OpenLDAP Benchmark v1.2, December 2007

Version: 1.2

Status: draft

Table of Contents

Agreed Terms of Use

Background.

CIS provides benchmarks, scoring tools, software, data, information, suggestions, ideas, and other services and materials from the CIS website or elsewhere ("Products") as a public service to Internet users worldwide. Recommendations contained in the Products ("Recommendations") result from a consensus-building process that involves many security experts and are generally generic in nature. The Recommendations are intended to provide helpful information to organizations attempting to evaluate or improve the security of their networks, systems and devices. Proper use of the Recommendations requires careful analysis and adaptation to specific user requirements. The Recommendations are not in any way intended to be a "quick fix" for anyone's information security needs.

No representations, warranties and covenants.

CIS makes no representations, warranties or covenants whatsoever as to (i) the positive or negative effect of the Products or the Recommendations on the operation or the security of any particular network, computer system, network device, software, hardware, or any component of any of the foregoing or (ii) the accuracy, reliability, timeliness or completeness of any Product or Recommendation. CIS is providing the Products and the Recommendations "as is" and "as available" without representations, warranties or covenants of any kind.

User agreements.

By using the Products and/or the Recommendations, I and/or my organization ("we") agree and acknowledge that:

- No network, system, device, hardware, software or component can be made fully secure;
- We are using the Products and the Recommendations solely at our own risk;
- We are not compensating CIS to assume any liabilities associated with our use of the Products or the Recommendations, even risks that result from CIS's negligence or failure to perform;
- We have the sole responsibility to evaluate the risks and benefits of the Products and Recommendations to us and to adapt the Products and the Recommendations to our particular circumstances and requirements;
- Neither CIS, nor any CIS Party (defined below) has any responsibility to make any corrections, updates, upgrades or bug fixes or to notify us if it chooses at it sole option to do so; and
- Neither CIS nor any CIS Party has or will have any liability to us whatsoever (whether based in contract, tort, strict liability or otherwise) for any direct, indirect, incidental, consequential, or special damages (including without limitation loss of profits, loss of sales, loss of or damage to reputation, loss of customers, loss of software, data, information or emails, loss of privacy, loss of use of any computer or other equipment, business interruption, wasted management or other staff resources or claims of any kind against us from third parties) arising out of or in any way connected with our use of or our inability to use any of the Products or Recommendations (even if CIS has been advised of the possibility of such damages), including without limitation any liability associated with infringement of intellectual property, defects, bugs, errors, omissions, viruses, worms, backdoors, Trojan horses or other harmful items.

Grant of limited rights.

CIS hereby grants each user the following rights, but only so long as the user complies with all of the terms of these Agreed Terms of Use:

- Except to the extent that we may have received additional authorization pursuant to a written agreement with CIS, each user may download, install and use each of the Products on a single computer;
- Each user may print one or more copies of any Product or any component of a Product that is in a .txt, .pdf, .doc, .mcw, or .rtf format, provided that all such copies are printed in full and are kept intact, including without limitation the text of this Agreed Terms of Use in its entirety.

Retention of intellectual property rights; limitations on distribution.

The Products are protected by copyright and other intellectual property laws and by international treaties. We acknowledge and agree that we are not acquiring title to any intellectual property rights in the Products and that full title and all ownership rights to the Products will remain the exclusive property of CIS or CIS Parties. CIS reserves all rights not expressly granted to users in the preceding section entitled "Grant of limited rights." Subject to the paragraph entitled "Special Rules" (which includes a waiver, granted to some classes of CIS Members, of certain limitations in this paragraph), and except as we may have otherwise agreed in a written agreement with CIS, we agree that we will not (i) decompile, disassemble, reverse engineer, or otherwise attempt to derive the source code for any software Product that is not already in the form of source code; (ii) distribute, redistribute, encumber, sell, rent, lease, lend, sublicense, or otherwise transfer or exploit rights to any Product or any component of a Product; (iii) post any Product or any component of a Product on any website, bulletin board, ftp server, newsgroup, or other similar mechanism or device, without regard to whether such mechanism or device is internal or external, (iv) remove or alter trademark, logo, copyright or other proprietary notices, legends, symbols or labels in any Product or any component of a Product; (v) remove these Agreed Terms of Use from, or alter these Agreed Terms of Use as they appear in, any Product or any component of a Product; (vi) use any Product or any component of a Product with any derivative works based directly on a Product or any component of a Product; (vii) use any Product or any component of a Product with other products or applications that are directly and specifically dependent on such Product or any component for any part of their functionality, or (viii) represent or claim a particular level of compliance with a CIS Benchmark, scoring tool or other Product. We will not facilitate or otherwise aid other individuals or entities in any of the activities listed in this paragraph.

We hereby agree to indemnify, defend and hold CIS and all of its officers, directors, members, contributors, employees, authors, developers, agents, affiliates, licensors, information and service providers, software suppliers, hardware suppliers, and all other persons who aided CIS in the creation, development or maintenance of the Products or Recommendations ("CIS Parties") harmless from and against any and all liability, losses, costs and expenses (including attorneys' fees and court costs) incurred by CIS or any CIS Party in connection with any claim arising out of any violation by us of the preceding paragraph, including without limitation CIS's right, at our expense, to assume the exclusive defense and control of any matter subject to this indemnification, and in such case, we agree to cooperate with CIS in its defense of such claim. We further agree that all CIS Parties are third-party beneficiaries of our undertakings in these Agreed Terms of Use.

Special rules.

The distribution of the NSA Security Recommendations is subject to the terms of the NSA Legal Notice and the terms contained in the NSA Security Recommendations themselves (http://nsa2.www.conxion.com/cisco/notice.htm).

CIS has created and will from time to time create special rules for its members and for other persons and organizations with which CIS has a written contractual relationship. Those special rules will override and supersede these Agreed Terms of Use with respect to the users who are covered by the special rules. CIS hereby grants each CIS Security Consulting or Software Vendor Member and each CIS Organizational User Member, but only so long as such Member remains in good standing with CIS and complies with all of the terms of these Agreed Terms of Use, the right to distribute the Products and Recommendations within such Member's own organization, whether by manual or electronic means. Each such Member acknowledges and agrees that the foregoing grant is subject to the terms of such Member's membership arrangement with CIS and may, therefore, be modified or terminated by CIS at any time.

Choice of law; jurisdiction; venue.

We acknowledge and agree that these Agreed Terms of Use will be governed by and construed in accordance with the laws of the State of Maryland, that any action at law or in equity arising out of or relating to these Agreed Terms of Use shall be filed only in the courts located in the State of Maryland, that we hereby consent and submit to the personal jurisdiction of such courts for the purposes of litigating any such action. If any of these Agreed Terms of Use shall be determined to be unlawful, void, or for any reason unenforceable, then such terms shall be deemed severable and shall not affect the validity and enforceability of any remaining provisions. We acknowledge and agree that we have read these Agreed Terms of Use in their entirety, understand them and agree to be bound by them in all respects.

Introduction

LDAP stands for Lightweight Directory Access Protocol, which is based on X.500 directory services. The protocol is defined in the RFC 4510 series.

LDAP servers are very popular, including commercial servers such as Microsoft Active Directory, IBM Tivoli Directory Server, Novell eDirectory, and Sun Java System Directory Server. OpenLDAP is the most popular of the open source LDAP servers.

LDAP servers are just one part of a network infrastructure, and their security depends in part on the security of the rest of the network. This benchmark will focus primarily on the secure configuration of slapd, the OpenLDAP server.

Applicability

The benchmark was developed and tested using OpenLDAP versions 2.3.39 and 2.4.6, but most of the content will apply to other versions of OpenLDAP. Many of the recommendations and principles will also apply to other LDAP servers.

Additional Resources

OpenLDAP Web Site - http://www.openldap.org/

OpenLDAP Security Considerations - http://www.openldap.org/doc/admin23/security.html

OWASP LDAP Injections - http://www.owasp.org/index.php/LDAP_injection

RFC 4511 LDAP: The Protocol - http://www.ietf.org/rfc/rfc4511.txt

RFC 4510 LDAP Technical Specification Road Map - http://www.ietf.org/rfc/rfc4510.txt

1 Operating System Level Configuration

1.1 OS Hardening

Security of the LDAP service depends on having a secured operating system as a foundation.
To achieve compliance:

Harden the operating system using the appropriate CIS benchmark from http://www.cisecurity.org/

1.2 Non-privileged LDAP Account and Group

The-uand-goptions to slapd are typically set by the startup script or a file read by the script. This user account and group must not be used for any purpose other than running slapd, the LDAP server.
To achieve compliance:

Configure slapd to run as a dedicated, non-privileged user and group by starting slapd with the -u and -g options, such as "-u ldap -g ldap".

1.3 slapd.conf File Ownership and Permissions

Ownership by other users would allow the file permissions to be changed. Non-LDAP users and/or groups do not need access to the slapd configuration.

Note: slapd.conf is being phased out in favor of the slapd.d directory hierarchy .

To achieve compliance:

The slapd.conf configuration file must be owned by root, read-only for the ldap group, and have no access for other. That is:

```
-rw-r----- 1 root ldap slapd.conf
```

1.4 slapd.d Configuration Ownership and Permissions

slapd's configuration can be changed on the fly using standard LDAP search, modify, and delete operations. Configuration changes can usually take effect without a restart. In order to make these configuration changes persist across restarts, the slapd.d directory hierarchy (on disk) must be used to store slapd's configuration instead of the slapd.conf configuration file.

When a configuration change is made using LDAP operations, configuration files in this directory hierarchy are modified by slapd to commit changes to permanent storage. Without write access to the slapd.d hierarchy, online configuration changes are lost when slapd is restarted.

More information on slapd online configuration can be found in the OpenLDAP Software Administrator's Guide .

To achieve compliance:

The slapd.d directory hierarchy must be owned by the ldap user.

The ldap user must have read/write access to the slapd.d directory hierarchy. All other users and groups should have no access. For example:

```
drwx------ 1 ldap root slapd.d
drwx------ 1 ldap root cn=config
-rw------- 1 ldap root cn=config.ldif
```

1.5 Client conf File Ownership

Ensuring the system client ldap.conf file is owned by root helps prevent unauthorized changes of the file permissions.
To achieve compliance:

The LDAP client file, ldap.conf, must be owned by root.

1.6 Client conf File Permissions

Prevent unauthorized changes to the files.
To achieve compliance:

The LDAP client file and directory, ldap.conf, must not be writable except by root. For example:

```
drwxr-xr-x 4 root root /etc/ldap/
-rw-r--r-- 1 root root ldap.conf
```

1.7 LDAP Schema Files

The LDAP schema files define the types and semantics of the stored data. It is important to prevent write access to the schemas except by the root user.
To achieve compliance:

The LDAP schema files must not be writable except by root. For example:

```
drwxr-xr-x 3 root root /etc/ldap/schema/
-rw-r--r-- 1 root root corba.schema
-rw-r--r-- 1 root root core.ldif
-rw-r--r-- 1 root root core.schema
```

1.8 Permissions for rootdn Password

Protect the powerful root Distinguished Name password from improper disclosure.
To achieve compliance:

The database or configuration file that contains the root Distinguished Name password should be owned by root, must read-only for the ldap group, and not readable by other users. That is:

```
root:ldap 0640
```

1.9 Protect the LDAP Database

Inappropriate direct access to the on-disk database makes it easy to copy and crack passwords, or to directly update the database by modifying the on-disk files. Keep in mind that slapd may rely on other services, such as saslauthd (for SASL authentication) or external SQL or LDAP servers (if the back-sql or back-ldap backends are used), which must also be secured appropriately.
To achieve compliance:

The directory and files containing the on-disk database should be owned by the ldap user and the ldap group, with no access for any other user or group. For example:

```
drwx------ 2 ldap ldap 4096 /var/lib/ldap/
-rw------- 1 ldap ldap 24576 __db.001
```

1.10 Protect LDAP Export/Import Files

Rather than attack the LDAP database directly, it's often easier to obtain the information through import/export files (typically stored in LDIF format) or backup copies of the OpenLDAP on-disk database. Another attack vector could be access to the Berkeley DB log files generated as part of normal slapd operation when configured with the "bdb" or "hdb" backends. Look for such files on the local system and review any import, export, or backup processes that are being used.
To achieve compliance:

Import, export, or backup files containing any or all of the LDAP database must be removed when no longer needed as defined by site policy, and protected by enforcing minimal read access, such as root ownership with permissions 0600. Encrypting these files may also be worthwhile, especially if the LDAP directory contains highly sensitive information.

Additionally, OpenLDAP 2.4.8 and up will offer optional Berkeley DB encryption, which automatically encrypts slapd's database files for the "bdb" and "hdb" backends.

Berkeley DB log files for the on-disk database should be removed as defined by site policy. The OpenLDAP Software Administrator's Guide has more information on Berkeley DB log rotation .

1.11 Dedicated System

In order to reduce the risk of potential vulnerabilities in other services jeopardizing the LDAP server, consider installing OpenLDAP on a dedicated system, or a server with minimal services.
To achieve compliance:

The server should be dedicated to running OpenLDAP. If possible, only administrative services like SSH and other closely related authentication services, such as RADIUS, should be running on the same system.

1.12 Restricted File System Access

If the slapd service is exploited remotely, file system access restrictions will prevent the exploit from getting access to system files and executables.
To achieve compliance:

Consider running slapd in a chroot environment, or using SELinux or Solaris RBAC, to restrict access to system files for the user slapd runs as.

1.13 Restricted Network Access

If only local access is required, consider listening on only the loopback interface or using Unix domain sockets. If network access is required, consider restricting access based on source IP address as an added layer of security. Remember that IP addresses can be spoofed, so restricting connections based on source IP address is no guarantee.
To achieve compliance:

Restrict network access using host-based IP filtering to only the networks or systems requiring access. Depending on the platform, this may be accomplished with tcpwrappers, iptables, or ipf.

2 slapd Configuration File - slapd.conf

2.1 LDAPv2 Bind

Note the bind_v2 option is not enabled by default. It allows the old version 2 bind, and does not enable support for the entire LDAPv2 protocol.
To achieve compliance:

Do not allow the older LDAPv2 bind request unless compatibility with old LDAP clients is absolutely necessary. Do not specify "allow bind_v2" in slapd's configuration.

2.2 rootdn Password Storage

Since the root Distinguished Name password provides unrestricted access to the LDAP data, it needs to be carefully protected with a salted hash value. The slappasswd(8) command may be used to generate the hash. Administrators will need to update (and may need to share) the contents of the slapd.conf file; placing the rootdn password in a separate file helps protect it from accidental disclosure.

The rootpw directive may be omitted, which prevents all access by the root DN (the rootdn directive should still be present since other features, such as syncrepl, require its presence). Instead, set up explicit ACLs for individual administrative accounts, granting each account the minimal access required.

To achieve compliance:

The rootpw directive, if present, must not be stored directly in the slapd.conf file. It may be included from a separate, protected configuration file using a secure hash (such as SSHA, the OpenLDAP default), or stored in an alternative database or service (such as SASL).

2.3 rootdn Password Policy

The LDAP rootdn password is easily overlooked with regard to password policies for administrative access.
To achieve compliance:

The password used for the rootdn must comply with site policies for administrative passwords, especially with regard to password complexity and periodic changes.

Note that the password policy overlay (ppolicy) , does not apply any of its restrictions to the root DN, as they could unintentionally lock the root DN out.

2.4 Administrative Access

Individual accounts are necessary for accountability and reduce the risk of password disclosure.
To achieve compliance:

Administrators and operators with privileged access should use individual accounts for day-to-day tasks rather than the rootdn or a shared account.

2.5 Application Access

Individual accounts are also necessary for applications to ensure minimal access, and to help track possible compromise or abuse.
To achieve compliance:

When application accounts are required, the accounts should provide minimal required access and not be shared with other applications or users.

2.6 Log Configuration

To configure detailed logging, add the values for the desired subsystems and specify the resulting sum with the LogLevel directive. Log entries are syslogged.
1: Trace function calls
2: Debug packet handling
4: Heavy Trace debug
8: Connection Management
16: Print Packets sent and received
32: Search Filter processing
64: Configuration File processing
128: Access Control List processing
256: Statistics for connections, operations and results
512: Statistics for entries sent
1024: Print communication with shell backends
2048: Entry parsing

The accesslog and/or auditlog overlays, available in OpenLDAP 2.3.4 and later, are better alternatives to increasing slapd's LogLevel.

To achieve compliance:

From a security standpoint, the most useful subsystems are:

256: Statistics for connections, operations and results

Logs connection sources, authentication attempts, and search information.

Sample output:

```
conn=0 fd=13 ACCEPT from IP=127.0.0.1:43534 (IP=0.0.0.0:389)
conn=0 op=0 BIND dn="cn=admin,dc=example,dc=com" method=128
conn=0 op=0 BIND dn="cn=admin,dc=example,dc=com" mech=SIMPLE ssf=0
conn=0 op=0 RESULT tag=97 err=0 text=
conn=0   op=1   SRCH   base="dc=example,dc=com"   scope=2   deref=0
filter="(objectClass=*)"
conn=0 op=1 SEARCH RESULT tag=101 err=0 nentries=3 text=
conn=0 op=2 UNBIND
conn=0 fd=13 closed
```

512: Statistics for entries sent

Logs the DN of LDAP entries sent to clients.

Sample output:

```
conn=0 op=1 ENTRY dn="dc=example,dc=com"
conn=0 op=1 ENTRY dn="uid=example,dc=example,dc=com"
```

Verbose debugging logs may adversely affect slapd performance.

2.7 SSL/TLS Configuration

In an OpenLDAP context, "SSL" refers to LDAPS on port 636, in which the entire LDAP protocol is encapsulated in an encrypted connection from the very moment it is established, even before the LDAP protocol itself comes into play. In this regard, LDAPS is similar to HTTPS, which also establishes an encrypted connection immediately, before the HTTP protocol is involved.

In contrast, "TLS" refers to the Start TLS extension, which allows a client to open a non-encrypted LDAP connection on port 389 and then issue an LDAP protocol request to enable TLS. Once TLS is successfully enabled, all further data sent over the connection is encrypted. The client holds the responsibility for issuing this request (and thus enabling encryption).

In both cases, slapd can be configured to require encryption of a certain strength before processing operations, such as bind or search, that transfer sensitive data.

To achieve compliance:

OpenLDAP must be configured to support LDAPS or Start TLS, in order to protect authentication credentials and other information sent over LDAP connections.

2.8 slapd SSL/TLS Certificate

Clients connecting to slapd perform a validation procedure on the certificate the server presents. Clients may optionally supply a client certificate when negotiating an encrypted LDAP connection, and slapd may require that a valid client certificate be presented before allowing an encrypted connection to be established. Client certificates must pass the same validation procedures as server certificates.

For a certificate to be valid, it must not be expired (i.e., the current date must be between the certificate's issue and expiration dates), and the certificate must be signed by a trusted Certificate Authority (CA). A site-local CA may be used if LDAP clients and servers are configured to trust it.

Additionally, to prevent man in the middle attacks , SSL/TLS certificates include the fully qualified domain name (FQDN) of the server in the certificate's common name (CN) or subjectAltName fields. Typically, when a client connects to slapd, it will look for the hostname it used to connect to the server in the server certificate's CN and subjectAltName fields. If the hostname is not found in at least one of these fields, the client will refuse to continue.

To achieve compliance:

All SSL/TLS certificates must be non-expired and signed by trusted Certificate Authority (CA). A certificate's common name (CN) field must match the host's fully qualified domain name (FQDN).

To require that clients present valid client certificates, slapd may be configured with the TLSVerifyClient directive:

```
TLSVerifyClient demand
```

2.9 slapd SSL/TLS Certificate File Permissions

The slapd private key file must be protected, since an attacker can use it to decrypt intercepted SSL/TLS traffic. OpenLDAP Software as recent as version 2.4 does not support encryption of the key itself, so the key must be decrypted and the file protected carefully.

To achieve compliance:

The certificate authority certificates, slapd certificate, and slapd private key file are configured in slapd.conf as shown below, and must be owned by root and writable only by root. The slapd private key file should be readable only by the ldap user, while the CA certificates and slapd certificate are public information and may be readable by anyone.

```
TLSCACertificateFile /etc/ldap/cacert.crt
TLSCertificateFile /etc/ldap/ldap.example.com.crt
TLSCertificateKeyFile /etc/ldap/ldap.example.com.key
```

```
root:ldap 644 cacert.crt
root:ldap 644 ldap.example.net.crt
root:ldap 640 ldap.example.net.key
```

2.10 Strong SSL/TLS Ciphers

The TLSCipherSuite directive controls the parameters (key exchange methods, encryption algorithms, key lengths, and hash functions) slapd will allow on SSL/TLS connections. The recommended value is specific to the SSL toolkit that slapd is linked with.

For OpenSSL, the HIGH and MEDIUM cipher suites include those encryption algorithms that use 128 bit keys or longer. Specifying !ADH prevents the weak Anonymous Diffie-Hellman key exchange from being used, and -SSLv2 removes the weaker ciphers used with the SSLv2 protocol. It is recommended that you use this openssl command to verify the list of ciphers defined by any cipher string used, as different compilations or revisions of OpenSSL may differ:

```
openssl ciphers -v 'cipher-string'
```

For GNUTLS, it is recommended to disable the following suites: TLS_RSA_NULL_MD5, since it uses the NULL encryption algorithm which does not encrypt data, and any suites using the ARCFOUR encryption algorithm, as they use the less secure RC4 encryption algorithm. Unfortunately, it is not possible to explicitly exclude individual cipher suites when using GNUTLS; instead, a list of cipher suites must be configured while omitting the suites listed here. A complete list of the cipher suites supported by GNUTLS can be found in the GNUTLS manual .

To achieve compliance:

slapd must be configured to not allow weak ciphers.

For OpenSSL, the following configuration allows only 128 bit or stronger ciphers and is expected to be compatible with most LDAP clients.

```
TLSCipherSuite HIGH:MEDIUM:!ADH:-SSLv2
```

For GNUTLS, choose a list of cipher suites while being careful to omit the TLS_RSA_NULL_MD5 suite and any suites using ARCFOUR/RC4.

2.11 Security Strength Factors

Allowing bind operations with password authentication without requiring SSL/TLS encryption would open clients up to man-in-the-middle attacks as well as allow for eavesdropping of passwords and other information.

To achieve compliance:

The Security Strength Factors (SSF) indicates the relative strength of protection required for different types of access, such as bind, read-only, or update access. The SSF flag indicates the overall strength required, while simple_bind is the strength factor required for a simple bind operation. Refer to the slapd.conf(8) man page for details. The number indicates the strength of the encryption in terms of the approximate number of effective cipher bits. For example, 112 allows usage of triple DES or stronger, while 128 requires stronger 128-bit ciphers such as Blowfish. Minimal recommended setting is 112 for all operations.

```
security ssf=112 update_ssf=112 simple_bind=112
```

2.12 Disallow Anonymous Access

Although bind_anon is disabled by default in current OpenLDAP implementations, is recommended that the directive be included for additional assurance.

To achieve compliance:

Use the following directive to disallow anonymous binds to the LDAP server, which would provide access with no user name or password.

```
disallow bind_anon
```

2.13 Disallow Unauthenticated Access

Allow access only for those authenticated with a valid DN and password.

To achieve compliance:

Use the "require authc" directive to disallow unauthenticated access, in which a remote user has provided a valid DN, but no password.

```
require authc
```

2.14 Password Access Controls

Read access to LDAP password attributes (such as userPassword) should not be granted except in special circumstances, which should require additional mitigating controls. For example, a RADIUS server using LDAP accounts for authentication may need to read passwords to support CHAP authentication. In addition to requiring SSL/TLS, additional controls such as client certificates and IP address filtering can further reduce the risks of password disclosure in situations like this.

To achieve compliance:

Access to password attributes must be limited to authentication for anonymous, write access for self and administrators authorized to reset user passwords. For example:

```
access to attrs=userPassword
by self write
by dn="cn=joeadmin,dc=example,dc=com" write
by * auth
```

2.15 Password Hash

Usage of a secure hash algorithm such as SSHA (which uses a random salt) is best practice. A random salt makes precomputed dictionary attacks more difficult, but if the hash is disclosed, password guessing and brute force attacks can still be used to crack the password. Therefore, protecting the hash from disclosure and using strong passwords are still required.

To achieve compliance:

The password-hash directive defines the default hash algorithm used to store passwords. The default value is {SSHA}, which is the SHA-1 algorithm with a random salt. For example:

```
password-hash {SSHA}
```

2.16 Search Size Limits

Reducing the search size limit may be useful to increase the work required in attacks like LDAP injection where the intent is to match as many entries as possible. Care should be taken to not prevent legitimate queries by setting the size limit too low.

To achieve compliance:

The sizelimit directive specifies the maximum number of entries to return in response to search requests, except for those performed by the root DN.

Different limits can be applied to anonymous vs. authenticated users. To limit search results to 5 entries for anonymous connections and 100 entries for authenticated connections:

```
limits anonymous size=5
limits users size=100
```

2.17 Idle Timeout

Reduces the resources consumed due to idle sessions.
To achieve compliance:

Time connections out after 120 seconds of inactivity.

```
idletimeout 120
```

2.18 Search Time Limits

When properly configured with indexes corresponding to expected searches, slapd will process most requests in less than one second. Placing a time limit on searches may be useful to increase the work required in attacks like LDAP injection, as well as to limit the impact of injecting intentionally expensive queries. However, care must be taken to not prevent legitimate usage that would exceed the limit.
To achieve compliance:

If possible, reduce to 15 seconds the maximum time slapd will spend processing a search request.

```
timelimit 15
```

Different limits can be applied to anonymous vs. authenticated users. To limit search time to 15 seconds for anonymous connections and 60 seconds for authenticated connections:

```
limits anonymous time=15
limits users time=60
```

2.19 Redundant LDAP Servers

Consider deploying redundant replicated LDAP servers if LDAP services are a necessary part of your organization's infrastructure.
To achieve compliance:

Additional LDAP servers can be deployed, using syncrepl to automatically replicate changes from master server(s). All LDAP servers should be configured to the same security standards.

slurpd-based replication is deprecated and no longer supported in OpenLDAP 2.4 and later. The LDAP Sync Replication Engine (syncrepl) is more reliable, flexible, and featureful. syncrepl also allows the initial LDAP database load for a new slave to be pulled/pushed while the master remains fully online and able to perform read and write operations.

More information about replication can be found in the OpenLDAP Software Administrator's Guide.

Master slapd sample configuration:

```
database bdb
index objectclass,entryCSN,entryUUID eq
```

```
overlay syncprov
syncprov-checkpoint 100 10
syncprov-sessionlog 100
limits dn.exact=cn=syncuser,dc=example,dc=com size=unlimited
limits dn.exact=cn=syncuser,dc=example,dc=com time=unlimited
```

Note that this configuration overrides any size or time limits that may apply to the replication user. Since LDAP slaves use persistent searches against the master LDAP server to find updated entries to replicate, any search size or time limits would adversely affect replication.

Slave slapd sample configuration:

```
syncrepl rid=123
provider=ldap://master.example.com
bindmethod=simple
binddn="cn=syncuser,dc=example,dc=com"
credentials=secret
type=refreshAndPersist
searchbase="dc=example,dc=com"
```

Note that the replication user (in this case, cn=syncuser,dc=example,dc=com with password "secret") must be created in the LDAP directory before configuring replication, and ACLs must be set up to grant it read access to the portions of the LDAP tree that are being replicated.

3 Useful slapd Overlays

3.1 Password Policy (ppolicy)

The password policy (ppolicy) overlay enforces configurable controls on passwords, such as maximum age, minimum length, or maximum number of authentication failures before lockout.
To achieve compliance:

Add the ppolicy overlay to each "database" section in slapd.conf to enforce password controls according to site policy.

3.2 Access Logging (accesslog)

The access logging (accesslog) overlay allows slapd to write log entries to a dedicated LDAP database when certain LDAP operations are performed. These LDAP entries can be searched using normal LDAP operations. Old log entries can be purged automatically to prevent the database from growing indefinitely.
To achieve compliance:

The following example logs all LDAP read and write operations into a Berkeley DB database under "cn=log". A complete guide to configuring this overlay is beyond the scope of this benchmark; more information is available in the slapo-accesslog man page .

```
database bdb
# [other database-specific settings]
overlay accesslog
logdb cn=log
logops writes reads
database bdb
suffix cn=log
```

3.3 Audit Logging (auditlog)

The Audit Logging (auditlog) overlay writes a log file of all changes made to a given slapd database. Log entries are standard LDIF, with comments indicating the time of the change and the identity of the user making it.

The accesslog overlay is generally preferred, since it can also log read operations, makes its output available via LDAP for easy access by remote clients, has configurable criteria for selecting which events to log, and has support for automatically purging log entries older than a certain age.

To achieve compliance:

Configure the auditlog overlay for each "database" section in slapd.conf:

```
database bdb
# [other database-specific settings]
```

```
overlay auditlog
auditlog /var/log/slapd-audit
```

Revision History

Original Version, 1.0 -- Jan-Apr 2007 -- Editor Ralph Durkee

Consensus Updates, Version 1.1 -- Aug 2007 -- Editor Ralph Durkee

Updates, Version 1.2 -- Dec 2007 -- John Morrissey

References