Central Authentication System Using CAS in Distance Learning Application

Idris Winarno, SST, M.Kom, Akhmad Alimudin, S.ST, Anang Siswanto
Politeknik Elektronika Negeri Surabaya Kampus ITS Sukolilo

Abstract

Presently, most applications still use a separate authentication system for each application. It makes it difficult for developers to synchronize the authentication system. The solution to the problem is Single Sign On. Out of various Single Sign On applications, CAS (Central Authentication System) is chosen as the Single Sign On application. LDAP used in Zimbra is used for users' database which is to be integrated with distance learning application such as Moodle. Implementing CAS in distance learning application such as Moodle makes it easier for users in the login process in which they do not need to repetitively do it. They just need to do it once and all the applications which support distance learning can be accessed. In addition to that, the authentication system becomes more secure because CAS makes use of HTTPS protocol which guarantees higher security because it is supported by a third-party certification system provided by Digicert which has been tested on a CAS server at PENS.

Keywords: *CAS, LDAP, Single Sign On, authentication*

1. INTRODUCTION

1.1. Background

The universality of HTTP protocol has attracted many developers in developing an application. It is shown in the present application development which is mostly web-based in the interface and most of the applications require authentication to enter the administration system [1].

One of the methods which is frequently used is LDAP [3]. LDAP offers ease of use in password management and usage in which users only need to remember one password for various applications. Still, users have to enter their names and passwords when doing the authentication in each application. LDAP method has several drawbacks [1]:

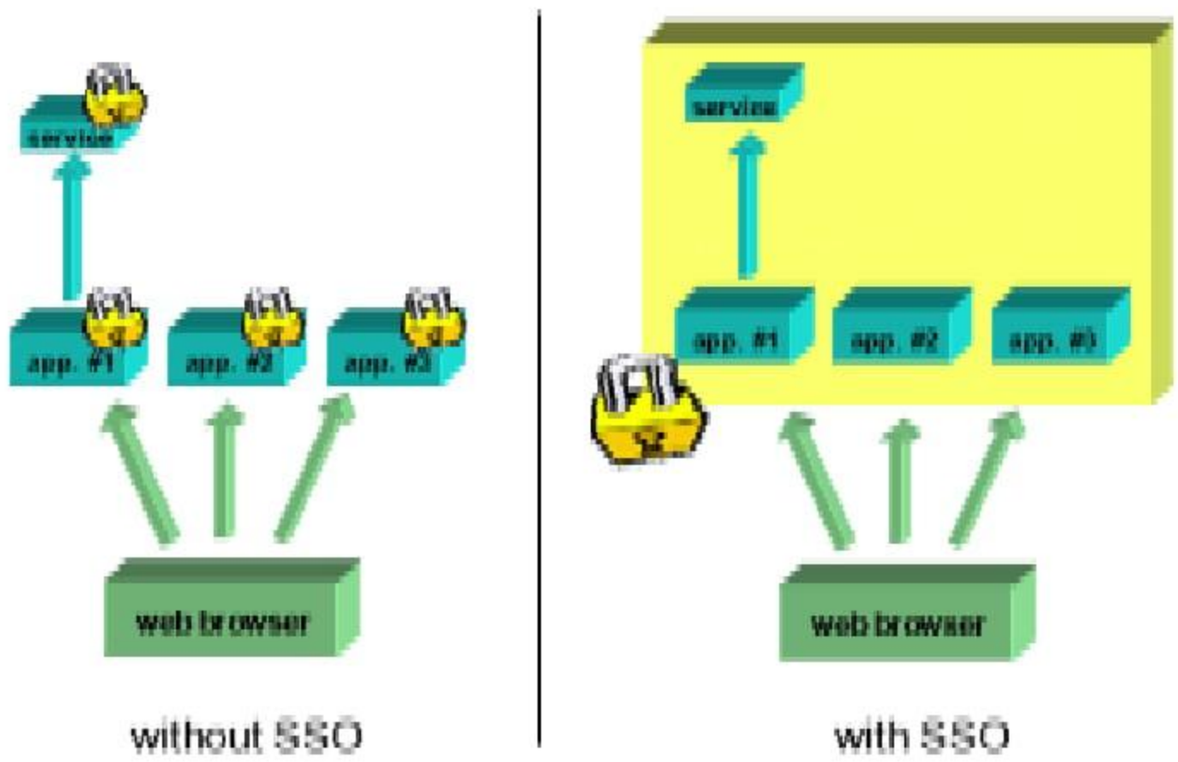
- Repeated authentication. Users still need to provide their user names and passwords for each application.
- Security. Because every time users log in to the applications they need to provide user names and passwords, there are more user name and password forms which possibly lead to password theft.

To overcome those drawbacks, a new authentication mechanism is developed which is called Single Sign On (SSO) [1] in which the methods used are:

- Central authentication on a single server which is the only engine with a login page through an encrypted protocol (HTTPS).
- The use of HTTP Redirection from the application to the authentication server for an unauthenticated user and return to the application when the user has been authenticated.

- The information on the user is sent from the authentication server to the application using cookies or CGI parameters.

The SSO used in the research is Central Authentication Service (CAS). Picture1.1. illustrates the differences between services which are integrated with CAS and services which are not integrated with CAS. In the applications which are integrated with CAS, authentication is managed by a service which is separated from the application. The application which utilizes the service uses a ticket generated by the authentication service to authenticate.



Picture 1.1: With and Without SSO System

1.2. Statement Of Problem

Because there are many applications used in distance learning processes, users have to undergo an authentication process in each service. This causes difficulties and causes the operation to be slower when more supporting services are used in the distance learning processes.

1.3. Objective

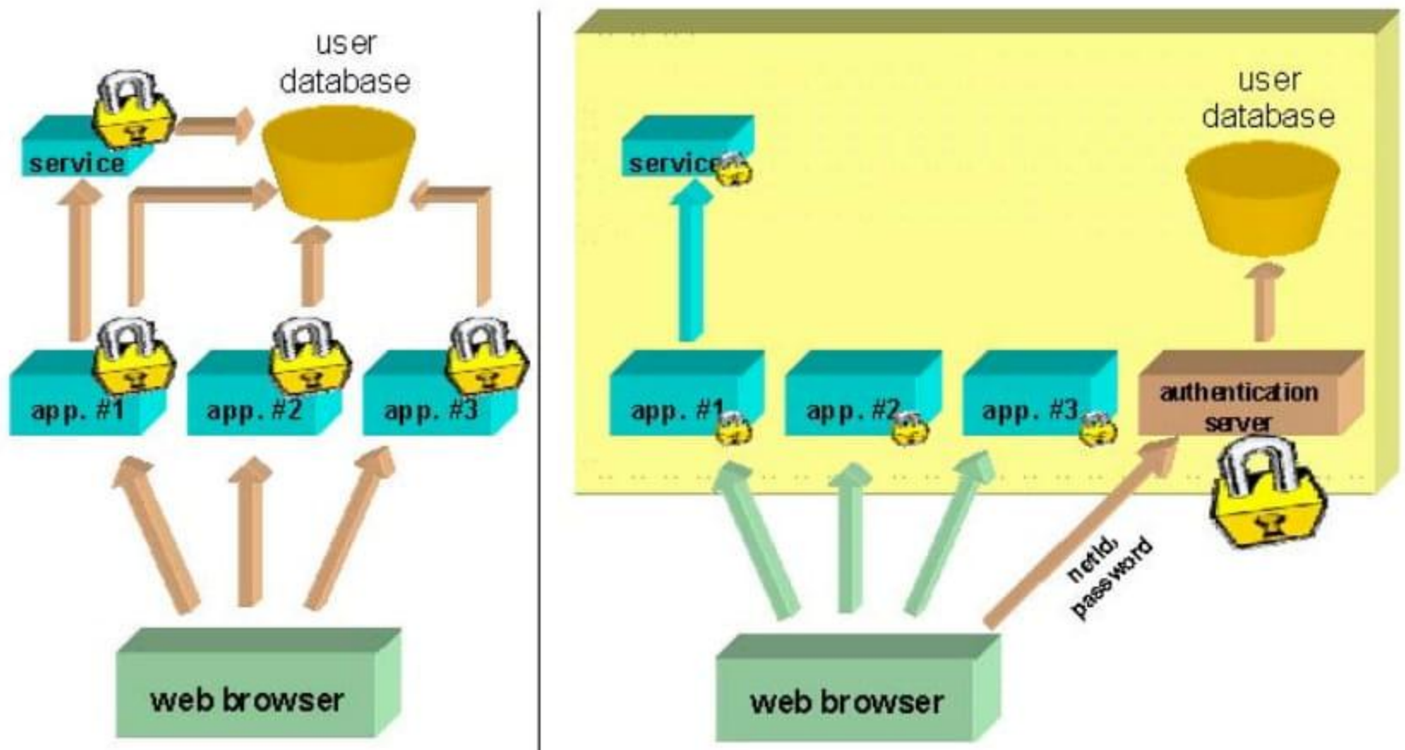
The research is aimed to providing a more user-friendly authentication system to service users by the means of:

- Centralized authentication in each service in which when a user authenticates in one service, in other services the user will automatically be authenticated.
- Secure authentication in which not many forms have to be repeatedly entered and also by the means of secure line (HTTPS) service.

2. LITERATURE REVIEW

2.1. CAS Server

CAS consists of a collection of Java servlets which can run on almost all servlet engines (JSP spec 1.2 compliant) which offer web-based authentication. CAS offers advantages such security, feature proxying, flexibility, endurance and various client libraries [1]. Picture 2.1 illustrates an authentication process in an application without CAS and an application which is integrated with CAS. In the application without CAS, the authentication occurs in each application and each application accesses the user database in which the process is prone to login data hacking. Whereas in the application with CAS, the authentication process and the user database access take place only in the authentication server. The application makes use of ticket produced by authentication server to authenticate the user.



Picture 2.1 Application with and without CAS

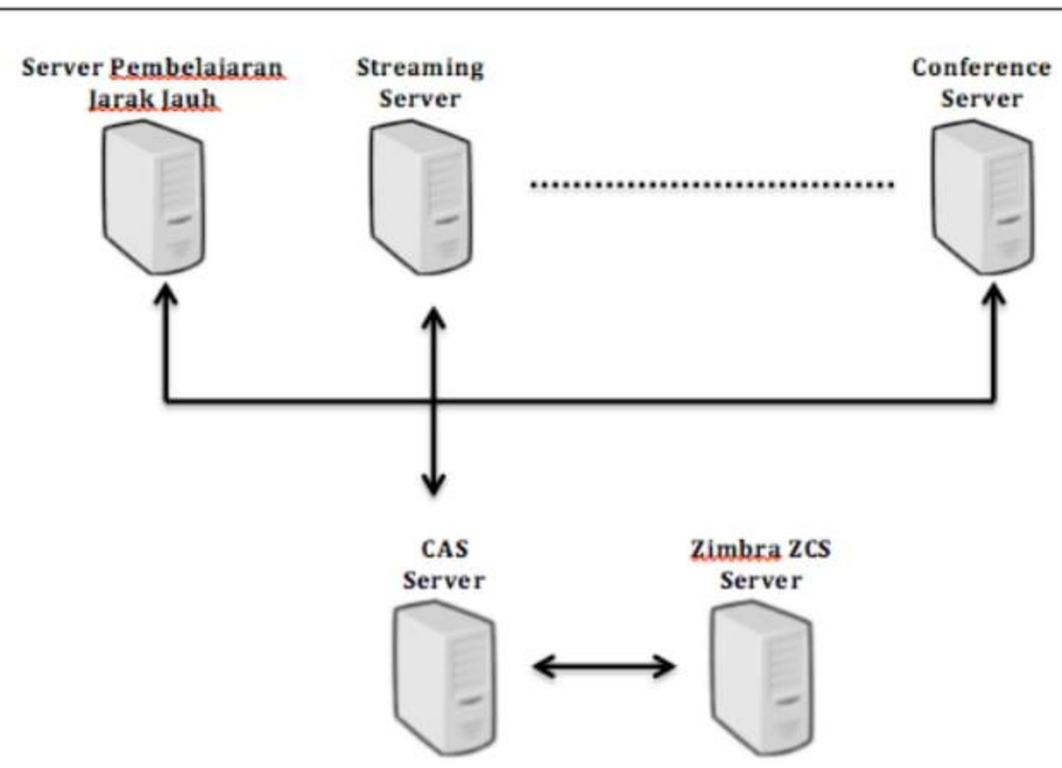
2.2. Client Library, LDAP and Zimbra

The basic protocol code can easily be written on the client's side. Additional libraries are available for Perl, Java, ASP and PL/SQL. Besides that libraries for PHP are available. The libraries for PHP make it easier for website developers to integrate one service to the other. All of the libraries offer excellent flexibility to apply CAS in an application just by adding several lines of codes [1]. A module (mod_cas) is also available to authenticate the static content in Apache Web Server. PAM (pluggable Authentication Modules) Module is available for authentication in low-level applications which do not have web interface.

3. SYSTEM DESIGN

3.1. General Design

Picture 3.1 illustrates how several servers representing a service or application support distance learning server (in PENS it is called "Dosen Jaga"/Standby Lecturer), Streaming Server, Conference Server and so on. The services are separate from one another in which authentication for the services uses the same user account which exist in Zimbra ZCA. However, the authentication processes in the services are not integrated to one another.

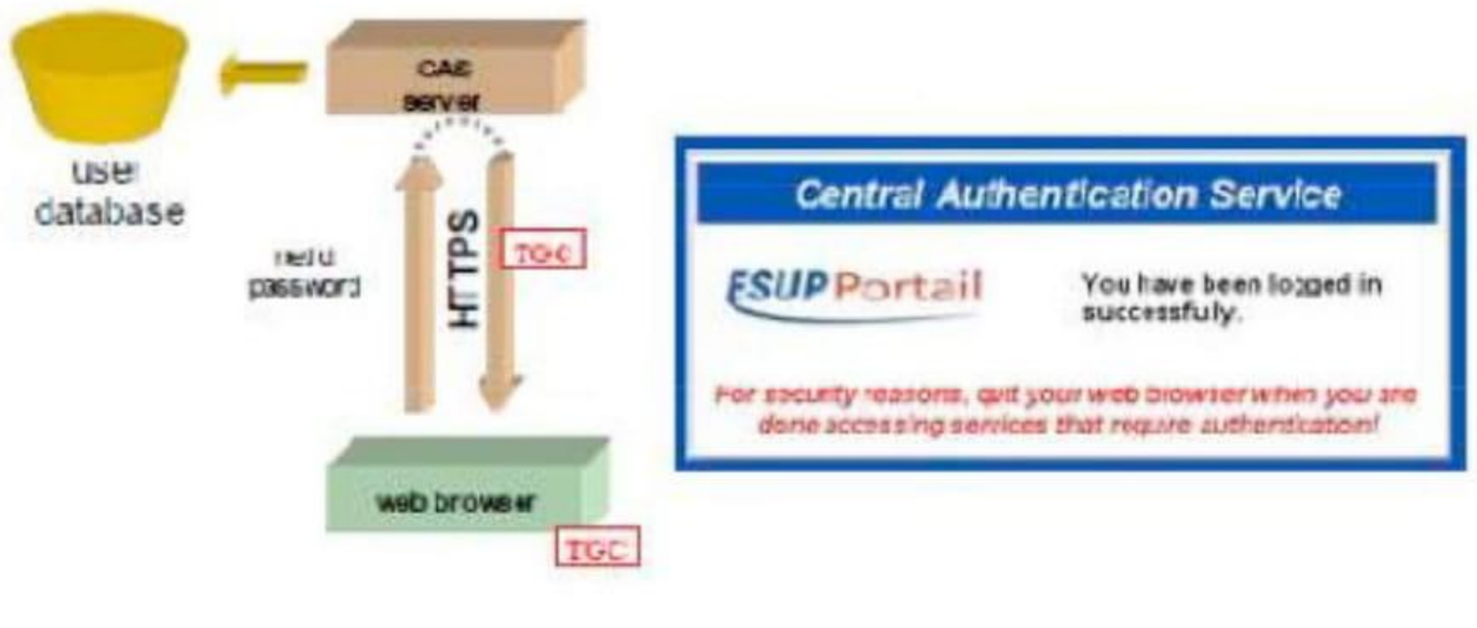


Picture 3.1: Implementation of SSO System Design

Server Pembelajaran Jarak Jauh = Distance Learning Processes Server

3.2. Software

The research uses CAS Single Sign On software package which is developed by Jasig. Picture 3.2 shows the block diagram illustrating a browser authentication process to CAS. Unlike other Single Sign On applications like Microsoft.NET Passport and Sun One Identity Server which are not freeware, CAS is a Single Sign On system which is freeware and open-source and has been used in many countries [4]. A CAS server consists of a collection of Java Servlets and Java Server Pages so that in the deployment process the server must use a servlet container. In this research, the servlet container used is Tomcat 5.5. The user database used is OpenLDP of Zimbra ZCS. For the purpose of integration to PGO-based applications, Apache 2.2.3 is used.



Picture 3.2: Browser-to-CAS Authentication Process

3.3. CAS Library

CAS has a library which is tested in this research. The most frequently used is phpCAS library on the grounds that the applications or services developed in distance learning are web-based. The phpCAS is most frequently used in the SSO-based authentication process.

The phpCAS itself can be integrated to both PHP4 and PHP5 making it more flexible in terms of integration to several types of applications which are ready-made or open-source like Wordpress, Moodle, Joomla! and the like.

3.4. Security System

The security system in the CAS technology which is tested in this research makes use of the security system which is readily available in the web. The security system used is HTTPS Protocol in port 443 where a server must have a certificate which is self-sign or commercial like the ones released by Verisign, Commodo, Digicert and the like. The certificate is used as communication media not only between a client (browser) and a CAS server but also between a CAS server and LDAP server, in which case Zimbra ZCS which uses LDAPS in port 636 is utilized.

4. TESTING AND SYSTEM INTEGRATION

4.1. LDAP

The testing and analysis of the LDAP server checks if the LDAP server runs accordingly. The tested LDAP server has been integrated to Zimbra and query in the CAS server engine. The testing process some parts of the operations taking place in LDAP used. The operations used in the testing are search and query which do not require a bind process. Picture 4.1 shows the query result towards the LDAP server used by Zimbra.

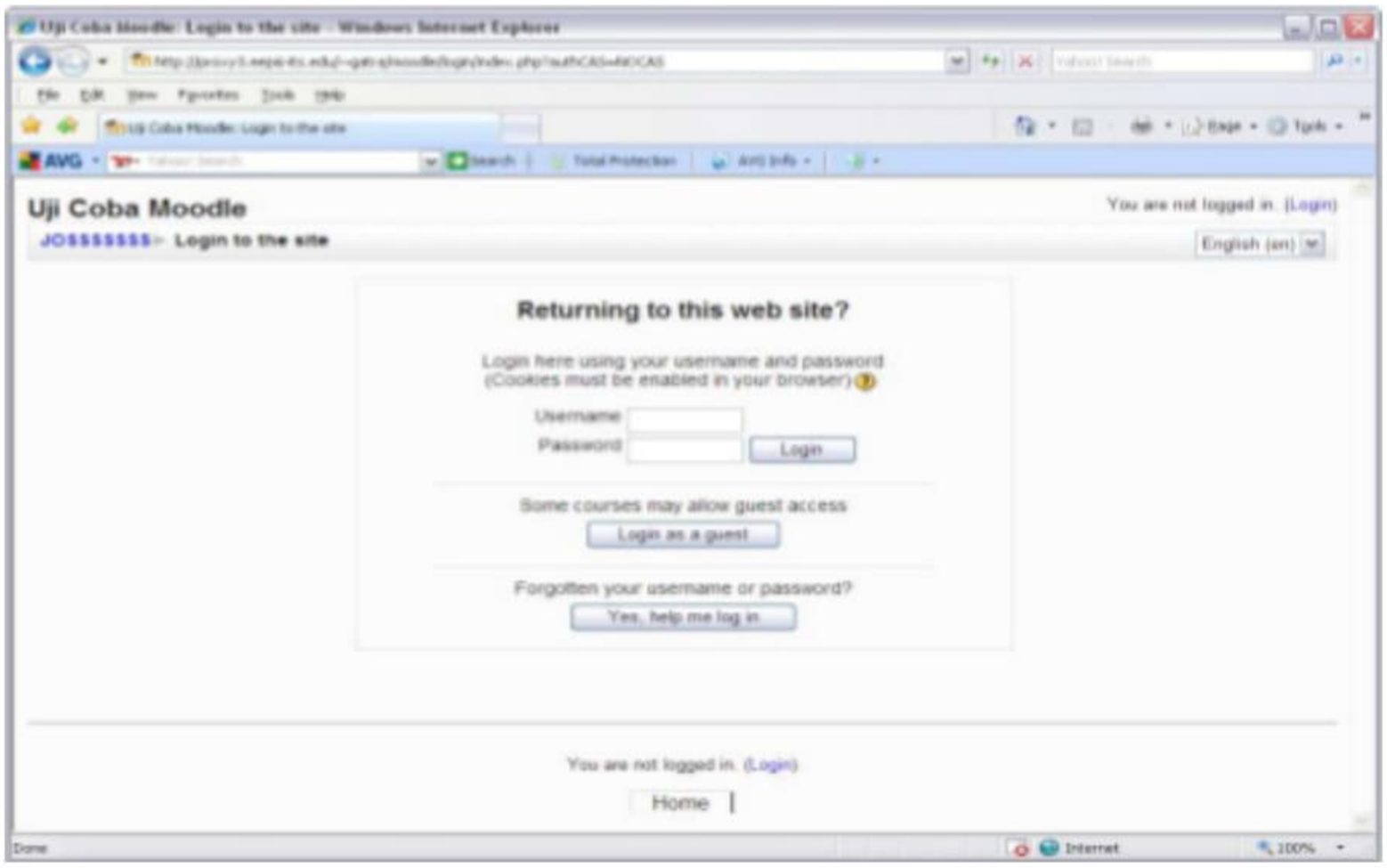
Picture 4.1: LDAP Query in Zimbra

4.2. Integration of CAS to Moodle

In this research, Moodle is used as the e-learning application. The integration of the application to CAS authentication system is relatively simple compared to that of other applications. It so happens because basically Moodle supports CAS.

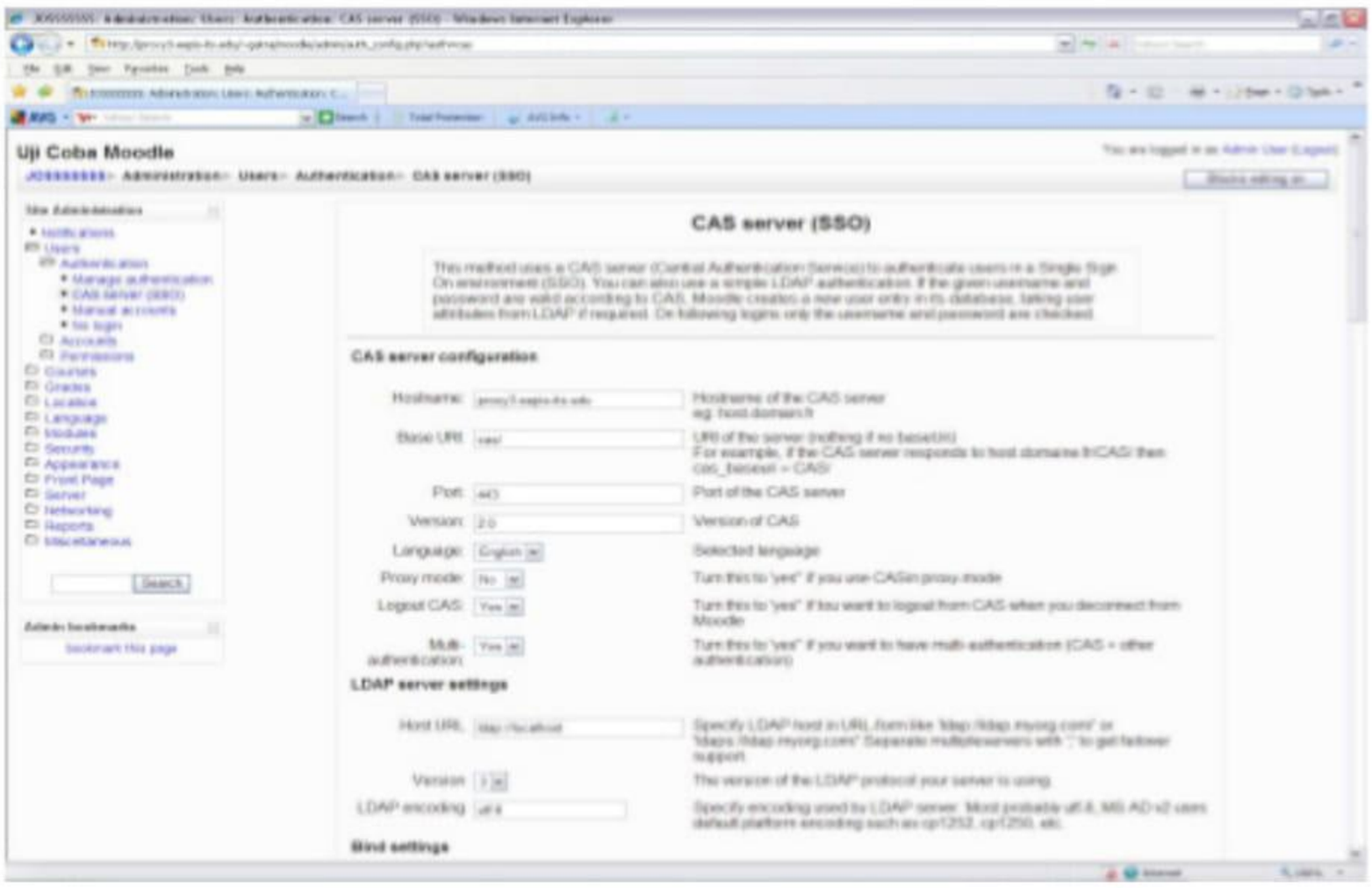
In addition to being user-friendly, the application is flexible in terms of authentication. Take, for instance, in the integration to CAS. Moodle provides a default login form from the application which provided for users who are not registered in the CAS server.

After all installation processes are complete, the next step is Moodle configuration so as to enable it to be integrated to CAS. To be able to configure Moodle to be integrated to CAS, users must have the administrator privileges. The first step is to login as a user with administrator privileges through Moodle default login page shown in picture 4.2.



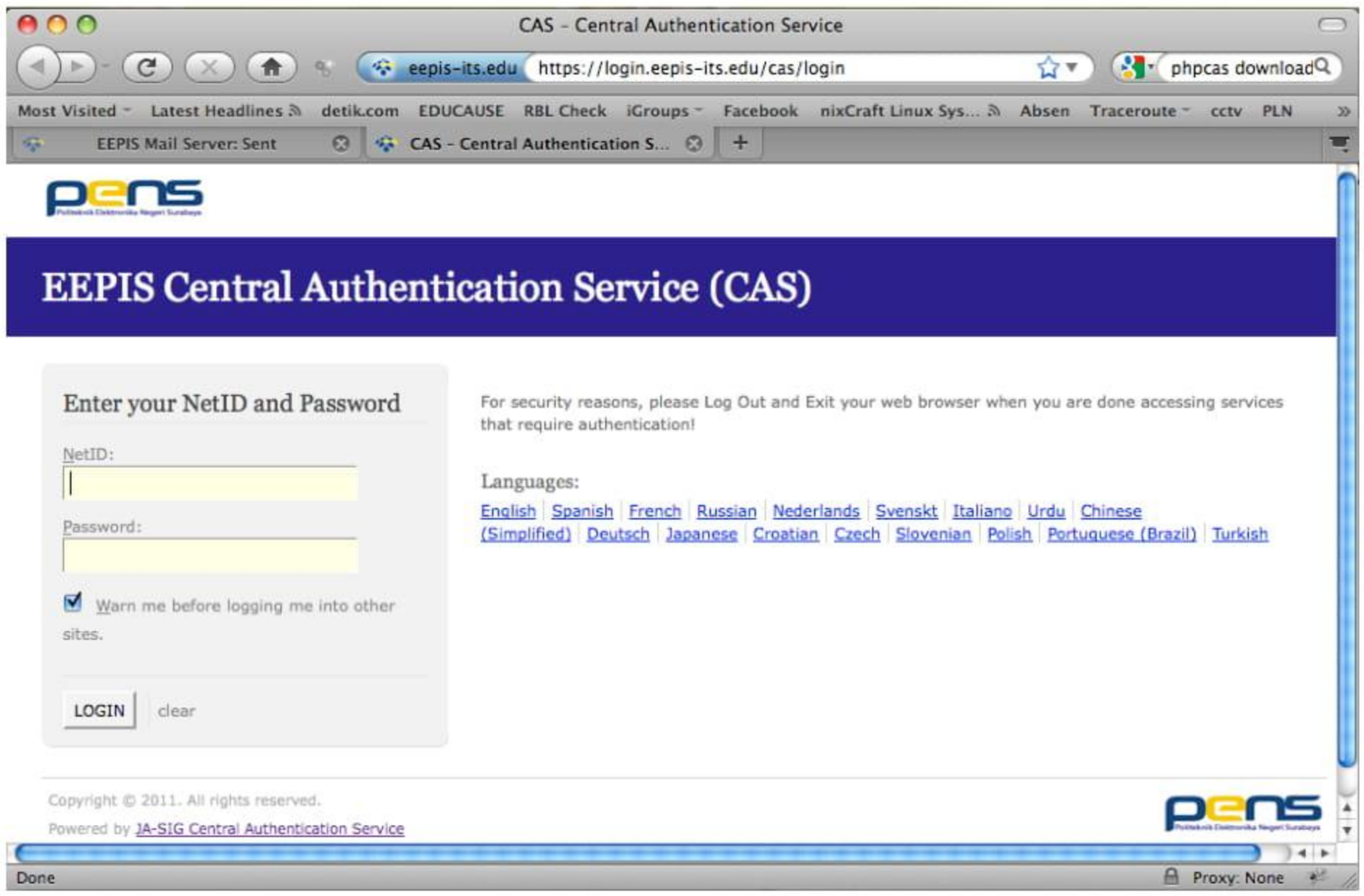
Picture 4.2: Moodle Login Page without CAS

To activate CAS authentication, select Users → Authentication → CAS Server (SSO) on the administrator page. Picture 4.3 shows what appears.



Picture 4.3: CAS Administrator Page in Moodle

To test the Moodle, the login layout is directed to CAS login address as shown in picture 4.4.



Picture 4.4: Login Page after the use of CAS

5. CONCLUSION

From the system testing, several conclusions can be drawn:

- Users' data are more secure in the login process because the authentication process runs on the HTTPS protocol
- System can run well by using SSL certificates released by trusted sources.
- One of the weaknesses of CAS system is when a user cannot maintain his/her password confidentiality. If the confidentiality is breached, all services can easily be accessed.

References

[1] Aubry Pascal, Julien Marchal, Vincent Matheieu,ESUP-Portail: open source Single Sign-On with CAS (Central Authentication Service) Paper On EUNIS2004, 2004

[2] Aubry Pascal, Julien Marchal, Vincent Matheieu,ESUP-Portail: open source Single Sign-On with CAS (Central Authentication Service) Presentation On EUNIS2004, 2004

[3] Berg Alan, Bas Toeter, Single Sign On and Single Sign Off in a non-homogeneous portal front ended environment Paper On Central Computing Services, Universiteit van Amsterdam, Amsterdam, 2005