Multimedia Flavor in Directory Services Environment

Emanuil REDNIC, Andrei TOMA
Academy of Economic Studies, Bucharest, Romania
emanuil.rednic@gmail.com, andrei.toma@ie.ase.ro

The present article aims to present the extended functionality of LDAP based systems (known as Directory Services) of handling multimedia information such as photographs. The most frequently used are JPEG encoded photos. Furthermore this article will explain how this multimedia functionality can be used to increase the security of identity management based systems.

Keywords: *database security, distributed activities, multimedia information, multimedia processing, watermarking.*

1 Introduction

The problematic issue of managing multimedia information in LDAP based systems was encountered in almost the same time frame as the issue of managing multimedia information in databases. This is because of the large amount of resources required by database features involving multimedia processing.

In the beginning, database systems applied the workaround (used at the storage level) of keeping only a logical reference to the multimedia resource which was actually stored separately on the disk. In the same vein, LDAP systems included a similar workaround by offering the possibility of creating an attribute in the schema which would hold a reference to the JPEG/photo resource. This could be an attribute after labeledURI or a subtype used by the objectClasses attribute found in the subschema subentry. An example of this is presented below.[3]

```
attributetype ( 1.1.2.1.3 NAME
'myPhotoURI'
        DESC 'URI and optional label
referring to a photo'
        SUP labeledURI)
```

Another example of how this resource is handled by LDAP is the one provided by Oracle Internet Directory Services. First the schema defines the class which contains the the multimedia feature [4]:

```
objectclasses: (
0.9.2342.19200300.100.4.3 NAME
'pilotObject' SUP top STRUCTURAL MAY (
jpegPhoto $ audio $ dITRedirect $
lastModifiedBy $ lastModifiedTime  $
uniqueIdentifier $ manager $ photo $
info ) )
```

where jpegPhoto is one of the attributes contained in the class. This is defined as follows:

```
attributetypes: (
0.9.2342.19200300.100.1.60 NAME
'jpegPhoto' SYNTAX
'1.3.6.1.4.1.1466.115.121.1.28' )
```

The main function of Directory Services is Identity Management. In order to achieve this in the best possible way, the system has to define specialized classes for each type of identity used in Directory Services. For example, let's assume that one of the used identities is “person”. In order to manage it, a set of classes is defined in the manner presented below.[6][7]

```
objectclasses: ( 2.16.840.1.113730.3.2.2
NAME 'inetOrgPerson' SUP organization
 alPerson STRUCTURAL MAY ( audio $
businessCategory $ carLicense $
department
 Number $ displayName $ employeeNumber $
employeeType $ givenName $ homePhone
  $ homePostalAddress $ initials $
jpegPhoto $ labeledURI $ mail $ manager
$
 mobile $ pager $ photo $
preferredLanguage $ roomNumber $
secretary $ uid $
 userCertificate $ x500UniqueIdentifier
$ userSMIMECertificate $ userPKCS12 $
  o ) )
```

or

```
objectclasses: (
2.16.840.1.113894.1.2.58 NAME
'orclSubscriber' SUP top
AUXILIARY MAY ( orclSubscriberFullName $
orclSubscriberType $ orclContact $
orclHostedDunsNumber $
orclHostedPaymentTerm $
orclHostedCreditCardType $
orclHostedCreditCardNumber $
orclHostedCreditCardExpireDate $ c $
jpegPhoto $ orclversion ) )
```

or

```
objectclasses: (
0.9.2342.19200300.100.4.3 NAME
'pilotObject' SUP top
STRUCTURAL MAY ( jpegPhoto $ audio $
dITRedirect $ lastModifiedBy $
lastModifiedTime
  $  uniqueIdentifier $ manager $ photo
$ info ) )
```

Oracle Internet Directory also has some derivations in the schema for multimedia features, such as: photo, thumbnailphoto[5]

```
attributetypes: (
0.9.2342.19200300.100.1.7 NAME 'photo'
SYNTAX
'1.3.6.1.4.1.1466.115.121.1.23{250000}'
)
```

and

```
attributetypes: (
1.3.6.1.4.1.1466.101.120.35 NAME
'thumbnailPhoto' SYNTAX '1.
3.6.1.4.1.1466.115.121.1.28' )
```

In each Directory Service, attributes and object classes defined have to be unique, requirement which is achieved by the existence of a unique number each has in the schema definition, such as *1.3.6.1.4.1.1466.101.120.35*. There is no specific annotation for this unique key, nor any specific rules of generation; the only condition is that it is used for a single attribute or objectclass.
Another well-known Directory Service, which use multimedia flavor, is Microsoft Active Directory; in Active Directory the definition of this attribute is slightly different as shown in figure 1 [8]:

CN	jpegPhoto
Ldap-Display-Name	jpegPhoto
Size	-
Update Privilege	-
Update Frequency	-
Attribute-Id	0.9.2342.19200300.100.1.60
System-Id-Guid	bac80572-09c4-4fa9-9ae6-7628d7adbe0e

Fig. 1. Attribute jpegphoto in MS AD

This attribute is used to store one or more images of a person using the JPEG File Interchange Format [JFIF]. This LDAP is similarly focusing on “person” identity in order to implement the multimedia flavor; consequently, the objectclass used in this case is “user”, “person”.
The derivation of objectclasses, and multimedia-based attributes varies from Directory Services to Directory Services, but all share the main goal, which is to offer the multimedia usage functionality in identity management at the process level.
In fact the majority of LDAP's used on the market are using multimedia features out of the necessity, in identity management processes, to handle photos, badges or any graphical code or rule for the employees. The aforementioned operations are necessary in order to provide flexibility, portability and

respond to all the enterprise based applications requests.

2 Using multimedia flavor

The main problem of multimedia flavor starts from the point of importing/exporting this feature into LDAP. This raises a series of questions

- Is it possible to use a LDIF (LDAP Data Interchange Format)?
- How to use a LDIF file?
- How to create a LDIF file for this action?

Before starting to describe these actions, the main factor of these is the LDIF file. We will thus first attempt a short description of the LDIF file structure.

The LDIF format is used to convey directory information or a description of a set of changes made to directory entries. An LDIF file consists of a series of records separated by line separators.

A record consists of a sequence of lines describing a directory entry or a sequence of lines describing a set of changes to a directory entry.

A LDIF file specifies a set of directory entries, or a set of changes to be applied to directory entries, but not both. There is a one-to-one correlation between LDAP operations that modify the directory (add, delete, modify and modrdn) and the types of changerecords described below ("add", "delete", "modify", and "modrdn" or "moddn"). This correspondence is intentional, and permits a straightforward translation from LDIF changerecords to protocol operations. The rules defined for creating a proper LDIF are described in the *RFC 2849 LDAP Data Interchange Format*, [9] from this point the standard format for LDIF file have to be respected and used by all LDAP Directory Services.

```
*******FILE_STARTS_HERE**********
dn: cn=Rednic
,cn=Users,dc=phd,dc=ro,dc=com
changetype: add
add: jpegPhoto
jpegPhoto:: /path/to/photo
********FILE_ENDS_HERE************
```

Fig. 2. The sample of multimedia.ldif file

The file can be created using a simple text editor and it is not mandatory that it has the *ldif* extension. The only requirements are that it respects the RFC 2849 specification.

The LDAP command used for adding the multimedia resource to the Directory Services is:

```
> ldapmodify -D cn=orcladmin -w
"welcome1" -p 389 -f multumedia.ldif
```

After this command, the multimedia feature is added to the LDAP Directory Services. It then be modified, in which case the only thing required is to change the operation from LDIF file from add to modify as seen in figure 3 below.

```
*******FILE_STARTS_HERE**********
dn: cn=Rednic
,cn=Users,dc=phd,dc=ro,dc=com
changetype: modify
add: jpegPhoto
jpegPhoto:: /path/to/photo_new
********FILE_ENDS_HERE************
```

Fig. 3. New multimedia LDIF file for altering this resource

The command remains the same in this case, just in case separate files must be created in order to keep a file for each kind of operation.

The main problem, which appears at this point, is the export action. How do we export binary information in the character based file? Until the introduction of the multimedia feature the LDAP command for exporting an entry would be:

```
ldapsearch -h host -D
"Administrator@phd.ro.com" -w welcome1 -
s sub -b
"cn=candidate1,cn=users,dc=phd,dc=ro,dc=
com" objectclass=*  >userdump.ldif
```

from AD or from OID:

```
ldapsearch -h host -p port -D
"cn=orcladmin" -w welcome1 -s base -b "
cn=candidate1,cn=users,dc=phd,dc=ro,dc=c
om " objectclass=* userdump.ldif
```

The only difference is the super user definition, which varies from LDAP to

LDAP.
In the case that the above entry does not have any multimedia flavor this operation will be fine. If, however, we export an entry with a photo resource by the above mentioned commands, the export will produce the same results for character fields, but for jpegphoto it will looks like:

```
jpegphoto::
/9j/4AAQSkZJRgABAgAAZABkAAD/7AARRHVja3kA
AQAEAAAAXwAA/+4ADkFkb2JlAGTAAAAAAf/bAIQA
AQEBAQEBAQEBAQEBAQEBAQEBAgEBAQEBAgIC
AgICAgICAgICAwICAgMDAwMDAwUFBQUFBQUFBQUF
BQUFBQEBAQECAQIDAgIDBAQDBAQFBQUFBQUFBQUF
BQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUFBQUF
BQUFBQUFBQUF/8AAEQgAyACgAwERAAIRAQMRAf/E
AKsAAAAHAQEBAAAAAAAAAAAAAAIDBAUGBwgBCQAB
AAEFAQEBAAAAAAAAAAAAAAMBAgQFBgcACBAAAgEC
BAUDAgMHAgUFAQAAAQIDEQQAIRIFMUEiEwZRYQdx
MoGRFKGxwUIjFQjRUuFicjMk8PGSQxY0EQABBAAD
BAkDA
```

The command takes any input and formats it with the correct line continuation and appropriate attribute information. The command also assesses whether the output requires base–64 encoding. The above examples take an entry containing a JPEG image and convert it into LDIF format for the attribute named jpegPhoto. The output can be saved as userphotoout.ldif [10].
At this point the question is how to visualize the multimedia output. If we are not using it, for example in identity management applications, it is useless to overload the database with unused information even if it is only referenced in the application. This information will be reported using a web application.
Another series of questions appear here. How will this influence the load of the web pages (considering all the image traffic involved in a n-tier enterprise application). Who will be affected the most? The server? The client? The database? Only one of these? Canthe browser influence the issue? Or the API used by application? Or the objectclasses? How can we find optimal rule(s) of the multimedia resources used in order to make a lightweight way of managing these resources?
One of the most used web based applications, which can report the multimedia information is the DAS (Delegated Administration Services) application. This application can be installed with the Identity Management Infrastructure package from Oracle Application Server kit. The application can be easily accessed as: http://infrahost:infraport/oiddas and gives the administrator the functionality of managing photos easily from OID, from user creation phase till the deletion one. The web control used when reporting this field appears in the user management page, no matter if the resource was loaded for the user or not.[11]

3 Usability of the multimedia feature in LDAP

The limitation of the multimedia feature is visible during the retrieval and search operation. Unfortunately, this attribute cannot be used as a filter for the search command, in the manner text based attributes can be used, for example [12]:

```
ldapsearch -h host -p port -D
"cn=orcladmin" -w welcome1 -s base -b "
cn=candidate1,cn=users,dc=phd,dc=ro,dc=c
om" “objectclass=*” dn
```

The following command cannot be used, due to binary base of the attribute:

```
ldapsearch -h host -p port -D
"cn=orcladmin" -w welcome1 -s base -b "
cn=candidate1,cn=users,dc=phd,dc=ro,dc=c
om" “objectclass=*” jpegphoto
```

So the question is, use is an attribute, which can’t be used in search actions? How can we alter this attribute, manipulate, update or change its physical characteristics such as resolution, RGB or color depth.
In order to achieve the above, the only approach is to make the search by user information such as: userid, dn, cn. We then save the multimedia information separately in a buffer variable. The buffer can then be programmatically changed according to our requirements.
From this phase onward, Oracle gives the opportunity to work with multimedia using Oracle Intermedia. Even here there is a limitation in that a picture can’t be searched using a pattern from it, a subdivision, or some pixels, directly (figure 4).

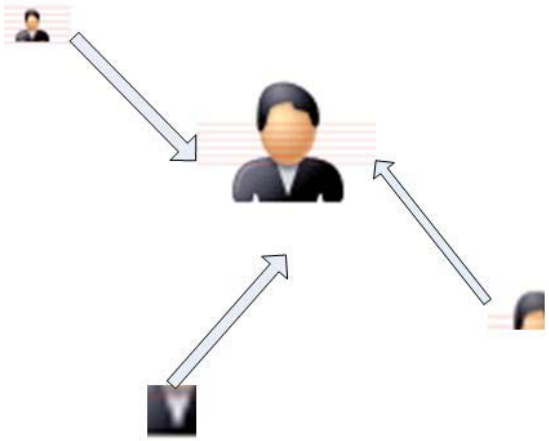

Fig. 4. Search using patterns

Another limitation in multimedia management using Directory Services is the compare operation.

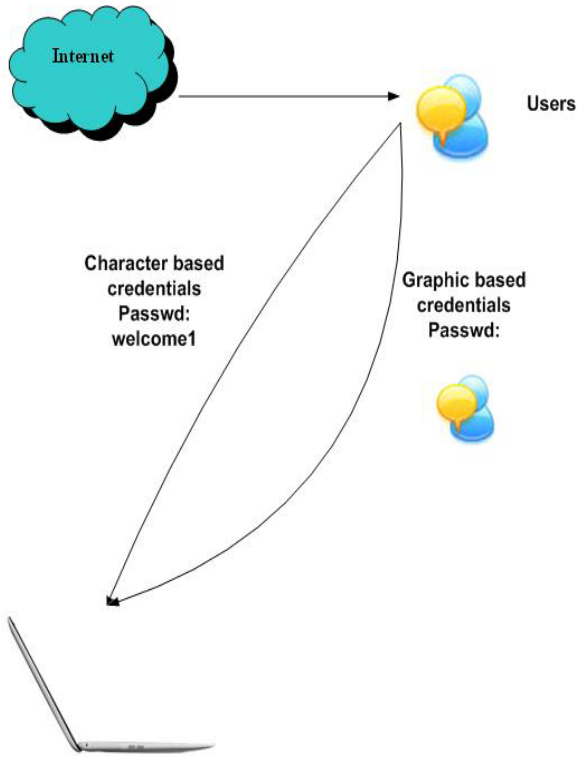

Fig. 5. Future plans for graphical authentication

This operation is most frequently used at the login phase, where the password in encrypted in the LDAP. The compare is done after user fills in the password (figure 5). LDAP command *ldapcompare* can do the compare between the plain text filled by end-user and the encrypted one saved in the database. The error codes used for the ldapcompare results respect ***RFC 4511 Lightweight Directory Access Protocol (LDAP): The Protocol[13]*** and can be either:
compareFalse (5) or compareTrue (6).
In fact accordingly to RFC 4511, these are called "non-error" results codes, because these don't indicate an error condition.

4 Portability of multimedia feature

More and more Directory Services include in their identity management package the multimedia feature, which makes it easier to import/export stored information containing multimedia between LDAP's. Below we present a series of considerations on how this can be achieved.
One of the most used features is *synchronization*. This process is used in LDAP integration with 3rd parties' products. Just as the character based information can be synchronized by the usage of the profile's mapping file, so can the multimedia information, which gains its correspondent from the source LDAP to the target one. The problem of the difference in attribute names between the source and the target is solved, making the binary information transferable easily and in real time.
Another method to achieve similar results is to use *replication*. If the synchronization works on the LDAP level, replication goes a level down to the database. Now the source and destination are the databases on which underlie the two LDAP's. All the transactions are made on this level, using database jobs. Replications can be done using the same databases. Oracle has implemented this method with a lot of success, and this can be done not only on databases, synchronously, but also on LDAP level, asynchronously.
Oracle does not support replication to non-Oracle databases like DB2, Informix, Sybase, SQL Server, etc. Other vendors provide products that can replicate heterogeneous databases. One such company is Sybase.
The simplest method is using the LDIF file, but the level of information risk is quite high. Dummy characters can appear in the file,

such as BLANK, CR, LF etc. which can corrupt the information. As the multimedia feature is exported in base-64 and it is represented as a long char string, as mentioned in a previous section, any extra character added to it corrupts irrecoverably the multimedia information. As blank, CR, LF etc. can be traceable in the LDIF, any extra characters in multimedia attribute make the search more complex. Only an initial backup LDIF file can eliminate this problem, through comparison of the exported data, for example the backup took at second 1 after entry was exported from the LDAP.

5 Multimedia LDAP metrics

In order to create metrics systems for Directory Services Identity Management it is mandatory to establish the entities involved in it. Starting from the matrix presented in article *Security Management in a Multimedia System*: the main entity is the user, followed by the graphics resources.

Also, a main factor involved for creating the graphics metrics are the letters of Latin alphabet, as it is described in (1):

$$GPIM = (TU^* FL(U,L) + TUX)^*NC \quad (1)$$

where:

- *GPIM* – graphical process index for identity management
- *L* – total letters of Latin alphabet
- *TU* - total number of users involved in the system
- *TUX* – total number of users integrated from other directory systems
- *NC* – number of colors used by multimedia resource
- *FL* – is a function defined on the users set, and the total letters of the alphabet, and it shows how many letters are necessary to write a user identifier

This can be extended with a multiple factor, used to define the number of multimedia attribute, an user entry has, as presented in (1')

$$GPIM = (TU^* FL(U,L) + TUX)^*NC^*M \quad (1')$$

M - multiple factor, this M has to be constant for all LDAP entries of one Directory Services.

Of course that some entries can have loaded the multimedia information in all M attributes, but others do not, making the multimedia-loading factor variable. In this case M(m) , where m is the number of multimedia attributes which do not have null information

In the Directory Services management, TU, presented in formula (2) depends as well by TO-total operations made in the LDAP system, as it is described in article [1] and [2].

$$TU(TO) = TO + \Phi^*TO \quad (2)$$

Φ is the index that can measure the instability of the LDAP system, due to big amount of managing users, and takes values in the interval [0,1], where:

$$TO = TCN + TCN + TF + TB + TU + TS + TCC + TM + TMD + TA + TD \quad (3)$$

$$TBV = TB + TU \quad (4)$$

$$TR = TCN + TF \quad (5)$$

$$TMO = TS + TCC + TM + TMD + TA + TD \quad (6)$$

$$TO = TBV + TR + TMO \quad (7)$$

and

- TO – Total operations
- TCN – Total connections
- TF – Total authentication failures
- TB – Total binds
- TU – Total unbinds
- TS – Total searches
- TCC – Total compares
- TM – Total modifications
- TMD – Total modrdns
- TA – Total additions
- TD – Total deletions

6 Conclusions

In an enterprise organization the necessity of managing multimedia information in the

LDAP Directory Services, not only the textual information appears more and more frequently. Consequently, the necessity of using multimedia databases in identity management has thus increased as well. Solutions for the using multimedia information are taken into consideration for more and more Directory Services like: Microsoft Active Directory, OpenLDAP.

Why use multimedia databases? First of all because of the scalability advantages stemming from storing large amounts of multimedia data, secondly because of the flexibility of accessing the information, short time for either DDL or DML operations, easily achieved import/export of the information, either by using the facility of the relational databases or using databases files.

Multimedia databases can also be used in various fields such as medical, cadastral, shipping, mailing, geographical, geodesic, transportation activities and the list goes on; a common point for these activities is that these involve processing, in a significant percentage, multimedia information and to a lesser extent alphanumerical one like in the case of financial/banking activities.

Besides the advantages, the multimedia databases also have some limitations, such as those related to searching the information using a multimedia filter (a filter is an amount of pixels, either random or consecutive, from an image or a specific division or subdivision of the image). For the moment the only way to search for specific multimedia information is by interrogating the metadata information of the multimedia information.

This feature of interrogating the databases records by a multimedia filter would increase the security of multimedia information, for example through the possibility of a user to create a password based on specific pixels from an image; should the picture be altered, the password would be altered as well; in order to change the password the RGB information of a pixel would be changed. This topic is however outside the bounds of the present article and will be discussed in a future paper.

Introducing and increasing the usage of multimedia information in the Directory Services, specific for identity management increases the level of granularity for user management. Now enterprise application can use in their identity management: photos of employees, badge scanned images. With this feature the security level increases. Also a higher level of security than can be achieved is to use multimedia resource at authentication/authorization level? Is it possible to use a multimedia password? Add a multimedia picture for the password, beside the classic text password. This will be discussed in a future article, which will include all aspects of this extension for login operation, how to integrate this on enterprise web based applications, all in distributed environments.

References

[1] E. Rednic and A. Toma, *Security Management in a Multimedia System*, www.jaqm.ro, June 2009.

[2] M. Velicanu and E. Rednic, *Identity Management in University System*, Revista Informatica Economica, nr. 2/2008, pp. 71-74

[3] S. Saha, *Oracle Application Server 10g Administration I, II*, ORACLE, U.S.A., 2004.

[4] S. Saha, *Oracle Application Server 10gr2 Administration I, II*, ORACLE, U.S.A., 2004.

[5] L. Dhananjavan, *Oracle Identity And Management – all in one*, ORACLE, U.S.A., 2007.

[6] S. Mavria, Oracle *inter*Media Feature Overview, ORACLE, S.U.A., 2005.

[7] J. Mauro, Oracle *inter*Media Managing Multimedia Content, ORACLE, U.S.A., 2008.

[8] http://msdn.microsoft.com/en-us/library/ms676813(VS.85).aspx

[9] http://www.ietf.org/rfc/rfc2849.txt

[10] http://docs.sun.com/app/docs/doc/820-2493/6ne3feem5?l=Ja&a=view

[11] http://download.oracle.com/docs/cd/B28196_01/idmanage.1014/b15998/ldif_appendix.htm#CHDIFIEG

[12] http://download-uk.oracle.com/docs/cd/ B28196_01/ idmanage.htm

[13] http://www.faqs.org/rfcs/rfc4511.html

Emanuil REDNIC has a background in computer science and is interested in database related issues. He has graduated the Faculty of Cybernetics of the Academy of Economic Studies in Bucharest. He is currently conducting doctoral research at the Academy of Economic Studies. Other fields of interest include multimedia, directory services, and identity management.

Andrei TOMA has a background in both computer science and law and is interested in an interdisciplinary approach to IT Law related issues. He has graduated the Faculty of Cybernetics of the Academy of Economic Studies in Bucharest and the Faculty of Law of the University of Bucharest. He is currently conducting doctoral research at the Academy of Economic Studies. His fields of interest include IT Law related issues, as well as various artificial intelligence topics.