Security Issues in Distributed Transaction Processing Systems

R. A. Haraty
Lebanese American University, Lebanon

INTRODUCTION

Transaction-processing systems (TPS) are becoming increasingly more available as commercial products. However, the approaches to the issues associated with using TPS in multilevel secure environments are still in the research stage. In this article, we address the issues of multilevel security in distributed transaction-processing systems. A distributed transaction-processing system (DTPS) is a collection of a finite number of centralized transaction-processing systems connected by a computer network. Each of these transaction-processing systems is controlled by a software layer and can be accessed both remotely and locally. Properties of a DTPS, such as data replication, may have a substantial effect on the security of the system. The security policies and integrity constraints adopted at each site may result in global security having inconsistent states. We address the issues of achieving a multilevel secure DTPS, and discuss the security constraints and data replication.

In this work, we address the issues of achieving a multilevel secure DTPS system and discuss the security constraints and the replication of data items. The next section provides some background. Then, next, an overview of a distributed transaction-processing system is presented. In the fourth section, security-related issues are discussed. In the fifth section, a multilevel secure distributed transaction-processing system is presented. Then, in the next section, future trends are presented. The final section concludes the article.

BACKGROUND

Several commercial and military applications require a multilevel secure transaction-processing system (MLS/TPS). In an MLS/TPS, users are assigned classification levels that we denote by “clearances,” and data items are assigned sensitivity levels. There are three interesting architectures that have been used to build MLS/TPSs from untrusted ones. These architectures are known as the integrity lock architecture, the kernelized architecture, and the data distribution architecture (Air Force Studies Board, 1983). While most of the techniques for TPS security are developed for traditional centralized TPSs, more TPS researchers are making substantial contributions to the development of a distributed TPS (Getta, 2003; Haraty, 1999; Haraty & Rahal, 2002; O’Connor & Gray, 1988).

A DTPS is a collection of a finite number of TPSs connected by a computer network (Ozsu & Valduriez, 1999). Each of these TPSs is controlled by a transaction management software layer and can be accessed both remotely and locally. A DTPS integrates information from the local TPS and presents remote users with transparent methods to use the total information in the system. An effective TPS system serves to maintain the ACIDity properties (i.e., atomicity, consistency, isolation, and durability) of transactions and must be superimposed on the preexisting local TPSs (Gray & Reuter, 1993).

One proposed architecture for MLS/TPS is the replicated architecture. This approach is being explored in several ongoing research efforts, including the Naval Research Laboratory Secure Information through replicated architecture (SINTRA) project (Thuraisingham, 1987). Data replication in DTPS has several implications for the security of the system. Replication allows data items in different local TPSs to be identified as logically belonging to the same entity. The security policies adopted by each site may result in global security having inconsistent states, because of the difference of local representation and management.

OVERVIEW OF DISTRIBUTED TRANSACTION-PROCESSING SYSTEMS

A DTPS consists of a set of preexisting local TPSs \( \{LTPSi\}_{i=1}^{m} \), distributed among several interconnected sites. Each LTPSi is a software layer on a set of data items Di. Figure 1 depicts the architecture of a DTPS.

SECURITY ISSUES

Processes that execute on behalf of users are referred to as subjects. Objects, on the other hand, correspond to a data item. Objects can be files, records, or even fields. In this section, we present the notion of object classification with emphasis on the problem of conflicting security constraints due to replication.