Introduction to LDAP and Tivoli Directory Server overview

- Pradeep Lawande.

13/09/2010
Contents

- Define LDAP
- Define Directory
- Difference between LDAP directories and database
- Advantages of using Directories
- LDAP Implementation
- Tivoli Directory Server
- Features of Tivoli Directory Server
What is LDAP?

- LDAP defines a standard method for accessing and updating information in a directory.
- LDAP is a lightweight implementation of the OSI X.500 Directory Access Protocol.
- LDAP requires the TCP/IP protocol stack
  - Lightweight
  - Simple
- LDAP requires an LDAP server and an LDAP client.
Continued...

- LDAP Server and LDAP Client
  - LDAP client makes a request.
  - LDAP server accesses the directory and sends the information to the client.
Define Directory

- A directory is composed of entries, this entry is the basic unit of the directory.
- These entries usually contain a similar kind of information.
- Directories contain data that is intended to be read much more often than written.
- For example, the phone number of a person usually does not change very frequently. Compare this to the daily sales of a company which most likely changes very frequently.
- Directories are optimized for read access.
- Stores information about objects in a hierarchical structure.
- Directories help people by organizing information.
- The directories can streamline your business processes.
Directory Example

Directory

Objects

  o=ibm

    cn=john
    cn=erica
    cn=david
      sn=freeman
      telephoneNumber=512-286-2000
      userPassword=******
      mail=dfreeman@ibm.com
    cn=jane
Differences between an LDAP Directory and a Database

- A directory usually contains information which is static.
- A database often contains information which is dynamic.

- A directory is optimized for read access.
- A database is designed for providing data that can be easily manipulated both for reading from the database and writing to the database.

- Information in the directory is organized in a hierarchical structure.
- Information in the database is organized in the form of rows and columns.

- Each entry in the directory has a name which identifies its location in the directory.
- One or more of the columns in the database is identified as a key.
Continued...

- Most directory implementations do not support transactions (all or nothing operations).
- Databases support transactions.

- LDAP directories use simplified and optimized access protocol to retrieve or update information from the directory.
- Databases use Structured Query Language (SQL) to retrieve information from the database.
Advantages of Using a Directory

- Manage information centrally.
  - Information about people.
  - Information about user accounts.
  - Information about network resources such as computers.
- Quickly search information.
- Applications can take advantage of centrally located Data.
- LDAP standard is widely accepted.
- An LDAP directory can be distributed across multiple servers. This design distributes the load and provides other management benefits.
- LDAP is an open standard, with multiple-platform support. An LDAP client on any platform can communicate with any LDAP server. So there is less reliance on a single vendor.
LDAP Implementation

- LDAP is supported by a growing number of vendors
  - IBM Tivoli Directory Server
  - Netscape iPlanet Directory
  - Lotus Domino Server
  - Microsoft Active Directory
IBM Tivoli Directory Server (TDS)

IBM Tivoli Directory Server provides an identity data foundation for rapid development and deployment of Web applications, security and identity management on zSeries Linux and distributed platforms

Key features

- LDAP support ensures compatibility with industry standard LDAP based applications
- Built-in, high-performance LDAP Proxy with reliable IBM DB2® Universal Database engine offers massive scalability to hundreds of millions of entries, as well as groups with hundreds of thousands of members
- Robust replication capability for both master/subordinate replication, gateway, cascaded and peer-to-peer replication with up to dozens of master servers
- Eases management and usability with Web Administration GUI and features such as Dynamic and Nested Groups, along with Sorted and Paged Search Results
- 64-bit client support, with ability to run multiple directories on single machine
- Tight integration with IBM operating systems, WebSphere® middleware, and Tivoli identity management and security products
- Provides industry-certified openness and dynamic, extensible schema for broad and flexible application support
Tivoli Directory Strategy

- **Open**
  - TDS does not limit platform choice by offering support for all leading operating systems:
  - TDS embraces open standards, such as LDAP v3 and web services

- **Scalable, Performance and Availability**
  - Built on the DB2 engine for unsurpassed reliability and scalability
  - Replication/Multi-Master provide competitive advantage for availability and failover

- **Building block of IBM’s leading middleware portfolio**
  - TDS is a key part of IBM’s leading middleware portfolio, and complementary to IBM platform-optimized directories, offering complete solutions
Tivoli Directory Server extends scalability, performance, availability advantages

**Business Value**

- Provides the high-performance, highly-available identity data infrastructure required for global on demand services with robust DB2 engine, as well as rich multi-mastering and replication capabilities

- Provides industry-certified openness and dynamic, extensible schema for broad and flexible application support

**Features**

- Built-in, high-performance LDAP Proxy – offering massive scalability to hundreds of millions of entries – at no additional cost!

- 64-bit client support, with ability to run multiple directories on single machine

- Excellent security, performance, manageability, deployment, and ease-of-use!
**Virtually Unlimited Scalability**

**Directory Service scaling to hundreds of millions of entries**

TDS automatically partitions directory tree
The partitions are transparent to LDAP clients
Proxy load balances requests across each partition’s backend servers
Protocol aware routing

LDAP Clients

Proxy

Read/Write Requests in Partition 1

Read/Write Requests in Partition 2

DIT Partition 1

Server 1

Server 2

Server 3

DIT Partition 2

Server 4

Server 5

Server 6

LDAP Requests

Directory Service scaling to hundreds of millions of entries

Protocol aware routing
High Availability

- Provides proxy LDAP protocol aware routing for peer-to-peer (multiple master server) configurations and for 24/7 data availability

If Peer1 fails, WebSphere Edge Server automatically fails over to Peer2. Peer2 queues updates for Peer1 until it is back online.

Reads are balanced across replicas

HW/SW Load Balancer balances reads across replicas
Thank You