been an exponential increase in the volume of digital content available to consumers. Apple iTunes (Distributed Computing Industry Association 2004) and YouTube (2006) exemplify the paradigm shift in music and video distribution. Users are increasingly obtaining digital content through downloads. Given the tremendous popularity of digital content, exploring new channels to enable content distribution and creating new non-traditional marketplaces is a logical step forward.

P2P networks are currently popular vehicles for digital content distribution. With the continued proliferation of P2P networks such as Kazaa...
Digital Rights Management for Untrusted Peer-to-Peer Networks

(2006) and Gnutella (Gnutella clients 2006, Gnutella 2006), industry and academia are beginning to realize the potential of such networks in the dissemination of digital information. However, at present, P2P networks are rife with risks of copyright infringement. P2P networks lack many of the security features inherent in client-server networks that can be used to protect the rights of content owners.

This chapter focuses on exploring solutions that enable large-scale distribution of digital content in P2P networks such that intellectual property rights are not violated and the content creators are able to collect profits. We augment the basic distribution of content from creators to consumers with distribution through authorized resellers. As part of our distribution model, we apply appropriate digital rights management (DRM) technologies to the content in an effort to ensure that the P2P networks benefit creators and legitimate customers, not just pirates.

This chapter is organized as follows. Section 2 gives relevant background information on P2P networks and DRM technologies. In particular, we describe the requirements of a DRM system that is suitable for P2P networks. We also outline the design goals for our system. This section concludes with a brief review of related work.

Section 3 provides the detailed design of our proposed system. We include justification of our design decisions. We also provide a description of the system architecture and the functional flow in our proposed system. Section 4 includes the implementation details of the components of our system. We do not cover the details of the security-related features, which are discussed in the subsequent section. Section 5 covers the security features in the proposed system. We discuss the implementation aspects and analyze the strengths and weaknesses of each significant security feature.

In Section 6 we consider the testing of our working prototype system. This section illustrates the underlying functionality of the prototype by going through the steps of a sample use case in detail. We conclude the chapter with Section 7 where we summarize the achievements of the project and present various ideas that could be developed as extensions of this project.

BACKGROUND

P2P Networks

In traditional client-server network architectures, all nodes communicate to and from a central server whereas in a P2P network the nodes communicate in a relatively ad-hoc manner. Figure 1 illustrates the difference between the topologies of P2P and client-server networks.

Each P2P network has its own procedure for connecting the P2P clients. For example, the classic P2P network Napster (2006) had a central server used to index all content that peer users had to offer. This central server based approach has evolved into more decentralized networks which do not require a centralized server to connect peers in the network (Kazaa 2006, Gnutella 2006). Kazaa and Gnutella are well-known examples of decentralized P2P networks. Although Kazaa does not have any central indexing servers, it relies on a subset of peers, called supernodes, to perform the analogous function of an indexing server. Supernodes are peers with more advanced machines and faster connections which host a list of files their neighborhood Kazaa users are willing to sharing. Ordinary Kazaa users connect to the supernodes to search content. Gnutella uses a more distributed protocol (sometimes known as “pure P2P”) whereby search requests are progressively routed to directly connected “neighbors” (in an overlay network), effectively creating a query flood.

P2P networks provide several advantages over traditional client-server based network model, including the following.