高品質インターネット　6－14
(2003. 2. 7)

異機種環境におけるディレクトリサービスを用いたユーザ管理機構システムの提案

能城　茂雄†　中村　豊†　藤川　和利†　砂原　秀樹†

† 奈良先端科学技術大学院大学　〒630-0101 〒630-0101 奈良県生駒市高山町 8916－5
E-mail: †shigeo-n@is.aist-nara.ac.jp, ††{yutaka-n,fujikawa,suna}@itc.aist-nara.ac.jp

あらまし　企業や学校などの UNIX や Windows などが混在した異機種環境では，ユーザ管理やシステムリソースの管理は，多くの人の手間と時間を必要とする．本学でも，NIS を中心としたユーザ管理が行われている．しかし NIS を中心としたユーザ管理やシステム管理にはセキュリティとコスト面で大きな問題があり，なりすまし問題などのセキュリティ上の課題を抱える．また異機種環境においては，相互運用性での対応が不十分であるため，非常に管理コストがかかる．これらの問題を回避するためには，異機種環境における統一した枠組みのなかで必要な情報への容易なアクセス方法を提供し，不要な情報をネットワーク上に流出しないことが重要となる．ひとつの解決策としてはディレクトリサービスを用いた手法が考えられる．しかし既存のディレクトリサービスには，セキュリティ面での懸念や既存環境からの移行などが問題となる．これらの問題を解決するために，代表的なディレクトリサービスである LDAP (Lightweight Directory Access Protocole) を検証し，LDAP 統合環境の開発を行い，本学における新アカウント管理システムを構築しその有効性を示す．
キーワード　OpenLDAP，PAM，NIS

Proposal of the user control mechanism system using the directory service (LDAP) in operating system mixture environment

Shigeo NOSHIRO†, Yutaka NAKAMURA†, Kazutoshi FUJIKAWA†, and Hideki SUNAHARA†

† NARA Institute of Science and Technology　8916-5 Takayama-cho,Ikoma-shi,Nara,630-0101,Japan
E-mail: †shigeo-n@is.aist-nara.ac.jp, ††{yutaka-n,fujikawa,suna}@itc.aist-nara.ac.jp

Abstract　Heterogeneous environment of Unix and Windows in busineses and schools, we are necessary to many time of administrator that maintain system resource and user management. In our campus, we control user account based on the NIS. However, NIS-based system has much cost of user account control and system management. These systems have also security problems that someone can impersonate other person. In heterogeneous environment, we do not have enough to correspondence in mutually operational, because management cost is much. To avoid these problems, we should provide easy way to access to necessity information into a unification framework. We make disused information flow out to network. One solution is directory service. However, conventional directory service has problem shifting from existing environment. To solve these problems, we verify the LDAP that is typical method of directory service. We construct a new account control mechanism and show that validity.
Key words　OpenLDAP，PAM，NIS

1. はじめに

企業や学校などの UNIX や Windows などが混在した異機種環境では，ユーザ管理やシステムリソースの管理は，多くの人の手間と時間を必要とする．ある規模を越えた利用者を一元的に管理する作業は，管理者のコストを増加させ，日々の管理業務を圧迫する．これまでも，こうした管理コストを軽減するために，NIS(Network Information Services) などが利用されてきた．NIS は，その利便性から広く利用されてきた，しかし NIS にはいくつかの問題点がある．第一に NIS では全ての情報がマスター・サーバに一元化されることである．ネットワークの規模が大きくなると，こうした一元的な管理は，情報

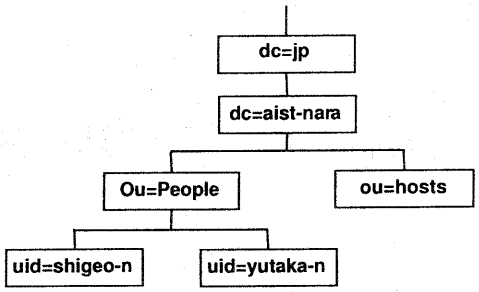

図 1 Directory Information Tree

の更新にタイムラグが生じる．第二の問題はセキュリティ上の問題である．NIS を用いたユーザ認証では，NIS クライアントはネットワークにブロードキャストにより NIS サーバを探索する．この方法では，NIS クライアントは，最初の呼びかけに答えたマシンをサーバとみなす．したがって，NIS サーバを装うことによりシステムを不正に使用することが可能となる．こららの問題を踏まえ，我々は NIS に変わる新たなユーザ認証の枠組みとして注目されている LDAP [1] に着目する．LDAP は，RFC2251 に基づき，開発されているスタンダードなディレクトリサービスのひとつである．LDAP 製品としては，Microsoft の Windows2000 Active Diretory [2] や Novell の eDirectory [3] Oracle の Oracle Internet Directory [4] などが存在する．またオープンソース実装として The Open LDAP Project が開発している OpenLDAP [7] もある．本研究では，LDAP サーバと LDAP クライアント環境を構築し，その性能を評価した．計測はクライアントソフトウェアに変更を加え，NIS との処理時間の比較をもって LDAP の性能を比較する．また NIS と比較した LDAP の優位性を述べる．

2. ディレクトリサービス

コンピュータを利用するうえで，検索はよく利用される機能のひとつである．例を挙げると，ユーザ名からパスワードデータベースを検索し，ユーザ ID やホームディレクトリ，シェルや GECOS などが取得できる．このように，ある情報をキーとして、それに付随する情報を検索できるようなサービスをディレクトリサービスと呼ぶ．ディレクトリサービスは，コンピュータシステムにおいては，ファイルシステムやドメイン情報 (DNS)，日常生活では，電話帳や住所録などに活用されている．ディレクトリはデータベースに似ており，さらに記述的で属性ベースの情報を含んでいる．ディレクトリ中の情報は書き込みより読み出しの方が多い．そのため，ディレクトリサービスでは，複雑な更新を行うためのトランザクション処理やロールバック機構は実装されず，大量の照会や検索操作に最適化されている．

ディレクトリサービスを汎用的に扱えるようにするプロトコルが LDAP(Lightweight Directory Access Protocole) である．LDAP は，TCP/IP に限定し，ディレクトリサービスで必要となる操作が可能となるプロトコルが定義されている．LDAP を使用すれば，様々なアプリケーションから共通のインターフェースを使ってネットワーク上のユーザを特定したり，認証できる．LDAP はプロトコル自体にアクセス制御の機構を持つため，ディレクトリに対して，だれが何にアクセスしてよいかを制御できる．ディレクトリに格納する情報は，UNIX のファイルシステムのディレクトリや DNS のドメイン名のように，階層構造に配置される．このツリーは DIT (Directory Information Tree) と呼ばれ，図 1 のように表現する．ディレクトリ内の情報を一意に表現するには識別名 (Distingunish Name:DN) を使用する．DN は，ファイルシステムでのフルパスに相当し，図 1 であれば，ユーザ shigeo-n の DN は次のように表現する．

uid=shigeo-n,ou=People,dc=aist-nara,dc=ac,dc=jp

これらの情報を利用して，LDAP は必要な情報を参照したり認証したりするためのデータベースとして使うことができる．これまでもユーザアカウントや他の情報をネットワーク経由で参照するために，NIS などが利用されてきた．LDAP は，NIS と同じ機能を提供でき，かつ LDAP の方が優れている点がいくつかある．

（1） LDAP サーバ上の情報は容易に複数の用途に利用できる．LDAP データベース上の同じエントリは，さまざまな他のアプリケーションに使用でき，データの重複や矛盾を避けることができる

（2） LDAP は複雑なアクセスコントロールリストをデータベースに適用できる

（3） SSL(Secure Socket Layer) を使用して，LDAP サーバとクライアント間にセキュアな転送経路を確保できる

（4） LDAP ではサーバ情報の複製 (レプリカ) を作成し，負荷分散を図ることができる

（5） LDAP は，NIS に比べ Windows や他のソフトウェアとの連携をすることが容易にでき，Samba [8] や各 MTA でも LDAP 対応が進んでいる

以上の優位性を確認するため，次章で述べる実験環境を構築した．本研究では (2)，(3) の項目について検証を行った．

3. LDAP 実験環境の構築

3.1 測定環境

定量的なデータを測定するために，図 2 に示す実験環境を表 1 に示した機材を用いて構築した．アカウント情報などは本学の情報をそのまま利用した．

表 1 使用した機材

	サーバ	クライアント
CPU	PentiumIII 1GHz	
Memory	512MB	
NIC	Intel EtherExpress Pro100	
OS	Debian GNU/Linux 3.0r0	
OpenLDAP	2.0.23	
DB	Berkeley DB 3.2.9	—
LDAP module	—	pam_ldap 140

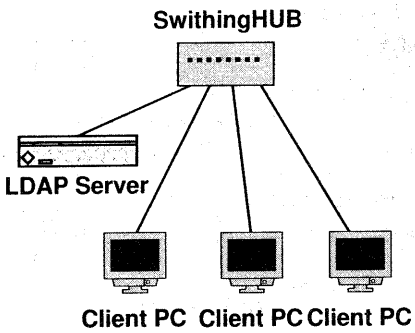

図 2 実験環境

3.2 LDAP Server

LDAP サーバソフトウェアとしては，オープンソース実装のひとつとして The Open LDAP Project が開発している OpenLDAP [7] を使用した．OpenLDAP には，LDAP サーバ，LDAP レプリケーションサーバ，LDAP プロトコルライブラリ群，LDAP クライアントユーティリティが含まれている．実装としては，LDAPv2 プロトコルを採用した 1.2.x 系列と，LDAPv3 プロトコルを採用した実装した 2.x 系が存在する．LDAPv3 は LDAPv2 であり，以下の機能が追加されている．

- SASL を利用した強力な認証
- TLS(SSL) を利用した一貫性と機密性の保護
- Unicode 利用による国際化対応
- 紹介 (referral) と継続 (continuation)
- 拡張性 (コントロールと拡張操作)
- スキーマ開示

現在では，LDAPv2 プロトコルを使用するメリットはないため，LDAPv3 プロトコルが実装された 2.x 系を利用した．OpenLDAP ではデータベースライブラリとして，LDBM(LDAP DataBase Manager) 用のデータベースバックエンドが必要となる．利用可能なデータベースバックエンドは，Sleepycat Software [9] の Berkeley DB や，GNU Database Manager [10] などが存在する．今回は NIS との性能比較を行うため，NIS でも利用されている Berkeley DB を利用した．

3.3 LDAP client

各クライアントでのユーザ管理機構を LDAP に切り替えるためには，認証したいユーザの情報を NSS(Name Service Switch) という枠組みをつかってどこから参照するか指定する必要がある．Linux や Solaris では，図 3 のようなイメージでどこからユーザ情報を参照するか制御することができる．NSS でユーザを検索する場合には nss_ldap [11] を導入する必要がある．

NSS によって検索されたユーザは，PAM(Pluggable authentication Modules) [12] と呼ばれるモジュールをシステムに適切に設定することで LDAP によるユーザ認証が有効になる．

4. 実　　験

本章では LDAP 実験環境を用いた，検証実験について述べる．

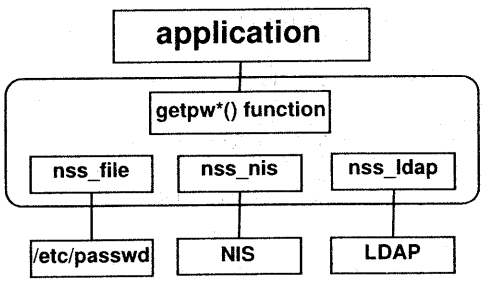

図 3 Name Service Switch

```
iwc% ldapsearch -x -D'uid=shigeo-n, \
ou=People,dc=aist-nara,dc=ac,dc=jp' \
-W -b 'dc=aist-nara,dc=ac,dc=jp' uid=shigeo-n
Enter LDAP Password:
<snip>
objectClass: posixAccount
objectClass: top
userPassword:: a3NygZBofUkvM2NYAXhiDL1IOFU=
loginShell: /bin/csh
<snip>
```

図 4 ユーザ自身による検索

```
iwc% ldapsearch -x -h ldb.aist-nara.ac.jp \
 -b 'dc=aist-nara,dc=ac,dc=jp' uid=shigeo-n
<snip>
objectClass: posixAccount
objectClass: top
loginShell: /bin/csh
<snip>
```

図 5 匿名ユーザで検索

4.1 アクセスコントロール

NIS では，暗号化されているとはいえ，他人のパスワードが取得可能であった．LDAP では，TCP Wraper とは別に，個別のアクセス制限が可能である．ただし UNIX のファイルパーミッションとは異なり，書き込みは可能で，読み込みは不可という設定はできない．ディレクトリに対するアクセス権を適切に設定することによって 図 4 の実行例や図 5 の実行例のように，ユーザのパスワードを，そのユーザだけが読み書きでき，ほかの人は何もできないようにすることでセキュリティを高められる．

4.2 LDAP を利用したユーザ認証

LDAP の基本的な性能を評価するために，tcpdump により，クライアントサーバ間の TCP の状態遷移から処理時間を計測する方法と，LDAP クライアントソフトウェアに処理の開始時間と，終了時間を計測するよう変更を加える手法を考察した．

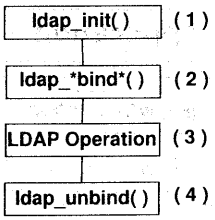

図 6 LDAP によるオペレーションの概要

表 2 検索時間

データエントリ数	LDAP	NIS
初期状態	6.425[ms]	—
1000 件	218.232[ms]	—
2000 件	468.555[ms]	0.8903[ms]
2000 件 (SSL)	498.173[ms]	—

LDAP オペレーションでは，OS に組み込まれている Name Service Cache が有効になるため，TCP の状態遷移では，正確な処理時間が計測できない．したがって，LDAP クライアントソフトウェアに修正を加える方法を用いた．LDAP は図 6 に示したように，初期化，認証 (bind)，切断 (unbind) というステップを必要とする．(2) では同期認証を行う ldap_simple_bind_s() 関数や，非同期認証の ldap_simple_bind() 関数など数十個の関数が存在する．検索，修正，追加などの処理は (3) で行われる．実験では図 6 中の (1) を開始してから (4) の処理が終わるまでの時間を計測した．

LDAP サーバのハードウェア的性能に左右されないように，LDAP サーバ構築時の初期状態における性能を計測した．ここでいう初期状態とは LDAP ディレクトリに対して基本情報のみを登録した状態を指す．測定結果を表 2 に示す．

次に，実際に本学の全アカウント情報 (約 2000 件) を全て LDAP に登録し計測した．検索はランダムに抽出した uid 20 個を 3 回検索し，その平均時間を測定した．参考までにデータ数を約 1000 件登録した場合と，NIS に置ける測定結果も表 2 に示す．NIS でもランダムに抽出した uid 20 個を 3 回検索し，ypmatch の処理の開始と終了の時間を計測した．NIS における性能はネットワークや環境の影響を受けないよう図 7 に示したように同じ状況下になるように同種のデータ，同種の機材を用いて測定した．

表 2 に示すように LDAP と NIS の処理時間の差は 500 倍程度となった．これまで，LDAP が普及してこなかった原因は，この性能差に起因すると考えられる．

次に，通信路を SSL によって暗号化した場合のオーバヘッドを測定した．LDAPv3 では，LDAPv2 までと異なり，TLS/SSL を利用したセキュアな通信を確保することができる．今回は，OpenSSL を使用し，簡易 CA(Certificate Authority) を使用して通信の暗号化を行った．表 2 に示したように，5%程度の速度低下が見られた．

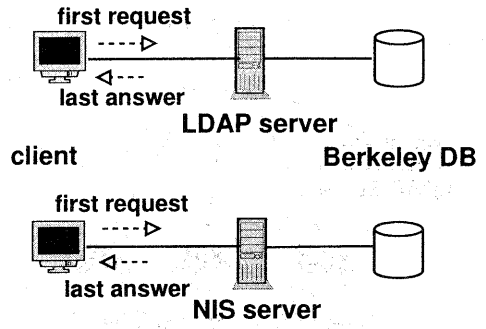

図 7 計測のイメージ

5. ま と め

本研究では，ディレクトリサービスを提供する OpenLDAP を使用して，NIS を使用せずにアカウントを一元管理するクライアントサーバシステムを構築した．ディレクトリサービスは，NIS や NIS+に変わる新たなユーザ認証の枠組みとして注目されている．特に NIS の置き換えとしての用途とした場合を想定し，既存の NIS 環境からの必要な情報を検証した．LDAP は NIS が提供していたサービスを提供でき，かつ NIS にはない柔軟なアクセス制御が可能である．しかし，NIS に比べ LDAP は，処理時間が数百倍と遅く，NIS に比べより高性能なハードウェアが必要となる．今回は LDAP サーバに対してチューニングを行わなかったため，今後は処理速度低下の要因を特定し，改善する必要があると考えられる．今後の課題としては，他のアプリケーションでの LDAP の活用である．たとえば，Samba では次期リリース版で LDAP との連携によって，Windows クライアントのアカウントを一元的に管理できる実装が進んでいる．

文 献

[1] Lightweight Directory Access Protocol (v3) RFC 2251
[2] “Microsoft Active Directory”, http://www.microsoft.com/japan/windows2000/techinfo/howitworks/activedirectory/adarch.asp
[3] “Novell eDirectory” http://www.novell.co.jp/products/edirectory/
[4] “Oracle Oracle Internet Directory” http://technet.oracle.com/products/oid/
[5] ティム・ハウズ+マーク・スミス 著，“LDAP インターネットディレクトリ アプリケーション プログラミング” ピアソンエデュケーション，
[6] 山口英，砂原秀樹 “ネットワークサービス集中管理型環境の試み” 電子情報処理学会研究報告 IN98-128
[7] “The OpenLDAP Project ” http://www.openldap.org/
[8] “SAMBA” http://www.samba.org/
[9] “Sleepycat Software ” http://www.sleepycat.com/download.html
[10] “GNU Database Manager ” http://www.gnu.org/software/gdbm/
[11] “NSS module for using LDAP as a naming service ” ftp://ftp.padl.com/pub/
[12] “Linux-PAM ” http://www.kernel.org/pub/linux/libs/pam/