EVIDEN

Identity and Access Management

DirX Directory

Introduction

Version 9.1, Edition June 2025

Table of Contents

Preface

The *DirX Directory Introduction* describes the concepts of DirX Directory. The book is structured as follows:

- Chapter 1 provides an introductory description of the material to be covered in the guide.
- Chapter 2 provides a glossary that defines terms and concepts that relate to DirX Directory.

DirX Directory Documentation Set

DirX Directory provides a powerful set of documentation that helps you configure your directory server and its applications.

The DirX Directory document set consists of the following manuals:

- *DirX Directory Introduction*. Use this book to obtain a description of the concepts of DirX Directory.
- *DirX Directory Administration Guide*. Use this book to understand the basic DirX Directory administration tasks and how to perform them with the DirX Directory administration tools.
- *DirX Directory Administration Reference*. Use this book to obtain reference information about DirX Directory administration tools and their command syntax, configuration files, environment variables and file locations of the DirX Directory installation.
- *DirX Directory Syntaxes and Attributes*. Use this book to obtain reference information about DirX Directory syntaxes and attributes.
- *DirX Directory LDAP Extended Operations*. Use this book to obtain reference information about DirX Directory LDAP Extended Operations.
- *DirX Directory External Authentication*. Use this book to obtain reference information about external authentication.
- *DirX Directory Supervisor*. Use this book to obtain reference information about the DirX Directory supervisor.
- *DirX Directory Plugins for Nagios*. Use this book to obtain reference information about DirX Directory plugins for Nagios.
- *DirX Directory Disc Dimensioning Guide*. Use this book to understand how to calculate and organize necessary disc space for initial database configuration and enhancing existing configurations.
- *DirX Directory Guide for CSP Administrators*. Use this book to obtain information about installing, configuring and managing DirX Directory in the context of a Certificate Provisioning Service operating in accordance with regulations like the German “Signaturgesetz”.
- *DirX Directory Release Notes*. Use this book to install DirX Directory and to understand the features and limitations of the current release.

Notation Conventions

Boldface type
In command syntax, bold words and characters represent commands or keywords that must be entered exactly as shown.

In examples, bold words and characters represent user input.

Italic type
In command syntax, italic words and characters represent placeholders for information that you must supply.

[ ]
In command syntax, square braces enclose optional items.

{ }
In command syntax, braces enclose a list from which you must choose one item.

In Tcl syntax, you must actually type in the braces, which will appear in boldface type.

|
In command syntax, the vertical bar separates items in a list of choices.

...
In command syntax, ellipses indicate that the previous item can be repeated.

install_path
The exact name of the root of the directory where DirX Identity programs and files are installed. The default installation directory is *userID_home_directory**/DirX Identity* on UNIX systems and **C:\Program Files\DirX\Identity** on Windows systems. During installation the installation directory can be specified. In this manual, the installation-specific portion of pathnames is represented by the notation *install_path*.

1. Introducing DirX Directory

Directory services are critical components of today's highly interconnected business environment. Directory services provide the foundation for identity and access management across the ever-widening boundaries of the enterprise:

- In the intranet environment, the directory service provides a global repository for shared information about employees, organizations, and resources such as applications, network devices, and other distributed services, accommodating hundreds of thousands of users.
- In the extranet environment, the directory service maintains profile information about customers, trading partners, and suppliers, accommodating millions of users.

For both these environments, the directory service must be able to manage user identities and control access to the information and services offered to its users, and it must provide fast, always available, authenticated access to the information and services, potentially to a huge number of users. The ideal directory service for today's enterprise:

- Integrates the disparate service- and platform-specific databases, profiles, policies, and provisioning processes within the enterprise's infrastructure into a centralized service- and platform-independent model that is always up-to-date
- Supports advanced, complex identity management services to guarantee data security
- Provides the performance and scalability required to manage user identities and control access to information and services by potentially tens of millions of users

1.1. What is DirX Directory?

DirX Directory is a directory service that provides a standards-compliant, high-performance, highly available, highly reliable securable identity management platform with very high linear scalability.DirX Directory can act as the identity store for employees, customers, trading partners, suppliers, subscribers, and other e-business entities.DirX Directory can also serve as a metadirectory store, to provide a single point of access to the information within disparate and heterogeneous directories available in today's enterprise network for user management and provisioning.DirX Directory provides:

- Full and comprehensive directory service standards compliance—DirX Directory implements the Lightweight Directory Access Protocol (LDAP) v3 and X.500 (1993) directory standards and supports Hypertext Transfer Protocol (HTTP) access with a custom HTTP API.It also permits third-party LDAP-enabled applications to manage the DirX Directory schema over LDAP, which allows them to integrate their schema information into the directory schema simply and quickly.
- High-performance directory access—DirX Directory uses an innovative database kernel that is optimized for directory access, allowing for sub-second response times and high throughput rates for parallel queries.
- High availability and reliability via software- and hardware-based solutions—DirX Directory supports "floating master" directory replication for high availability and failover configurations that is based on X.500 replication (also called "shadowing")

protocol.It also supports high-availability hardware solutions like clusters and RAID arrays.For backup and recovery, DirX Directory supports full and differential saving in parallel with directory operations without any interruption of service on retrieval or update operations.Transaction processing in the DirX Directory database provides guaranteed recovery after crashes without data loss.

- A centralized identity management store—DirX Directory is designed to be the foundation of an identity management system.It can store and manage user profiles (where users are employees, suppliers, trading partners, services, and customers), digital certificates for public key infrastructures (PKIs), authorization and authentication information, access permissions and other relevant attributes for users that control access to information, network resources, or distributed services.DirX Directory provides a single point of administration—one overall database that stores the complete set of user information for the enterprise – and offers the ability to create unique, integrated, cross-service, cross-platform user profiles with up-to-date account information.
- Advanced security mechanisms to control access to data and user authentication—DirX Directory supports Secure Socket Layer/Transport Layer Security (SSL/TLS) for LDAP and HTTP server and client authentication, X.500 Directory Access Protocol (DAP) authentication, authorized user access control, server-side policies for local security management, and the creation and enforcement of password policies to control how passwords are used and administered in the enterprise network.DirX Directory also provides LDAP server and DSA audit logging facilities to record information about interactions with the directory service for traffic analysis or accounting and billing operations.
- Very high scalability—The DirX Directory database is designed to permit linear scalability in a single directory server, providing the ability to scale from workgroup to enterprise to e-business directory roles.The DirX Directory database provides the potential for future growth on existing hardware configurations and can scale quickly to accommodate a huge number of users in an extranet deployment.
- Powerful administration tools—DirX Directory offers both graphical and command-based scriptable tools for centralized administration of a distributed directory system, including auditing, monitoring, and logging functions, from all supported operating system platforms.

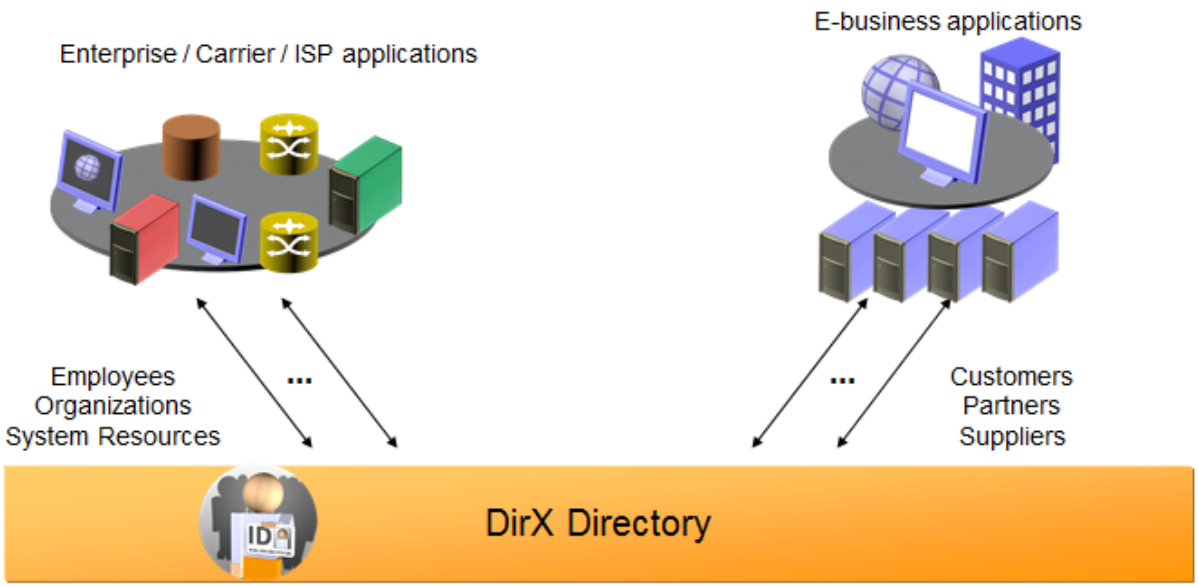

Figure 1. DirX Directory Service

1.1.1. How Do Clients Access DirX Directory?

The data stored in DirX Directory is accessible through:

- Any LDAP client and LDAP-enabled application.
- Any HTTP client and HTTP-enabled application.
- A command-line administration interface with full LDAP functionality, which can also be controlled via Tcl scripts; for example, **dirxcp**.
- DirX Directory Manager, a Java-based LDAP management client that provides configurable platform-independent graphical administration interface for local and remote administration of DirX Directory.

1.1.2. What are the Main DirX Directory Components?

The DirX Directory service consists of the following main architectural components:

- The LDAP server—the component that provides access to the directory service over the LDAPv3 protocol (and LDAPv2, for legacy clients).
- The Directory System Agent (DSA)—the component that implements the X.500 (1993) directory service standard: the directory information, schema, administrative, and access control models and the following X.500 protocols:
- The Directory Access Protocol (DAP), which defines the exchange of queries between DSAs and Directory User Agents (DUAs).
- The Directory System Protocol (DSP), which DSAs use to forward queries and administration requests that they cannot answer to other DSAs for possible resolution.
- The Directory Information Shadowing Protocol (DISP), which DSAs use to replicate directory information from one DSA to another.
- The Directory Basic Access Method (DBAM) database kernel—the component that manages the directory service data storage and retrieval.The DBAM database stores the

Directory Information Base (DIB).

- The DirX Directory Progsvr—the component that provides a safe and reliable way for executing external procedures specified by LDIF policies in LDIF agreements on generated LDIF files.
- The DirX Directory HTTP server—the component that provides access to the directory service over the HTTP 1.1 protocol.

The following figure illustrates the DirX Directory components, including the protocols they support, the databases they manage, and the files they can generate.The component-specific sections in this chapter provide architectural and functional details about each DirX Directory component shown in the figure.

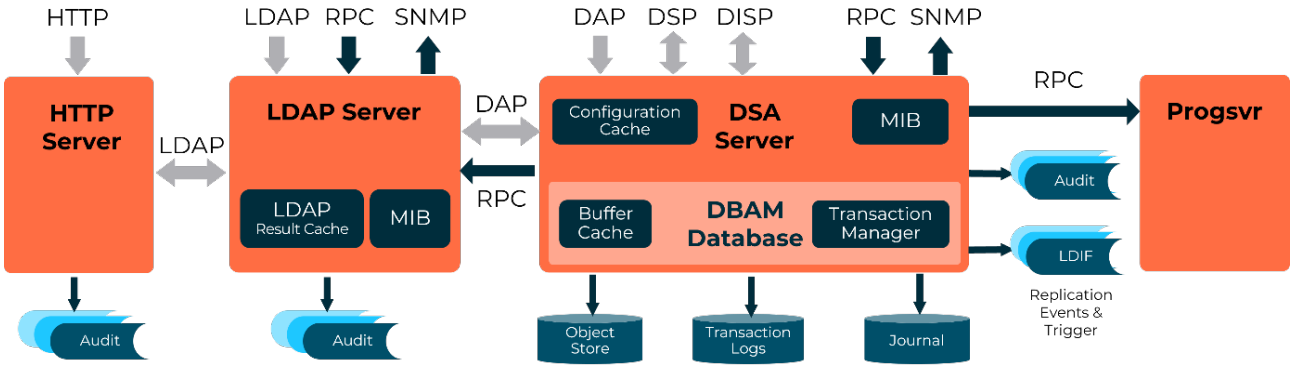

Figure 2. DirX Directory Service Components

1.2. The LDAP Server

DirX Directory provides extensive support for LDAP access to its directory service.The DirX Directory LDAP server implements the complete LDAPv3 protocol to provide full support for access to the directory service by LDAP client applications.It also implements the LDAPv2 protocol to support directory access by older LDAPv2-compliant applications.The LDAP server implementation is high-performance, highly configurable, and supports state-of-the-art security features.

The LDAP server runs in a separate process from the DSA.It can run on the same server machine as a DSA or a different (remote) machine.The LDAP server and DSA are tightly integrated by an internal, proprietary LDAP-like protocol.The LDAP server process (and the DSA process) is also accessible using this protocol, which allows access by co-located or remote applications.The protocol's interface definition represents a management API that is applicable to a variety of DirX Directory administration tools and is for internal use only.

1.2.1. LDAP Server LDAPv3 Conformance

The DirX Directory LDAP server conforms to the Internet Engineering Task Force (IETF) LDAPv3 core standards, including:

- All operations, object classes and attribute types specified in the series of LDAP RFCs.
- LDAPv3 sessions and operations without an initial bind. An "association" between an LDAP client and an LDAP server normally starts with a **bind** operation, which allows the

client to be authenticated to the directory service. LDAPv3 permits an LDAP client to perform search operations on the directory database without having to bind first to an LDAP server. The LDAP server handles requests of this type as if the LDAP client has performed an "anonymous" bind. The section on Security Services provides more details about authentication over LDAP.

- Full UTF-8 support. LDAPv3 is based on UTF-8 character set encoding rather than restricting character encoding to printable strings, as is the case with LDAPv2. UTF-8 provides a more general character encoding mechanism that reflects the need for special character sets for languages such as Kanji and Hiragana.
- The "binary" option for attribute values, which can be used to override a string-based representation defined for an attribute. The binary option allows an attribute value to be transferred from server to client in ASN.1 binary format. It is used for attributes with complex ASN.1 data type syntax in cases where clients need the structure of values of that type; for example, Certificate and CertificateList attribute types.
- Support for LDAP referrals. In the DAP protocol, a DSA that is unable to satisfy a particular query can return a message to the requesting DUA that identifies the name and communications address of a DSA that is better able to handle the operation. These messages are called continuation references, and are returned in a referral when the entire operation must be handled by another DSA. Further handling of the referral is left to the client. LDAPv3 provides full support for the passing of referrals to clients. LDAP clients can request referrals, and LDAP servers can return requests to clients for subsequent handling by other LDAP servers. In DirX Directory, the tight integration between the LDAP server and the DSA provides the support for LDAP referrals. DSAs can be configured to automatically try to resolve referrals, including LDAP referrals.
- Support for the attributes of the LDAP root DSA-Specific Entry (DSE). The LDAP root DSE provides information about the individual LDAP server that is accessible to LDAP clients, such as the naming contexts it holds, its schema subentry, and alternative servers to use in case this one is not available.
- Support for LDAP schema publishing through the LDAP global schema subentry, which allows LDAP clients to read and adapt dynamically to the data model (the schema elements) that the LDAP server supports.

LDAP operations are atomic: they’re either successful or they’re not. For most operations, the LDAP server returns a single response to a request, which is either the result of a successful operation or an error, if only a part of the operation succeeds. A search request can result in one or more responses to the LDAP client. The LDAP server maps each matching entry found in a successful search (or each referral) onto a single search response. The search is terminated by a final search response that indicates the success of the operation or an error, if only part of the operation succeeds. The DirX Directory LDAP server supports the following LDAPv3 search control extensions:

- The simple paging of search results (PR), which permits an LDAPv3 client to request search results in "pages" according to RFC 2696. Each page of the result consists of a client-specified number of entries. The simple paged results extension allows the LDAP client to control the rate at which an LDAP server returns the results of a search operation, and is useful when the client does not have the resources or the bandwidth to process the entire result set at once.

- The server-side sorting of search results (SSS), which enables an LDAPv3 client to request the server to return a search result ordered by a particular attribute—for example, the "surname" attribute—in ascending or descending order, according to RFC 2891. The SSS extension is useful when the LDAP client is unable to sort the results itself but still needs them to be sorted.

1.2.2. LDAP Server Implementation Features

DirX Directory LDAP server is implemented in a single process, using a threads-based architecture. Threads permit multi-tasking and provide a convenient solution to the requirement for the LDAP server to support many simultaneous activities, some of which may be suspended while awaiting the result of queries to other LDAP servers (or DSAs) or the completion of database accesses. As a result, directory queries made in parallel are processed in parallel, which provides genuine parallel processing on a multi-processor system. Using threads to implement parallel programs leads to better system resource use (shared memory, files) than a comparable implementation in a multi-process architecture.

The DirX Directory LDAP server maintains a pool of threads for running LDAP client requests in parallel. The thread pool is configurable: an administrator can adjust the number of threads in the pool according to the available resources on the target system and the expected search profiles (large results or small results, result cache on or off).

The use of the LDAP server's result cache can dramatically improve performance in situations where LDAP clients are sending frequent identical search requests. By default, the result cache is disabled. When the result cache is enabled, the LDAP server first searches the cache for a query before sending it on to the DSA for processing. The results cache is both configurable and monitorable. Using DirX Directory administration tools, an administrator can configure the cache for maximum number of entries, maximum size, and other parameters, enable and disable the cache, and view cache statistics.

Administrators use DirX Directory administration tools to set up and configure an LDAP server; the resulting configuration information is stored in the directory database and can be easily retrieved and updated using the same administration tools. Each LDAP server can have its own specific configuration, which allows administrators to create special-purpose LDAP servers. Multiple LDAP servers on different machines can also be set up as "front ends" to a DSA to provide load-balancing of directory service queries.

The LDAP server is started within the DirX Directory service together with the DSA. The LDAP server immediately binds to the default DSA specified in the LDAP configuration file **dirxldap.cfg**. This is only possible if the DSA has its own access point (specified in the environment variables **DIRX_DSA_NAME** and **DIRX_OWN_PSAP**) and is attached to the network. If the bind is not possible, the LDAP server exits.

Use the command **dirxldapv3 –V** to display the LDAP server's version and build number.

1.2.3. LDAP Server Security Features

The LDAP server supports the Secure Socket Layer 3.0/Transport Layer Security 1.0/1.1/1.2 (SSL/TLS) protocols for encrypted LDAP communications, with both server and client authentication supported. The LDAP server also supports TCP/IP address filtering for LDAP

client access control. These features are described in more detail in the "Security Services" section of this chapter.

In addition, the DirX Directory LDAPv3 implementation has been successfully tested for protocol implementation errors that could compromise security using the PROTOS LDAPv3 test suite developed by the University of Oulu, Finland.

1.2.4. LDAP Server Monitoring, Auditing and Logging

The LDAP server supports a Management Information Base (MIB) based on RFC 2605 and RFC 2780 for the storage of usage statistics about the LDAP server's operation. The DirX Directory administration tools provide access to this MIB for viewing and evaluating the usage information. The LDAP server also supports auditing and diagnostic logging of its operations. The section on "Monitoring, Auditing, and Logging Services" provides further details about these services.

1.2.5. LDAP Server Extended Operations

The LDAP server supports extended operations. (See section 4.12 "Extended Operation" in the *Lightweight Directory Access Protocol (v3)*, RFC 4511, June 2006 for details on extended operations.) The extended operations provided by DirX Directory are designed mainly for support and monitoring purposes. They provide management information base (MIB) table data and other diagnostic data of the DirX Directory server processes. Use the **dirxextop** command or the **Monitoring** view in DirX Directory Manager to perform an LDAP extended operation.

1.2.6. LDAP Proxy

Running the DirX Directory LDAP server in proxy mode allows two different modes:

- Routing incoming LDAPv3 requests from LDAPv3 clients to any remote target LDAPv3-compliant directory server and associates responses back to the requesting client.
- Modify the content of incoming LDAPv3 client requests and outgoing LDAPv3 results before sending them on for further processing. This feature allows for adding, replacing or removing unwanted or incompatible data from LDAP requests and results – for example, to hide certain data from LDAP clients – or to provide a transparent naming schema between legacy clients and the directory back end server.

The main advantages of an LDAP proxy are the following:

- It creates a central access point for LDAP clients.
- It hides the actual processing server knowledge.
- It hides server outages (the feature implies automatic request forwarding to other available servers in the event of network problems or server outages).
- It transparently provides load balancing to the calling LDAP client.
- It provides an additional access control layer to the directory service.

Key features of the DirX Directory LDAP proxy include:

- Rule-based request routing: The LDAP Proxy selects the target server based on the evaluation of a configurable set of rules.Both user-routing rules and operation-routing rules are supported.
- Load Balancing: A round-robin mechanism selects the target server for the next connection to be routed to, considering the results of the rule evaluation of the request routing.
- Failover: If a target server is unavailable due to connection error, the request fails over to the next appropriate target server according to configuration settings.
- Rule-based LDAP request and result rewriting: The rewriting feature includes schema and attribute mapping to provide schema compatibility that is helpful for clients that use an outdated schema or that contain a hard-coded schema.This feature can also be used to implement an additional access control layer.The actions that users are allowed to perform can be defined by denying certain requests or hiding or blocking critical subtrees, entries, and attributes of search operations and their results.It also allows protecting the directory infrastructure from specific users.

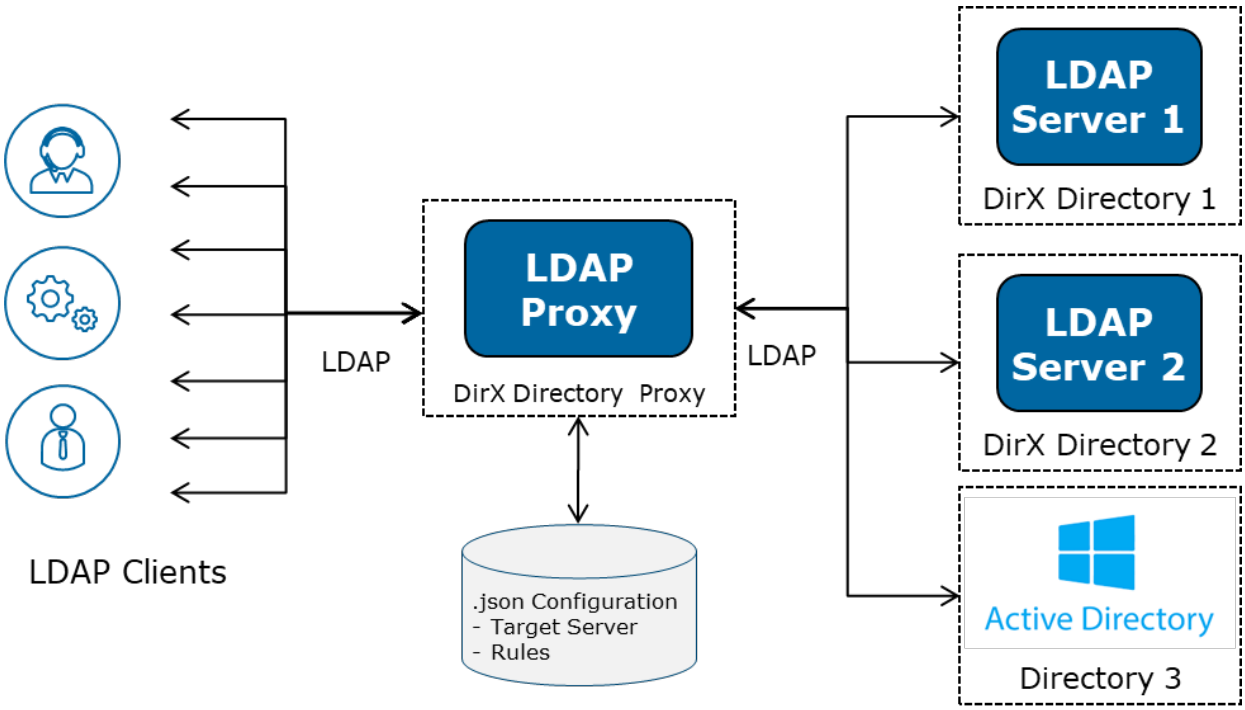

Figure 3. DirX Directory LDAP Proxy in a Heterogeneous Environment

1.3. DSA

The DirX Directory DSA is the X.500 component of the DirX directory service: its main function is to support the directory protocols DAP, DSP and DISP and to act as the "query manager" for the DBAM database system.The DSA runs in its own server process and, like the LDAP server process, is accessible via RPC to support operations from co-located or remote administration tools that cannot be performed with DAP.

The DirX Directory DSA has no hard limits in the following areas:

- Number of simultaneous associations (DAP, DSP)

- Number of simultaneous users (DAP, DSP)
- Number of entries
- Number of naming contexts
- Number of subordinate references

Any limits applied in these areas are a factor of the host operating system. However, DirX Directory administrators can in some cases apply administrative limits using the DirX Directory administration tools.

1.3.1. X.500 1993 Standards Conformance

The DirX Directory DSA provides a full-scale implementation of the X.500 1993 standards, including the directory information model and schema, and the X.500 directory protocols.

1.3.1.1. Directory Information Model Support

The DSA conforms to the information model specified in the X.500 1993 standards. This support includes:

- Collective attributes - identical attributes of several directory entries which are accessed like normal attributes but which are stored only once and managed at a central location. Note that DirX Directory does not support the use of collective attributes in filters.
- Access control rules for parts of a directory tree (administrative areas)
- Attribute subtyping - the option of accessing specific attributes by referencing generic attributes (for example, Private TelephoneNumber as a subtype of Telephone Number)
- Operational attributes - attributes used for directory-internal purposes or which, like the timestamp, are generated by the directory itself

1.3.1.2. Directory Schema Support

The DirX Directory DSA supports a flexible and customizable directory schema, including support for:

- All attribute types and syntax rules defined in X.520
- All object classes and matching rules defined in X.521, including support for phonetic rules for finding entries with similar values (DirX Directory uses the soundex coding rules for phonetic matching.)
- All object classes, attribute types, and syntaxes defined by LDAP RFCs 2252, 2256, and 2798 (inetOrgPerson)
- All security objects and attributes defined in X.509 1997 edition

The DirX Directory DSA also permits the definition and administration of user-defined object classes and attributes.

1.3.1.3. X.500 Protocol Support

The DirX Directory DSA is accessible over DAP using any standards-compliant DUA. These DUAs are not required to support the X.500 1993 protocol features, although some of the DSA's capabilities will only be available to DUAs that do provide this support. Remote DUAs access the DirX Directory DSA either directly or through other DSAs.

The DirX Directory DSA supports the operations of the X.500 standards not only in accessing the local database, but also in passing on queries to other DSAs and consolidating the result, in accordance with X.500 procedures for distributed operations.

The DirX Directory DSA supports shadowing with DISP, which permits the automatic shadowing of naming contexts and subtrees within naming contexts from one DSA to another. The DSA can inter-operate in shadowing with other standards-compliant DSAs that support DISP.

The DirX Directory DSA supports the X.500 DAP, DSP, and DISP directory protocols directly over TCP/IP through the Internet Directly Mapped (IDM) protocol, as specified in ISO/IEC 9594-5: 2001 (E). IDM is a connection-oriented and fully asynchronous "convergence" protocol layer that operates between the X.500 protocols and TCP/IP. IDM enables the support of X.500 directory service elements without the implementation overhead of supporting the full OSI stack. It also enables DirX Directory DSAs to use SSL/TLS-protected DAP, DSP and DISP connections, known as secure IDM (IDMS) in DirX Directory.

1.3.2. DSA Implementation Features

The main features of the DirX Directory DSA server implementation include its implementation as a multi-threaded server and its ability to export the directory database into LDIF files.

1.3.2.1. Multi-Threaded Architecture

Like the LDAP server, the DirX Directory DSA is implemented in a single process using a threads-based architecture.

1.3.2.2. LDIF Support

An important feature of the DirX Directory DSA is the ability to export its database into standard LDAP Directory Interchange File format (LDIF) and to import this file format into the DBAM database.The DSA's ability to import LDIF file format allows it to support bulk-loading of directory content via LDIF files into the DBAM database.Its ability to export into LDIF file format allows it to support the synchronization of directory change events from the DBAM database to other non-DirX Directory services through the generation of LDIF change files.More information about the DSA's LDIF mechanism is provided in the section on Replication Services.

1.3.3. DSA Security Support

The DirX Directory DSA supports the authentication and access control models defined in the X.500 1993 standards.In addition to the X.500 security model support, the DSA supports DSA policy operational attributes that can be used to establish trust relationships between

DSAs.The DSA can also store certificates and revocation lists generated by Certification Authorities (CA) as part of its support for the DirX Directory X.509 Public Key Infrastructure (PKI).The DSA can also use the SSL/TLS protocol for encrypted X.500 DAP, DSP and DISP communication over IDM (IDMS).The section on Security Services provides more information about these features.

1.3.4. DSA Monitoring and Auditing

The DSA supports a Management Information Base (MIB) based on RFC 2605 and RFC 2788 for the storage of usage statistics about the DSA's operation.The DirX Directory administration tools provide access to this MIB for viewing and evaluating the usage information.The DSA also supports auditing and diagnostic logging of its operations.The section on "Monitoring, Auditing, and Logging Services" provides further details about these services.

1.4. DirX Directory Progsvr

DirX Directory Progsvr is a specialized server for the execution of procedures defined by LDIF policies.The implementation is small sized, high-performance, and secure.It runs as a separate process that must run on the same server as the DirX DSA process.The DirX Directory Progsvr is enabled by default.(See the environment variable **DIRX_USE_PROGSVR** in the *DirX Directory Administration Reference* for details.)

1.5. Progsvr Implementation Features

The DirX Directory Progsvr is implemented using a thread-based architecture.It contains a main thread that listens for incoming RPC requests, RPC threads, and a pool of worker threads for the execution of commands.Administrators can adjust the number of worker threads according to the available resources and the number of expected LDIF agreements.The recommended thread count is the number of LDIF agreements using LDIF policies that specify procedures to be executed on generated LDIF files.(See the section on shadow operational binding (SOB) policies in the *DirX Directory Syntaxes and Attributes* document for details.)

1.5.1. Progsvr Security Support

The messaging protocol between the Progsvr and the DSA performs several security checks, including sequence numbering and a proprietary hashing algorithm to ensure that only the DSA can execute a command or retrieve executed commands results.In addition, commands are written to a file with a unique name in a protected folder and only the name of the generated file is sent to the Progsvr.Transmitting the file name instead of the actual command provides security to the Progsvr against man-in-the-middle attacks, as intercepting the RPC call only reveals knowledge about a file name but not about the actual command.

1.5.2. Progsvr Logging

The Progsvr provides diagnostic logging of its operations.The section on "Monitoring,

Auditing, and Logging Services" provides further details about this topic.

1.6. The DirX Directory HTTP Server

The DirX Directory HTTP server implements a custom API for accessing the directory service using the HTTP protocol.The implementation is small sized, performant, and secure.The DirX Directory HTTP server is enabled by default.(See the environment variable **DIRX_USE_HTTP** in the *DirX Directory Administration Reference* for details.)

1.6.1. HTTP Server Implementation Features

The DirX Directory HTTP server acts as a protocol translator from the custom JavaScript Object Notation (JSON) schema to LDAP requests.This implementation preserves all the benefits of the LDAP server, like the LDAP cache and user policies, while providing easy access from a web browser via HTTP to the data stored in the directory service database.The DirX Directory HTTP server uses a built-in Swagger UI component to provide an easy to use, industry standard, interactive user document where users can read about the available operations and configuration options and manage the interface.

1.6.2. HTTP Server Security Support

Messaging between the DirX Directory HTTP server and the LDAP server can be configured to use SSL/TLS to provide a secure channel between these servers.The HTTP server can also be configured to provide access though the HTTPS protocol using SSL/TLS on the client side.

1.6.3. HTTP Server Logging

The DirX Directory HTTP server provides diagnostic logging of its operations.The section on "Monitoring, Auditing, and Logging Services" provides further details about this topic.

1.7. The DBAM Database

The Directory Basic Access Method (DBAM) database is a database kernel specialized for the handling of directory data and directory applications environments.

1.7.1. Implementation Features

The DBAM design makes extensive use of caching and indexing mechanisms that have been optimized for directory access and directory tree modeling.The design also supports:

- High-performance name resolution: finding an entry based on its distinguished name
- High performance name retrieval: finding the distinguished name of an entry from its internal database record

1.7.1.1. Index Management

DBAM supports two types of attribute type indexing schemes to its database:

- A simple prefixed B*-tree implementation for indexing attribute types. B*-tree implementations provide for fast search operations for queries on attributes with a large number of possible values (such as customer name or telephone number).
- Optimized bitmapped indexing of attribute types. A bitmapped index on an attribute creates a bit string of references for each value in the indexed attribute. Bitmapped indexing provides optimal performance for complex queries in LDAP search operations.

Supporting both indexing schemes maximizes query resolution performance for all types of directory search operations.

1.7.1.2. Directory Tree Modeling

Directory systems are intrinsically required to manage a directory tree data structure that is not easily mapped onto traditional relational data models or on a B*-tree, and to manage the data with optimal performance. The DBAM database addresses this challenge by mapping the directory tree directly to the physical disk.

1.7.1.3. Cache Management

The DBAM database supports a high-performance cache for buffering portions of the database on disk in main memory. DBAM cache management is directory-specific: the goal is at most one disk I/O access for one directory access (read or search).

To achieve this goal, DBAM caches the upper parts of the directory tree and some parts of the attribute index B-trees permanently in cache memory and uses replacement policies based on time-to-live (TTL) for other data.

1.7.1.4. Direct Data Access

The DBAM database manages the persistent storage of data on the physical, or "raw" device(s), rather than through the file system. This design streamlines access to the data in the cache and on disk by reducing the number of data copy operations. Data access goes directly from the DSA process to the DBAM cache, then directly to the data on disk. In contrast, databases that use standard components such as ISAM or the operating system file system cache cannot directly access the cache without having to copy the data first.

The DBAM database is designed for high availability: it can support a hardware configuration of RAID devices as the DBAM data store. This design also offers a huge storage capacity potential, for many millions of entries.

1.7.1.5. Contiguous Disk Allocation

Sequential I/O operations are considerably more efficient than random I/Os because they significantly minimize disk arm movements. However, this feature can only be leveraged when the distribution of logical records over the physical disk can be completely controlled. An operating system’s file system maps to physical records spread non-contiguously over the disks. In contrast, DBAM directly controls the physical placement of the logical records on the disks, which permits directory data to be clustered on disk according to their logical relationships. There is no fragmentation of data over physical disk clusters with the DBAM model. This optimization is critical for obtaining high performance of bulk data operations

such as initial loading, indexing, and database reorganization.

1.7.1.6. Transaction Support

The DBAM database provides transaction support for all directory modification operations with consistency checks and rollbacks.

1.7.2. DBAM Database System Components

A typical database system consists of the following components:

- A query manager, which takes a high-level query or data manipulation request (such as a schema modification or a modification to the real data) and turns it into a sequence of requests to the storage manager for the stored data.
- A storage manager, which retrieves or modifies the stored data. The storage manager controls storage on the disk directly, not through the operating system's file system.
- A transaction manager, which is responsible for the integrity of the database system.

A storage manager typically consists of two functional elements:

- A volume manager, which keeps track of the location of data on disk and obtains or writes back the requested block(s)
- A cache manager, which obtains accessed blocks of data from the volume manager and stores them in memory pages

The transaction manager enforces the proper execution of database transactions by enforcing the principles of atomicity, consistency, isolation, and durability (ACID) on all database transactions, as follows:

- Atomicity means that all transactions are performed completely or not at all (all or nothing)
- Consistency means that a transaction transforms a consistent database state into another consistent state
- Isolation means that transactions are isolated from each other. Although many transactions run concurrently, any given transaction's updates are concealed from all other transactions until the transaction is committed.
- Durability means that once a transaction commits, its updates survive, even if there is a subsequent system failure

To enforce these principles, the transaction manager manages:

- Database object locking, to lock other transactions from access to the object
- Transaction commitment, which is the process of making a successful transaction permanent in the database
- Transaction rollback, which is the process of undoing all the updates made for an unsuccessful transaction to maintain database consistency
- Transaction recovery, which is the process of recovering (from logs) any committed

transactions whose updates have not made to the physical stored data due to a system failure

The following figure illustrates the typical components of a database system.

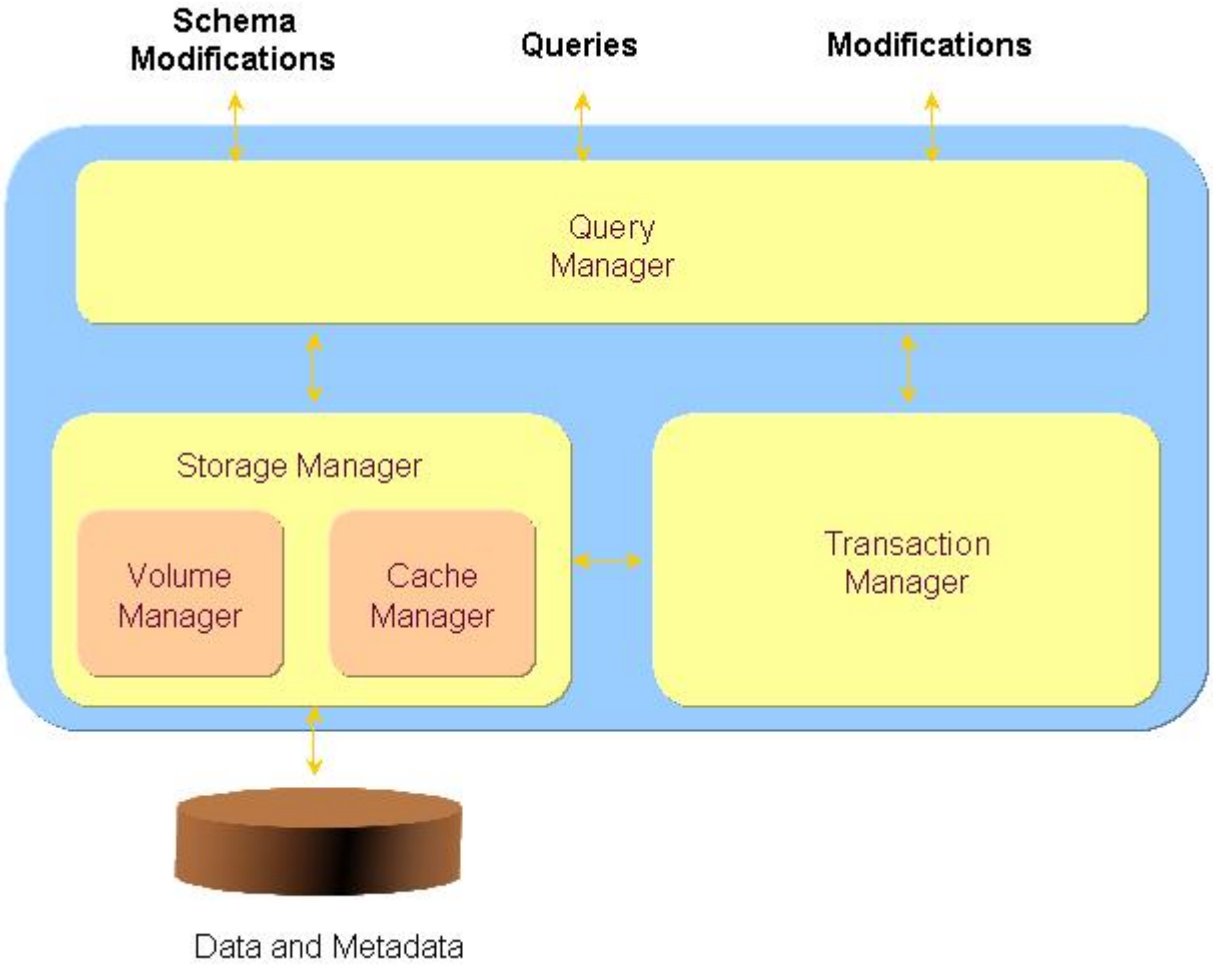

Figure 4. Database System Components

The following figure illustrates the DBAM database system and how it maps to the typical system just described.

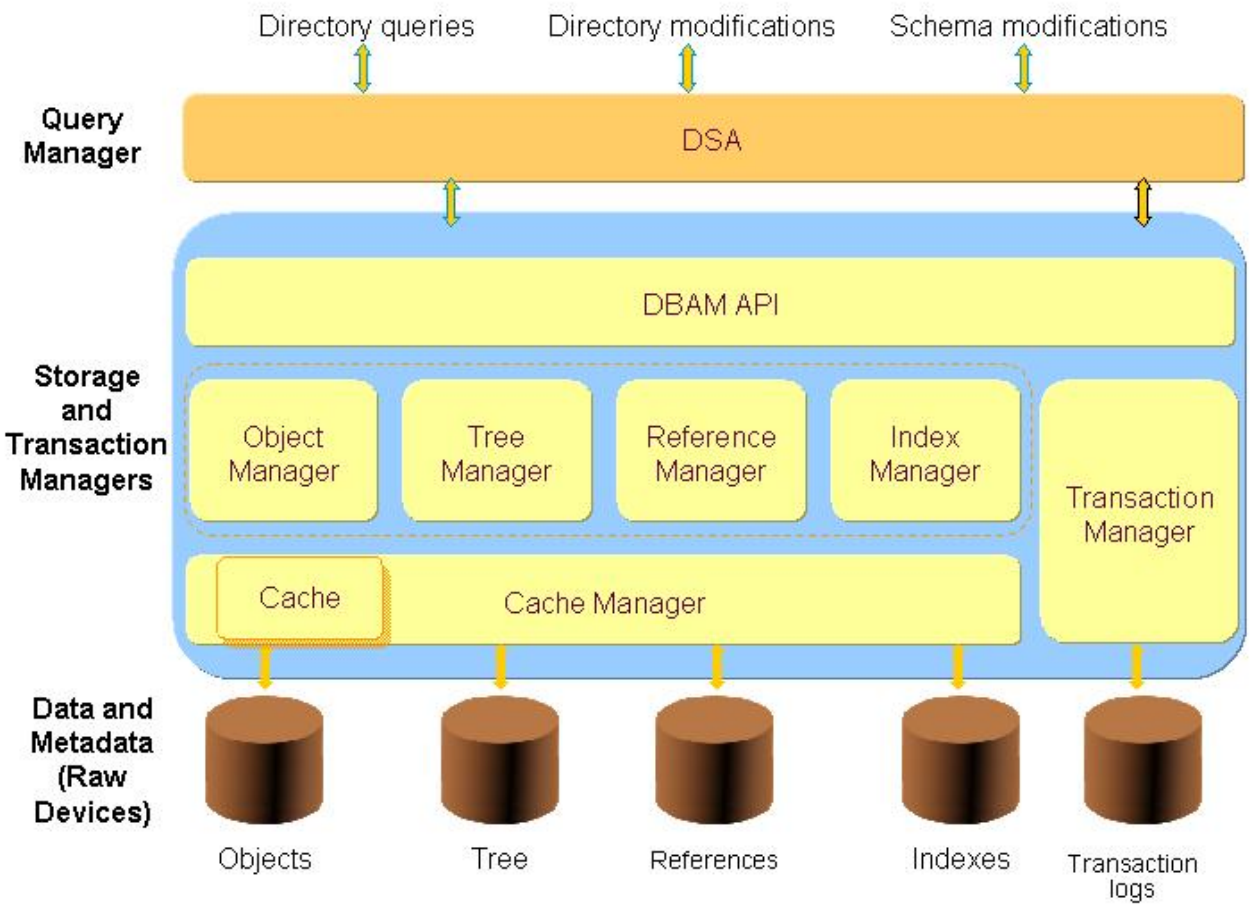

Figure 5. DBAM Components

1.7.2.1. DSA and DBAM API Functions

As illustrated in the figure, the DirX Directory DSA is the DBAM database's query manager. It communicates with the storage manager and the transaction manager over a proprietary database API called the DBAM API, and translates directory queries and modifications to a series of requests to the storage manager.

The DBAM API is the communications path between the DSA—which can be considered as a DBAM "application"—and the storage and transaction managers. DirX Directory administration tools that need to access the DBAM database directly also use this API.

1.7.2.2. Storage Manager Functions

Like the typical database system, the DBAM storage management model comprises a volume manager and a cache manager. The volume manager in the DBAM model consists of separate functions for the management of specific types of directory data, including:

- An object manager, for controlling the storage and retrieval of actual, or "real" directory objects
- A tree manager, for controlling the storage and retrieval of the hierarchical relationships between directory objects
- A reference manager, for controlling the storage and retrieval of object reference

elements called "pseudo objects" and bit string references of bitmapped attribute indexes

- An index manager, for controlling the storage and retrieval of attribute value indexes

The storage manager and the transaction manager have direct access to the directory data on the physical (or "raw") disk devices. They do not go through the operating system's file system to access the data.

1.7.2.3. Transaction Manager Functions

Like the storage manager, the DBAM transaction manager has direct access to the data on the physical disks. In addition, the cache manager is tightly integrated with directory transaction handling and logging functions. The following figure illustrates the process of transaction management and storage.

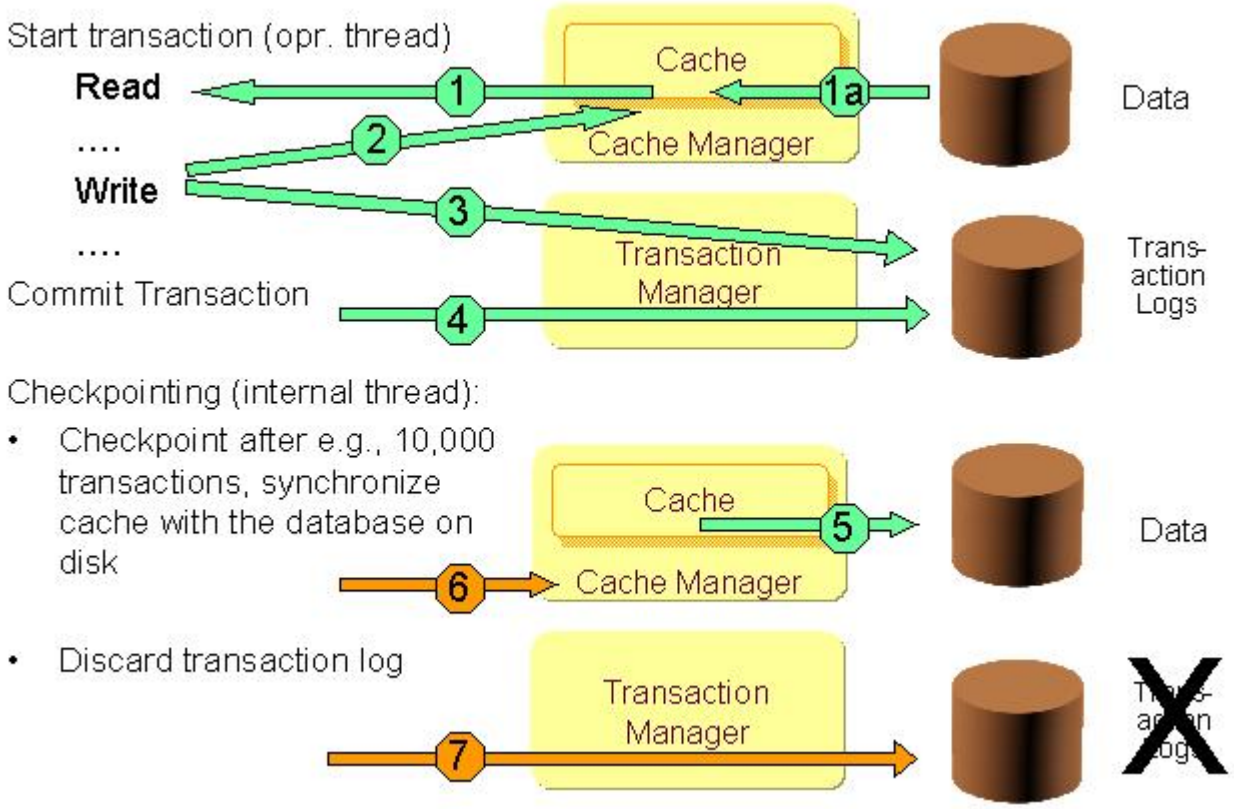

Figure 6. Transaction Management and Storage

As illustrated in the figure:

1. The cache manager satisfies read requests directly from the cache, if the data is available there (step 1); otherwise, it retrieves the data, stores it in the cache, and presents the data to the volume manager (step 1a).
2. For a write operation, the volume manager writes the updates first to the cache (step 2).
3. The transaction manager writes the same updates to the transaction logs persistently on disk (step 3).
4. The transaction manager commits the transaction and writes the termination mark

into the transaction logs on disk (step 4).

5. At a checkpoint interval (for example, after 10,000 transactions) the cache manager (in an internal thread, step 5) writes the committed update to disk in the process of synchronizing the data in the cache with the database on disk (step 6) and notifies the transaction manager, which discards the transaction log for the committed update (step 7).

1.7.3. DBAM Management and Monitoring Services

DirX Directory provides utilities for the following DBAM management tasks:

- Configuring the database
- Post-indexing one or more attributes for optimizing performance
- Saving, restoring, and recovering the database
- Checking the database consistency

The section on the DirX Directory administration tools provides more information about these utilities.

1.8. DirX Directory Security Services

Advanced, state-of-the-art security is a critical component of a carrier-grade directory service.DirX Directory addresses this requirement by supporting the following levels of security:

- "Wire-based" security for traffic across insecure networks like the Internet
- Authentication, to identify the originator of a query to the directory service
- Access control, to restrict access to authorized users
- Server-side policies, for local security management
- Two-factor authentication, for improved security for users

The next sections describe how DirX Directory implements these levels of security for LDAP and X.500 connections.

1.8.1. LDAP Security Features

DirX Directory supports the following types of security for LDAP client access to the directory service:

- Standard LDAPv3 authentication
- Authenticated, encrypted LDAP communications over SSL/TLS
- TCP/IP address and username filtering and the denial of anonymous user access

1.8.1.1. LDAPv3 Authentication

The LDAP server supports anonymous and simple unprotected authentication by LDAP clients—authentication by username and a password in plain text—as defined in the LDAPv3 standards. The LDAP server also supports encrypted LDAPv3 authentication (anonymous and simple unprotected) via the SSL/TLS protocol, which is described in the section "SSL/TLS Security".

Administrators can also configure an LDAP server to deny access to the directory by anonymous users, thereby limiting directory access to authorized users.

1.8.1.2. SSL/TLS Security

DirX Directory supports the Secure Socket Layer 3.0/Transport Layer Security 1.0/1.1/1.2 (SSL/TLS) protocols for secure LDAP client-server communications between LDAP clients and LDAP servers. The SSL/TLS protocol provides the foundation for authenticated and encrypted LDAP client-server communications over the Internet and other TCP/IP networks. SSL/TLS session encryption ensures LDAP traffic confidentiality at the network level.

An encrypted SSL/TLS connection requires all information sent between a client and a server to be encrypted by the sending software and decrypted by the receiving software, thus providing a high degree of confidentiality. All data sent over an encrypted SSL connection is also protected with a mechanism for detecting tampering.

LDAP authentication over SSL/TLS can be:

- Authentication by name and password (anonymous and simple unprotected) through an encrypted SSL connection, called "server-side SSL". Server-side SSL enables encrypted LDAPv3 authentication between LDAP clients and servers.
- Authentication by public-key user certificate, called "client-side SSL" or "strong LDAP authentication" through an encrypted SSL connection. Strong authentication can be server-based or client-based:
- Server-based authentication allows a user to confirm a server’s identity. SSL-enabled client software can use standard techniques of public-key cryptography to check that a server’s certificate and public ID are valid and have been issued by a Certification Authority (CA) listed in the client’s list of trusted CAs. This confirmation might be important if the user, for example, is sending a credit card number over the network and wants to check the receiving server’s identity.
- Client-based authentication allows a server to confirm a user’s identity. Using the same techniques as those used for server authentication, SSL-enabled server software can check that a client’s certificate and public ID are valid and have been issued by a CA listed in the server’s list of trusted CAs. In addition, the client’s identity from the certificate can be used as a strong authentication in terms of the directory protocols. This form of authentication might be desired if the server, for example, is a bank sending confidential financial information to a customer and wants to check the recipient’s identity.

1.8.1.3. Proxied Authorization Control

DirX Directory supports the proxied authorization according to the RFC 4370. Proxy authorization allows a client to request that an operation is processed under a provided authorization identity instead of the current authorization identity associated with the authentication identity provided with the DN in the context of the bind operation. The Proxy Authorization Control provides a mechanism for specifying an authorization identity on a per-operation basis. This enables a user to perform operations efficiently on behalf of multiple users. The model of trust in such a proxy environment is a Single-Sign-On scenario: The LDAP client – typically service like applications – has performed the authentication of the end user and uses the proxy authorization control to transport the authorization ID to the DSA. The DSA trusts this authorization ID based on the policy stored in a special subentry.

1.8.1.4. IP Address and User Filtering

Administrators can configure an LDAP server to grant access to LDAP clients on the basis of their IP address and/or user DNs and can also configure it to deny access to LDAP clients based on these same criteria. The lists of specified "permitted" and "denied" IP addresses and/or user DNs are stored in the LDAP server configuration subentry and can be displayed by viewing the LDAP server's MIB static information table.

1.8.2. HTTP Security Features

DirX Directory supports the following types of security for HTTP client access to the directory service:

- Standard LDAPv3 authentication methods (anonymous and simple bind)
- Encrypted HTTPS communication over SSL/TLS

1.8.2.1. Client Authentication

The DirX Directory HTTP server supports anonymous and simple authentication—authentication by user DN and a password—by HTTP clients. The authentication can be performed using plain-text HTTP or encrypted HTTPS.

1.8.2.2. SSL/TLS Security

DirX Directory supports the SSL/TLS protocols for secure HTTP client-server communications between HTTP clients and the DirX Directory HTTP server and for secure HTTP Server—LDAP server communications. The SSL/TLS protocol provides the foundation for encrypted communications over the Internet and other TCP/IP networks. SSL/TLS session encryption ensures HTTP and LDAP traffic confidentiality at the network level.

An encrypted SSL/TLS connection requires all information sent between a client and a server to be encrypted by the sending software and decrypted by the receiving software, thus providing a high degree of confidentiality.

1.8.3. X.500 Security Features

DirX Directory X.500-based security consists of the following elements:

- Support for the X.500 1993 security standards for authentication and access control
- Support for the X.509v3 1997 standards for secure management of public key infrastructure (PKI) objects
- Support for security policies through DSA server-side data structures
- Support for encrypted X.500 DAP, DSP and DISP communication via the SSL/TLS protocol over the IDM protocol stack, known as secure IDM (IDMS)

1.8.3.1. X.500 Authentication

DirX Directory supports the following X.500-prescribed methods for the authentication of DUAs to DSAs using DAP and for the mutual authentication of DSAs using DSP or DISP:

- Anonymous authentication ("anonymous" "anonymous" user and plain text password strings)
- Simple unprotected authentication - authentication with a user name and a plain text password
- Simple protected authentication - authentication with a user name and an encrypted password. The algorithm used is derived from the "RSA Data Security, Inc. MD5 Message-Digest Algorithm".
- External authentication – authentication using an external service with an external authentication ID and a password.

DirX Directory supports strong authentication for LDAP clients over SSL/TLS. Since nearly all directory access is through LDAP, this implementation of strong authentication should provide a sufficient level of security, if it is required.

The plain text passwords used in simple unprotected authentication are stored as attributes of the user entry in encrypted form to protect them against unwanted examination (for example, by hex-dumping the database).

DSA passwords used for DSA-to-DSA authentication are usually not stored in directory entries. Instead, administrators supply them when they establish an "operational binding agreement" for a DSA, which then stores them in encrypted format. The DSA's own passwords are similarly stored (a DirX Directory DSA can maintain a specific password for each DSA with which it co-operates using DSP or DISP).

In a distributed directory—where multiple DSAs manage pieces of the directory information tree (DIT)—simple client authentication proceeds as follows:

- The client is authenticated by the first DSA it encounters.
- The client's identity is passed from one DSA to the next when a chaining operation occurs.
- Each DSA maintains a list of DSAs that it trusts in its DSA policy operational attribute and grants the appropriate access rights to clients that have been authenticated by one

of its trusted DSAs. (The section on policy-based security discusses policy attributes in more detail).

- Clients who have been authenticated by an untrusted DSA are granted the access rights that the DSA gives to anonymous clients.

1.8.3.2. X.500 Access Control

The DirX Directory DSA supports both Basic Access Control (BAC) and Simplified Access Control (SAC) at the entry and attribute level, as defined in the X.500 1993 directory administration model. All three Access Control Information (ACI) operational attributes are supported to control access to entries, subentries, and attributes:

- Prescriptive-ACI
- Entry-ACI
- Subentry-ACI

Administrators can use DirX Directory administration tools to access these operational attributes (subject to their own access control).

The DirX Directory DSA does not currently support access control on specific values of an attribute. For example, different access rights to different values of a recurring attribute cannot be defined.

Because all directory operations must retrieve access control information, and because quick access to this information is essential for optimum directory performance, the DirX Directory DSA stores all access control information in local cache memory and uses short-cut techniques to access the information that is relevant for a given operation. In addition, the DSA keeps each entry's access control information at the beginning of its DSE so that it can be easily and quickly accessed.

Directory access control procedures require not only the identity of the user but also an assessment of the quality of authentication. The assessment can include factors such as:

- The identity of the requestor
- The original means of authentication (if known)
- Whether the DSAs that have forwarded the request are trusted DSAs

The DirX Directory DSA implements procedures to assess the reliability of the identification process based on the originator and authentication-level elements of the DSP protocol as well as on locally available information. These procedures are regulated by the DSA policy operational attribute, which allows DirX Directory administrators to have close control of the authentication process.

1.8.3.3. X.509 PKI Support

DirX Directory supports the X.509v3 (1997) standard for the secure management of Public Key Infrastructure (PKI) certificates and their standardized extensions. As a result, DirX Directory supports the storage of certificates and revocation lists produced by Certification Authorities (CAs). DirX Directory has been successfully tested with products from leading

CA vendors.

1.8.3.4. Policy-Based Security

DirX Directory DSAs support a set of policy attributes, held in the root DSE, which can be used to regulate specific operational behavior for the DSA. Two of these attributes—the DSA policy attribute and the user policy attribute—can be used to apply security policies for the local DSA that relate to other DSAs and to the users of its directory information.

The DSA policy attribute controls a DSA's methods of interaction with other DSAs. Security-related policies that can be defined in this attribute include:

- Whether a remote DSA is a "cooperating" or "trusted" DSA for chaining operations
- The accepted authentication method for DSA-to-DSA authentication
- The passwords to be used in DSA-to-DSA authentication
- Whether read and/or modify operations are permitted over chaining

The user policy attribute specifies a set of limits to be placed on specific users or on groups or on users in particular subtrees of the DIT, over and above any access control permissions that they may have, for example:

- The users' required authentication method
- Whether the users can initiate chained operations or modify operations
- Timeout, size and priority limits

Administrators use DirX Directory administration tools to define DSA and user policy operational attributes for a local DSA. (These attributes are not accessible over the standard X.500 protocols.)

DirX Directory DSAs also allow administrators to define policies for user password-based authentication and for the modification of user passwords, including:

- Password syntax-checking policies, which can prevent users from choosing passwords that are easy to guess or from re-using password values
- Password aging policies, which force users to change their passwords on a regular basis
- Account lockout policies, which permit administrators to react to a series of failed logins

Password policies are stored in the password policy subentry. (See "Attributes of the Password Policy Subentry" in the *DirX Directory Syntaxes and Attributes* for details.) During initialization, the local DSA reads the password policy subentry to determine the password policies in force. The DSA uses per-user password policy-specific operational attributes to track and respond to the state of each user's password against the specified policies.

DirX Directory also supports proxied authorization according to RFC 4370. The proxy authorization control subentry stores the policy that controls which authenticated user may use the respective proxy authorization control. Based on the attributes of this subentry the DirX Directory DSA performs its decision whether to grant or deny the usage of authorization proxy for a particular operation. (See "Attributes of the Proxy Authorization

Control Subentry” in the *DirX Directory Syntaxes and Attributes* for details.)

Like the other subentries (access control, collective attribute, and schema), password policy subentries can be replicated to shadow DSAs via DISP to provide homogeneous password policies across shadowed DSAs. The DirX Directory password policy implementation is based on the IETF draft RFC "Password Policy for LDAP Directories" (document: draft-behera-ldap-password-policy-07.txt).

1.8.3.5. Encrypted X.500 Communication over IDM (IDMS)

DirX Directory supports encrypted X.500 DAP, DSP and DISP protocol exchanges over the IDM stack (but not the OSI stack) through the application of the SSL/TLS protocols over the transport layer.While SSL/TLS permits mutual authentication between communicating peers, its purpose in secure IDM (IDMS) is to encrypt the data exchanged between communicating peers.IDMS provides the foundation for encrypted X.500 communications between DUAs and DSAs over DAP and between DSAs over DSP (chaining) and DISP (replication), ensuring X.500/IDM protocol traffic confidentiality at the network level.

1.8.4. Two-factor Authentication (2FA)

DirX Directory also provides security features that are not restricted to a specific protocol.One of these features is two-factor authentication (2FA). 2FA enhances service security by requiring two forms of identification for user access to a service via simple binds.DirX Directory implements the time-based one-time password (TOTP) type of two-factor authentication for DirX Directory users.

In a TOTP-based 2FA DirX Directory configuration, TOTP 2FA-enabled DirX Directory users are expected to supply their passwords plus a 6-digit TOTP issued by a TOTP authenticator app on the user side when binding to a TOTP 2FA-configured DirX DSA.The user TOTP is based on the current time and a secret key shared between the DSA and the user.

For general information about TOTP 2FA, see RFC 6238: TOTP: Time-Based One-Time Password Algorithm.For information about configuring TOTP 2FA for DirX Directory, see the section “Using Two-Factor Authentication” in the *DirX Administration Guide*.

1.9. Replication Services

DirX Directory provides two mechanisms for the replication of directory information:

- LDIF file synchronization
- X.500 shadowing

In addition, DirX Directory augments the X.500 shadowing model to support a "floating master" replication configuration that can provide for uninterrupted service in the face of system maintenance or failure.DirX Directory supports both synchronous and asynchronous shadowing modes for use in floating master replication configurations.

1.9.1. LDIF File Synchronization

To provide further integration with LDAP-enabled applications and services, DirX Directory supports the high-performance export of DSA database content into LDAP Data Interchange Format file format (LDIF), as defined in RFC 2849.

Administrators can use the DirX Directory administration tools to set up the DirX Directory DSA to export the entire DirX Directory database or selected subtrees directly into files in LDIF content format. They can also set up the DSA to export changes to the database either periodically or on change into files in LDIF change format. Triggers can be associated with the creation of an LDIF file to initiate further processing.

Exporting DirX directory content into LDIF file format permits administrators to carry out bulk data transfers, integrate DirX Directory with other non-DirX directories, and especially to use DirX Directory as a "meta directory store".

1.9.2. X.500 Shadowing

Shadowing is the controlled replication of directory information. The X.500 1993 Directory Standards define shadowing as a standard method of replication of information from one DSA to another. Because shadowing ensures that directory information is available in more than one place, it is an important tool for balancing directory service communications traffic and for achieving a higher reliability of service.

Shadowing results from a shared shadowing agreement between two DSAs: the information source is the shadow supplier and the recipient is the shadow consumer. Shadowing uses its own specialized protocol, called Directory Information Shadowing Protocol (DISP), to pass information from the shadow supplier to the shadow consumer, passing either all the information (total refresh) or just the information that has changed since the last protocol exchange (incremental refresh). The shadow relationship is implemented as a Shadow Operational Binding (SOB).

DirX Directory DSAs support the DISP shadowing protocol, as described by the X.500 1993 standards, for the replication of information between two DSAs, including access control, collective attribute and schema information. DirX Directory DSAs support the main features of shadowing:

- Units of replication can be shadowed, provided that they comprise one naming context including subordinate references. DirX Directory supports selection of entries by object class within the subtree or attribute selection.
- Supplier-initiated shadowing is supported (coordinate-shadowupdate and request-shadow-update)
- Both incremental-refresh and total-refresh operations are supported
- Periodic update and update on change are supported

DirX Directory DSAs also support shadowing without a total refresh. With this method, the administrator performs the total update by saving and restoring the complete directory database. A shadowing agreement is then established to propagate incremental changes to the database over DISP.

1.9.3. Floating-Master Replication

"Floating master" replication is a technique for providing high availability for all directory service operations. A floating master configuration permits the directory service to be "always available" and ensures that there is a master for directory update operations during maintenance periods.

Unlike multi-master replication architectures, floating-master replication preserves the concept of a unique owner of the directory data. With floating-master replication, there is only one master of directory information at any given time, so data ownership is unambiguous. This design avoids the potential for data collision inherent in the multi-master configuration.

To create a floating master configuration, the DirX Directory administrator creates a clone of the master DSA by replicating all of the directory information residing on the master to a specific shadow DSA that can take over the master's role if it fails or needs to be maintained. The replication can be performed by media (via **dirxbackup** save and restore) or by the shadowing protocol.

If the master DSA subsequently fails or needs to be taken out of service, the administrator can use the DirX Directory administration tools to switch the shadow DSA to the master role. The tools force the propagation of all outstanding incremental updates for all enabled shadowing agreements to the new master and automatically update the shadowing agreement information to reflect the new master-shadow DSA configuration to all other replicas in the configuration.

Users can continue to retrieve and update the directory information on the new master DSA while the old master is serviced (or replaced). The administrator can then restore the old master to the configuration as a shadow DSA and optionally switch it back to master status without interrupting directory service operation.

1.9.4. Synchronous and Asynchronous Shadowing

Floating-master replication supports two kinds of shadowing protocols:

- Asynchronous shadowing, where a DAP or LDAP client's update operation returns immediately after the master DSA commits the operation and writes it to the journal.If the master DSA fails, this protocol can lead to loss of recent update operations at the consumer DSAs.
- Synchronous shadowing, where a DAP or LDAP client's update operation does not return until the master DSA and all synchronous consumer DSAs have committed the operation.If the master DSA fails, acknowledged update operations to the DAP/LDAP client are safely stored at the synchronous consumer DSAs and there is no data loss.

Synchronous shadowing provides for high data integrity between master and shadow DSAs even in the event of a master failure.Synchronous shadowing makes DirX Directory shadow DSAs suitable for DAP and LDAP clients that perform read operations immediately following successful modify operations.As long as there is no network outage, the clients receive the correct result.Because it ensures data synchronicity, a synchronous shadow DSA is an ideal candidate for the role of a stand-by master in a floating master

configuration.

In contrast to synchronous shadowing, which always replicates the entire master DIT and supports a lower rate of update operations between supplier and consumer DSAs in favor of synchronized data, asynchronous shadowing provides full flexibility for defining the replication area and allows a high rate of update operations between supplier and consumer DSAs, even over long distances.Asynchronous shadowing allows for total updates via media or DISP, while synchronous shadowing supports only DISP.

1.10. Recovery Services

The DirX Directory DBAM database provides complete transaction support via transaction commitment, rollback, and recovery operations for all directory modifications, as described in the section "The DBAM Database".

The DirX Directory service also provides an administration tool that permits the DBAM database to be saved and restored while the DirX Directory service is online.The section "DirX Directory Administration Tools" in the *DirX Directory Administration Guide* provides more information on the DirX Directory backup and restore utility (**dirxbackup**).

In master-shadow configurations, the DirX Directory service provides recovery from failed master DSAs via the floating-master replication scenario described in the previous section.

If an administrator overwrites DirX Directory working directories—for example, by restoring a hard-disk backup— the DirX Directory service is not able to run.Therefore, it is strongly recommended to use only the DirX Directory backup utility to restore DirX Directory data.

1.11. Monitoring, Auditing, and Logging Services

DirX Directory supports three types of monitoring facilities:

- LDAP server extended operations
- Management Information Base (MIB) monitoring
- SNMPv2 traps
- Audit logging
- Diagnostic logging

1.11.1. Extended Operations

The LDAP server supports extended operations. (See section 4.12 “Extended Operation” in the *Lightweight Directory Access Protocol (v3)*, RFC 4511, June 2006 for details on extended operations.) The LDAP extended operations provided by DirX Directory include:

- Operations for support and monitoring. These operations provide management information base (MIB) table data and other diagnostic data of the DirX Directory server processes. Use the **dirxextop** command or the **Monitoring** view in DirX Directory Manager to perform these operations.

- An operation to change an LDAPv3 connection to the LDAP server from a plain bind to a secure (SSL/TLS) bind. Use the **dirxcp startTLS** command to perform this operation.

1.11.2. MIB Monitoring

The DirX Directory service supports LDAP and DSA Management Information Bases (MIBs) for the storage and retrieval of LDAP and DSA usage statistics.

The DirX Directory LDAP server MIB is based on RFC 2605 and RFC 2780. MIB data is stored in memory only. The MIBs contain static information (such as configuration details) and dynamic information (such as cache hits and resource consumption) about the LDAP server.

The DirX Directory DSA supports the following MIBs:

- The Network Services Monitoring MIB (RFC 2788), which contains the elements common to the monitoring of any OSI network service application. This information includes a table of all the network service applications that can be monitored, a count of the associations (connections) to each application, and basic information about the parameters and status of each application-related association.
- The Directory Service Monitoring MIB (RFC 2605), which covers the portion that is specific to the DSA application. The information contained in this MIB includes the process-related aspects—resource utilization of the DSA—and the network related aspects, for example, inbound-associations, outbound-associations, operational status, DSA operation, and performance.
- The DSA also provides statistical information about the DBAM database in the DBAM MIB, which is a proprietary MIB that contains DirX Directory implementation-specific information. You can access the DBAM MIB tables through LDAP extended operations used by the Monitoring view of DirX Directory Manager and with the **dirxextop** command. For details about the DBAM MIB, see the appendix "DBAM MIB Tables" in the *DirX Directory Administration Reference*.

DirX Directory administrators can use the data in these MIBs to profile the DSA and LDAP server usage and evaluate their configurations. The *DirX Directory Administration Reference* provides further details about LDAP and DSA MIB contents.

1.11.3. SNMPv2 Traps

The DirX Directory service supports sending of Simple Network Management Protocol (SNMP) v2 traps (also called SNMP alarms) according to RFC 1155 and RFC 1905 standards. A trap is an unsolicited message that is used to notify an SNMPv2 entity of a specific situation, for example, that the DSA has been started, or an exceptional situation, like exceeding the maximum duration of an operation. The user must provide an application that can receive these SNMPv2 traps. (See RFC 1155 and RFC 1905 for details.)

1.11.4. Audit Logging

Audit logging is the process of recording information about interactions with the directory service for later use, for example, to analyze directory service traffic or for accounting and

billing purposes. The recorded information includes:

- The session ID, which is used to identify the connection
- The protocol used to access the server
- The user's identity, if known
- The type of directory operation performed, the time is started, its duration, and a summary of its arguments
- The result of the directory operation or the reason for its failure

The DirX Directory audit logging service stores this information as an entry into an audit log file. Both the DirX Directory servers (LDAP server and DSA) support auditing of their operations.

The DirX Directory audit logging service is highly configurable. DirX Directory administrators can select the audit file overflow strategy they want to use (for example, to stop auditing or to wrap or move the audit log file) and configure the level of auditing performed (the operations that are audited and the level of auditing detail). DirX Directory provides an audit decoding and evaluation tool that provides extensive filtering capabilities on the audit information (for example, operation types, connection ID, client TCP/IP address or distinguished name).

1.11.5. Diagnostic Logging

Diagnostic logging is the process of logging trace and exception messages for DirX Directory applications and the DirX Directory servers (Progsvr, LDAP server, and DSA) to isolate and repair problems and failures.

DirX Directory supports two levels of program monitoring:

- System error and status reports
- The monitoring of program execution

The DirX Directory administrator can configure which messages are logged, in which form they are logged, and where they are stored.

1.12. DirX Directory and High Availability Configurations

DirX Directory is intended to operate in environments where availability, scalability, and performance are of the utmost importance.Delivering this kind of platform requires a solution that combines and integrates software and hardware.Although there are various configurations that can be used for such a solution, DirX Directory is best deployed on a RAID-based disk configuration.

Redundant Arrays of Independent Disks (RAID) is a technology that is used to improve data protection and performance when storing large amounts of data.RAID provides a method of accessing multiple individual disks as if the array of disks is one large disk.Data access is

spread over the disks, reducing the risk of losing all data if one drive fails, and improving access time.The RAID system's manager component manages the multiple disk drives so that the system can withstand the failure of any individual member without a loss of data.The RAID Advisory Board specifies a series of RAID levels.RAID level 0 plus 1 (also called RAID level 10) is the recommended RAID level for a DirX Directory installation.

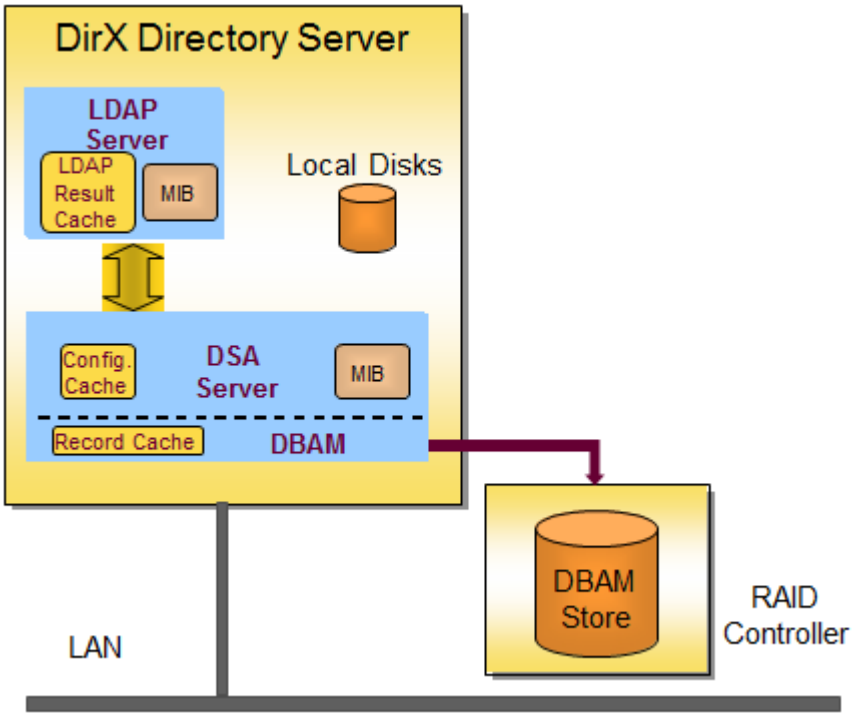

Figure 7. High Availability Configuration

The chapter "Understanding DBAM and Storage Management" in the *DirX Directory Administration Guide* provides more information about RAID disk configurations and planning a fault-tolerant disk configuration for a DirX Directory installation.

1.13. DirX Directory Administration Tools

DirX Directory provides a set of administrative tools and utilities for the configuration and management of the DirX Directory service.For directory administration, DirX Directory provides:

- DirX Directory Manager, a multi server, Java-based LDAP client for directory entry and schema administration
- The **dirxcp** program, a command-line user and administration tool for managing the DSA and LDAP server over DAP and LDAP
- The **dirxadm** program, a command-line "super administrator" tool for managing the DSA and LDAP server over an internal management API

For database loading and backup, DirX Directory provides:

- The **dirxload** program, a command-line tool for bulk-loading data into the DBAM database from an LDIF content file

- The **dirxbackup** program, a command-line tool for saving and restoring the DBAM database to and from a DBAM database archive
- The **dirxmodify** program, a command-line tool for loading an LDIF content or change file into the DBAM database over LDAP (v3 only)

For configuring and initializing the DBAM database, DirX Directory provides:

- The **dbamconfig** program, a command-line tool for creating a profile of a set of DBAM devices
- The **dbamboot** program, a command-line tool for initializing the DBAM database according to a DBAM profile
- The **dbaminit** program, a command-line tool for initializing a file-based DBAM database
- The **dirxconfig** program, a command-line tool that can be used instead of **dbamconfig** and **dbamboot** (or **dbaminit**) to configure and initialize a DBAM database. The **dirxconfig** program calls the **dbamconfig** command and the **dbamboot** command (or the **dbaminit** command) with pre-defined values specified in an input file. Note: if you use the **dirxconfig** command to set up the DBAM database, do not use **dbamconfig** or **dbamboot**/**dbaminit**.

For monitoring the DBAM database, DirX Directory provides:

- The **dbamverify** program, a command-line tool for verifying the consistency of the DBAM database or a DBAM database archive written by **dirxbackup**.
- The **dbamdevinfo** program, a command-line tool for periodically checking the capacity of the database to make sure there is still enough space available for incoming data.

For DirX Directory monitoring, auditing and diagnostic logging, DirX Directory provides:

- **dirxadm** operations for monitoring DSA and LDAP server MIBs and configuring and controlling DSA and LDAP server auditing
- The **dirxextop** program, a command-line tool for running LDAP extended operations for diagnostics and monitoring.The **dirxextop** program performs some of the same monitoring and auditing functions as **dirxadm**, but does so over LDAP instead of over RPC.The **dirxextop** program is useful in DirX Directory installations deployed in firewall configurations because it uses a well-known port that is usually open in a firewall, rather than relying on RPC ports, which are dynamic and cannot be predicted.
- The DirX Directory Manager Monitoring view
- The **dirxauddecode** program, a command-line tool for evaluating DSA and LDAP server audit log files
- The **dirxdumplog** program, a command-line tool for displaying binary directory service trace log messages that provide diagnostic information for DirX Directory support.

The command-line programs and DirX Directory Manager run on both Windows and Linux systems.

The DBAM configuration tools are discussed in greater detail in the chapter "Understanding DBAM and Disk Storage Management" in the *DirX Directory*

Administration Guide.The chapter "Using the DirX Directory Administration Tools" in the *DirX Directory Administration Guide* provides further details about the directory administration and database load and backup tools.The chapter “Monitoring DirX Directory” in the *DirX Directory Administration Guide* provides more information about DirX Directory monitoring, auditing and diagnostic logging tools.The *DirX Directory Administration Reference* provides descriptions of the command-line administration programs and their syntax.The DirX Directory Manager online help and the *DirX Directory Manager Guide* provide details about DirX Directory Manager’s graphical user interface.

1.14. Setting up the DirX Directory Service

The procedure to set up the DirX Directory service consists of two main tasks: setting up the DBAM database and setting up the DirX Directory service.

Setting up the DBAM database consists of the following tasks:

1. Installing DirX Directory
2. Planning the disk configuration for the DBAM database
3. Configuring the DBAM disks
4. Creating the DBAM profiles
5. Initializing the DBAM database

The *Release Notes* describe how to install DirX Directory. The chapter "Understanding DBAM and Storage Management" in the *DirX Directory Administration Guide* describes how to perform tasks 2-5.

Setting up the DirX Directory service consists of the following tasks:

1. Planning the service
2. Setting up the DSA, LDAP and HTTP servers
3. Loading the data
4. Starting the service

Planning the serviceSetting up the DSA, LDAP and HTTP serversLoading the dataStarting the serviceThe chapter "Setting up the DirX Directory Service" in the *DirX Directory Administration Guide* describes how to perform these tasks.

1.15. Maintaining the DirX Directory Service

DirX Directory maintenance tasks include:

- Monitoring the LDAP and DSA MIBs
- Setting up HTTP, LDAP, and DSA audit logging
- Setting up HTTP, LDAP, and DSA diagnostic logging
- Checking the DBAM database

- Performing backup and recovery operations

The chapter "Monitoring DirX Directory" in the *DirX Directory Administration Guide* describes how to perform these tasks.

Glossary

A

abbreviation

A symbolic identifier that represents an OID or a component of a structured attribute. An OID abbreviation can be used for matching rules, object classes, supported application contexts, administrative roles, access control schemes, encoded information types and attribute types. DirX Directory provides a set of abbreviation-element mappings in the **dirxabbr** file, which is provided with the DirX Directory product.

access control

A security mechanism that regulates access to information on the basis of identity.

access point

A DSA name and address that can be used to establish a communication association.

administrative area

A subtree of the DIT viewed from the perspective of administration.

administrative authority

The agent of a Domain Management Organization (DMO) that has administrative control of the entries stored within DSAs belonging to the organization.

administrative entry

An entry located at an administrative point. Administrative entries are the only kind of entry that can have subentries as immediate subordinate entries, and are distinguished by having an administrative-role attribute that regulates their relationship with for example access control, and collective attributes.

administrative point

The root vertex of an administrative area.

alias

An alternative name for an object.

anonymous bind

A bind operation that does not use authentication (no credentials are passed in the bind).

asynchronous shadowing

A type of DirX Directory replication protocol in which a DAP or LDAP client's update operation returns immediately after the master DSA commits the operation. Asynchronous shadowing allows a high rate of update operations between supplier and consumer DSAs, even over long distances, but can lead to loss of recent update operations at the consumer DSAs if the master DSA fails. See also synchronous shadowing.

attribute

Information of a particular type to be associated with an object, and typically accessible within a directory entry. An attribute consists of an attribute type and one or more attribute values.

attribute type

The attribute component that identifies the class of information given by the attribute.

attribute syntax

The information on how an attribute's value is to be represented. The **dirxabbr** file contains the mappings between the attributes defined for the DirX Directory product and their default attribute syntaxes.

attribute value

An instance of the class of information indicated by the attribute type.

authentication

A security mechanism that verifies the identity of a user or directory service component.

autonomous administrative area

A subtree of the DIT whose entries are all administered by the same administrative authority.

auxiliary object class

An object class that describes entries or classes of entries and is not used for the structural specification of the DIT. An auxiliary object class is typically associated with objects of a variety of classes.

B

bind

The operation that initiates an association between a DUA and a DSA, or between two DSAs. A bind optionally authenticates a user to the directory service. Within the association, one or more operations can be performed. An association is terminated by an unbind operation or an abort.

C

certificate

An attribute value that is used by the directory as a highly reliable means of publishing the public key of some party (e.g. a user or other object). It contains the name of the certification authority issuing the certificate, the name and public key of the party, together with expiration time and other information. The certificate is made tamper-proof by being signed using the private key of the issuing certification authority.

certification authority

A reliable authority for the publication of public keys to be used for authentication and other purposes. A certification authority guarantees that the certificate is truly

associated with the party named by the certificate, and, in particular, that the named party owns the corresponding private key.

certification path

An ordered sequence of certificates of objects in the DIT that, together with the public key of the initial object in the path, can be processed to verify the ownership of the public key of the final object in the path, based on a chain of trust. For example, if the verifying party P trusts a certification authority A, which trusts a certification authority B, which trusts a certification authority C, which issued a certificate to the party Q, this certificate may be considered reliable in authenticating information signed by Q.

chaining

A type of DSA-to-DSA communication in which a DSA forwards an operation to another DSA for execution, then returns the result to the original requester.

collective attribute

An attribute whose values can be associated with a defined set of entries. A collective attribute is accessed as if it were a normal attribute of the entries.

collective-exclusions operational attribute

An operational attribute that specifies one or more collective attributes to be excluded from an entry.

context prefix

The sequence of Relative Distinguished Names (RDNs) that lead from the root of the DIT to the starting point of a naming context. A context prefix corresponds to the distinguished name of the starting point of a naming context.

continuation reference

A data object that describes how the performance of all or part of an operations can be continued at a different DSA or DSAs. **Continuation references** can be returned embedded in partial results by a DSA that can only partially process an operation, to indicate the DSA or DSAs that can help to complete the operation. See also **referral**.

credentials

Information used to establish the identity of a user or resource. Credentials usually consist of a username and password; more reliable credentials (“strong credentials”) involve passing a certificate or a name with an associated public key.

cross-reference

A knowledge reference that contains information about a DSA that holds a naming context frequently used by the DSA that holds the cross-reference.

D

database profile

A structure that links physical "raw" devices to the DirX Directory DBAM data storage model. A profile contains configuration information and is not part of the database data.

Profiles are stored in the registry on Windows and in a file on Linux.

DBAM

See Directory Basic Access Method.

DBAM database

The DirX Directory database component. The DBAM database stores the Directory Information Base (DIB).

DIB

See Directory Information Base.

digital signature

A mechanism to ensure the integrity and authenticity of the originator of a piece of electronic information.

directory

A repository of information about objects that also provides services to its users that allow access to the stored information.

Directory Access Protocol (DAP)

The protocol that a Directory User Agent uses to communicate with the Directory System Agents that provide the directory service.

Directory Basic Access Method (DBAM)

The database kernel of DirX Directory that is tailored to the handling of directory data and directory applications environments.

Directory Information Base (DIB)

The collection of information held by the directory as a whole (typically in many DSAs).

Directory Information Model

The X.500 standards specification that describes directory service entries, their contents, and the way in which the entries are named. It also describes the schema and other aspects of the information to which the directory provides access.

Directory Information Shadowing Protocol (DISP)

The protocol that passes entry information from a shadow supplier DSA to a shadow consumer DSA.

Directory Information Tree (DIT)

The Directory Information Base viewed as hierarchical tree-structure.

Directory Management Domain (DMD)

The collection of DSAs and DUAs owned by a specific organization (see Domain Management Organization).

directory operational attribute

An operational attribute that stores directory service-specific information within an

entry, for example, access control information or the time the entry was last modified.

Directory Operational binding management Protocol (DOP)

The protocol that serves the pair-wise automatic coordination of DSAs, for example to maintain a hierarchical operational binding or to coordinate shadowing agreements.

directory schema

The set of rules and constraints governing object classes, attribute types, attribute syntaxes, and matching rules which characterize the Directory Information Base.

directory service

The service that provides access to the Directory Information Base.

Directory Service Markup Language (DSML)

A method for expressing directory information, directory queries and updates and the results of these operations as XML documents. The DSML standard is defined by the Organization for the Advancement of Structured Information Standards (OASIS).

Directory System Agent (DSA)

The component that provides the directory service. The collection of entries that comprise the Directory Information Base is distributed between the DSAs in the directory.

Directory System Protocol (DSP)

The protocol that a Directory System Agent uses to communicate with other Directory System Agents that provide the directory service.

Directory User Agent (DUA)

The component that represents users in accessing the directory; it communicates user requests to the DSAs providing the directory service and passes their responses back to the user.

dirxadm

The DirX Directory command line-driven program that system administrators can use to manage DSAs.

dirxcp

The DirX Directory command line-driven Directory User Agent (DUA) that users and system administrators can use to communicate with a DSA.

DirX Directory (DirX)

The standards-compliant, high-performance, highly available and reliable securable identity management platform with very high linear scalability for workgroup, enterprise, and e-business applications. DirX Directory implements the LDAPv3 and X.500 directory standards.

DirX Directory Manager

The DirX Directory graphical user interface (GUI) that system administrators can use to configure and manage DirX DSAs over LDAP on Windows and Linux systems.

Distinguished Name (DN)

The sequence of Relative Distinguished Names (RDNs) leading from the root of the DIT to a specific object. In DirX Directory, the string representation uses forward slashes (/) to separate the RDNs, for example /C=DE/O=PQR. See also Relative Distinguished Name.

DIT

See Directory Information Tree.

DMD

See Directory Management Domain.

DMO

See Domain Management Organization.

DNS

See Domain Name System.

Domain Management Organization (DMO)

The organization that owns and manages a collection of DSAs and DUAs.

Domain Name System (DNS)

A service that translates names into Internet Protocol (IP) addresses.

DSA

See Directory System Agent.

DSA-shared operational attribute

An operational attribute used to store information needed by DSAs to operate a distributed directory. Values of a particular DSA-shared operational attribute should be the same on each DSA. Specific knowledge information is of this kind. See also **operational attribute**.

DSA-specific entry (DSE)

An entry in the DIT viewed from the perspective of a single DSA.

DSA-specific operational attribute

An operational attribute used to store information needed by DSAs to operate a distributed directory. Values of a particular DSA-specific operational attribute are different on each DSA. Superior knowledge information is of this kind. See also operational attribute.

DOP

See Directory Operational binding management Protocol.

DSE

See DSA-specific entry.

DSP

See Directory System Protocol.

DUA

See Directory User Agent.

E

entry

A part of the Directory Information Base that contains information about an object.

F

first-level DSA

A DSA that holds a naming context immediately beneath the root of the DIT.

first-level reference

The context prefix and access point of a DSA that holds a naming context immediately beneath the root of the DIT.

floating master

A software technique for providing high availability for all directory service operations, in which all directory information on a master DSA is replicated to a specific shadow DSA that can operate as the master should the master fail or be taken out of service for maintenance. A floating master configuration permits the directory service to be "always available" and ensures that there is a master for directory update operations during maintenance periods. See also master and shadow.

H

Hierarchical Operational Binding (HOB)

A relationship between two DSAs that hold (as masters) naming contexts, one immediately subordinate to the other. The superior DSA holds a subordinate reference to the subordinate DSA. The information held by the subordinate reference is maintained within the scope of the HOB as well as policy information (e.g. access control) held by the superior DSA but relevant to the subordinate DSA.

I

incremental update

The DISP operation that provides the shadow consumer DSA with updated copies of those entries that have changed in the unit of replication since the last update (and not the entire set of entries). Incremental updates can be configured to occur immediately on a change or at a predefined time. Also called incremental refresh. Contrast with total update.

invoke

A ROSE service element that contains a user request.

K

knowledge reference

Pieces of information that one DSA has about another DSA and the directory information it holds.

L

LDAP Data Interchange Format (LDIF)

A type of tagged data file format specified in "The LDAP Data Interchange Format (LDIF) - Technical Specification". LDIF format consists of an LDIF content format and an LDIF change format. LDIF content format contains a list of directory entries and their attributes. LDIF change format contains a list of directory modifications.

LDIF agreement

A type of Shadow Operational Binding (SOB) that is analogous to a shadowing agreement. In an LDIF agreement, the DSA is the LDIF file supplier and a directory synchronization tool is the LDIF file consumer.

Lightweight Directory Access Protocol (LDAP)

A simplified version of the Directory Access Protocol (DAP) that provides X.500 access to platforms supporting TCP/IP. LDAP is the proposed industry-standard protocol for providing directory services on the Internet. It makes direct use of TCP/IP services (that is, without using an OSI upper-layer stack).

M

master

A DSA that holds the original copy of a directory entry. A DSA is master for all entries in a naming context. The term master is also used for the original copy of a directory entry.

matching rule

A directory schema element that corresponds to a predefined rule or algorithm for comparing attribute information. A matching rule allows entries to be selected by making a matching rule assertion concerning their attribute values.

matching rule assertion

A proposition, which may be true, false, or undefined, that concerns the presence in an entry of attribute values that meet the criteria defined by the matching rule and user-supplied attribute information. For example, an entry containing a surname “Kitto” may be found by specifying an entry with a surname that sounds like “cat”. The object identifier for the algorithm specifying “sounds-like” is the matching rule; “sounds like ‘cat’” is the matching rule assertion.

N

naming attribute

An attribute type used by an entry in the attribute value assertion (or assertions) that form its relative distinguished name (RDN).

naming context

A partial subtree of the DIT that is entirely self-contained within a single DSA and mastered by it (i.e. all entries in the naming context are master entries). A naming context begins at a starting point in the DIT and extends downward to leaf entries or references to subordinate naming contexts.

NAT

See Network Address Translation.

Network Address Translation

A technology used to maintain private IP addresses (in a LAN) separately from public IP addresses, for example to increase security or to share Internet connections.

non-specific hierarchical operational binding (NHOB)

A relationship between two master DSAs holding naming contexts, one of which is immediately subordinate to the other, in which the superior DSA holds a non-specific subordinate reference to the subordinate DSA.

O

object class

An identified family of objects which share certain characteristics; alternatively a special attribute of an entry whose values are object class identifiers that define or describe the object that the entry represents. For example, an organizational-person is a human being in the context of an organization, while a directory entry representing an organizational-person has an object class attribute which contains three values: top, person, organizational-person. Each such value defines mandatory or optional attributes. Every directory entry possesses an object class attribute.

object identifier (OID)

A unique sequence of integers separated by periods (**.**). Object identifiers permit the global registration of objects and are assigned to attribute types, object classes, and matching rules (schema elements), etc. Object identifiers form a tree; registration authorities own particular sequences of integers, and can register their own objects by extending their sequence.

operational attribute

An attribute that represents information used to control the operation of the directory (e.g. access control information), or used by the directory to represent some aspects of its operation (e.g. knowledge references). See directory operational attributes, DSA-shared operational attributes, DSA-specific operational attributes for the specific kinds of operational attributes.

P

policy

An expression by an administrative authority of general goals and acceptable procedures.

policy attribute

A generic term for an operational attribute that expresses policy (for example, an attribute that defines the type of access control which is to apply in an area of the DIT).

policy object

An entity with which a policy is associated (for example, an entry to which an access control policy can be directed).

private key

The key of a key pair for public key cryptography known only by the owning user. The other part of a key pair is the public key. The private key is often called the secret key.

public key

The publicly-known key of a key pair. Contrast with private key.

R

referral

A method of DSA communication in which a DSA that cannot completely perform an operation returns a continuation reference which specifies how far it has been able to proceed with the operation, together with the name and communications address of one or more DSAs that may be able to complete the operation. See also continuation reference.

Relative Distinguished Name (RDN)

The portion of a distinguished name that uniquely names an entry relative to its immediately superior entry. Each RDN consists of one or more attribute value assertions which specify an attribute type and an attribute value for the entry. The RDN is selected to achieve uniqueness. In DirX Directory, the string representation for attribute value assertions is of the form *type*=*value* and a comma (,) is used to separate attribute value assertions, for example CN=Lynch,O=SNI.

root

The topmost node of the Directory Information Tree. The root of the DIT has no name (its DN is an empty sequence); it also has no corresponding entry, since each entry in the DIT must belong to some owning organization, and no such organization can own the root. In DirX Directory the string representation for the root is the slash character (**/**).

root context

The collection of context prefixes and access points of DSAs that hold naming contexts immediately beneath the root of the DIT; in other words: the complete collection of first level references.

root DSE

A DSE that contains DSA-specific attributes that relate to the DSA as a whole, for example, the my-access-point attribute, which holds the name and address of the DSA itself.

S

schema publication

The provision made within the directory standards whereby operational attributes are defined to describe the schema (attributes, object classes, etc.) applicable to the DIT. The schema publication attributes are held in the schema subentry; reading these attributes informs a DSA, DUA, LDAP client and server about the schema.

Secure Socket Layer/Transport Layer Security (SSL/TLS)

A protocol for enabling secure communications between clients and servers communicating over TCP/IP and other network protocols. The DirX Directory LDAP server supports SSL/TLS to ensure secure LDAP communication.

shadow

A copy of one or more directory entries, created by the use of the Directory Information Shadowing Protocol (DISP). Contrast with master.

shadow consumer

A DSA that receives shadowed information by means of the Directory Information Shadowing Protocol (DISP). Contrast with shadow supplier.

shadowing

The process of maintaining a copy of a set of DIT entries by means of the Directory Information Shadowing Protocol (DISP).

shadowing agreement

An agreement made between administrators of the two DSAs in a shadow operational binding that specifies the information to be shadowed, when it is to be shadowed, and the roles (supplier or consumer) to be played by each DSA. A shadowing agreement is represented by information within each DSA, and is maintained by a shadow operational binding.

shadow operational binding (SOB)

A relationship between two DSAs in which one DSA acts as a supplier of replicated information and the other DSA acts as the consumer of the replicated information. A shadow operational binding maintains the shadowing agreement between the two DSAs.

shadow supplier

A DSA that supplies shadowed information by means of the Directory Information Shadowing Protocol (DISP). Contrast with shadow consumer.

shadow-supplier reference

A knowledge reference held by a shadow consumer DSA that contains information about the shadow supplier DSA.

Simple Network Management Protocol

A set of standards that defines the communication between agents and their management stations to monitor and control devices in an IP network.

simple protected authentication

An authentication method for binding to a DSA in which a name and a protected (one-way-encrypted) password must be supplied when invoking the bind operation.

SNMP

See Simple Network Management Protocol.

SNMPv2-trap

An unsolicited, unconfirmed message from an agent in a network to a manager notifying a specific event.

strong authentication

An authentication method for binding to a DSA that uses digital signatures employing public key cryptography.

subentry

A special entry used to hold policy information. There are several kinds of subentries, for example: access control subentries and collective attribute subentries; a subentry can be of more than one kind. Subentries can only be subordinate to administrative entries.

subordinate DSA

In a hierarchical operational binding the DSA that holds the subordinate naming. Contrast with superior DSA.

subordinate reference

A knowledge reference that contains information about a DSA that holds a subordinate naming context. Subordinate references represent the transition from one DSA's naming context to another. Contrast with superior reference.

subtree

A collection of entries that represent a subset of the DIT.

superior DSA

In a hierarchical operational binding the DSA that holds the superior naming context. Contrast with subordinate DSA.

superior reference

A knowledge reference that contains information about a DSA that holds a superior naming context. A superior reference is used by a DSA as a reference for operations for which no more specific DSA can be identified. First-level DSAs must hold the root context, and so have no need of a superior reference. A superior reference normally

specifies a DSA that has at least one entry closer to the root than any entry held in the present DSA. Contrast with subordinate reference.

synchronous shadowing

A type of DirX Directory replication protocol in which a DAP or LDAP client's update operation does not return until the master DSA and all synchronous consumer DSAs have committed the operation. Synchronous shadowing provides for high data integrity between master and shadow DSAs even in the event of a master failure because acknowledged update operations to the DAP/LDAP client are safely stored at the synchronous consumer DSAs. See also asynchronous shadowing.

system schema

The set of rules that regulate the use of operational attributes, administrative entries and subentries.

T

top

The special object class of which every other class is a subclass.

total update

The DISP operation that replicates all the entries in the unit of replication on the shadow consumer DSA. Also called a total refresh. Contrast with incremental update.

U

unit of replication

The specification of the information to be shadowed or written to an LDIF file, including policy information in administrative entries and subentries, the replicated area containing the entries to be shadowed, and (optionally) subordinate knowledge information.

user attribute

An attribute that represents user information.

X

X.500 Directory Standards

A set of standards that describe how a global directory service can be built.

DirX Product Suite

The DirX product suite provides the basis for fully integrated identity and access management; it includes the following products, which can be ordered separately.

DirX Identity

DirX Identity provides a comprehensive, process-driven, customizable, cloud-enabled, scalable, and highly available identity management solution for businesses and organizations. It provides overarching, risk-based identity and access governance functionality seamlessly integrated with automated provisioning. Functionality includes lifecycle management for users and roles, cross-platform and rule-based real-time provisioning, web-based self-service functions for users, delegated administration, request workflows, access certification, password management, metadirectory as well as auditing and reporting functionality.

DirX Directory

DirX Directory provides a standards-compliant, high-performance, highly available, highly reliable, highly scalable, and secure LDAP and X.500 Directory Server and LDAP Proxy with very high linear scalability. DirX Directory can serve as an identity store for employees, customers, partners, subscribers, and other IoT entities. It can also serve as a provisioning, access management and metadirectory repository, to provide a single point of access to the information within disparate and heterogeneous directories available in an enterprise network or cloud environment for user management and provisioning.

DirX Access

DirX Access is a comprehensive, cloud-ready, scalable, and highly available access management solution providing policy- and risk-based authentication, authorization based on XACML and federation for Web applications and services. DirX Access delivers single sign-on, versatile authentication including FIDO, identity federation based on SAML, OAuth and OpenID Connect, just-in-time provisioning, entitlement management and policy enforcement for applications and services in the cloud or on-premises.

DirX Audit

DirX Audit provides auditors, security compliance officers and audit administrators with analytical insight and transparency for identity and access. Based on historical identity data and recorded events from the identity and access management processes, DirX Audit allows answering the “what, when, where, who and why” questions of user access and entitlements. DirX Audit features historical views and reports on identity data, a graphical dashboard with drill-down into individual events, an analysis view for filtering, evaluating, correlating, and reviewing of identity-related events and job management for report generation.

For more information: support.dirx.solutions/about

EVIDEN

Legal remarks

On the account of certain regional limitations of sales rights and service availability, we cannot guarantee that all products included in this document are available through the Eviden sales organization worldwide. Availability and packaging may vary by country and is subject to change without prior notice. Some/All of the features and products described herein may not be available locally. The information in this document contains general technical descriptions of specifications and options as well as standard and optional features which do not always have to be present in individual cases. Eviden reserves the right to modify the design, packaging, specifications and options described herein without prior notice. Please contact your local Eviden sales representative for the most current information. Note: Any technical data contained in this document may vary within defined tolerances. Original images always lose a certain amount of detail when reproduced.