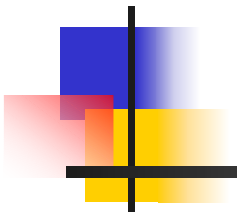
OpenLDAP
トラブル対処の第一歩

日本LDAPユーザ会
伊藤忠テクノソリューションズ株式会社
菊池研自

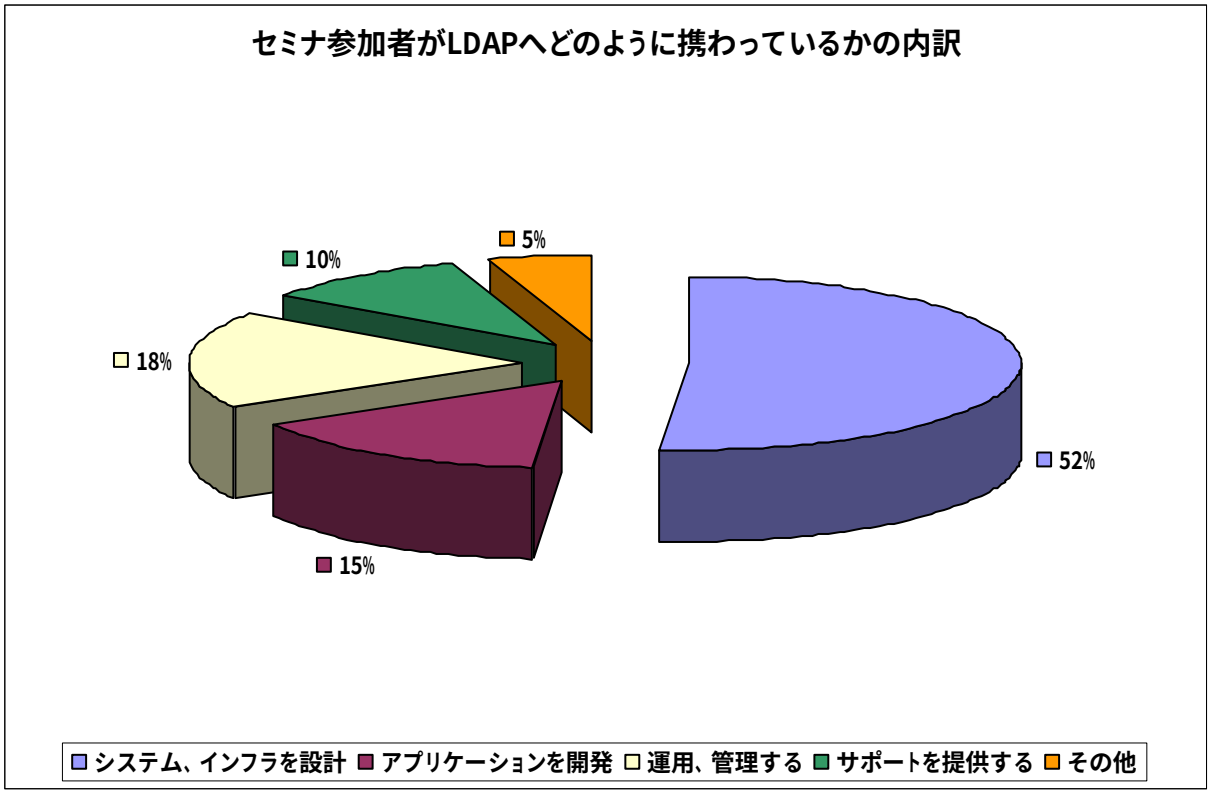


オープンソースカンファレンス2008 Tokyo/Springアンケート結果

昨年のOSCで、ご協力頂いたアンケートの集計結果を、まだ皆様にフィードバックしておりませんでした…

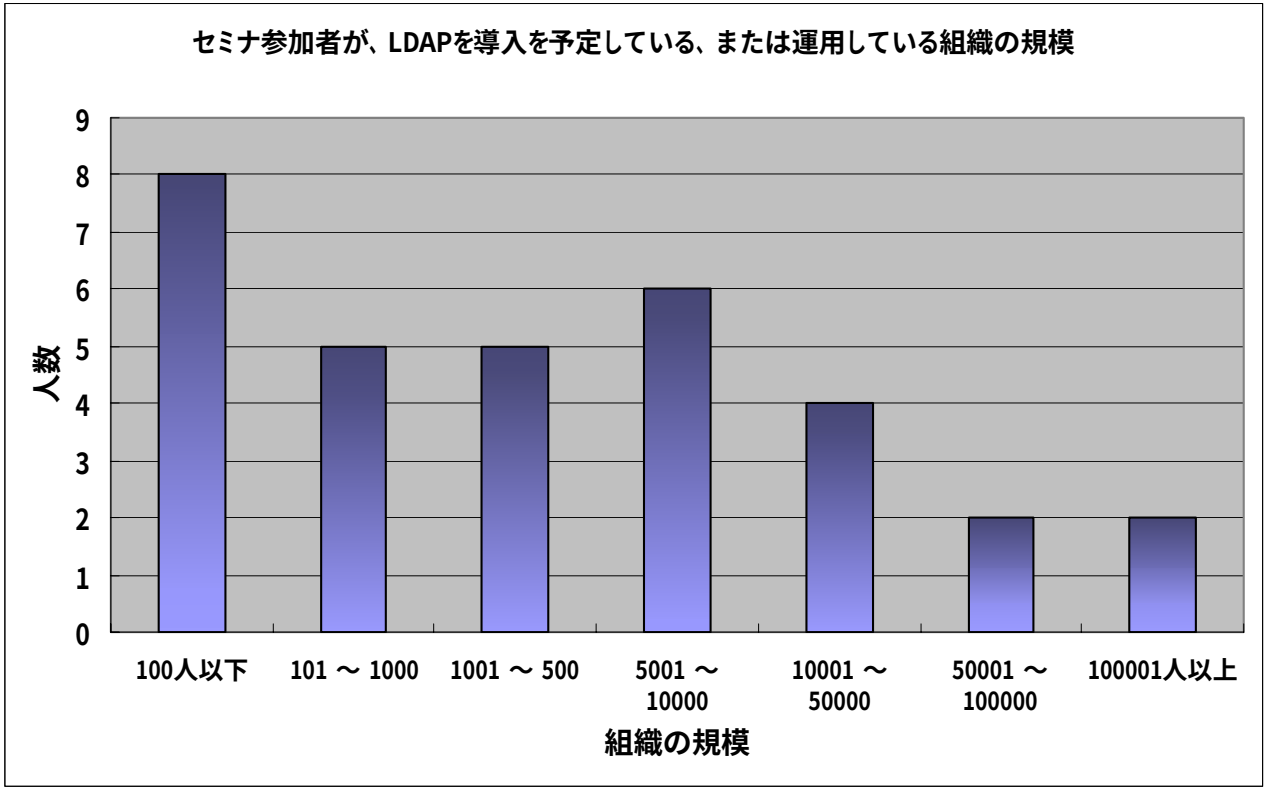
セミナ参加者の内訳

■ LDAPには、どのような立場からたずさわっていますか？



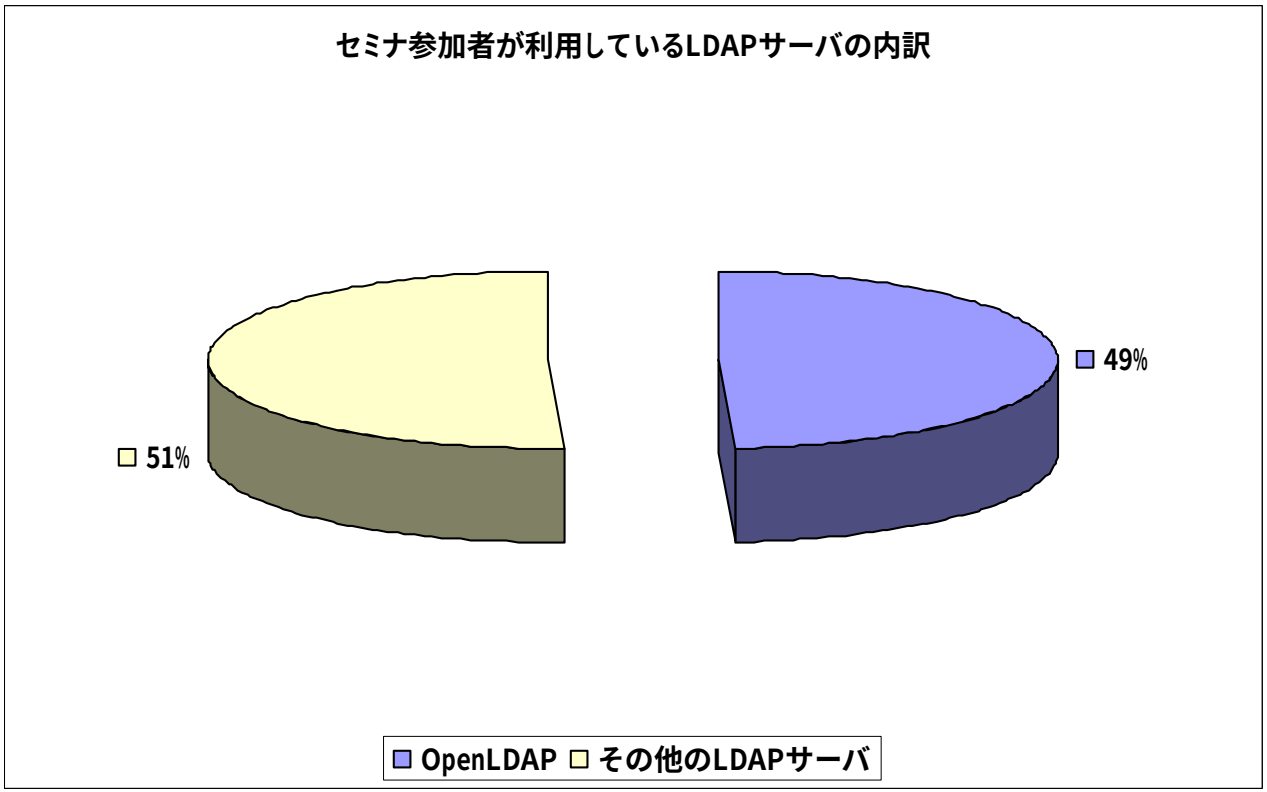
LDAPを利用する組織の規模

■ LDAPを運用している、または導入を検討している組織の規模を教えて下さい。



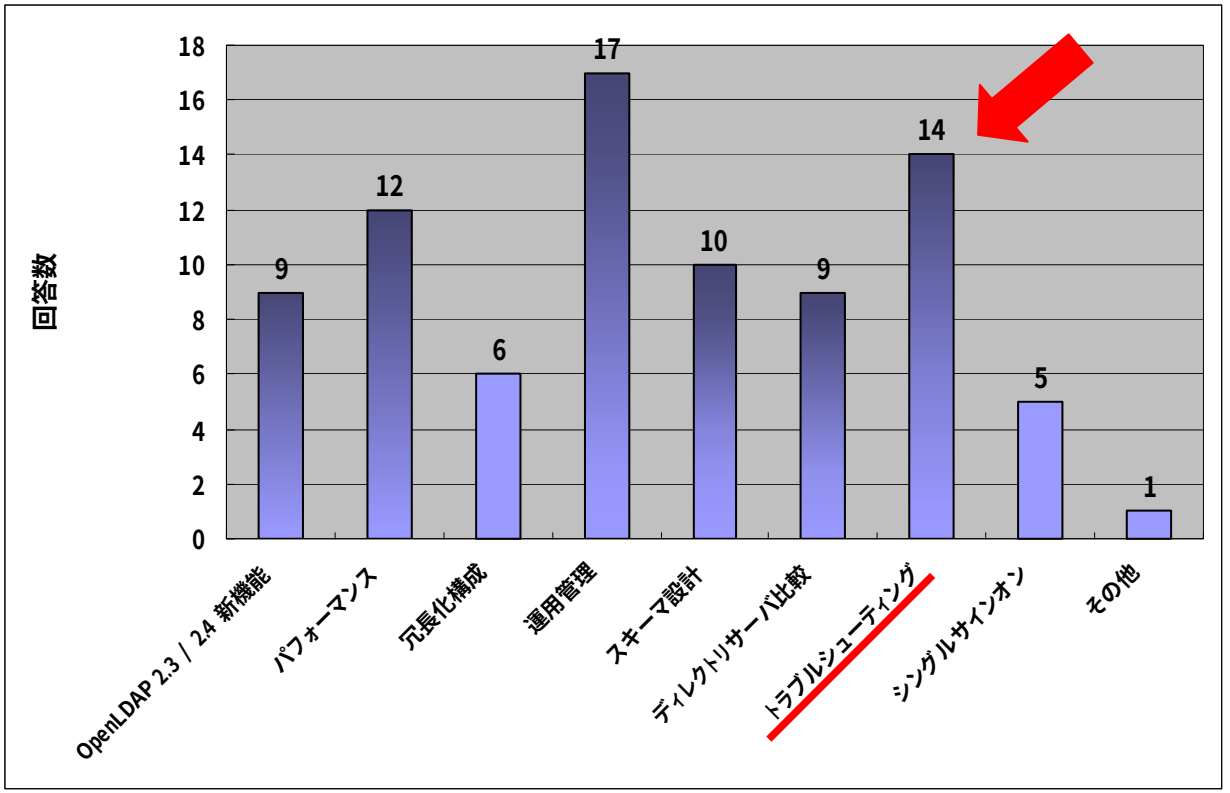
利用しているLDAPサーバ

■ご利用のLDAPサーバを教えてください。



次回、聞きたいテーマ

■次回セミナーで聞いてみたいテーマは何ですか？



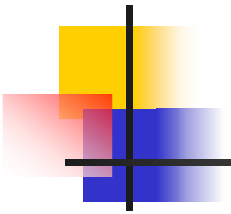
アンケートのお願い

今回もアンケートへのご協力、よろしくお願いします！

OpenLDAP
トラブル対処の第一歩

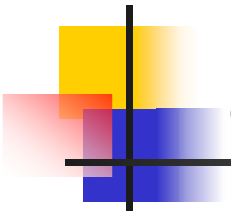
今回、コマンド等の表記は、
- Cent OS 5.x **及び、**
- OpenLDAP 2.4.x **を**

ベースにしております。



目次

- **よくある、トラブル**
- OpenLDAP**でのログの取得方法**
- loglevel 256 **でのログ調査 (概要)**
- loglevel 256 **でのログ調査 (詳細)**
  - Accept、close
  - BIND、BIND RESULT
    - Case 1: **認証に失敗する**
  - SRCH、SEARCH RESULT
    - Case 2: **検索が上手くいかない**
  - **その他**(ADD/MOD、slapd.conf)
    - Case 3: **エントリを登録させてくれない**
    - Case 4: **設定が上手くいかない**
- **その他**



よくある、トラブル

- 使い始めのころ、よく起きるトラブル
  - Case 1: 認証に失敗する
  - Case 2: 検索が上手くいかない
  - Case 3: エントリを登録させてくれない
  - Case 4: 設定が上手くいかない
  - …
- 多くの場合、ログ等に出力されるメッセージに、解決のヒントが隠されている
  - 業務であればログ等の情報を根拠に、次のアクションを検討しなくてはならない場面も…

目次

- よくある、トラブル
- **OpenLDAPでのログの取得方法**
- loglevel 256 **でのログ調査 (概要)**
- loglevel 256 **でのログ調査 (詳細)**
  - Accept、close
  - BIND、BIND RESULT
    - Case 1: **認証に失敗する**
  - SRCH、SEARCH RESULT
    - Case 2: **検索が上手くいかない**
  - **その他**(ADD/MOD、slapd.conf)
    - Case 3: **エントリを登録させてくれない**
    - Case 4: **設定が上手くいかない**
- **その他**

OpenLDAPでのログ取得方法

- Syslog**経由で取得**
- Access_log overlay**機能で取得**

Step 1. ログ取得の準備

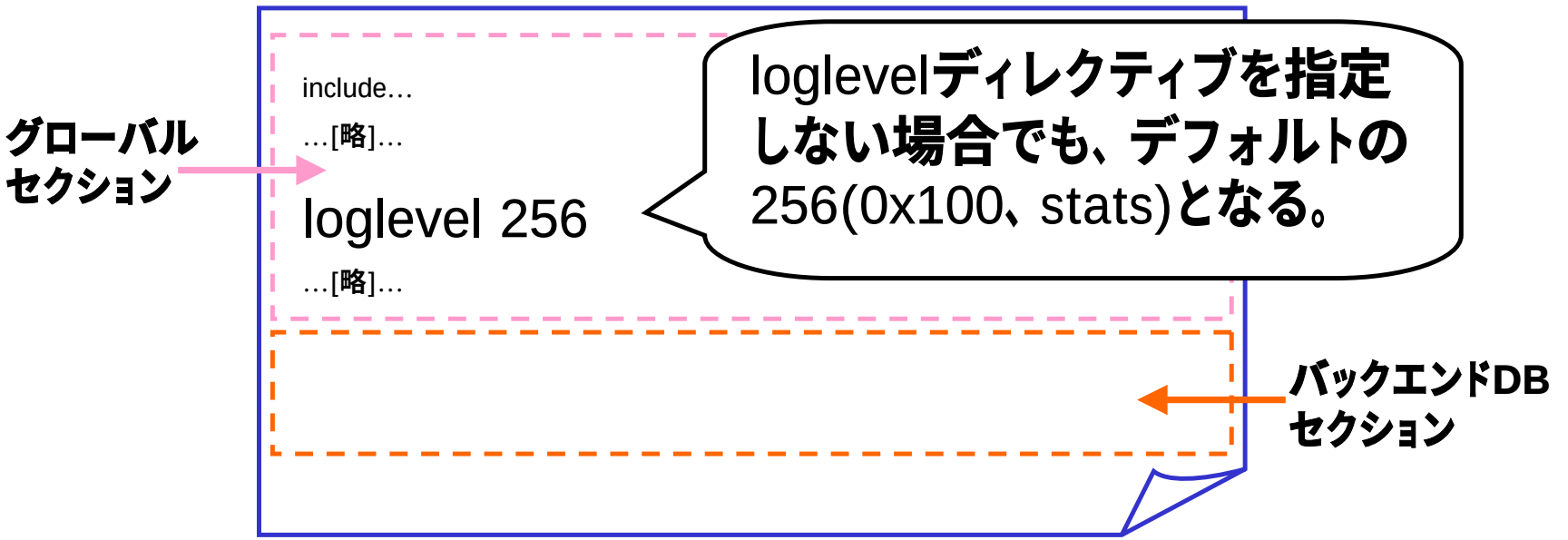
- OpenLDAPは、デフォルトでLOCAL4ファシリティを用いてログメッセージを送付
- Syslog側で、 LOCAL4ファシリティログを受け取る設定が必要
  - # vi /etc/syslog.conf

```
…[略]…

local4.*                        /var/log/ldap.log

…[略]…
```

  - Syslog 再起動

Step 2. ログ送信の準備

- **ログの出力内容(レベル)を設定**
  - # vi slapd.conf



  - Slapd **再起動**

ログ出力内容(レベル)一覧

- ログレベル

Level (10進数)	Level (16進数)	Level (文字列)	説明
-1		any	enable all debugging
0	0x0		no debugging
1	0x1	trace	trace function calls
2	0x2	packet	debug packet handling
4	0x4	args	heavy trace debugging (function args)
8	0x8	conns	connection management
16	0x10	BER	print out packets sent and received
32	0x20	filter	search filter processing

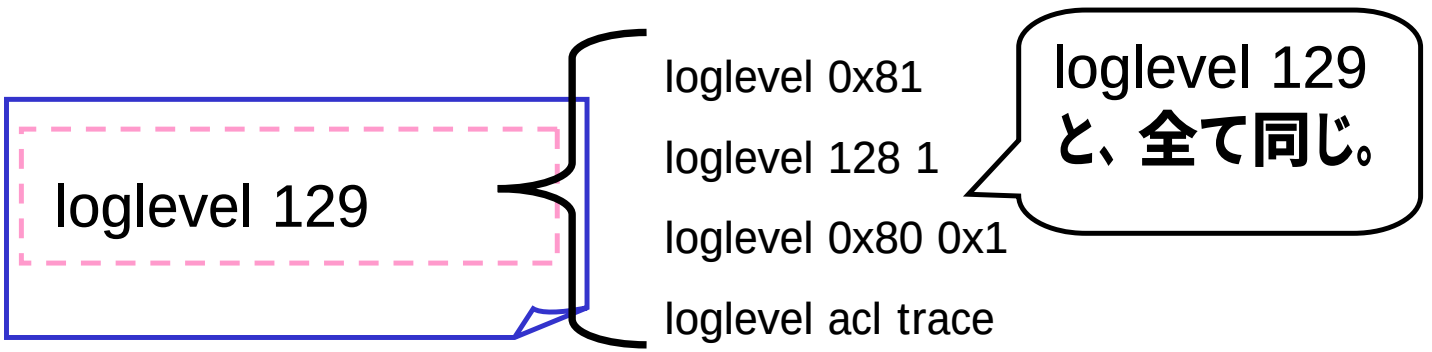
ログ出力内容(レベル)一覧(2)

- ログレベル

Level (10進数)	Level (16進数)	Level (文字列)	説明
64	0x40	config	configuration file processing
128	0x80	ACL	access control list processing
256	0x100	stats	stats log connections/operations/results
512	0x200	stats2	stats log entries sent
1024	0x400	shell	print communication with shell backends
2048	0x800	parse	entry parsing
16384	0x4000	sync	LDAPSync replication
32768	0x8000	none	only messages that get logged whatever log level is set

組合せの指定方法

- loglevelディレクティブには、大まかに、2つの表記での指定が可能
  - 表記方法1: 10進数、または16進数でログ出力したいコンポーネントの値を足し合わせて指定
  - 表記方法2: 10進数、16進数、または文字列を空白区切りでログ出力したいコンポーネントを指定



参考: ログ出力のデメリット

- **ログの取得(ログファイルへの出力)は、即ち**Disk**への書き込み処理**
  - Battery Back Write Cache**なし/仮想環境**(DiskI/O**の弱い環境)にて、**# time ldapsearch -x **で計測**

localhost**に接続し、**24**エントリを取得**

OpenLDAP	Syslog設定	処理時間
loglevel –1	**同期 書込み**	80.601(sec)
loglevel –1	**非同期 書込み**	0.647(sec)
loglevel 256	**同期 書込み**	0.151(sec)
loglevel 256	**非同期 書込み**	0.133(sec)

参考: ログ出力のデメリット(2)

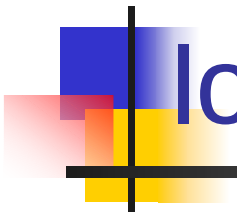
- **過剰なログ出力は、性能の劣化、ディスク領域の圧迫、欲しいログの見落としの要因につながるので注意が必要**

- **運用環境では、デフォルト (または出力が少なめのログレベルに)設定し、syslogでの非同期書き込みを設定するなど、トラブル対処時の設定とは分けるべき**
  - # vi /etc/syslog.conf

```
mail.*                  -/var/log/maillog
...[略]...
local4.*                -/var/log/ldap.log
```

目次

- よくある、トラブル
- OpenLDAPでのログの取得方法
- **loglevel 256 でのログ調査 (概要)**
- loglevel 256 でのログ調査 (詳細)
  - Accept、close
  - BIND、BIND RESULT
    - Case 1: 認証に失敗する
  - SRCH、SEARCH RESULT
    - Case 2: 検索が上手くいかない
  - その他(ADD/MOD、slapd.conf)
    - Case 3: エントリを登録させてくれない
    - Case 4: 設定が上手くいかない
- その他



loglevel 256 でのログ調査 (概要)

ここでは、OpenLDAPの基本的なアクセスログ(loglevel 256)の読み方を説明し、LDAPクライアントがリクエストした処理の概要を把握できるようになって頂ます。

loglevel 256 でのログ出力

- ldapsearchを実行しログ出力を確認
  - ldapsearch -x -D DN -w passwd filter attr

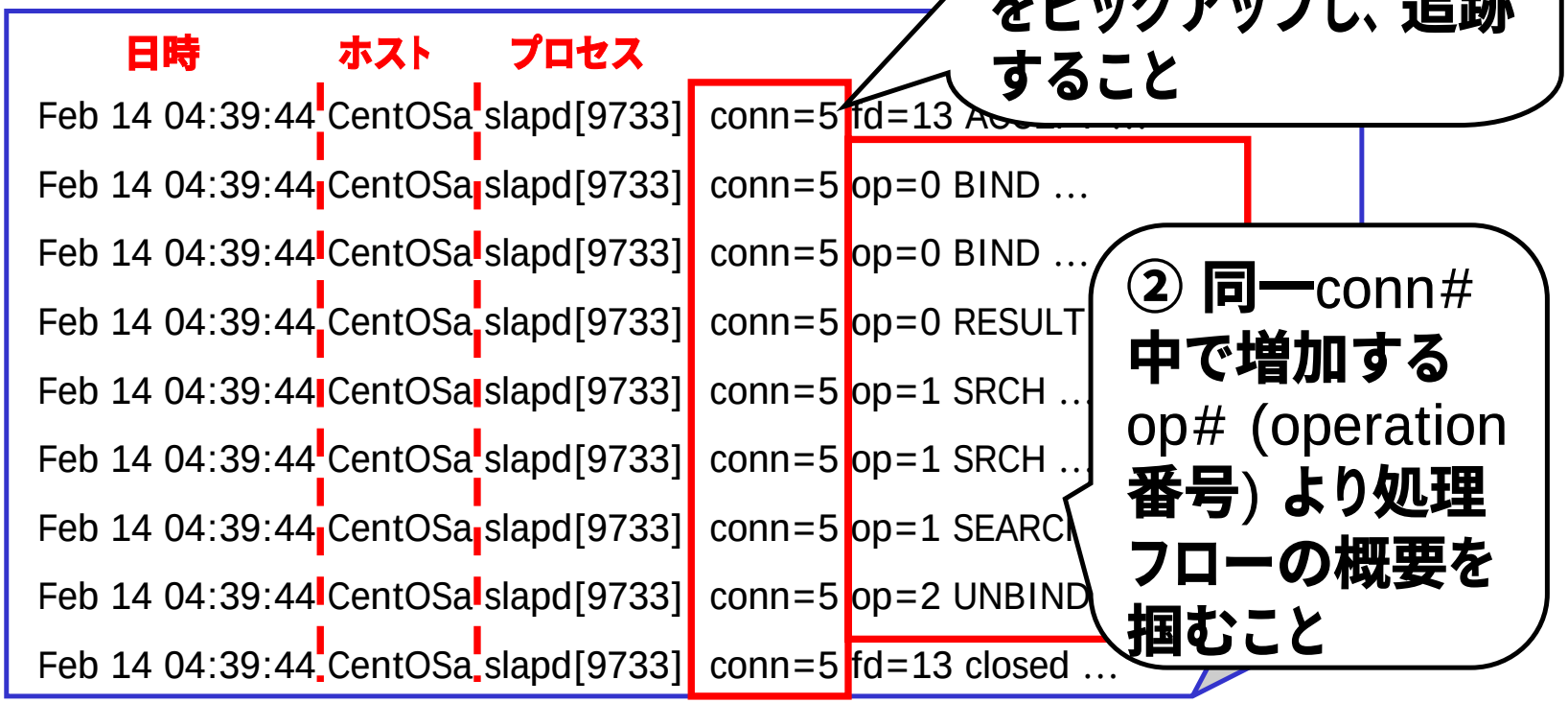
```
# ldapsearch -x -D uid=test1001,ou=people,dc=my-domain,dc=com -w 3edcvfr4 uid=test1001 dn -LLL
dn: uid=test1001,ou=People,dc=my-domain,dc=com
```

  - tail -f /var/log/ldap.log

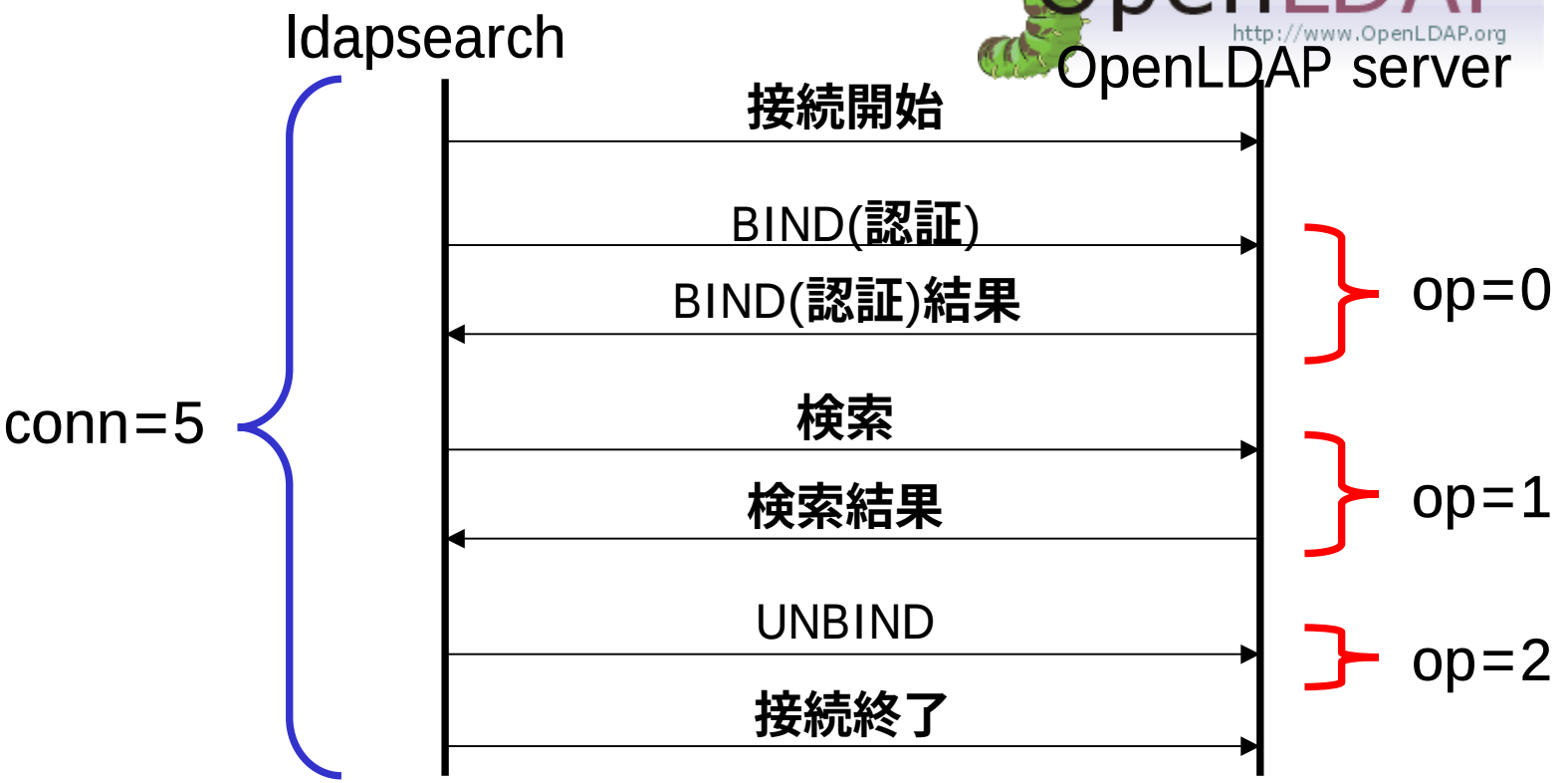
```
Feb 14 04:39:44 CentOSa slapd[9733]: conn=5 fd=13 ACCEPT from IP=127.0.0.1:38379 (IP=0.0.0.0:389)
Feb 14 04:39:44 CentOSa slapd[9733]: conn=5 op=0 BIND dn="uid=test1001,ou=people,dc=my-domain,dc=com" method=128
Feb 14 04:39:44 CentOSa slapd[9733]: conn=5 op=0 BIND dn="uid=test1001,ou=People,dc=my-domain,dc=com" mech=SIMPLE ssf=0
Feb 14 04:39:44 CentOSa slapd[9733]: conn=5 op=0 RESULT tag=97 err=0 text=
Feb 14 04:39:44 CentOSa slapd[9733]: conn=5 op=1 SRCH base="dc=my-domain,dc=com" scope=2 deref=0 filter="(uid=test1001)"
Feb 14 04:39:44 CentOSa slapd[9733]: conn=5 op=1 SRCH attr=dn
Feb 14 04:39:44 CentOSa slapd[9733]: conn=5 op=1 SEARCH RESULT tag=101 err=0 nentries=1 text=
Feb 14 04:39:44 CentOSa slapd[9733]: conn=5 op=2 UNBIND
Feb 14 04:39:44 CentOSa slapd[9733]: conn=5 fd=13 closed
```

Step 1. 処理フローの把握

- **ログの左側部分に着目**

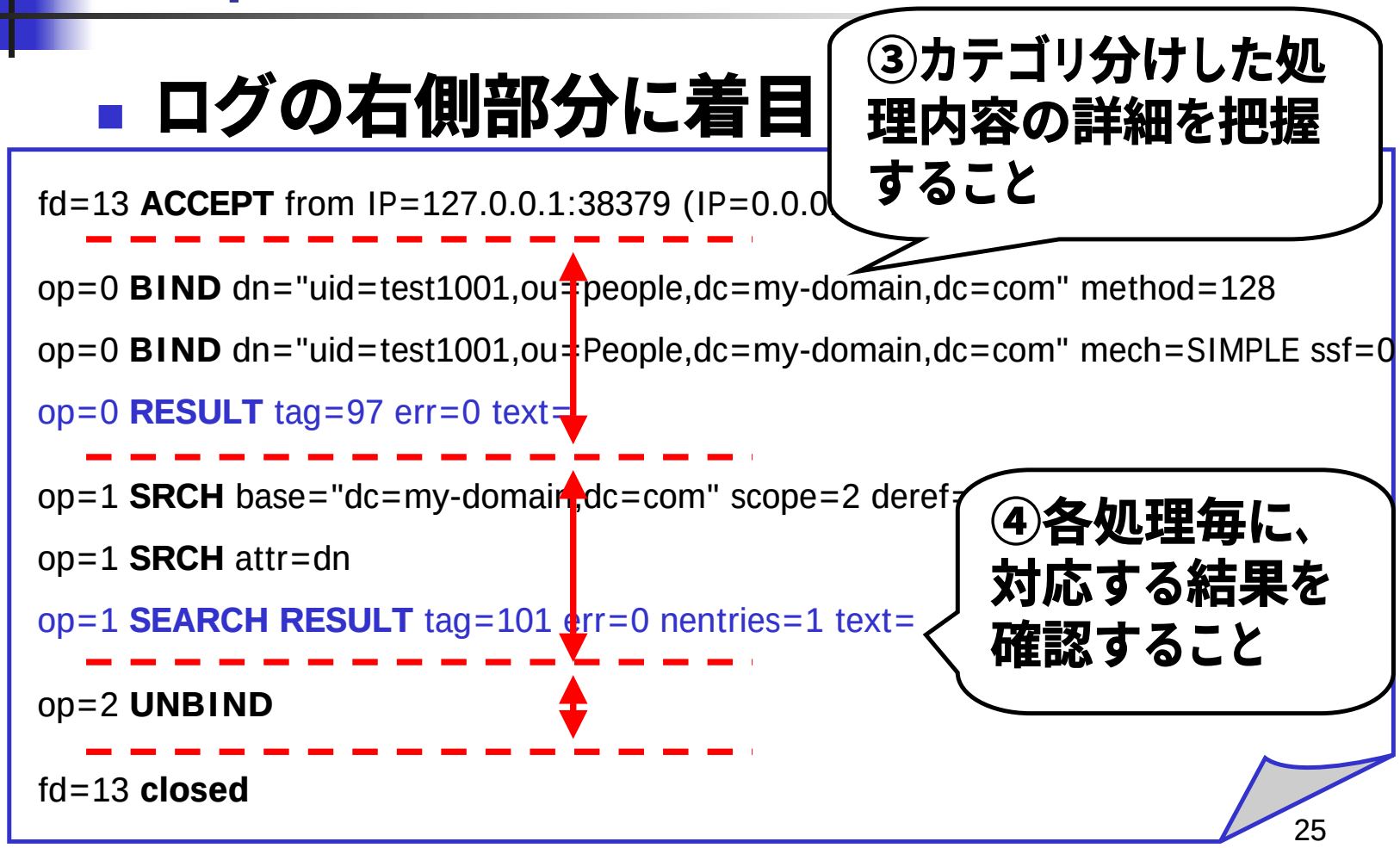


Step 1. で確認できた処理フロー

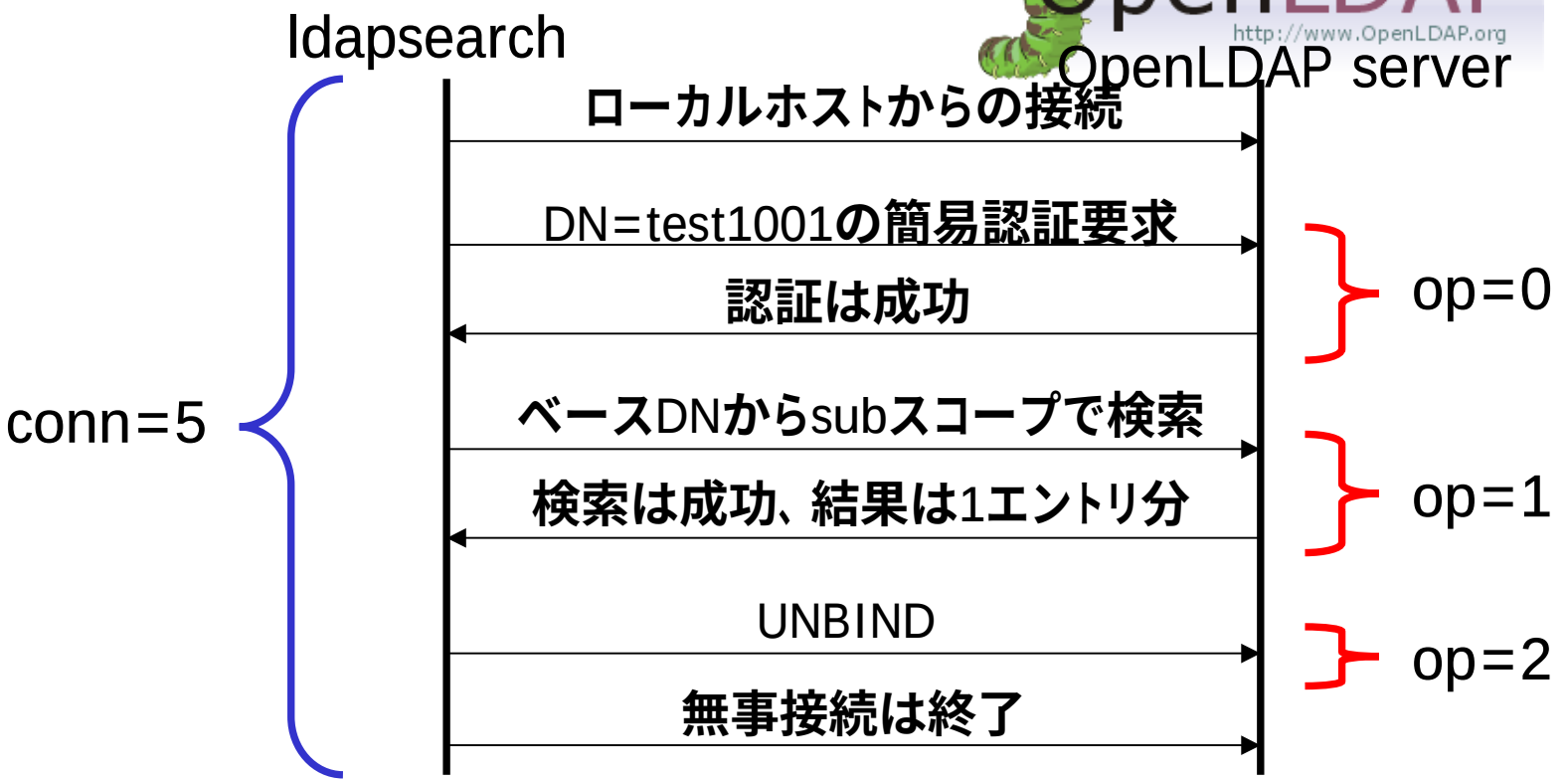


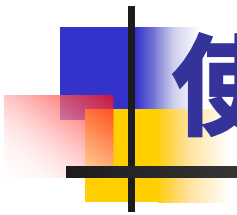
Step 2. 処理内容と結果の把握

- ログの右側部分に着目



Step 2.で確認できた処理と結果





使える便利ツール

- statslog

statslog

- **ログを整形する便利ツール**
  - contrib/slapd-tools **ディレクトリにあるperlスクリプト**
  - loglevel 256 (stats)**で出力されたログをコネクション#をベースにソートし直す便利ツール**
    - **コネクションが入り乱れているログで便利**
  - **正規表現を用いて、欲しいログを選択可能**
- **使い方は、**

```
# cd openldap-2.4.XX/contrib/slapd-tools
# ./statslog --usage
```

statslog (2)

- **例) 2月に認証に失敗しているTLS接続は…**

```
# cd openldap-2.4.XX/contrib/slapd-tools

# ./statslog -i ^Feb.*err=49 -i TLS /var/log/ldap.log
Feb  7 00:28:51 CentOSa slapd[7376]: conn=3 fd=15 ACCEPT from IP=127.0.0.1:52996 (IP=0.0.0.0:636)
Feb  7 00:28:52 CentOSa slapd[7376]: conn=3 fd=15 TLS established tls_ssf=256 ssf=256
Feb  7 00:28:52 CentOSa slapd[7376]: conn=3 op=0 BIND dn="cn=Manager,dc=my-domain1,dc=com" method=128
Feb  7 00:28:52 CentOSa slapd[7376]: send_ldap_result: conn=3 op=0 p=3
Feb  7 00:28:52 CentOSa slapd[7376]: conn=3 op=0 RESULT tag=97 err=49 text=
Feb  7 00:28:52 CentOSa slapd[7376]: connection_closing: readying conn=3 sd=15 for close
Feb  7 00:28:53 CentOSa slapd[7376]: conn=3 fd=15 closed (connection lost)

Feb 14 05:57:14 CentOSa slapd[9978]: conn=0 fd=15 ACCEPT from IP=127.0.0.1:50294 (IP=0.0.0.0:636)
Feb 14 05:57:14 CentOSa slapd[9978]: conn=0 fd=15 TLS established tls_ssf=256 ssf=256
Feb 14 05:57:14 CentOSa slapd[9978]: conn=0 op=0 BIND dn="uid=test1001,ou=people,dc=my-domain,dc=com" method=128
Feb 14 05:57:14 CentOSa slapd[9978]: conn=0 op=0 RESULT tag=97 err=49 text=
Feb 14 05:57:14 CentOSa slapd[9978]: conn=0 fd=15 closed (connection lost)
```

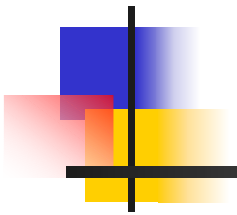
目次

- よくある、トラブル
- OpenLDAPでのログの取得方法
- loglevel 256 でのログ調査 (概要)
- **loglevel 256 でのログ調査 (詳細)**
  - Accept、close
  - BIND、BIND RESULT
    - Case 1: 認証に失敗する
  - SRCH、SEARCH RESULT
    - Case 2: 検索が上手くいかない
  - その他(ADD/MOD、slapd.conf)
    - Case 3: エントリを登録させてくれない
    - Case 4: 設定が上手くいかない
- その他

loglevel 256 でのログ調査 (詳細)

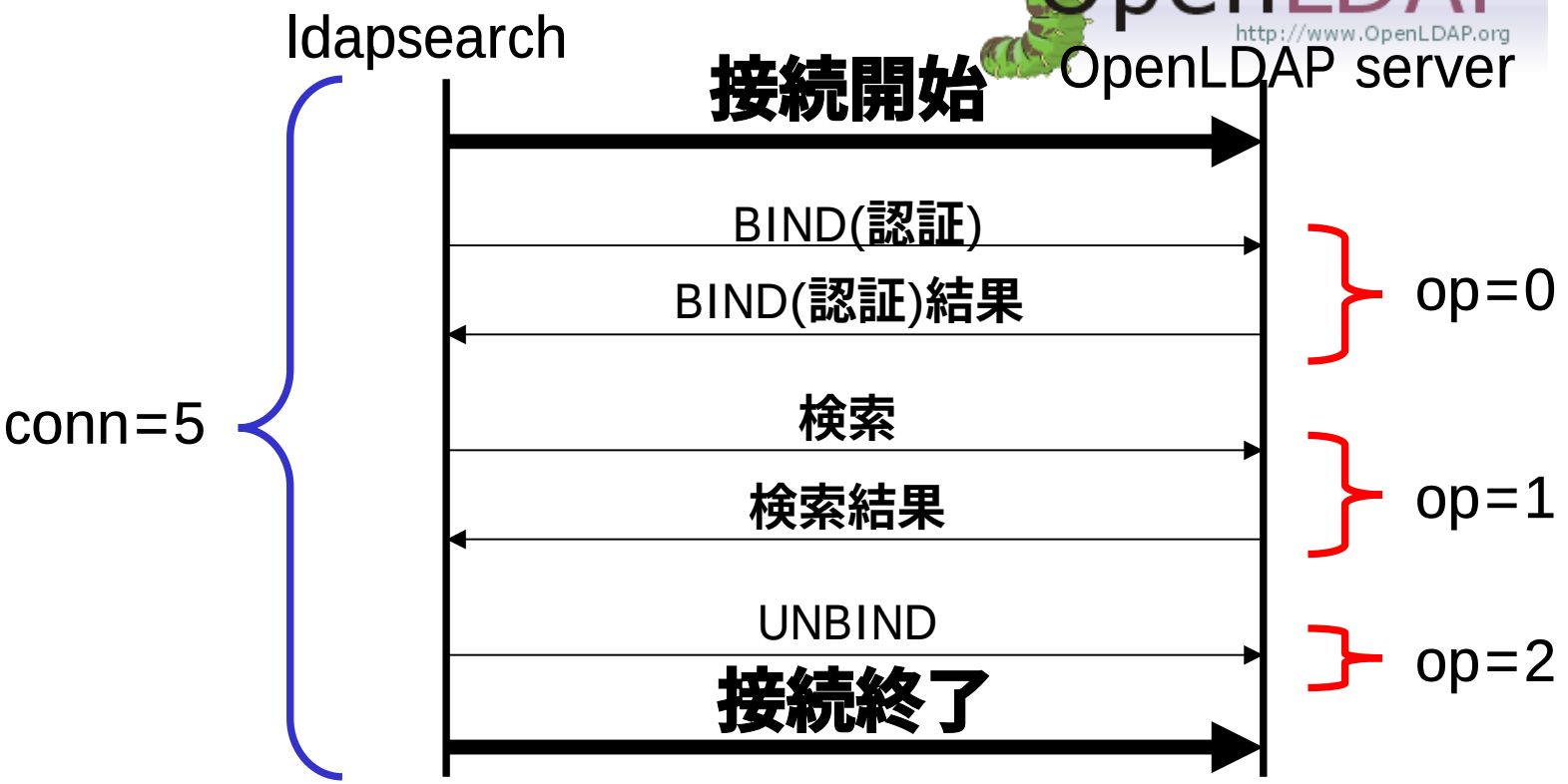
ここでは、取得したログを1行づつ説明して行きます。そして、以下3つのよくあるトラブルを解決するためのヒントをつかんで頂きます。

- Case 1: **認証に失敗する**
- Case 2: **検索が上手くいかない**
- Case 3: **エントリ登録させてくれない**



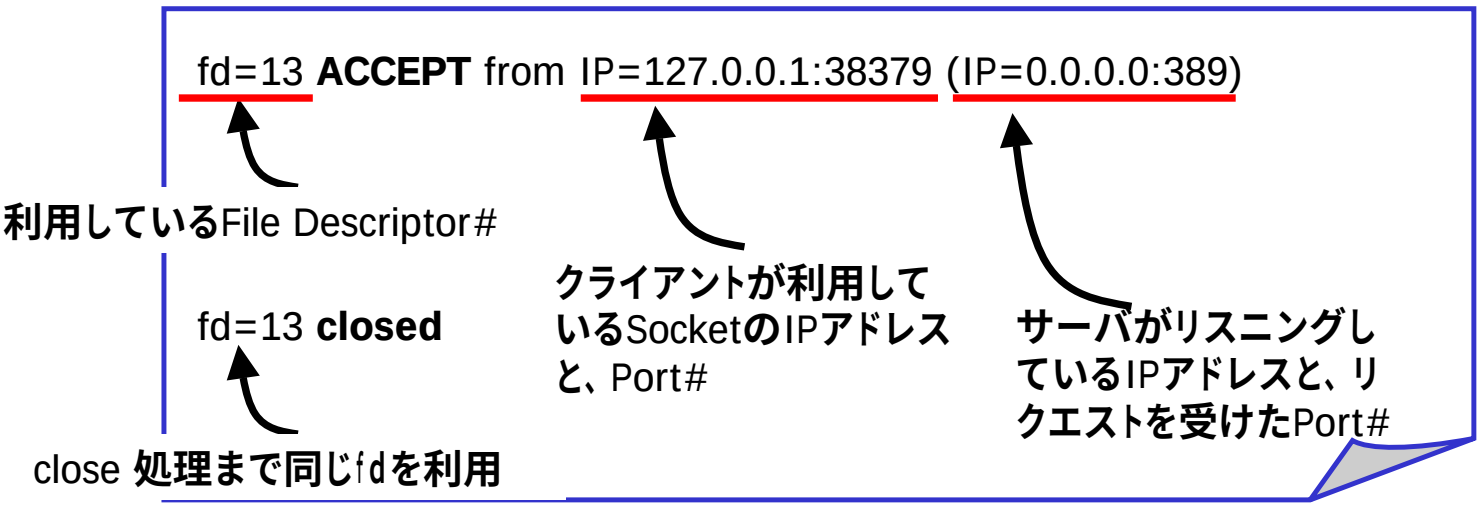
ACCEPT、close 处理

ACCEPT、close 処理



ログの確認

- ACCEPT、close



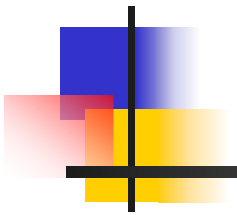
- SSL / TLS にてポート#636へ接続した場合

```
fd=14 ACCEPT from IP=127.0.0.1:38379 (IP=0.0.0.0:636)
fd=14 TLS established tls_ssf=256 ssf=256
```

ログが出力されない場合

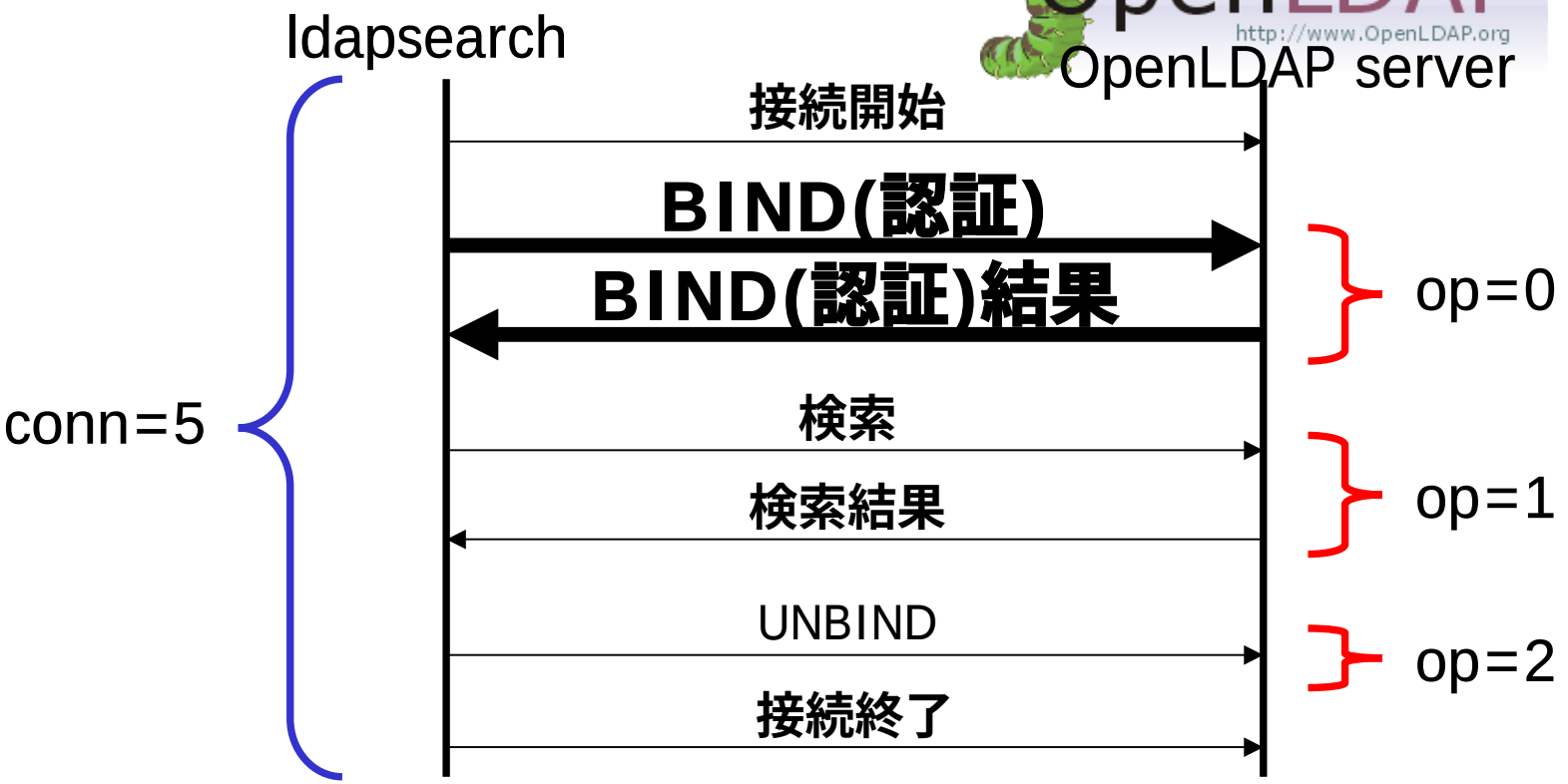
- syslog、slapd**の再起動**
- tcpdump
- lsof、fuser、netstat
- /var/log/messages
- TCP Wrappers **？**
- SSL/TLS**？**
- **ログレベルの変更 ？**

そもそも、有効なログが出力されていないと、直接の証拠を押さえられなくなるので、しっかり出力されない原因を確認し、修正



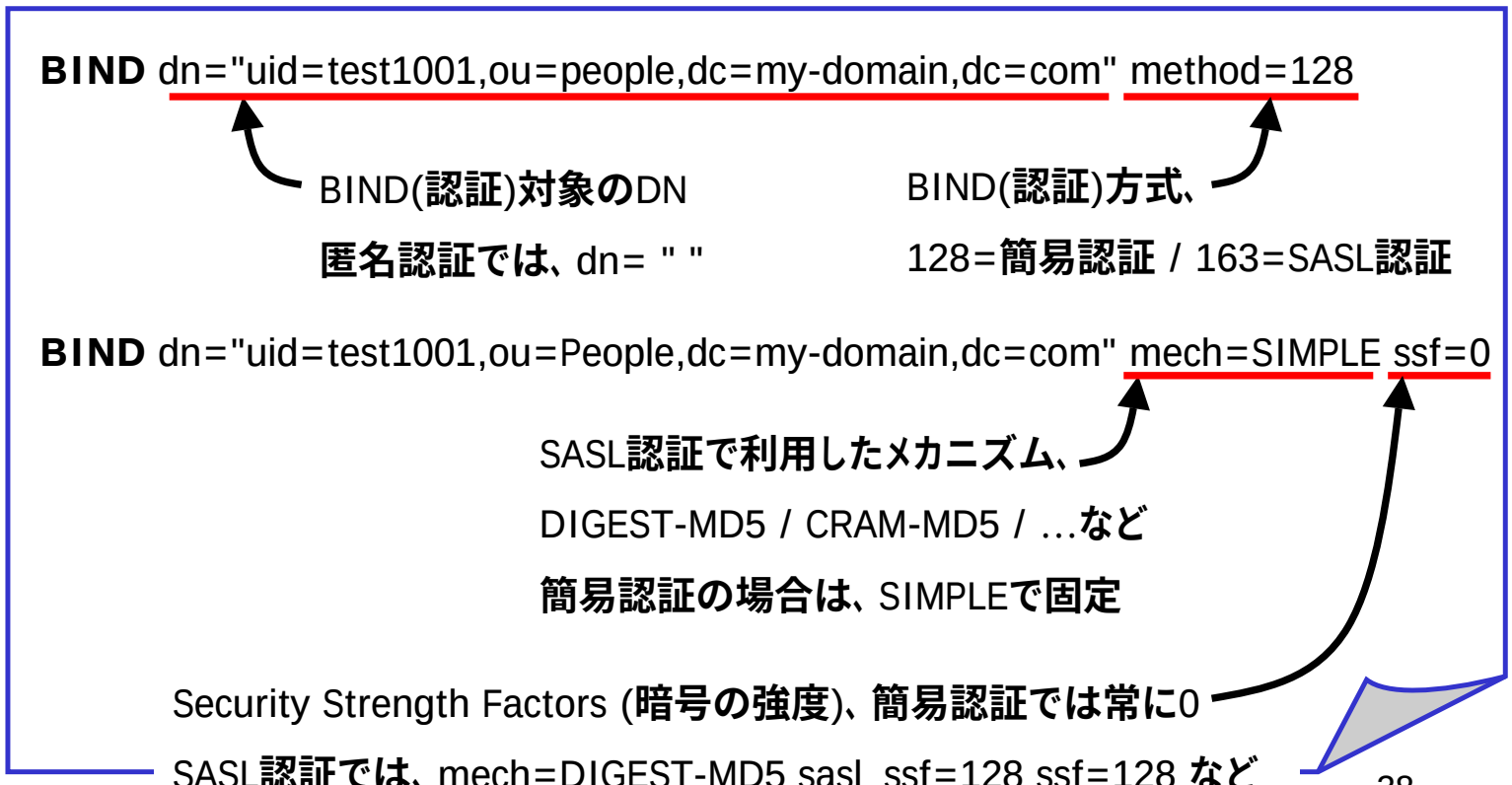
BIND、BIND RESULT 処理

BIND、BIND RESULT 処理



ログの確認

- BIND


BIND dn="uid=test1001,ou=people,dc=my-domain,dc=com" method=128
BIND(認証)対象のDN
匿名認証では、dn= " "
BIND(認証)方式、
128=簡易認証 / 163=SASL認証
BIND dn="uid=test1001,ou=People,dc=my-domain,dc=com" mech=SIMPLE ssf=0
SASL認証で利用したメカニズム、
DIGEST-MD5 / CRAM-MD5 / …など
簡易認証の場合は、SIMPLEで固定
Security Strength Factors (暗号の強度)、簡易認証では常に0
SASL認証では、mech=DIGEST-MD5 sasl_ssf=128 ssf=128 など

ログの確認(2)

- BIND RESULT

RESULT tag=97 err=0 text=

どのオペレーションに対する結果かを識別する為のタグ(参考: include/ldap.h)

エラー内容を識別するエラー#と、メッセージ(参考: include/ldap.h)

0 (0x0) = (Success **は表示なし**)

...

49 (0x31) = Invalid credentials

BIND REQUEST=96 (0x60)

BIND RESULT=97 (0x61)

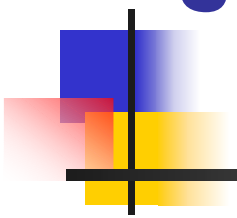
...

SEARCH REQUEST=99 (0x63)

SEARCH RESULT=101 (0x65)

...

49は、BIND処理にありがちなエラー#。DN名、パスワードに問題、または、ACL設定に問題があることも。この段回での断定は困難。



よくある、トラブル
Case 1: 認証に失敗する

= (49) Invalid credentials ?

認証が上手くいかなかったら

- DN(ID)の確認
  - 認証(BIND)処理に利用している、DN(ID)は正しいか？
    - statsレベルのログで、BIND処理関連を確認
- パスワードの確認
  - 認証(BIND)処理に利用している、パスワードは正しいか？
- ACL設定の確認
  - 認証前(anonymous)ユーザに、認証可能な権限(auth)を与えているか？
    - aclレベルのログを加え、BIND処理前後のログを確認

loglevel acl statsでのログ

- ACL設定が、良くない場合

access to userPassword
  by self write

認証前のユーザに認証権限(auth)がなければ、パスワードの一致／不一致に関わらず認証は失敗

: access_allowed: no res from state (userPassword)
: => acl_mask: access to entry "uid=test1009,ou=People,dc=my-domain,dc=com", attr "userPassword" requested
: => acl_mask: to value by "", (=0)
: <= check a_dn_pat: self
: <= acl_mask: no more <who> clauses, returning =0 (stop)
: => **access_allowed: auth access denied by =0**
: conn=0 op=0 RESULT tag=97 **err=49** text=
: conn=0 fd=15 closed (connection lost)

loglevel acl statsでのログ (2)

- ACL設定がOKで、パスワードがNGの場合

```
access to userPassword
  by self write
  by anonymous auth
```

認証前のユーザに認証権限(auth)があっても、
パスワードが不一致の場合は、認証は失敗

```
: access_allowed: no res from state (userPassword)
: => acl_mask: access to entry "uid=test1009,ou=People,dc=my-domain,dc=com",
attr "userPassword" requested
: => acl_mask: to value by "", (=0)
: <= check a_dn_pat: self
: <= check a_dn_pat: anonymous
: <= acl_mask: [2] applying auth(=xd) (stop)
: <= acl_mask: [2] mask: auth(=xd)
: => access_allowed: auth access granted by auth(=xd)
: conn=3 op=0 RESULT tag=97 err=49 text=
```

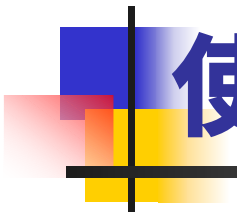
loglevel acl statsでのログ (3)

- ACL設定、パスワードともOKの場合

```
access to userPassword
  by self write
  by anonymous auth
```

認証前のユーザに認証権限(auth)があり、更に、パスワードが一致する場合には、認証は成功

```
access_allowed: no res from state (userPassword)
: => acl_mask: access to entry "uid=test1009,ou=People,dc=my-domain,dc=com",
attr "userPassword" requested
: => acl_mask: to value by "", (=0)
: <= check a_dn_pat: self
: <= check a_dn_pat: anonymous
: <= acl_mask: [2] applying auth(=xd) (stop)
: <= acl_mask: [2] mask: auth(=xd)
: => access_allowed: auth access granted by auth(=xd)
: conn=4 op=0 BIND dn="uid=test1009,ou=People,dc=my-domain,dc=com"
mech=SIMPLE ssf=0
: conn=4 op=0 RESULT tag=97 err=0 text=
```



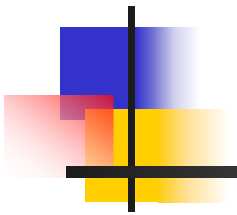
使える便利ツール

- slapacl

slapacl

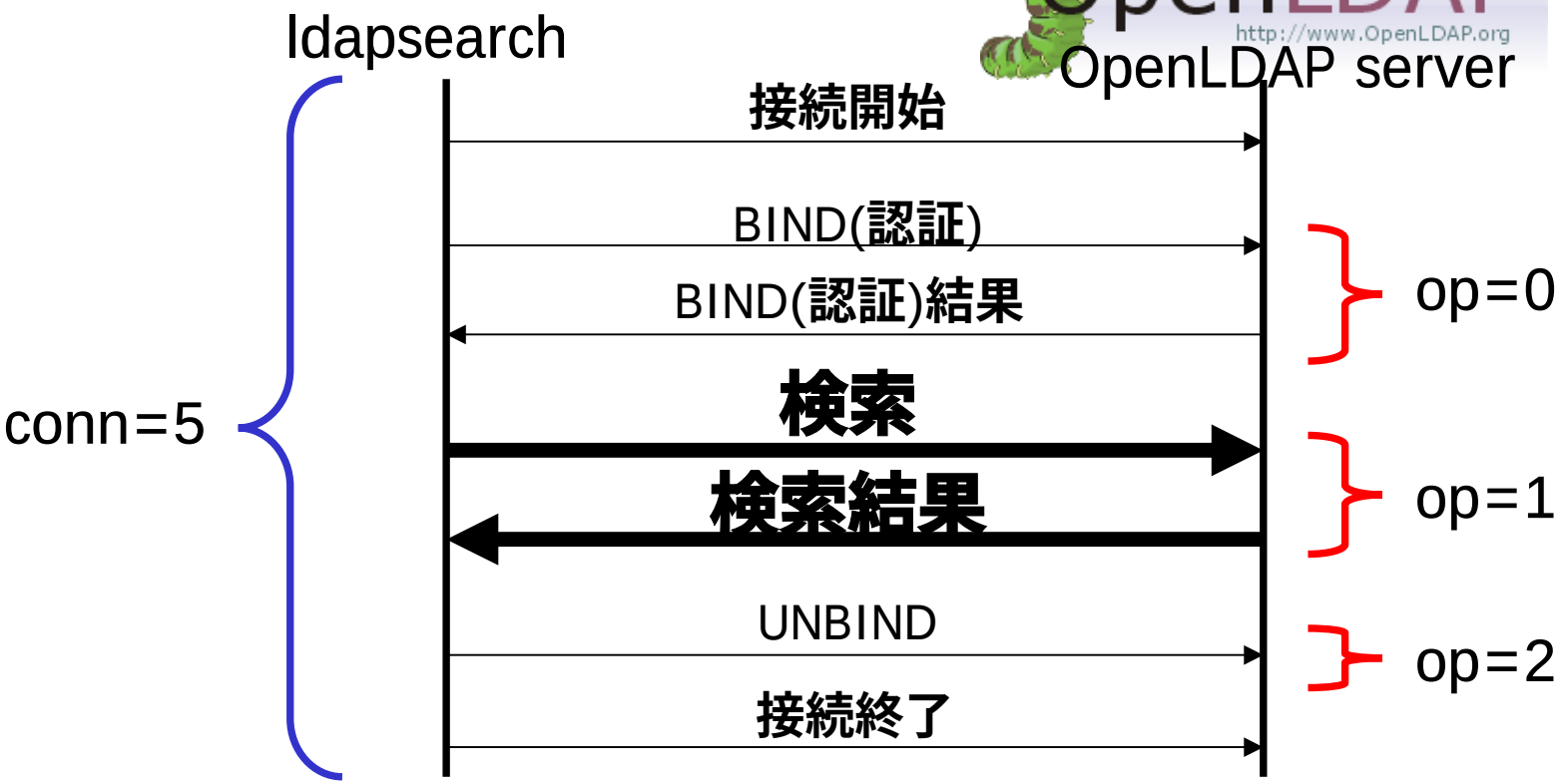
- slapacl を利用したACL確認
  - slapd.confのaccess to ディレクティブで設定したACLの振る舞いを確認可能
  - 再起動することなく、OpenLDAPサーバ起動中に、変更した場合の振る舞いを確認可能

```
# slapacl -b "uid=test1003,ou=people,dc=my-domain,dc=com" ¥
> "userPassword/auth" -d acl

…[略]…
<= check a_dn_pat: anonymous
…[略]…
=> access_allowed: auth access granted by auth(=xd)
auth access to userPassword: ALLOWED
```



SRCH、SEARCH RESULT处理

SRCH、SEARCH RESULT処理

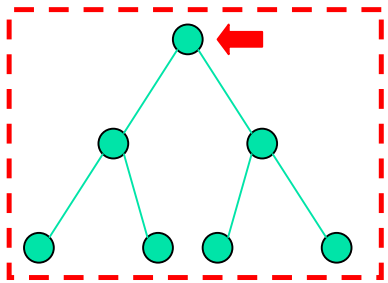


ログの確認

- SRCH

SRCH base="dc=my-domain,dc=com" scope=2 deref=0 filter="(uid=test1001)"

検索を開始するベース **検索スコープ** alias参照 **検索フィルタ**



0：base (**検索ベースのみ**)

1：one (**検索ベース以下1レベル**)

2：sub (**検索ベース以下全て**)

3：children (**検索ベースを除き以下全て**)

SRCH attr=dn

リクエストされた属性のリスト

この行がない場合は、全ユーザ属性がリクエストされた意味

attr=+ **は、運用属性がリクエストされた意味**

ログの確認(2)

- SEARCH RESULT

SEARCH RESULT tag=101 err=0 nentries=1 text=

どのオペレーションに対する結果かを識別する為のタグ(参考: include/ldap.h)

返されたエントリー数

エラー内容を識別するエラー#と、メッセージ
(参考: include/ldap.h、libraries/libldap/error.c)

0 (0x0) = Success

…

4 (0x04) = Size limit exceeded

…

32 (0x20) = No such object

BIND REQUEST=96 (0x60)

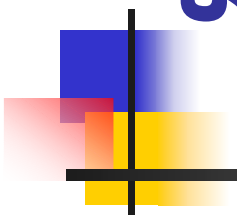
BIND RESULT=97 (0x61)

…

SEARCH REQUEST=99 (0x63)

SEARCH RESULT=101 (0x65)

…



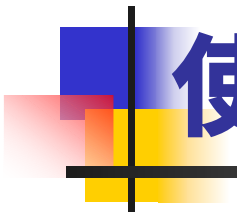
よくある、トラブル
Case 2: 検索が上手くいかない

- (3) Time Limit Exceeded ?
- (4) Size Limit Exceeded ?
- (32) No Such Object ?

…

検索が上手くいかなかったら

- ログより、検索処理内容と結果を確認
- エラー#の確認
  - http://www.openldap.org/doc/admin24/appendix-ldap-result-codes.html
  - http://tools.ietf.org/html/rfc4511#section-4.1.9
- ldapsearch に指定したオプションの確認
  - /etc/openldap/ldap.conf の設定の確認
    - ldapsearchなどOpenLDAPクライアントコマンドがデフォルトで利用する設定ファイル
  - /etc/ldap.conf の設定の確認
    - nss_ldap、pam_ldapが利用する設定ファイル



使える便利ツール

- ldapsearch -d <#>
- getent…など

LDAPクライアントからのdebug

- OpenLDAPクライアントに影響するレベル

Level (10進数)	Level (16進数)	Level (文字列)	説明
-1		any	e
0	0x0		no debugging
1	0x1	trace	tr
2	0x2	packet	d
4	0x4	args	h a
8	0x8	conns	c
16	0x10	BER	p re
32	0x20	filter	s

LIBRARY / SERVER

LIBRARY / SERVER

LIBRARY / SERVER

LIBRARY / SERVER

LIBRARY / SERVER

LIBRARY / SERVER

SERVER ONLY

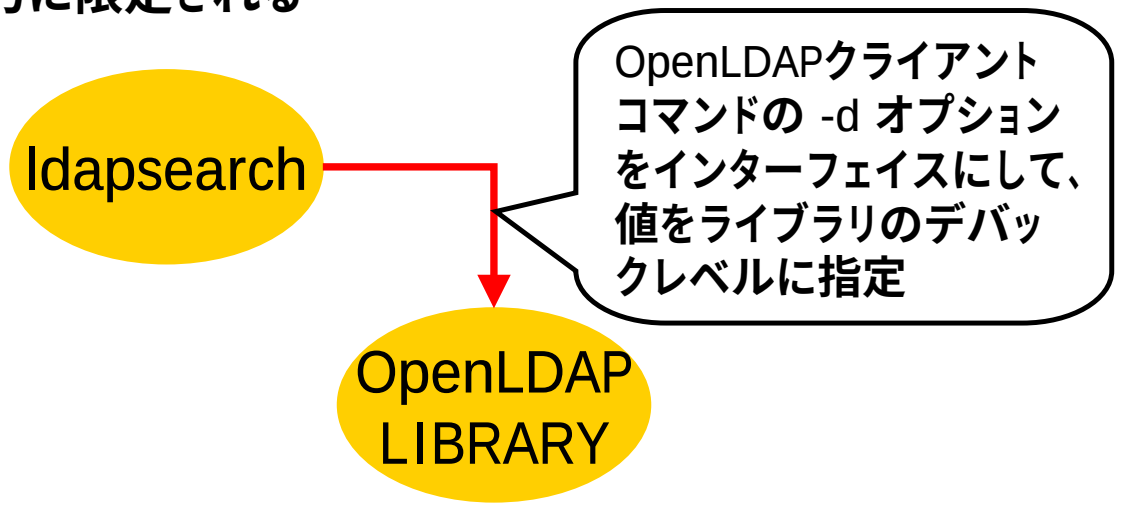
LDAPクライアントからのdebug(2)

- OpenLDAPクライアントに影響するレベル

Level (10進数)	Level (16進数)	Level (文字列)	説明
64	0x40	config	c SERVER ONLY ng
128	0x80	ACL	a SERVER ONLY ng
256	0x100	stats	s SERVER ONLY c sults
512	0x200	stats2	s SERVER ONLY
1024	0x400	shell	p SERVER ONLY th s
2048	0x800	parse	e LIBRARY / SERVER
16384	0x4000	sync	L SERVER ONLY
32768	0x8000	none	o LIBRARY / SERVER w

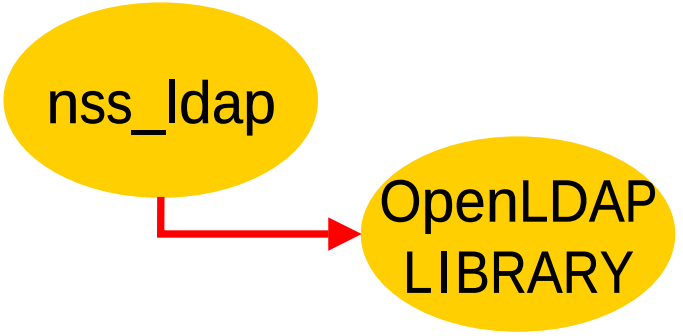
LDAPクライアントからのdebug

- ldapsearch にdebugオプションを指定
  - -d 1 などを指定し、OpenLDAPクライアント側のライブラリの動作を確認することが可能
  - 指定する値自体はslapd.confに指定するloglevelと同じ意味だが、クライアントコマンドが利用するOpenLDAPライブラリからのみの出力に限定される



NSS、PAMからのdebug

- nss_ldap、pam_ldapの場合は、/etc/ldap.conf にdebug 1 などを指定可能




```
# getent passwd test1001 2>/tmp/nss_ldap_debug.log
test1001:x:1001:1001:test1001:/home/test1001:/bin/bash

# less /tmp/nss_ldap_debug.log
…[略]…
ldap_connect_to_host: Trying 127.0.0.1:389
…[略]…
put_filter: "(&(objectClass=posixAccount)(uid=test1001))"
```

クライアント側から、接続先などを確認することが可能

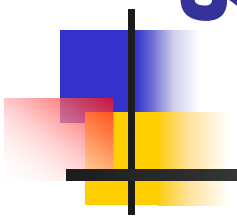
よくある、トラブル
Case 3: 登録させてくれない

- (21) Invalid Syntax ?
- (68) Entry Already Exists ?
- (80) Internal (…) error ?

…

登録(変更)が問題になったら

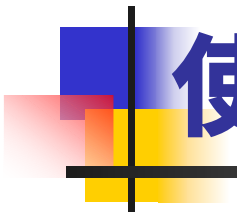
- **エラーメッセージの確認**
- **ログより、処理内容と結果を確認**
  - **エラー#の確認**
    - http://www.openldap.org/doc/admin24/appendix-ldap-result-codes.html
    - http://tools.ietf.org/html/rfc4511#section-4.1.9
- OpenLDAP FAQ、Common Errors
  - http://www.openldap.org/faq/data/cache/53.html
- LDAP**スキーマに関連するルールの再確認**
- **利用するスキーマの再確認**



よくある、トラブル
Case 4: 設定が上手くいかない

slapd.confの基本ルール

- OpenLDAPサーバがsldap.conf を解析する基本ルール
  - 先頭が、# で始まる行はコメント行として無視
  - 先頭がスペース、タブなど空白で始まる行は、それ以前の行からの継続行として扱う
  - それ以外の行は、ディレクティブ名と、それに続く、設定値として扱う



使える便利ツール

- slaptest -d 64
- slurpd -d 64

slaptest

- slaptest **を利用した**slapd.conf**の確認**
  - slapd.conf**に設定した内容を**OpenLDAP**サーバがどのように解釈するかを確認可能**
  - **再起動することなく、**OpenLDAP**サーバ起動中に、変更する設定が正しく解釈されることを確認可能**

```
# slaptest -f ./etc/openldap/slapd.conf -d 64
reading config file ./etc/openldap/slapd.conf
line 5 (include         /usr/local/openldap-2.4.11/etc/openldap/schema/core.schema)
reading config file /usr/local/openldap-2.4.11/etc/openldap/schema/core.schema
...[略]...
line 84 (rootdn         "cn=Manager,dc=my-domain,dc=com")
line 88 (rootpw ***)
...[略]...
config file testing succeeded
```

slurpd -d 64(OpenLDAP2.3まで)

- slurpd -d 64 を利用したslapd.confの確認
  - slurpdがslapd.confの内容をどのように解釈するかを確認可能
  - slurpd 向けのパラメータに関しては、slaptest -d 64より、slurpd -d 64のほうがわかりやすい

```
# slurpd -d 64 -o -r /tmp/none-exsiting-files
Config: (replogfile /var/lib/ldap/openldap-master-replog)
Config: (replica host=ldap-1.example.com:389 starttls=critical
bindmethod=sasl saslmech=GSSAPI     authcId=host/ldap-
master.example.com@EXAMPLE.COM)
Config: ** successfully added replica "ldap-1.example.com:389"
Config: ** configuration file successfully read and parsed
Processing in one-shot mode:
0 total replication records in file,
0 replication records to process.
slurpd: terminated.
```

目次

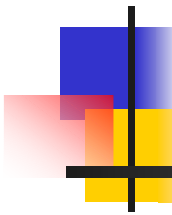
- よくある、トラブル
- OpenLDAPでのログの取得方法
- loglevel 256 でのログ調査 (概要)
- loglevel 256 でのログ調査 (詳細)
  - Accept、close
  - BIND、BIND RESULT
    - Case 1: 認証に失敗する
  - SRCH、SEARCH RESULT
    - Case 2: 検索が上手くいかない
  - その他(ADD/MOD、slapd.conf)
    - Case 3: エントリを登録させてくれない
    - Case 4: 設定が上手くいかない
- その他

トラブル対処時に利用できるドキュメント

- ✓『Administrator's Guide』
  - 22. Troubleshooting
    - http://www.openldap.org/doc/admin24/troubleshooting.html
- ✓『OpenLDAP Faq-O-Matic』
  - http://www.openldap.org/faq/data/cache/1.html
- ✓『Issue Tracking System』
  - http://www.openldap.org/its/
- ✓『Change Log』
  - http://www.openldap.org/software/release/changes.html
- 『Release Road Map』
  - http://www.openldap.org/software/roadmap.html
- 『Project Overview』
  - http://www.openldap.org/project/

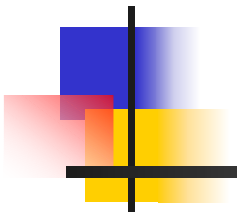
メーリングリストの利用

- OpenLDAPプロジェクト
  - http://www.openldap.org/lists/#archives
  - OpenLDAP-announce
  - OpenLDAP-bugs
  - OpenLDAP-commit
  - OpenLDAP-devel
  - ✓ OpenLDAP-software
  - ✓ OpenLDAP-technical
- 日本LDAPユーザ会
  - ✓ http://ml.ldap.jp/mailman/listinfo/ldap-users

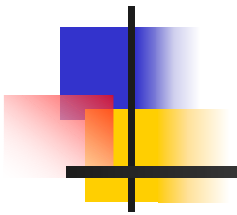


今回、説明できなかったこと

- 運用に関するトラブルシュート
- 性能に関すること
- Berkeley DBに関すること
- …などなど



それでも、OpenLDAPでの第一歩は踏み出せます！



ありがとうございました！