アクセス制御

ACL

openLDAPの場合、アクセス制御は/etc/openldap/slapd.confに記述する。アクセス制御は、アクセス・コントロール・リスト(ACL)という形式で静的に行う。なお、他のLDAPサーバなどではアクセス制御自体もLDAPディレクトリのエントリとして扱い、動的に制御できるものもある。

ACLは上から順に評価され、デフォルトでは最初にマッチした時点のアクセス権が適用されて評価が終了する。以下、ACLの一部を抜粋し、各項目ごとに内容を記述する。ACL全体についてはslapd.confを参照。

```
#
# cn=admin,ou=groupsに関するアクセス権
# Administratorおよびadminグループのオーナーにはwrite
# グループのメンバーにはread
access to dn="cn=admin,ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jp"
        by dn="cn=Administrator,o=kogaku,dc=kyoto-u,dc=ac,dc=jp" write
        by group/groupOfUniqueNames/owner="cn=admin,ou=groups, \
           o=kogaku,dc=kyoto-u,dc=ac,dc=jp" write
```

↑グループに属していることを条件としている。「group」=グループ所属が条件であることを示す。
「groupOfUniqueNames」=対象グループのオブジェクトクラス名
「owner」=メンバ
を登録している。属性名これらを「/」で区切り、続けて「=」でグループエントリのdnを記述する。
上記の場合、cn=admin・・グループのowner属性に登録されているメンバが該当し、それらのメンバに対して読み書き可能である指定となっている。

```
        by group/groupOfUniqueNames/uniqueMember="cn=admin,ou=groups, \
           o=kogaku,dc=kyoto-u,dc=ac,dc=jp" read
```

↑同様にuniqueMemberに登録されているメンバが該当し、読み込み可能である指定となっている。

```
        by * none
```

↑一般（認証なし）には読み込み不可能。

```
#
# cn=(.+),cn=(.+),ou=groupsに関するアクセス権
```

↑ou=groupsの階層構造の詳細はDITの構造を参照。

```
# 管理者(Administratorおよびadminグループのメンバ)とグループのオーナーにはwrite
```

```
# グループカテゴリの管理者(cn=$2,ou=groupsのメンバ)にはwrite
# グループのメンバーにはread
access to dn.regex="cn=(.+),cn=(.+),ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jp"
```

↑各グループにマッチするよう正規表現を用いることができる。

```
        by dn="cn=Administrator,o=kogaku,dc=kyoto-u,dc=ac,dc=jp" write
        by group/groupOfUniqueNames/uniqueMember="cn=admin, \
        ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jp" write
        by group/groupOfUniqueNames/uniqueMember.expand="cn=$2, \
        ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jp" write
```

↑グループ名を正規表現から引用して条件として使用できる。expandスタイルを明示する。

```
        by group/groupOfUniqueNames/owner.expand="cn=$1,cn=$2, \
        ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jp" write
        by group/groupOfUniqueNames/uniqueMember.expand="cn=$1, \
        cn=$2,ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jp" read
        by * none
```

↑当該エントリのオーナーは、オーナーとメンバの変更が可能。

アクセス制御仕様

工学研究科LDAPツリーに関するアクセス制御仕様を以下に定める。
アクセス制御はACLに記述されるように、以下の要素により定義できる。

対象者
　アクセス制御の対象者。ログインしたdnや所属するグループなどが該当する。

アクセス先
　アクセスする場所。あるエントリや、その配下全体、あるいは特定の属性など。

権限
　アクセス権限の種類を表す。具体的に次の４種類となる。{読み込み(r)、書き込み(w)、新規作成(n)、削除(d)}

ACLで(論理的に)定義して使用する対象者は以下のとおりとなる。

説明記号	名称	定義
m	LDAP管理者	cn=admin,ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jpのuniqueMemberに定義されているDN
mo	LDAP管理グループの所有者	cn=admin,ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jpのowerに定義されているDN
s	自分自身	あるエントリに対してそれ自身のエントリを指すものとする
wp	個人エントリ編集者	cn=w-people,ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jpのuniqueMemberに

		登録されているDN
wo	組織エントリ編集者	cn=w-organization,ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jpのuniqueMemberに登録されているDN
u	ユーザ	ログインしたユーザ全般を指す
1o	第1階層グループ所有者	cn=*,ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jpのownerに定義されているDN
1m	第1階層グループメンバ	cn=*,ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jpのuniqueMemberに定義されているDN
2o	第2階層グループ所有者	cn=*,cn=*,ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jpのownerに定義されているDN
2m	第2階層グループメンバ	cn=*,cn=*,ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jpのuniqueMemberに定義されているDN

アクセス先は次のとおり定義される。

説明記号	名称	定義
admin	LDAP管理者グループ	エントリcn=admin,ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jp
1g	第1階層グループ	エントリcn=*,ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jp
2g	第2階層グループ	エントリcn=*,cn=*,ou=groups,o=kogaku,dc=kyoto-u,dc=ac,dc=jp
userpass	ユーザパスワード属性	ou=users,o=kogaku,dc=kyoto-u,dc=ac,dc=jp配下の各エントリにおけるuserPassword属性
peoplepass	個人パスワード属性	ou=people,o=kogaku,dc=kyoto-u,dc=ac,dc=jp配下の各エントリにおけるuserPassword属性
people	個人エントリ（パスワード属性以外）	ou=people,o=kogaku,dc=kyoto-u,dc=ac,dc=jp配下の各エントリ
organization	組織エントリ	ou=organization,o=kogaku,dc=kyoto-u,dc=ac,dc=jp配下の各エントリ
others	その他	その他のエントリ

アクセス権定義
あるDNが複数の対象者に該当する場合はそれらのアクセス権を論理orしたものとする。

	m	mo	s	wp	wo	u	1o	1m	2o	2m
admin	r	rwnd								
1g	rwnd						rw	r		

2g	rwnd							rwnd	rw	r
userpass	rwnd		rwnd							
peoplepass	rwnd		rwnd	rwnd						
people	rwnd		rwnd	rwnd		r				
organization	rwnd		rwnd		rwnd	r				
others	rwnd		rwnd			r				

なお、ldapサーバ管理者としてのDN（cn=Administrator,o=kogaku,dc=kyoto-u,dc=ac,dc=jp）については、すべての操作が可能とする。

認証の概要

上記ACLと、ACL内に現れた各権限グループをデータベースに登録することにより、認証機構としてLDAPを使用できるようになる。認証手続きは主に、phpアプリ（フォームを使用して作成）やhttpのBASIC認証（ユーザ名とパスワード入力のダイアログ表示）、PAMによるユーザ認証などがある。ここでは、上記ACLをもとに住所録参照・更新phpアプリを作成する際の認証手順例などについて記述する。

[住所録・変更アプリ(php)]
住所録参照・変更アプリは、一般ユーザ（京大外部も含む）が参照できる「Web住所録アプリ」と、ou=peopleにエントリを持つログイン可能なユーザがアクセスできる「LDAPデータ管理アプリ」からなるものとする。

機能や特徴は以下のとおり。

(1)Web住所録アプリは、ユーザ認証なしで、ある程度（非公開と指定された項目以外）の個人・組織情報を参照できる。
(2)Web住所録アプリでは、ユーザ認証を行うと、ほとんどの個人・組織情報を参照できる。
(3)Web住所録アプリでは、ユーザ認証を行うと、本人の情報をある程度（氏名や主要な所属等以外）変更できる。
(4)LDAPデータ管理アプリでは、ユーザ認証を行うと、ほとんどの個人・組織情報を参照できる。また、本人や他の人の個人情報（氏名や主要な所属以外）を変更できる。
(5)LDAPデータ管理アプリでは、ユーザ認証を行い、そのユーザが特定グループのメンバーの場合、組織の情報のすべてを参照・変更できる。
(6)LDAPデータ管理アプリでは、ユーザ認証を行い、そのユーザが特定グループのメンバーの場合、個人の情報のすべてを参照・変更できる。
(7)LDAPデータ管理アプリでは、ユーザ認証およびLDAP書き込みのログを残す。

この場合、各特長を実現するための下準備や手順は以下の通りとなります。

(1)Web住所録用の専用エントリをou=usersに１つ作成しておく。（このエントリはou=groupsのLDAPに対するアクセス制限が定義された特定のグループには属さない）その専用エントリのアプリ名とパスワードを使用してLDAP認証(バインド)し、情報を取り出して表示する。

(2)ユーザ認証のときに入力されたユーザ名（ou=peopleのエントリに対応）とパスワードを使用してLDAP認証（バインド）し、情報を取り出して表示する。

(3)ユーザ認証のときに入力されたユーザ名（ou=peopleのエントリに対応）とパスワードを使用してLDAP認証（バインド）し、情報を取り出して表示する。本人の情報を変更する際も、同じエントリの権限を使用する。

(4)LDAPデータ管理アプリ用の専用エントリをou=usersに１つ作成しておく。（このエントリはou=groupsのLDAPに対する読み書きが可能な特定のグループに属する）その専用エントリのアプリ名とパスワードを使用してLDAPにアクセスし、本人や他の人の個人情報（氏名や主要な所属以外）を参照・変更する。また、組織情報を参照する。

(5)(4)とほぼ同様。ou=groupsにあらかじめ特定のグループを作成しておき、ログインユーザが特定グループに所属していれば、組織情報のすべてを参照・変更できる。

(6)(4)とほぼ同様。 ou=groupsにあらかじめ特定のグループを作成しておき、ログインユーザが特定グループに所属していれば、個人情報のすべてを参照・変更できる。

(7)LDAPデータ管理アプリではユーザ認証および書き込みのログを残すようにする。LDAPに書き込みを行う際、すべてLDAPデータ管理アプリ用の専用エントリを使ってアクセスする。よって、書き込みをトレースする場合、LDAPサーバログだけでは書き込んだ個人を特定できない。したがって、アプリ側でもユーザ認証時のログ（時刻とユーザ名、その他）および書き込みのログを残すこととする。

[BASIC認証]

BASIC認証を必要とするwebアプリごとに、ou=groups配下にグループを作成し、メンバーを登録しておく。apache等の認証モジュールの設定で、LDAPをアクセスするように設定し、指定グループに所属していれば認証OKとすればよい。詳細はweb認証（apache）を参照。

[PAMによる認証]

UNIXのシステム認証にLDAPを使用する。ただし、各ホストではユーザ一、グループを従来のようなpasswd,groupファイル等に格納し、shadowファイルに記述されるようなパスワードのみをLDAPに問い合わせることとなる。詳細はPAMによる認証を参照。