Journal of Theoretical and Applied Information Technology
20th June 2013. Vol. 52 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

SCAS: AN IMPROVED SINGLE SIGN-ON MODEL BASE ON CAS

[1,2]XIANG LIYUN, [1]FANG ZHIYI, [1]SUN HONGYU

[1]College of Computer Science and Technology, Jilin University, Changchun, China

[2]Department of Computer Engineering, ChangJi University, Changji, Xinjiang, China

E-mail: fangzy@jlu.edu.cn

Single sign-on is a strategy that is used to integrate user information resources, provide solutions for unified login and enhance the security of the user accounts. There exist many realizations of single sign-on model, such as Cookie based, passport based, CAS based and etc., in this paper, the merits and drawbacks of the various realizations have been analyzed. According to the analysis and the requirements of the enterprise, SCAS (Separated CAS) which is the improvement of CAS has been proposed to integrate the user information resources, Actual application of SCAS model shows that the database server pressure was reduced, the response speed was increased, the scalability of business logic was also better than CAS model.

Keywords: *Single Sign-on, CAS, SCAS, throughout*

1. INTRODUCTION

The technology of single sign-on is a popular solution used to integrate user information resources, realize unified login both in and abroad, users can access the pages freely as long as they have logged in one time in any of the pages in the related domain. As the widely used of WEB2.0 and the information flooding in the Internet, the problems form both users and enterprises themselves are serious; for the users, the user experience is worse and worse because more and more account and their passwords have to be remembered for the growing various system; and for enterprise, with the increasing of the service provider systems, the management of the redundant information and the synchronization between different account owned by the same user is more complex. So how to design an efficient single sign-on model to resolve these problems has great significance.

Currently, there are four main realizations of single sign-on solution: Cookie based realization, passport based realization, SAML based realization, and CAS [1] based realization. The Cookie[2,3,4] based one is easy to realize, but it doesn't support access across domain unless using additional technology such as script and etc.; passport based one with high security but the scalability is very limited since it is not open source; the SAML based one doesn't need the third authentication and easy to implement, but it not quite compatible with the existing equipment, and it will cost very much to configure new equipment; the CAS based is open source, can be re-exploited according to the requirements, and the information of users are stored with encryption code in order to enhance the security of the account.

On the basis of the analysis of existed single sign-on technologies, SCAS, a new single sign-on realization which is an improvement of CAS has been proposed in this paper. The new realization can be used to integrate the information of the enterprise, simultaneously; the model has been used to integrate the human resource system, office task system, financial system sales system and customer feedback system in a famous IT enterprise, the application manifest that the cost for exploiting the system has been cut down and it is more convenient for user to manage their own account information

2. RELATED WORK

2.1 Single Sign On

According to principles of Single Sign-On [5], related technology is divided into two, one is based on third party, second is not based on third party. Limited to the company's hardware, this paper

Journal of Theoretical and Applied Information Technology
20th June 2013. Vol. 52 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

researches the technology base on third party, which is showed in Figure 1:

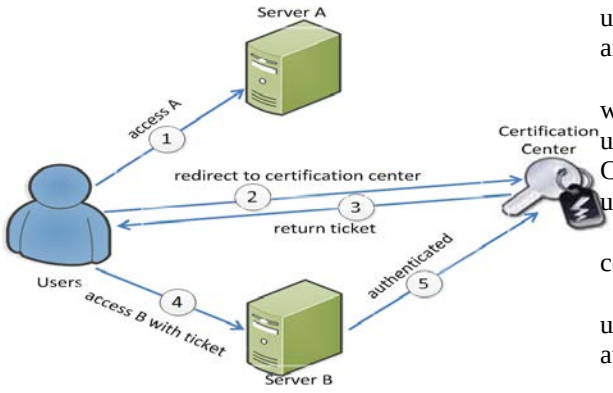

Figure 1: Principle of Single Sign On

As is shown in Figure 1, the process of SSO is:

(1) When the user accesses system A for the first time, system A detects whether the user has signed on.

(2) If the user has not signed on any business system, system A will guide the user to the signing on page.

(3) After the user signing on from the page, SSO generates a ticket to mark user identity, and send it back to the user.

(4) When the user accesses system B, the identity ticket is brought to be authentication credential.

(5) System B carries the ticket to the authentication center of SSO after it accepts the request, and the authentication center checks the legality of the ticket.

2.2 Central Authentication Service

CAS, developed by Yale University, is a typical single sign-on model. It is open source and supports agent function. CAS has a strong robustness, suitable for client applications developed by any language, such as JAVA, PHP, Ruby, C#. Data transmission between user browser and central authentication server is based on the HTTP protocol, while pure HTTP protocol is used between application system and central authentication server. In CAS, transmission of sensitive information all adopts SSL encryption channel technology, and information stored in cookie files of clients is processed by encryption, so the security is very reliable.

To illustrate working principles of CAS, we first make the following definition:

CAS Server: CAS Server is deploy separately, and mainly responsible for the validation of user identity. CAS provides a compatible way to extract users' information from database or XML files, and specific implementations can be customized.

CAS Client is deployed in each business system which needs to achieve single sign-on. When the user accesses business system through the browser, CAS Client redirects the user to CAS Server for user authentication, rather than business system.

In the CAS single sign-on model, there are two commonly used concepts.

TGC: The cookie written in CAS Client with user identity ticket, after user identity authentication.

ST: CAS Client intercepts requests for user accessing system service, and redirects them to CAS Server. Server generates a unique ST for Client according to TGC and certain allocation algorithm. Later, when Client requests business system, whether there is this ST in user cookie is checked.

SSL: The communication protocol between Client and Server.

Based on the above definition, Figure 2 shows working principles of CAS when the user accesses business system for the first time:

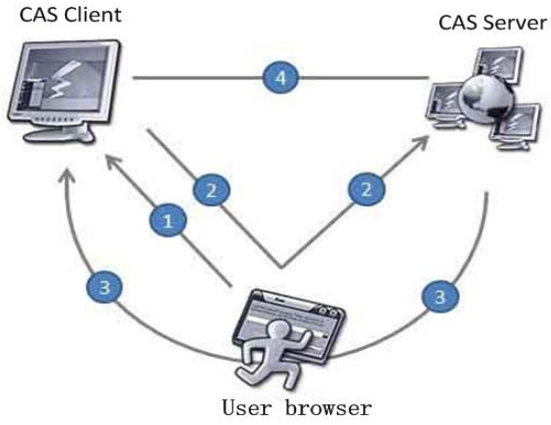

Figure 2: Working Principles of CAS

Shown in Figure 2, CAS certification process is as follows:

(1) The user accesses resources of business system though the client browser, CAS Client deployed in the business system can intercept request from the user. Then CAS Client authenticates whether the cookie in current business system contains ST information, if so, CAS Client directly carries the ST to interact with CAS Server; if not, turn to (2).

(2) CAS Client encodes the URL of business system resource as a parameter, and redirects the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

user's browser to CAS Server authentication, so as to get ST ticket.

(3) The user's identity needs to be authenticated, if the identity password is correct, CAS Server generates an only ST corresponding with current user and business system, though a set of encryption algorithm containing user identity information, and each business system can customize the validity of the ST according to their needs. After the generation of ST, the user's browser is redirected to the URL and the generated ST is transferred as a parameter, according to the above step. Then CAS generates a TGC ticket, which is equivalent to the credential to access business system, and writes the TGC into cookie files belonging to CAS Server. As a supplement, the TGC does have failure time, and the specific failure time can be customized. The TGC generated only has relationship with CAS Server, as long as TGC does not fail, CAS Server sides that the user has been carried out single sign-on. And only CAS Server can read the TGC of its domain own, so, the cross-domain problem is solved.

(4) CAS writes the ST, which is carried by the URL page redirected from CAS Server, into cookie files belonging to business system domain, and interacts with CAS Server to verify the legitimacy of the ST. After confirming, CAS Server returns the information of user login, if the validation is not passed, it returns an error message and redirects user page to the login one.

After the user browses pages, for the security, usually logs out, so the logout process is as follows:

(1) CAS Client deployed on system A sends logout request to CAS Server.

(2) CAS Server reads TGC in cookie file of its own domain to determine whether it is invalid. If so CAS-Server deletes the TGC, and returns error code that the user could not logout. Else if the TGC is valid, it deletes TGC and returns successful code.

(3) CAS Client displays the appropriate information on the browser according to the code returned, and finishes the logout operation.

2.3 Analysis of CAS

According to the above working principles, CAS Single Sign-On authentication process can be divided into three scenes: the user accesses business system for the first time, the user accesses other business system after logging in CAS Server, the user logs out. And specific functions of each scene have been described in detail in 2.2. CAS Single Sign-On model can be divided into the following parts by role entity: Client, Business System, CAS Client, CAS Server, and Data Storage.

(1) Client: The main function of Client is: 1) access resources needed though browser; 2) save cookie file written by Server, in order to support SSO; 3) interact with CAS using SSL protocol; 4) be transparent to users.

(2) Business System: Business System includes the existing system and the newly developed system, in both cases we need to do one thing, adding CAS Client package to Business System, to interact with CAS Server instead of it.

(3) CAS Client and (4) CAS Server has been defined above, we do not describe in detail here.

(5) Data Storage: The main function of this section is storing data information of users persistently. Forms of storage are diverse, like DB, XML, and LDAP server.

We can see that CAS Server is complex from the above analysis, CAS Server model is high coupling, and greatly reliant to each other, and not conductive to expansion of the business logic. Therefore, the paper is committed to optimize CAS Server, in order to make it better suit for business expansion.

3. SCAS

3.1 A separated server single sign-on model base on CAS

Though the analysis of Section 2 it can be seen that, functions of CAS Server are complex, including storage of user information, verification of user identity, generation of related tickets, operation of DB and so on. According to the actual needs of the application project, certain improvements to the model by CAS single sign-on design are what we should do, in order to meet the complex business logic. So, our system is divided into five parts, functions of each module and their mutual relationships are shown in Figure 3:

Journal of Theoretical and Applied Information Technology
20th June 2013. Vol. 52 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

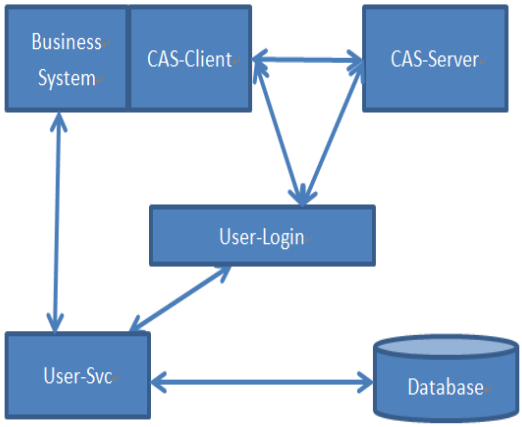

Figure 3: Design of SCAS

According to the actual situation and specific business needs, we decompose CAS Server of CAS Single Sign-On model into different functional modules, User-Login and CAS-Server. User-Login module is mainly responsible for users' login and control of users jumping between different Business Systems, while CAS-Server module is for generation of TGC and ST, also storage of them. Database stores all users' information attached to SSO system, which includes username, password, role, and permissions.

3.2 Implementation of SCAS

In Section 3.1 we propose an improved model SCAS, we achieve it in this part, and verify its superiority according to the testing and using in a well-known IT company.

SCAS integrates Client, CAS Client and CAS Server for this shortcoming of CAS, specific methods are as follows:

Remove authentication achieved at the user's browser, and transfer it to CAS Client.

CAS Client only retains the function of helping related business system to verify, the rest is compressed to CAS Server.

Because of step (2), CAS Server becomes more complex, so SCAS divides CAS Server into two parts. The first is single, only responsible for storing and maintaining users' identity tickets; another part achieves other functions. The two parts adopt ways shown in Figure 4 to communicate and work together, so as to achieve SSO.

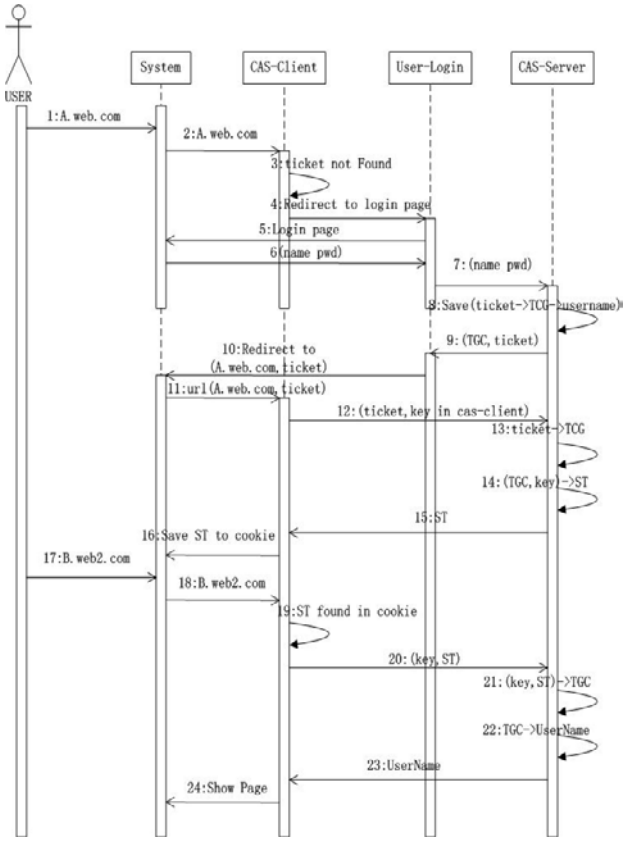

Figure 4: Work flow of SCAS

Here we analyze the user authentication process of improved SSO model. In order to clarify the process, the division is detail, so the steps are as follows:

(1) The user accesses business system A through the client browser.
(2) CAS-Client deployed in the business system intercepts the HTTP request of accessing A.
(3) CAS-Client detects whether there is ST information in cookie file, which belongs to the system A. Here it is the first time for the user's access, the answer is no.
(4) CAS-Client redirects the user's browser to User-Login module.
(5) User-Login module displays login page to the user, according to current state of the user.
(6) The user inputs its identity authentication information, which includes username, password and so on.
(7) User-Login module verifies the user's information, and passes related information to CAS-Server if it is correct.

Journal of Theoretical and Applied Information Technology
20th June 2013. Vol. 52 No.2

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

(8) CAS-Server generates unique TGC and ticket of user identity according to the information submitted, and saves them.
(9) CAS-Server passes the TGC and ticket generated to User-Login module.
(10) User-Login module redirects the user's browser to business system A that it requests, and splices the ticket behind the URL in the form of parameter.
(11) CAS-Client intercepts the request.
(12) CAS-Client carries the ticket to CAS-Server to verify.
(13) CAS-Server finds TGC according to the ticket and verifies.
(14) The verification is passed, it generates ST bill.
(15) The ST bill is returned to CAS-Client.
(16) CAS-Client writes the ST bill in cookie file of business system A. From Step (17) are the steps of after login:
(17) (18) Same as (1) (2).
(19) CAS-Client detects there is ST.
(20) It directly carries the ST to CAS-Server to verify.
(21) (22) CAS-Server verifies the user's bill.
(23) (24) Verification is passed, and CAS-Client displays the system resources to the user.

In the work flow, User-Svc module is mainly responsible for database interaction, and provides accessing interfaces to business system. User-Login handles user's login, registration, cancellation, and jumping between systems. The main two processes are login and jumping, because they coordinate the implementation of CAS-Server module.

CAS-Client interacts more with User-Login based on the original work. For CAS-Server does not need to the storage of related data in our SCAS model, response in the server is increased.

The model we improve in the paper has several advantages: responsibility of functional module is single; through interface to operate database; the foreign services are safe; it reduces the complexity of CAS-Server; maintainability of the whole system is improved.

4. PERFORMANCE ANALYSIS OF SCAS

From the above identity authentication we describe, when there is ST information in the business system, CAS-Client does not interact with User-Login module any more, this reduces the press on the access of database. Only when the ST does not exist or it fails, CAS-Client interacts with User-Login, this also reduces the press of CAS-Server. To test the changes of server pressure, we do a simulation, assuming that there are 100 visitors to access the site, and each visitor switches 300 times after login. The changes of login request that CAS-Server handled is shown in Figure 5:

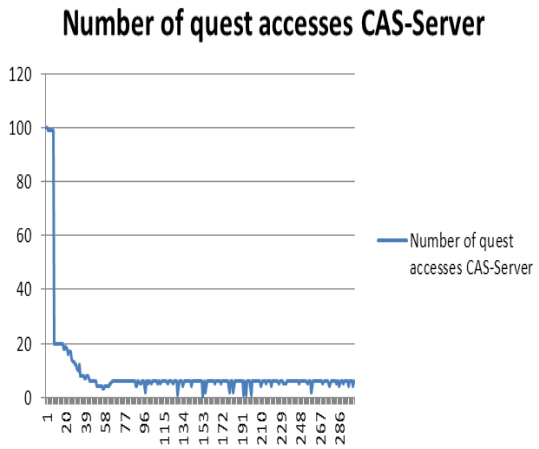

Figure 5: Number of quests of CAS-Server

We can see from Figure 5, at the time of user's login, it needs relocation to get ST, and the request of accessing CAS-Server is much. But with the increasing of login number, users access CAS-Server rarely access CAS-Server, the press of server reduces much, and it can support more users to do single sign-on.

Because we make the function of CAS-Server single, throughput of CAS-Server is larger, which means that the login request server can concurrently handle per unit of time is larger. We do test for the original and improved CAS-Server, assuming that the number of concurrent users is increasing, and the throughputs are as follows in Figure 6:

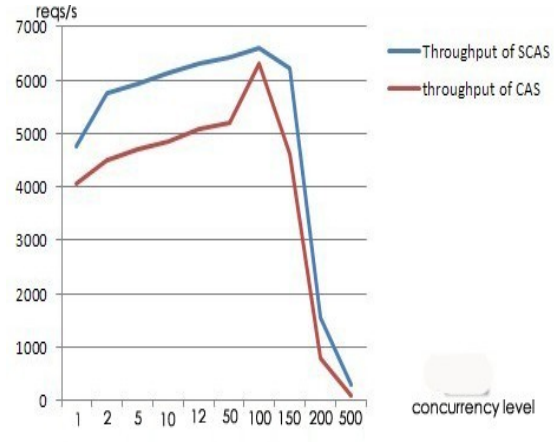

Figure 6: Throughputs of SCAS and CAS

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

The result shows that, the throughput of SCAS model is 12%-20% higher than the one of CAS. SCAS can handle more concurrent requests, so this improves server's capacity of withstanding pressure and response of login request. Meanwhile, to see the actual results, the core idea of SCAS model has down well in the real project.

5. CONCLUSION AND FURTHER WORK

This paper analyzes the advantages and disadvantages of current Single Sign-On models, and selects CAS according to the actual needs of the project. In order to increase the response speed and the throughput of our system, we design and implement an improved model, Separated Single Sign-On. The server pressure testing servers that the press of CAS-Server reduces significantly and the throughput of SCAS model is 12%-20% higher than the one of CAS in average situation.

The SCAS model only takes the problem of single sign-on into account, but the merger of current account numbers is not considered in the paper, so the further research is how to merge them.

6. ACKNOLEGMENTMENT

This work was supported by fund project of CHANG JI University named "Research on Load scheduling and equalizer software Based on Cluster Service": 2011YJYB08; fund project of National Nature Science Foundation of Jilin Province China (No.201215189)

REFERENCES

[1].L.X.Zhong. Disquisition on CAS Protocol-based Single Sign-on System.[M]. Sichuan university.2006:9-10.

[2].P.Wu, Y.Ji. Design of the SAML-based Security Service System.[M]. Application Research of Computers. 2004.11:1-4.

[3].D.M. Wen, A.D. Men, A.P. Wen. Design of Cross-domain Single Sign-on System Based on Cookie. Computer Knowledge and Technology. 2009.11:1-5:1-3.

[4]. Qing-Xiu Wu, un Ou, Li-Jun Wu, Xia Huang. Application of Cookie in PHP. Proceedings of 2010 International Conference on Future Information Technology and Computing (FITC 2010).2010.12:1-4.

[5]. Colin Boyd,Wenbo Mao.Single Sign-On Using Trusted Platforms[J].Information Security:6th International Conference,ISC 2003,Bristol,UK,Proceedings.October 1-3,2003.