Enhanced Authentication in Open ID Against Phishing Attacks

CH.V.S .Suryanarayana, M.V.S. N. Maheshwar, K.Prudvi Raju

Abstract: Multiple factors for authorization and authentication are essential for security of any software. To design and implement an Educational Academy Automation Software using OpenID and Role Based Authentication (RBA) System as dual layer of secure authentication techniques to ensure that only authentic users can access the predefined roles as per their Authorization level.

But the OpenID authentication suffers from phishing attacks. How the OpenId is affected by Phishing attack and technique to block phishing attack in OpenID authentication procedure are addressed.

Keywords: OpenID, RBA, Phishing.

I. INTRODUCATION:

In World Wide Web, we need to get authorized by website to get services from that website. So creating an account of a website is required to get its services. The user need to create an account for each and every service and have to remember all usernames and passwords for all websites is difficult and this authorization involves centralized authentication technique which is more vulnerable to attacks in World Wide Web.

The alternative to problems mentioned is OpenId authentication. **OpenID** is an open standard that describes how users can be authenticated in a decentralized manner. OpenID allows you to use an existing account to sign in to multiple websites, without needing to create new account. You may choose to associate information with your OpenID that can be shared with the websites you visit, such as a email address or name. With OpenID, you can manage how much of that information is shared with the websites you are visiting. With OpenID, your password is only given to your opnid provider, and that openid provider then confirms your identity to the websites you are visiting. Other than your OpenId provider, no website cannot get your password, so you don't need to worry about malicious website compromising your identity.

OpenID is rapidly gaining popularity on the web, with over millions of OpenID enabled user accounts and over 50k websites introduced OpenID for logins. So many esteemed organizations are come up with OpenID concept, including Face book, Google, Microsoft, France Telecom, Yahoo!, AOL, MySpace, Sears, Universal Music Group, Novell, Sun, Telecom Italia, and many more. OpenID is decentralized and not owned by any organisation. So everybody can choose to use an OpenID or become an OpenID Provider for free without having to register

1.1 OpenId Authentication procedure:

1. Let's say that you're visiting a new web site that supports OpenID. In the authorization process to get services from that website, it let you to choose any of the OpenId provider.

Figure:1

2. After choosing an OpenId provider, your browser takes you from the web site you are visiting to your OpenID provider's web site.

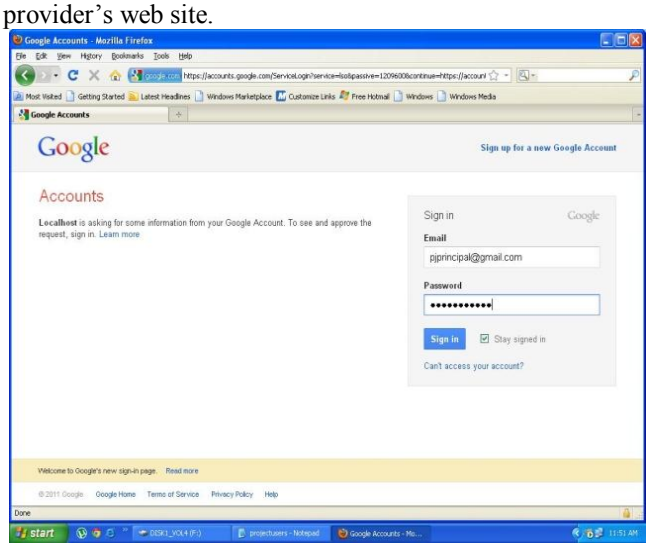

Figure:2

For example user has chosen Google OpenID provider. Then user redirected to google OpenId sign in page.

3. Log in to your OpenID provider with Your username and password.
4. After successful login. **Tell your OpenId provider that the original web site can use your identity**. You are then sent back to the original web site.
5. Then the website you have visted(i.e. service provider) check the user credentials received from OpenID provider and allow you to use service.

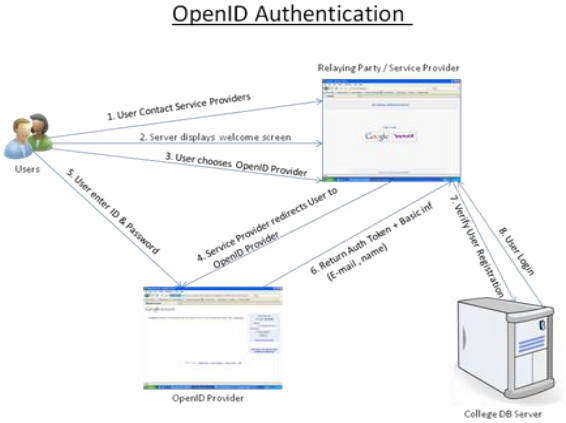

Figure 3:- The OpenID Authentication Process

Manuscript received December, 2013.

CH.V.S .Suryanarayana, Student, PVPSIT DEPT: CSE
M.V.S. N. Maheshwar, Asst.prof, PVPSIT, DEPT: CSE
K.Prudvi Raju, Asst.prof, PVPSIT DEPT: ECM

1.2 Role Based Authentication System

The Educational Academy Automation software includes Role based Authentication in addition to OpenId authentication. In this software predefined roles are assigned to users of Educational Academy. When the OpenId provider redirects the user to requested website also called relying party (i.e. Educational academy website) as authentic it checks to see the presence of user login information (i.e. openid url) in its database if the user existed it fetches role for particular user and provide access based on user role, if the user not existed in database it return with result as invalid user.

II. PHISHING

is the act of attempting to acquire information such as usernames, passwords, and credit card details (and sometimes, indirectly, money) by masquerading as a trustworthy entity in web communication. Communications purporting to be from popular social web sites, auction sites, banks, online payment processors or IT administrators are commonly used to lure the unsuspecting public.

2.1 Phishing attack in OpenId Authentication: A web site (i.e. a service provider) you're visiting uses loophole in the OpenId authentication process to launch phishing attack.

1. A website in attempt to steal user authentication details .It is pretend to give OpenId authentication support. But actually it does not give any OpenID support.
2. The malicious website, let user to choose any OpenId provider.
3. After user have chosen OpenID provider ,instead of redirecting to OpenId provider sign in page ,it redirects to some other fake sigin page which is exactly similar to original OpenId provider sign in page .The user think it is the OpenId provider sign in page and enters username and password.
4. Then the malicious website pretends to provide some kind of service through OpenId authorization processes.
5. The malicious website uses the stolen athentication details for harmful activities.

III. SOLUTION TO PHISHING

3.1 Introduction to third party

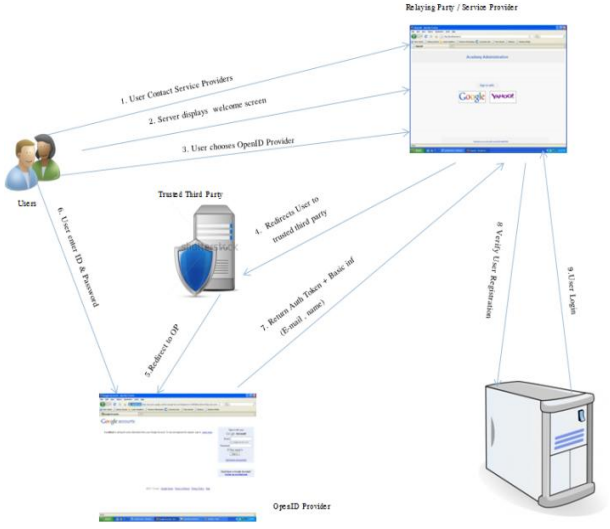

Figure: 4 The OpenID Authentication Process with third party

Aim: Make sure the user is redirected to original OpenId provider.

Third party is a trusted authority, positioned between service provider and OpenID provider in the authentication process.

Challenge: Making sure each and every redirection from client browser must be passed through trusted third party.

The steps for the OpenID authentication process with the introduction of third party are listed below

1) The service provider let the user to choose OpenID provider.

2) Upon the OpenID provider is chosen by user, the Third party check to see whether the user is redirecting to correct destination (i.e. Original OpenId provider).

3) Upon successful checking, the user is redirected to destination.

The steps involved in blocking phishing attack are listed below:

A website in attempt to steal user authentication details .It is pretend to give OpenId authentication support. But actually it does not give any OpenID support.

1) The malicious website, let user to choose any OpenId provider.
2) After user have choosen OpenID provider, the Third party check to see whether the user is redirecting to correct destination (i.e. Original OpenId provider).
3) In this case the malicious service provider is redirecting user to fake sign in page ,which get traced by third party ,found it as phishing attack and alerts the user about this information.
4) By this way user protecting from falling into phishing traps.

IV. OBSERVATION AND PROBLEM DESCRIPTION

The whole college academy automation consist of role's such as principal, head of the department, faculty, student, department non-teaching staff , administration non-teaching staff. Each and every role is having their own activities.

The activities of each role are as follows:

Principal: department creation, staff hiring, and course fee information.

Head of the department: preparing schedule, subject assigning.

Department non-teaching staff: syllabus creation, entering marks information for students, student attendance, and student credits calculation, and modifying student status.

Administration non-teaching staff: student admission and registration, course fee information, login information of users.

Faculty(teaching staff): can view schedule report.

Student: can view status, schedule and marks information.

When the user login is valid it fetches role for particular user and provide access to fetched role activities. For example, user login as head of the department cannot access activities of principal.

Problem Issues and Challenges:

There are following problems:

1. The information exchange between relying party and openId provider must be secured.
2. Proper encryption must be used for storing user login information.
3. Roles must be assigned carefully based on designation.
4. Control the information access using roles.
5. Communication between service provider and trusted

third party, trusted third party and OpenID server must be secure.

4.1 METHODOLOGY:

In Educational Academy Automation the role-id are assigned as follows, if the role is "principal" its role-id is "1", head of department role-id is "2".The full description of roles and role-id are given in table 1.

ROLE	ROLEID
Principal	1
Head of the department	2
Teaching staff	3
Student	4
Administration non-teaching staff	11
Department non-teaching staff	21

Table 1: various roles and their IDs

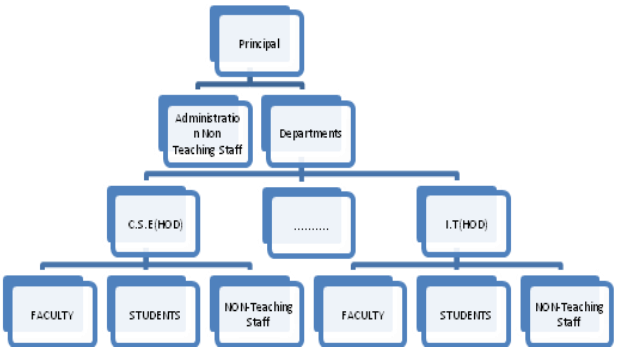

Figure 5: Basic Architecture of Educational Academy

V. ADVANTAGES OF OPENID

5.1 Increase registration and conversion rates:

Most websites ask for an extended, repetitive amount of information in order to use their application. OpenID accelerates that process by allowing you to sign in to websites with a single click. Basic profile information (such as your name, birth date and location) can be stored through your OpenID and used to pre-populate registration forms, so you spend more time engaging with a website and less time filling out registration pages.

5.2 Access rich user profile data:

Accepting OpenID gives access to a rich set of user data that would otherwise require the completion of lengthy registration forms to obtain. Many OpenID providers collect and share a wide range of demographic information, including name, date of birth, location, gender and an email address. This data allows you to optimize your marketing efforts and tailor your website to better target the needs of your core audience.

5.3 Reduce Frustration Associated with Maintaining Multiple Usernames and Passwords:

Most web users struggle to remember the multiple username and password combinations required to sign in to each of their favourite websites, and the password recovery process can be tedious. But using the same password at each of your favourite websites poses a security risk. With OpenID, you can use a single, existing account (from providers like AOL, Google, Yahoo or your own blog) to sign in to thousands of websites without ever needing to create another username and password. OpenID is the secure and simple method to joining new sites.

5.4. Gain Greater Control over Your Online Identity

OpenID is a decentralized standard, meaning it is not controlled by any one website or service provider. You can manage the amount of personal information you choose to share with websites that accept OpenIDs, and multiple OpenIDs can be used for different websites or purposes. If your email (Google, Yahoo, AOL), photo stream (Flicker) or blog (Blogger, Word Press, Live Journal) serves as your primary identity over world wide web.

5.5 Minimize Password Security Risks:

Internet users use the same password in different sites. Whereas conventional passwords are not centrally administered, if occurs any security compromise in any website, a hacker can get access to your password across multiple sites. With OpenID, passwords are never exchanged with any websites, and if a compromise happens, you can choose different password for your OpenID, thus we can stop a hacker from getting access to your accounts at any websites you go.

5.6 Reduce customer care and password recovery costs:

With OpenID, visitors to your site use an existing portable identity to sign in to your site. Because these users authenticate against an existing identity provider, there is no need to keep passwords and invest valuable time and resources into expensive account and recovering passwords. This makes you to pay more concentration on the core functions of your web application and achieve greater customer satisfaction by eliminating frustrations associated with forgotten passwords.

5.7 Link your site to the social web:

OpenID is the building block for several other open standards that allow you to enrich the experience for your users and connect your site to the social web. Protocols of open source such as Portable Contacts can be used with OpenID to offer your site access to a friend's address books and lists.

VI.OPENID IMPLEMENTATION

For Academy Automation, OpenId authentication was implemented by using java servlets.java provided OpenId authentication facility through package called OpenId" (version used in academy automation is jopenid-3.0). In academy automation, the login screen is changed (i.e. login screen is not as in introduction part).

6.1 How to design UI for sign on:

You may list all OPs in sign on page to let users choose their OpenID accounts. For example:
Please sign on from : Google Account
Yahoo Account
All OP that are listed in openid-providers.properties file(can add more op's in this file):
Google = https://www.google.com/accounts/o8/id Yahoo = http://open.login.yahooapis.com/openid20/www.yahoo.com/xrds

6.2 How JOpenId Works:

public void init() throws ServletException {
super.init();

```
//First, create an OpenIdManager instance and set your web
site domain
manager = new OpenIdManager();
//The setRealm is the realm that the Identifier returned from
OP contains.
manager.setRealm("http://localhost");
// The setReturnTo is the URL which handles the
information returned from OP. Usually it is a Servlet or
Action of an MVC such as Struts
manager.setReturnTo("http://localhost/ajax1/openid");
}
protected void doGet(HttpServletRequest request,
HttpServletResponse response)
throws ServletException, IOException {
String op = request.getParameter("op");
if (op==null) {
//this part of code will be called upon return from openid
provider
// check sign on result from Google or Yahoo:
checkNonce(request.getParameter("openid.response_nonce"
));
// get authentication:
byte[] mac_key = (byte[])
request.getSession().getAttribute(ATTR_MAC);
String alias = (String) request.getSession
().getAttribute(ATTR_ALIAS);
Authentication authentication =
manager.getAuthentication(request, mac_key, alias);
response.setContentType("text/html");
System.out.println("Identity: " + authentication. getEmail());
}
else if (op.equals("Google") || op.equals("Yahoo")) {
// redirect to Google or Yahoo sign on pages:
Endpoint endpoint = manager.lookupEndpoint(op);
Association association =
manager.lookupAssociation(endpoint);
request.getSession().setAttribute(ATTR_MAC,
association.getRawMacKey());
request.getSession().setAttribute(ATTR_ALIAS,
endpoint.getAlias());
String url = manager.getAuthenticationUrl(endpoint,
association);
response.sendRedirect(url);
}
else {
throw new ServletException("Unsupported OP: " + op);
}
}
```

VII. CONCLUSION

The research involves introduction of trusted third party for designing security model in Academy Automation process and provided solutions for phishing attacks.But, this kind process slows down authentication checking, so there is a need to improve the performance of the trusted third party algorithms to make this process fast.

REFERRENCE

[1] http://openid.net
[2] http://code.google.com/p/jopenid/wiki/QuickStart
[3] http://bkathir.wordpress.com/2009/12/10/how-to-use-openid-technology-in-our-web-application/
[4]. https://www.blackhat.com/presentations/bh-usa-
[5] http://en.wikipedia.org/wiki/Phishing
[6] http://www.informationweek.com/attacks/phishing-attackers-use-subdomain-registration-services/d/d-id/1097432?