

Webmail Authentication Using Shibboleth and Virtual Directory Server

Stefano Zanmarchi
Carlo Manfredi
Simone Marzola
Giorgio Paolucci

University of Padova - ITALY

TERENA Eurocamp
Athens – November 6, 2008

Introduction

Our infrastructure:

- POPS server for MUAs (Thunderbird, Outlook, ...)
- IMAP server only used by shibbolized webmail frontend
- LDAP server stores usernames and hashed passwords

Goal: Same username/pwd for POPS and webmail access

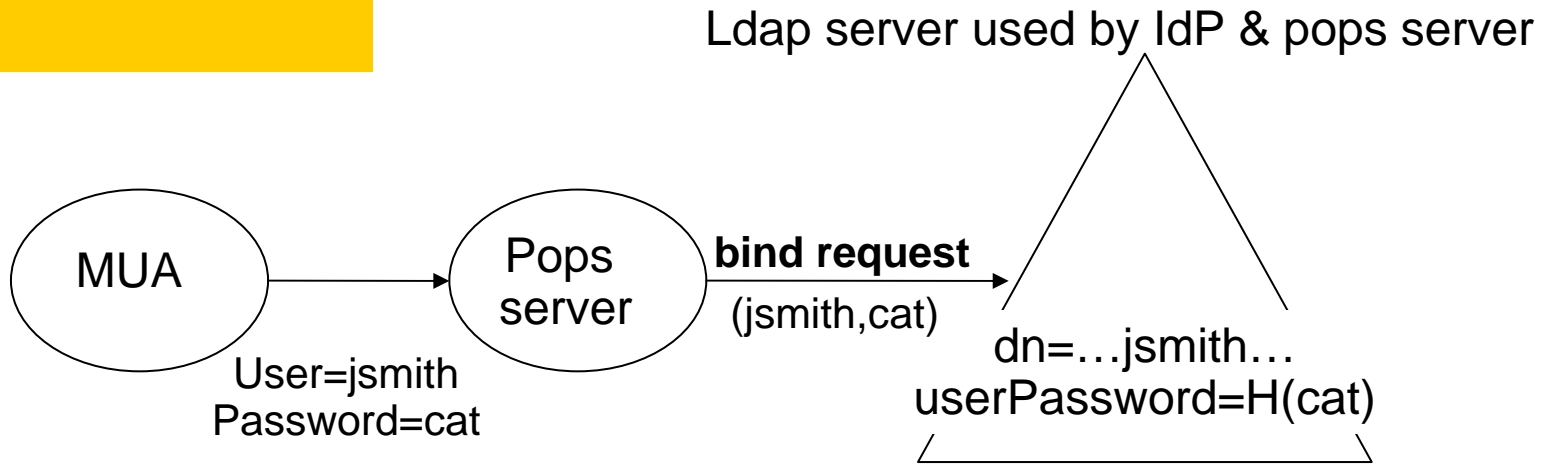
Requirements:

1. pwd stolen by possible webmail hacker unusable for SSO login
2. no anonymous access to the imap server by webmail frontend

Solution: the IdP sends the hash of the password as an attribute to the shibbolized webmail frontend (Horde).

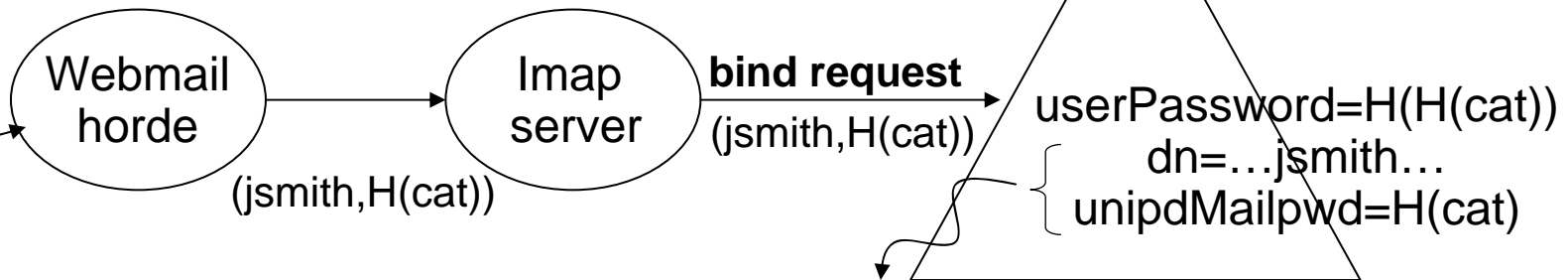
Yes, but how can the imap server bind using a hash?

Using a second LDAP server:



Attributes received by the AA:

User=jsmith
unipdMailpwd=H(cat)



sent as attributes
by the IdP

2nd LDAP needed because:

- LDAP serves bind requests only matching against *userPassword*, can't use another attribute
- Multiple *userPassword* attributes on first ldap server ($H(cat)$ and $H(H(cat))$) is no solution: works but doesn't meet requirement 1

Why *unipdMailpwd*?

On the second Idap server we introduced the *unipdMailpwd* attribute, copy of *userPassword* on first Idap server, to be sent as an attribute. Why?

Because Java apps (like the IdP) treat *userPassword* as an Octect String: the print method returns contents of the memory address: *userPassword* can't be directly sent as an attribute

2nd LDAP

Could be:

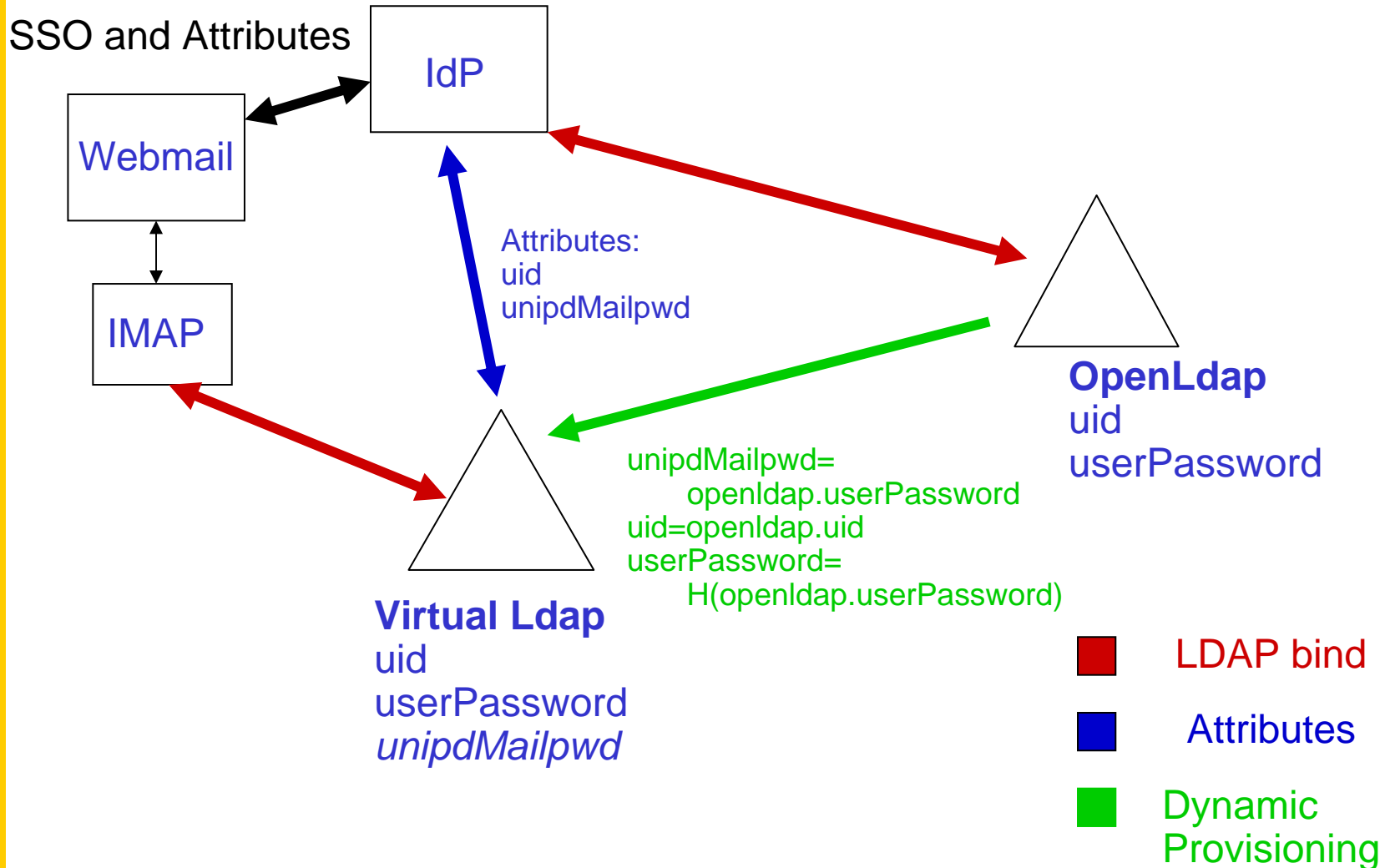
- “Real” ldap server: needs provisioning
- OpenLdap 2.4 overlay on first ldap server: OpenLdap and C coding needed
- Virtual ldap server (Penrose) in front of 1st ldap: our choice

Virtual Ldap

We use two Ldap servers:

- OpenLdap for IdP authentication
- Virtual Ldap for Imap authentication and Attribute Authority for Webmail. It uses OpenLdap as backend.

Virtual Ldap



OpenLdap

Only *uid* and *userPassword* attributes

Example:

```
ldapsearch -b "ou=people,dc=unipd,dc=it" -h db-openldap -p  
12312 -s sub uid=carlo.manfredi@unipd.it
```

dn: uid=carlo.manfredi@unipd.it,ou=people,dc=unipd,dc=it

objectClass: account

objectClass: simpleSecurityObject

uid: carlo.manfredi@unipd.it

userPassword: {SSHA}fEGNDYe8aLXDt+TJgTVJjGbYOaVNfZtF

Virtual Ldap

1. The attributes *uid* and *unipdMailpwd* of the virtual ldap are sent to the Webmail
2. Webmail passes the attributes to the IMAP server
3. Imap binds against the same virtual ldap

Example:

```
ldapsearch -x -h as1 -p 10385 -b "ou=people,dc=unipd,dc=it" -s  
sub uid=carlo.manfredi@unipd.it
```

```
dn: uid=carlo.manfredi@unipd.it,ou=people,dc=unipd,dc=it  
objectClass: account  
objectClass: top  
objectClass: unipdxmaileruid: carlo.manfredi@unipd.it  
userPassword:: {SHA}TRR2+vUYUYZ2E8N8qtelCfz4Bel=  
unipdMailpwd: {SSHA}fEGNDYe8aLXDt+TJgTVJjGbYOaVNfZtF
```

Penrose

We use **Penrose** as a java-based virtual directory server.

- Aggregates data from multiple heterogeneous sources:
 - Directories
 - Databases
 - flat files
 - web services
- Makes data available to identity consumers via LDAP
- Dynamic conversion and manipulation
- GUI client

<http://docs.safehaus.org/display/PENROSE/Home>

Penrose Server

From the website:

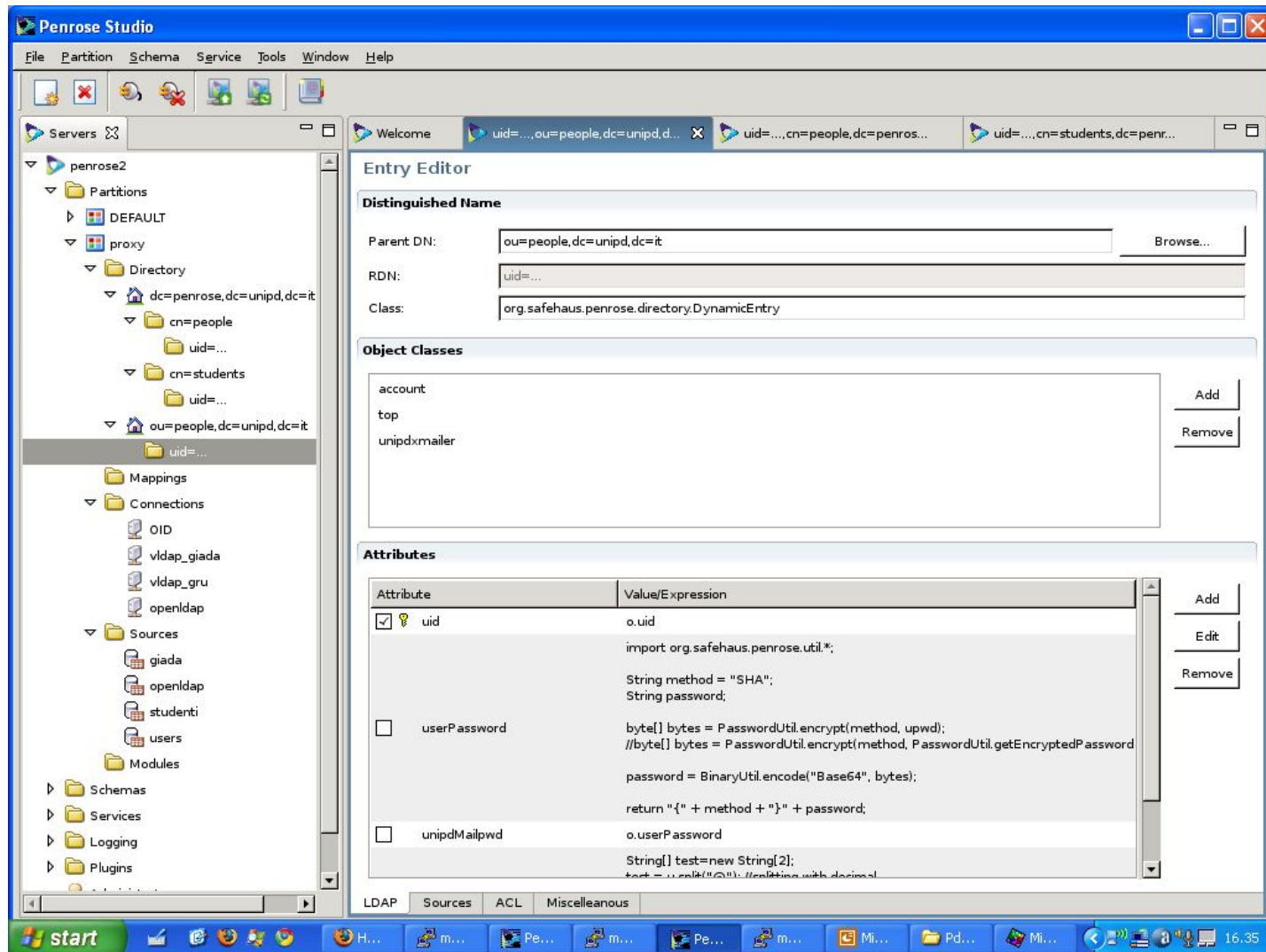
- Available on any platform where Java is supported
- Conversion and manipulation of Attribute values
- Namespace handling and Intelligent LDAP Queries routing
- Join and Cache engine
- Allow both in-memory and persistent cache
- Bi-directional synchronization via (Polling Connector and LDAP Sync) architecture
- Fine Grain Access Control Information
- Denial of Service protection
- Data Source Adapters for JDBC, JNDI, Active Directory, Web Services, etc.
- Configurable Fail-Over and Load-Balancing at the LDAP operation level
- Remote management via JMX.
- Extensible Plugin Architecture
- Run embedded in your application
- Run stand-alone or alongside with OpenLDAP, OpenDS or Fedora DS.

Penrose Studio

From the website:

- Enhanced Graphical Mapping Editor
- Access Control List (ACL) Editor
- Directory proxy and snapshot wizards
- Built-in Directory browser
- Off-line editing with one-click deployment
- Point and Click data source discovery wizards
- Live preview of your virtual directory
- Automated mapping validation and error checking

Penrose Studio screenshot



Attribute manipulation

In order to allow the Imap server to bind against the hash of the clear text password (unipdMailpwd, passed to the Webmail server as an attribute), we need to dynamically convert the userPassword attribute with a valid encryption method (ex. SHA). We use the PasswordUtil java library provided by Penrose

The userPassword attribute can then be defined with the following java code within penrose, where unipdMailpwd is the original hash of the clear text password:

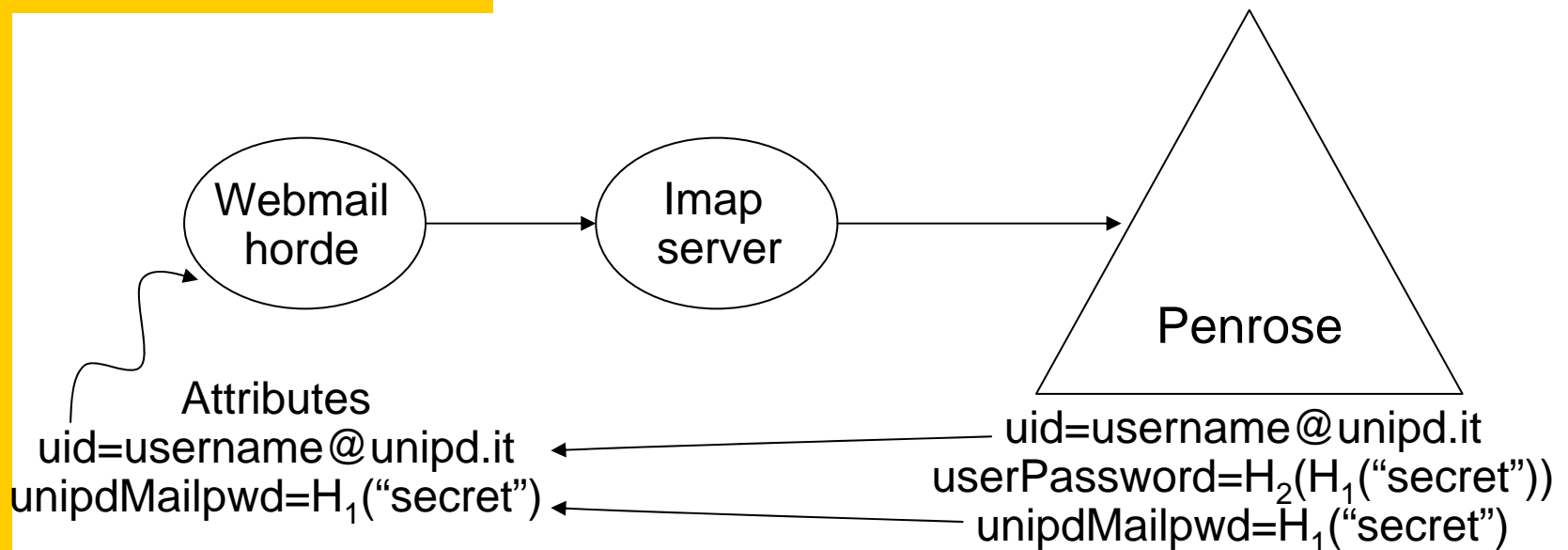
```
import org.safehaus.penrose.util.*;
String method = "SHA";
String password;
byte[] bytes = PasswordUtil.encrypt(method, unipdMailpwd);
password = BinaryUtil.encode("Base64", bytes);
return "{" + method + "}" + password;
```

Example:

Input: unipdMailpwd = {SSHA}fEGNDYe8aLXDt+TJgTVJjGbYOaVNfZtF

Output: userPassword = {SHA}TRR2+vUYUYZ2E8N8qtelCfz4Bel=

Conclusion



Webmail receives the attributes uid and unipdMailpwd from the IdP, passes them to the Imap server. Then the Imap server binds against uid and userPassword of the virtual Idap (penrose). Only the Imap server can bind against this Idap

H₁() and H₂() are valid Idap encryption method (MD5, SHA, SSHA, etc.)

Improvement

Caveat:

The webmail server receives the hash of the clear text password. A hacker who gets access to the webmail server can try a “brute force attack” to find the SSO password.

IMPROVEMENT:

Make unipdMailpwd temporary. It has to last only for the current webmail session or for a limited time (a day or a week). We can add a new hash function $H_3()$, time based:

Example, penrose might serve:

userPassword= $H_2(H_3(H_1(\text{“secret”})))$

unipdMailpwd= $H_3(H_1(\text{“secret”}))$

Where

$H_3()$: md5($H_1(\text{“secret”})$ +date)

Questions

Contacts:

stefano.zanmarchi@unipd.it

carlo.manfredi@unipd.it

simone.marzola@unipd.it

giorgio.paolucci@unipd.it