

Re-evaluating Single Sign-On System Design Risks: An Activity Theoretic Approach

Manish Gupta, Kranti Banala, and Raj Sharman
School of Management, State University of New York at Buffalo
Amherst, New York, USA, 14260

Abstract—Single Sign-On (SSO) systems provide users the convenience of accessing multiple applications and systems while having to provide credentials only once. Organizations across industries have started to evaluate and deploy Single Sign-On systems in their environment. SSO systems provide a range of benefits including improved productivity, reduced complexity, improved user convenience, facilitated business and improved compliance to security policies. While SSO systems have shown to provide many economic benefits, there are inherent risks that arise from the fact that in SSO environment, only one password or one set of authentication factor is needed. This creates a situation typically understood as ‘single-point of failure’. In an event the SSO password is breached, all of the applications covered under SSO will be exposed to huge risks. We use activity theory principles to understand how applications should be categorized to design SSO systems. The research develops a process guided by activity theory to unravel some of the hidden design tenets that should guide SSO deployments.

I. INTRODUCTION

AS IT systems elevate their role from “supporting” to “enabling” business processes, end users, system designers, managers and technicians are coping with an increasing complexity of maintaining an ever growing portfolio of applications and ensuring their system security. Users typically have to sign-on to multiple systems, necessitating an equivalent number of sign-on dialogues, each of which may involve different usernames and authentication information [12]. As a result users resort to committing serious security flaws like writing down passwords on paper or using passwords that can be easily guessed. Single sign-on (SSO) system enables the user to use a single user-id and password pair to access all authorized computer resources in a distributed, multiplatform computing environment, without authenticating multiple times [1]. Increased security and compliance, improved user productivity and convenience and real cost savings are few motivations that drive SSO implementation [16]. According to a Gartner report Single Sign-On system can save up to \$300 per user per year which can account to huge amounts [5]. By the use of single sign-on, user identity is consolidated to a single digital identity and this helps reduce administrative burden and meet regulatory and compliance needs of the organization like

HIPAA, Sarbanes-Oxley act, UK data protection act, European Union Privacy Act etc. Using SSO, users need to manage only one set of authentication credentials in order to log into the services they subsequently use [32]. A single sign-on system should provide secure storage of user passwords, support for more than one user password, as well as support for multiple target logon methods [4]. After authenticating to the SSO system, when users access target systems, the SSO agent passes the appropriate target system's credentials to those systems and logs in the user with no additional action required on their part. SSO has been proposed as a solution to improve employee productivity, reduce information systems administrative costs and increase system security [11]. Organizations that have five or more heterogeneous Windows, Web and terminal-based applications that users must sign on to every day are “feeling the pain” of user complaints via high numbers of help desk calls, stand to gain the most from SSO tools [20]. Over the years, enterprise-class Single Sign-On products have matured and today, they provide value for enterprises with users who must sign on to multiple applications. Users can be prompted for passwords and password changes. Passwords can be created as random character strings by the SSO tool and can be made as strong as allowed by the target systems, and changed without placing burden on the users. Through 2009, a Global 2000 enterprise that purchases an enterprise Single Sign-On tool will continue to use it for five or more years (0.8 probability) [20].

Password related threats including social engineering are an undeniable and pervasive threat to the security of information systems of an organization due to its reliance on the social nature of human beings [14]. Activity theory offers the options for understanding use and system design for computer applications as well as other parts of the work activity is constantly reconstructed to meet the dynamic demands of any organization. An explicit awareness of these hidden trends may change our way of doing design [11]. Researchers [19, 21] have proposed AT-based methodologies for software development. Several other disciplines have used Activity Theory to understand their processes and constructs. It is evident from the literature review that role engineering is an increasingly critical and vital process at any organization from both functionality and

TABLE I
EXAMPLE TECHNICAL AND FUNCTIONAL
REQUIREMENTS FOR A SSO SYSTEM

REQUIREMENT	DESCRIPTION
Single sign-on	One password to access multiple applications
Authenticator choice	Choice of multiple back-end password stores
Mobility support	Support for roaming profiles
Workstation sharing	Multiple users can share the same workstation
Automated password change	Product changes the password based on application level security
Event Logging	Automatically log events such as logons, password changes
Auto Prompt	Prompts for a new password-protected application
Common passwords	Multiple applications can share the same password
Central administration	All configurations and settings are centrally manageable
Automatic Backup/Restore	Automatically back up user credentials to a remote location
Customization	Modifiable templates for corporate policy
Secure architecture	The agent is designed to be highly secure and tamper-resistant.

security viewpoints. It is also revealed through literature review that activity theory has not been used, thus far, to analyze and understand role-engineering process to design effective and secure roles within an organization. With Activity theory's immense benefits, we analyze role-engineering process and principles to unravel some of the human and social facets that are not evident from traditional role engineering frameworks. The paper is organized as follows: In next section, we present background and preliminaries on different concepts used in the research. Section 3 presents activity theory guided evaluation of risks on SSO systems and applications. Section 4 presents an illustrative case study to show how activity has been used to uncover some risks hidden in user assignments to role in the context of two applications.

II. BACKGROUND AND PRELIMINARIES

A. Single Sign On

A single sign-on (SSO) system provides mechanisms and supporting technologies to support different authentication mechanisms including storing passwords and other credentials [4]. SSO system should have support for varying types of authentication mechanisms from simple passwords to complex biometrics. The SSO should be flexible enough to accommodate requirements of infrastructure and business based on agreed upon trust requirements, Authentication schemes (e.g., those based on passwords, certificates, biometric techniques, smart cards, etc.) are employed depending on the trust-level requirement(s) of an information resource (or information resources) to be accessed [31]. SSO has been shown to improve security, usability and infrastructure maintenance while improving the end user's convenience and trust [32]. Empowering the user with a Single Sign-On capability has multifold benefits. It greatly improves the user experience and relieves the user from the burden of remembering multiple user-id and

password pairs. Enterprises hope that Single Sign-On protocols will significantly decrease customer-care costs related to forgotten passwords and increase e-commerce transactions by enhancing the user experience [24]. "On the administrative side, help desk costs are noticeably reduced and security improved, as users are not tempted to 'store' multiple passwords in written form" [1]. With identity management systems, new user accounts can be setup and accounts of those leaving the organization can be deleted in a few minutes resulting in a huge productivity boost. SSO significantly improves convenience and ease of use of different kinds of systems [10]. According to Forrester Research, as much as 30% of helpdesk time is spent in dealing with password-reset issues [30]. With the cost of a single help desk call estimated at £13 to £20, these password problems can quickly add up to hundreds of thousands of pounds per year, for even mid-sized companies. When assessed against the cost of implementing and maintaining the Single Sign-On, the return on investment becomes apparent [28].

There are a plethora of SSO products available in the market. Oracles, IBM Tivoli access manager, Sun Microsystems Open SSO, Novell, Courion, CA are few of the leading SSO product vendors [4, 23]. Most of the identity management products fall into the four major areas- Identity intelligence, Identity administration, Identity verification and Access management. SSO systems perform authentication and authorization functions in the identity management systems. Most of the commercial identity management vendors contain an array of products for user provisioning, self administration and single sign-on. Oracle, IBM Tivoli, Sun, Novell and CA qualify under single provider portfolios as they have products required for a complete identity management system [23]. Table I presents some of most common requirements for a SSO System. Most of the identity management vendors charge either per user enrolled under the system or have a single product cost. Single Sign-On solutions find their application in wide variety of domains like healthcare, education, retail, finance, banking etc. Identity management as a service (also by *Symplified* [27]) is one of the latest developments in this arena. These activities involve huge costs and put IdM options out of the financial reach of small and midsized businesses [15].

B. SSO Standards and Architectures

Federated identity management is a version of single sign-on that spans across different organizational boundaries. It is enabled by sharing common authorization and authentication data between these organizations [9]. Remote systems control access to resources based on the roles assigned to the person trying to access them. Liberty Alliance project and *Incommon* Federation establish standards, specifications and policies for identity assurance and identity governance framework like SAML/2.0 – standardized XML documents for sharing identities, ITU-T X.509v3 – standardized digital certificate etc [17, 30]. Detailed information about

TABLE II
COMPARISON OF REQUIREMENTS ENGINEERING APPROACHES
(ADAPTED FROM [3])

Dimension	Focus	Goal Oriented	Function Based	Adapted Activity Theory
Goal	Intention	X		X
People	Individual (role)	X	X	X
	Community (group, role)			X
Process	Division of Labor (rule, task assignment)			X
	Activity and Activity Structure	X	X	X
	Object Hierarchy	X	X	X
Technology	Instrument (form)			X
Environment	Context Awareness	X	X	X
	Social Issues			X
	Environment Issues			X
Interaction	Contradictions	X		X

standards, policies and specifications can be found at Liberty Alliance specifications [17].

Single sign-on systems rely on other infrastructure like the authentication system, identity management/ registration, web servers etc [9]. Single Sign-On can be realized in many ways either by using browser cookies, HTTP redirects, user access tokens (Kerberos, SAML etc..) ticket granting systems, Digital certificates, by having a central authentication and authorization server, LDAP and active directories [22] Biometrics, smart cards or USB tokens with certificates/ login passwords, Radio badges using RFID technology can be integrated with SSO to provide multi-factor authentication [18].

C. Activity Theory

Activity Theory (AT) seeks to explain social and cultural work practices by relating them to the cultural and historic context in which the work activity, which is the basic unit for analysis, is taking place [2]. AT gives us guidelines and concepts to analyze “the actions and interactions with artifacts within a historical and cultural context” [25].

Activity theory offers the options for understanding use of system design for computer applications as well as other parts of the work activity that are constantly reconstructed to meet the dynamic demands of any organization. An explicit awareness of these hidden trends may change our way of doing design [11]. Researchers [19, 21] have proposed AT-based methodologies for software development. Several other disciplines have used Activity Theory to understand their processes and constructs. However, due to origins and wide applications of Activity theory in social and psychological areas, it is vastly underutilized in information security domain.

Activity theory has evolved through two generations of research [7] since introduced by several Russian scholars using Marx’s political theory [26]. The most common, by Engestrom [6, 7, 8] (Figure 1), is concerned with the process of social transformation and incorporates the structure of the

social world and aims to understand dialogues, multiple perspectives and networks of interacting activity systems.

In Engestrom’s original work [6], activity systems included subject, tool, object, rules, community, distribution of labor, and outcomes (as shown in Figure 2). *Subjects* are participants of activity and tools are resources that the subjects use to obtain the object or the goal. *Rules* are guidelines or restrictions that are considered during interactions and social dynamics amongst subjects and objects. The *community* is a group that subjects belong to and *division of labor* is the shared responsibilities determined by the community. Finally, the *outcome* is the consequence that the subject faces as a result of the activity. Activity systems analysis was developed to explore and document the sources of tensions in human individual or collective activities.

III. ACTIVITY THEORY GUIDED EVALUATION OF SSO RISKS

As we discussed above, there are several SSO benefits to individuals and organizations. However, there are various risks that arise from the fact that different applications share different characteristics and bringing them (a set of

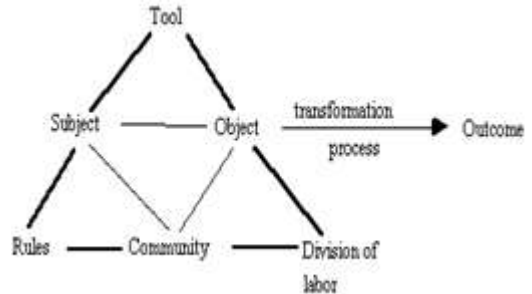


Fig. 1. Engestrom’s extended Activity System

applications) without considering those characteristics will expose the environment to severe threats. Activity theory (AT) is widely used for different applications in IT ranging from software development to secure role development [13]. Because of the unique insights and perspectives that AT offers in understanding not only technical and systemic features but also dynamic interactions amongst individuals and communities, it is a very apt approach for understanding and revealing some of the risks arising from SSO. Some of these critical risks cannot be brought under consideration for SSO design without the use of AT. Table II (adapted from [3]) shows different dimensions that are covered from under various approaches to under system behavior. As is evident from the table, use of AT brings more dimensions under the lens for evaluation of systems. More specifically, the ones that pertain most towards AT’s contributions are People, technology, environment and interaction dimensions.

Figure 2 shows adapted AT with different risk factors from application standpoint and from SSO system standpoint. In Figure 2, (A) denotes the risk factor is application specific and (S) denotes SSO denotes system specific. The figure uses AT as base and then represents

various risk issues and factors for each of the system components of AT. The risk factors are presented next to each system component, in italics within parenthesis. Using Figure 2 as guide we can analyze each application and SSO

system to lead to a risk-managed selection process. The end result is selection of applications for each SSO system given application characteristics and SSO system specific details.

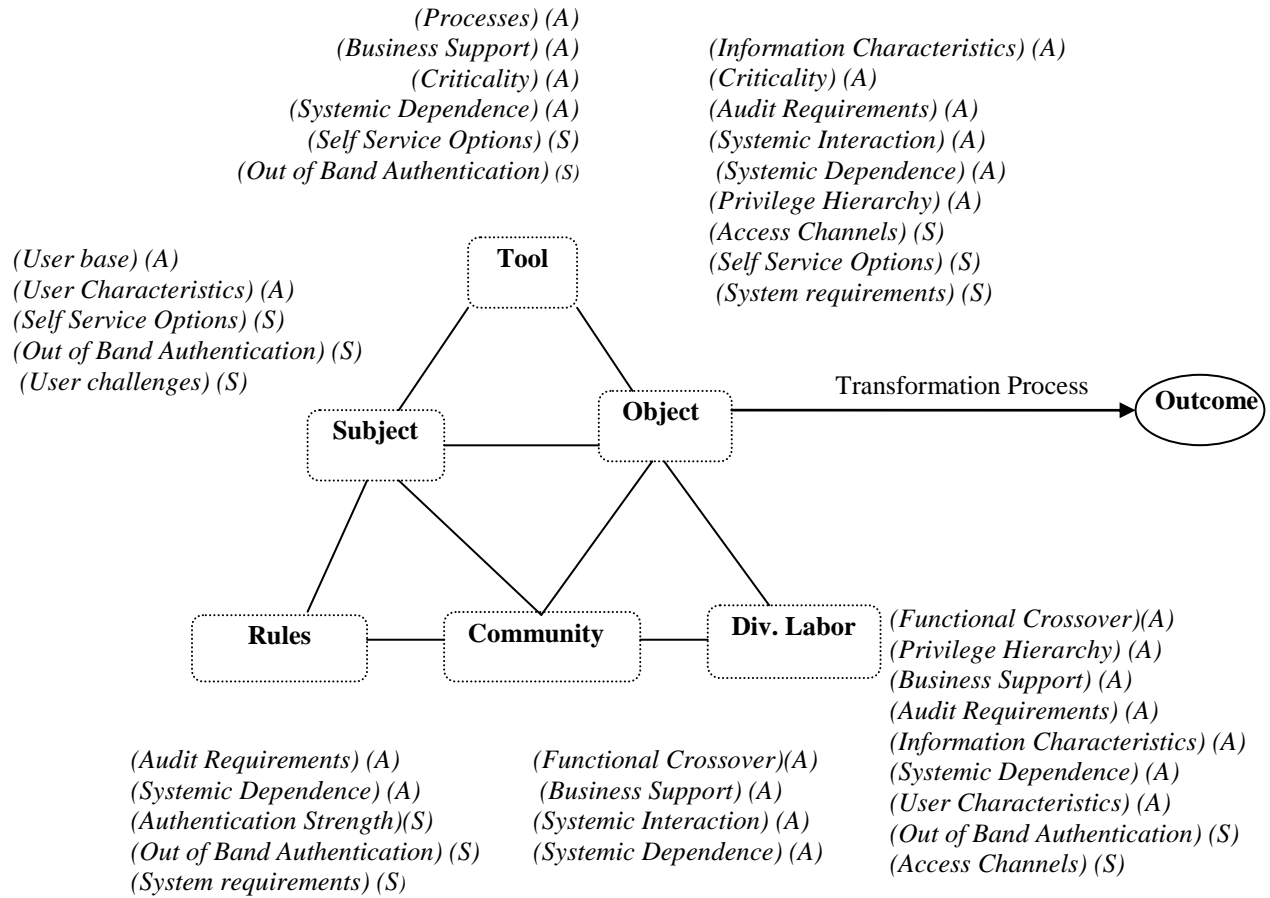


Fig. 2: Activity Theory Components with SSO and Application Risk Factors

A. SSO System Specific Risks

TABLE III
SSO SPECIFIC RISKS AND AT COMPONENTS

SSO Specific Risk	AT Components	Description
Authentication Strength	Rules	Strength of authentication factor such as simple passwords or complex passwords, password policies, number of factors, hardware or token.
Access Channels	Object, Div. Of Labor	Number of channels to access the user interface for SSO system
Self Service Options	Tool, Object, Subject	Options where users can reset their own passwords, modify profile or request additional accesses.
Out of Band Authentication	Tool, Div. Of Labor, Rules, Subject	Bypassing SSO system to access an application and having credentials reset over phone.
System Requirements	Object, Rules	Technical and functional requirements of the SSO system. Some common ones are listed in Table 1.
User Challenges	Subject	Ease of use of the SSO system and HCI issues.

There are several risks that emanate from various aspects of SSO systems including technology, environment, deployment architecture, people and processes (See Table III). Weak passwords in a single sign-on environment have been widely reported as one of the greater risks of SSO systems. Compromise of password can result in unauthorized access to information that can be potentially confidential and the incident can be detrimental to the organization. In the same light, applications processing sensitive information should implement stronger password policies or use multi-factor authentication such as token or biometrics so that the risk of SSO being a central point of attack is mitigated. There are several threats to privacy as well in an environment where a user's identity is shared amongst teams. Towards usability of SSO systems, user interface is considered to be one of most important and inadequately addressed component. Many of the phishing attacks are launched due to weakness in secure-proofing the user interface without putting undue burden on users. Many of the SSO systems that enable SSO amongst web-based applications heavily rely on cookies. There have been several reported vulnerabilities and threat specific to browsers and other technological components that bear severe consequences. Organizations should consider not only acquisition costs while selecting an SSO system, but also ongoing and maintenance costs. For example, while

selecting password based authentication which may appear simple and cost-effective choice, it should be considered that on an average, it costs about \$70 for each password reset. SSO helps lower these password-reset costs as there would be only one password for multiple applications which makes it easier for users to remember. The technical environment where SSO is going to be implemented should also be factored in selecting an SSO solution. Table III shows some of the most common SSO specific risks and how analyzing those using listed AT component(s) help understand the nature of risks and suggests ways to mitigate them. In absolute terms, authentication mechanism that uses more than one factor is considered stronger than a single factor (such as password), however careful risk assessment should be made to introduce additional factors which will presumably incur higher costs. Identified risks should justify higher costs and since the SSO can be used for multiple applications, this should drive motivation for multi-factor SSO. However, it should be considered that only applications that process high risk transactions such as financial systems or systems containing confidential information such as SSN should be included in multi-factor SSO.

B. Application Specific Risks

There are several characteristics of an application that give unique risks to authentication and authorization process for the application. Applications serve one or more business objectives for an organization, which can aid decision-makers in understanding the criticality of the application to the business. With that in light, both security and usability has to be balanced while selecting an authentication solution. Besides, while using SSO for authentication to that application, the authentication is delegated to the SSO system. Also, based on the user characteristics and nature of activities performed on the application, organizations must decide if additional layers of authentication are required and then an SSO system meeting their requirements should be deployed. More often than not, businesses will have applications sharing some of their prime features, which will further help organizations plan likewise considering the economies of scale that SSO systems provide. For example, enterprise applications that handle employees' sensitive information such as payroll, HR, medical and insurance records should have stronger authentication mechanisms. Besides information characteristics, user characteristics are also equally important. For example, a financial institution may choose to use multiple factor

TABLE IV
APPLICATION SPECIFIC RISKS AND AT COMPONENTS

Application Specific Risk	AT Components	Description
Business Support	Tool, Div of Labor, Community	Business functions and processes supported or enabled by the application.
Processes	Tool	Processes automated or facilitated by the application, usually electronic forms, work flows and transactions.
Privilege Hierarchy	Object, Div of Labor	Different roles within application that ascertains access to and responsibility for different classifications of information.
Criticality	Tool, Object	How critical is the application for the specific business function? Usually business continuity exercises determine recovery time objectives that assign criticality to applications.
Systemic Interaction	Object, Community	Extent of information exchange and communications with other systems, processes and people.
Functional Crossover	Community, Div. Of Labor	How many different functional units from the organizations perform their duties on the application?
Systemic Dependence	Tool, Object, Div of Labor, Community, Rules	Does this application relies on other application(s) for its functioning (processing, storing and transmitting information) or do other applications rely on this application for their successful operation?
Audit Requirements	Object, Div of Labor, Rules	Are there specific audit and compliance requirements or specific guidelines (or recommendations) for the application such as SOX, GLB, FFIEC etc.
Information Characteristics	Object, Div of Labor	What is classification of data that passes through the application? Which roles have access to data from which classification level?
User Base	Subject	How many users access the application and what is the frequency of usage?
User Characteristics	Subject, Div of Labor	What are the characteristics of users that access the application? Customers or employees? Important customers? Administrators?

authentication-based SSO system for commercial customers who execute financial transactions in large monetary terms, while retaining password-based authentication for retail customers (though enforcing stronger password policies). SSO systems provide organizations the agility to add more similar applications to SSO, which is scalable by design. This also creates a portfolio of applications for users for which they will have to authenticate only once. For many applications, multi-factor authentication options can be selected depending on some of the application-specific characteristics such as businesses that the application supports, privilege hierarchy, functional crossover, and audit and compliance requirements. There are four levels of authentication strength as proposed by NIST standard: SP 800-63.

The factors that NIST standard suggests to account for when selecting an authentication solution include use of tokens for identity verification, identity proofing during on-boarding and de-provisioning, remote authentication mechanisms and security information assertion. Some of the application specific risks and the way in which AT components help provide insight into their working is provided in Table IV. Description of each risk is also provided in the table to illustrate the contribution of the risk consideration. Analyzing application characteristics using AT and its components help managers and architects on making judicious decisions in selecting the most appropriate SSO system based on risk assessment, including cost.

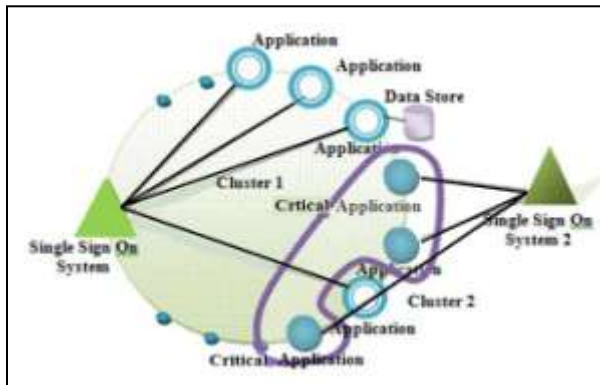


Fig. 3. Systemic View of interaction of SSO with applications

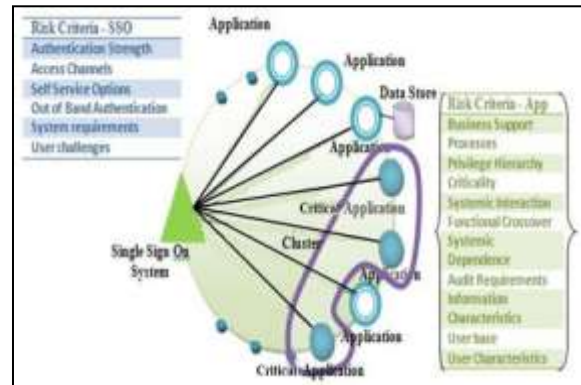


Fig. 4. Decoupling Applications to form separate SSOs

C. SSO Design: Application Selection

SSO systems provide a mechanism for users to authenticate only once and access a portfolio of applications. Figure 3 illustrates how SSO interacts with different applications for verifying the user's authentication and authorization statuses. The Figure shows criteria for risk assessment of each application to decide whether it should be brought under a specific SSO system. Also shown are some factors to consider when designing SSO system (the list on the top left corner). Most applications need a data store, temporal and persistent, for its operation. The data accessed and processed by application supports business goals and objectives. The more critical the data is for the business, the stronger the authentication that is required to access it. In the same vein, organizations should deploy more than one SSO system while putting applications that share risk characteristics under a specific SSO system. The user base and characteristics can also impose severe restrictions on selection of a specific SSO system. For example, applications used by customers who are available over public domain such as Internet should go through a rigorous risk assessment to select best risk-managed SSO solution. At the same time different SSO systems are built for varying loads of performance, so user base is also an important factor in its selection.

Similarly there are several other risk factors (Table IV) associated with the application that should be used to select the SSO system. AT concepts and component(s) shed light into each risk factor from a more comprehensive view while addressing some of the most ignored risk concerns such as the role of community in an organization and dynamics of interaction between people. For instance, by use of AT component, division of labor, we can delineate risks from social engineering, collusion and unauthorized escalated privileges. While most of the risk assessment methodologies only factor in technological, business and procedural aspects of business functions, AT, while covering those also incorporates community, division of labor, tools and rules. Use of these additional lenses in risk assessment in conjunction with traditional ones provides a holistic and more accurate posture of risks. Sometimes it makes better decision to decouple applications from under an existing SSO solution after performing an AT based risk assessment of applications. In Figure 4, we show how we can break an SSO system into 2, based on criteria presented above. For example, initially one SSO system was one-factor (password) based (SSO system specifics).

However, on review of applications under the system, it was unraveled that some of the applications (solid circles in Figure 4) process sensitive/private corporate information (based on internal organizational data classification). So it is imperative that a 2-factor authentication should be used for the SSO system (which is shown as SSO 2 in the figure).

IV. ILLUSTRATIVE EXAMPLE: A CASE STUDY ON APPLICATION RISKS

Next we present an example where Activity Theory is used to uncover some of the threats inherent in traditional role engineering process. These same issues surrounding users, communities, interactions and access channels as they relate to application specific risks apply to our discussions on SSO system design. Gupta [13] in an exploratory case study illustrates how role engineering is performed in organizations and how the activity theory principles and concepts can aid managers and designers engineer effective and secure roles at a mid-sized US financial institution. This paper leverages the work of Gupta [13] in the development of the constructs for this research. There were 6 people at the financial institution who aided us understand their role engineering process. Researchers met them to 1) educate them on activity theory and theory behind RBAC though they were

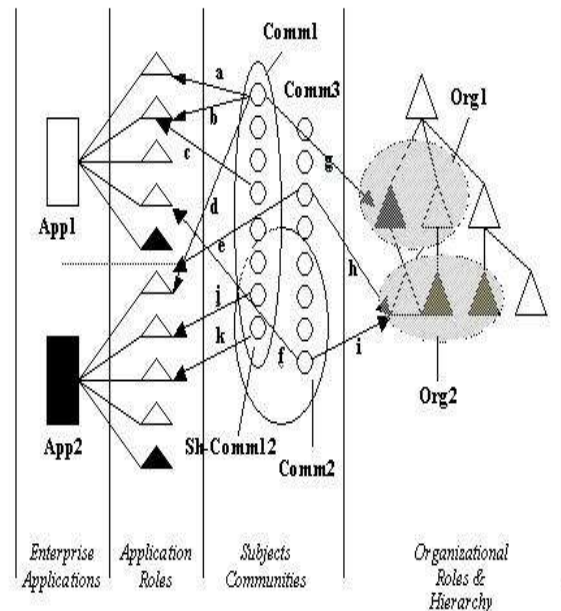


Fig. 5. Application Specific Risks as illustrated through example of two real world applications

not aware of terms used for formal RBAC representations. Based on discussions with the managers at the financial institution, we collectively came up with a situation where role engineering process can utilize activity theory principles to better understand the implications of role engineering process.

Also, this will aid managers understand and unravel social and community-based facets of environment that are ignored in traditional role engineering process. In the Figure 5, App1 and App2 are two applications for which the roles are to be engineered.

The triangles in the section right of applications are the application roles. In the subjects' section, circles are users and ovals represent user communities (Comm1-3). Shared communities are the communities that belong to the same users (Sh-Comm12). Only one such community is shown in the figure to keep the case simple. Right most section shows organizational roles within the enterprise (Org1 and Org2). These can denote same functional or positional roles in an organization. The small cased letters in Figure 5 denote assignments of application roles and organizational roles to users. Table V shows assignments with the exact mappings of users, communities, roles and applications. Third column (Emphasis) presents the linkage that is most vital in understanding applicability of activity theory

TABLE V
USER ASSIGNMENT TO ROLES IN APPLICATIONS

Assignments	Mappings	Emphasis
a	User(Comm1) ₁ - Role(App1) ₁	User-role
b	User(Comm1) ₁ - Role(App1) ₂	User-role
c	User(Comm1) ₂ - Role(App1) ₂	User-role
d	User(Comm1) ₁ - Role(App2) ₁	User-role
e	User(Comm3) ₁ - Role(App2) ₁	User-role
f	User(Comm2) ₁ - Role(App1) ₁	User-role
g	User(Comm1) ₁ -Org ₁	User-org
h	User(Comm2) ₁ -Org ₂	User-org
i	User(Comm2) ₁ -Org ₂	User-org
j	User(Sh-Comm12) ₁ - Role(App2) ₂	User-Community
k	User(Sh-Comm12) ₁ - Role(App2) ₃	User-Community

to role engineering for this scenario (Figure 5). Table VI below represents different interactions amongst the user assignments (column 2) and how they result in various threats to specific component of RBAC (column 3). Column 4 in Table VI represents unit of focus or the component that is most likely hit due to that particular interaction of assignments. The last column provides description of the interaction.

TABLE VI
APPLICATION RISKS FROM INTERACTION ASSIGNMENTS TO ROLES

SCENARIO	INTERACTION	RESULT	RISK FOCUS	DESCRIPTION
I	a + b	Multiple role	User	One user having more than 1 role in the same application
II	b+ c	Shared role	Role	Same application role assigned to multiple users from the same non-shared community
III	a+ d	Cross-application roles	User	One user assigned roles from multiple applications
IV	a+ c	Community Role	Community	Multiple users from the same community having access to multiple roles in same application
V	d + e	Across-community role	Role	Users from multiple communities having access to the same role
VI	j + k	Shared Community application	Application	Users from shared communities access multiple roles from same application
VII	e + h & d + g	Across Org. role	Role	Same application role assigned to users from multiple organizational role
VIII	a + b + g	Across application role	Application	Users from same organizational role have access to different application roles
IX	f + I & e + h	Across application and org role	Applications	Users from same organizational role have access to different roles from different applications

TABLE VII
ACTIVITY THEORY PRINCIPLES LEGEND

Abbrev.	AT System Elements (Engestrom's System Model, 1997)
Tool	To
Subject	S
Object	Ob
Rules	R
Community	C
Division of Labor	D
Transformation Process	Tr
Outcome	Ot

Based on discussions with the managers at the financial institution where this case study was carried out and analyses of the case study dynamics, managers consented on applicability of different principles and concepts of Activity Theory, as they would apply to role engineering process. Tables VII and VIII present abbreviations for RBAC and AT components used in Table IX. The threat scenario number from Table VI is presented in column 1 in Table IX. Table IX shows which component and consideration in the role engineering process (column 2) would be affected by Threat Scenario (Table VI). Next column in any row (read threat scenario) of the table present Activity theory principles (Engestrom's System Model, 1997) that should be used to further unravel any interactions that may arise inefficient and insecure roles.

For example, row 3 of the table represents Threat Scenario III (arising due to Cross-application roles). For this scenario user-assignment to roles, separation of duties and application role hierarchy are the most important components of RBAC that should be

TABLE VIII
RBAC COMPONENTS LEGEND

Abbrev.	Role Engineering Components
UA	User Assignment
PA	Permission Assignment
SOD	Separation of Duties Constraints
ARH	Application Role Hierarchy
ORH	Organizational Role Hierarchy
S	Sessions

closely scrutinized. At the same time, managers at the financial institutions, feel that Internalization/externalization and Object-orientedness are the principles from Kaptelinin and Nardi [33] Activity Theory Artifact that can aid in further understanding of the social dynamics within organization that can uncover some vital scenarios that should be

accounted for in role engineering. Similarly, last column shows Activity Theory System Elements, consideration of which will significantly mitigate the risks of insecure role creation by analyzing the context of the users and roles (both application and organizational).

V. DISCUSSION

In the paper we discussed concepts and

TABLE IX
APPLICATION OF ACTIVITY THEORY TO ROLE ENGINEERING DESIGN

Scenario	Role Engineering Components	Activity Theory System Elements (Engestrom's System Model, 1997)
I	UA, PA, SOD	S, To, D, Tr
II	SOD, ARH	To, S, Ob, C, D
III	SOD, UA, ARH	Ob, S, D, Tr
IV	UA, ARH, ORH	S, D, Tr, C
V	UA, ORH	S, Ob, T, C, D
VI	ARH, ORH, PA	S, Ob, T, C, D
VII	ORH, PA	S, T, C, Tr, To
VIII	ARH, ORH, UA	S, T, C, Ot
IX	ARH, ORH, UA	S, Ob, T, C, D, Ot

frameworks of single-sign-on systems and activity theory. Some common drawbacks of traditional SSO system design were presented. In light of unique insights that activity theory can provide in the SSO design process while considering specific risks from two perspectives (SSO system-specific and application-specific), we analyzed how different risk considerations and activity theory can be brought together for secure SSO system design. To illustrate relevance and utility of using AT in SSO design, we presented a case study where AT was used to help design secure role engineering process. The basic concepts and workings, we believe, will remain similar for SSO systems as well. The paper's main contributions are 1) application of activity theory to help identify risks and 2) identification of different types of risks that SSO system's deployment introduces in an environment (Table III and Table IV) and how they relate to different Activity theory principles. We are in process of gathering information on SSO design practices utilized by as many as fifteen different organizations representing various industries. Future work on this study is to show how the organizations can use activity theory to improve their SSO systems to mitigate risks arising from the deployment of SSO systems. Investigation will entail showing a system designed without considering AT principles and then analyzing how AT can be used to reveal risks.

REFERENCES

- [1]. Anchan, D. & Pegah, M. (2003) Regaining single sign-on taming the beast, September 2003, Proceedings of the 31st annual ACM SIGUCCS conference on User services SIGUCCS '03
- [2]. Bertelsen and Bodker. (2003). Activity Theory, HCI Models, Theories, & Frameworks: Toward a Multidisciplinary Science. Carroll, J (ed)
- [3]. Chen, R., Sharman, R., Chakravarti, N., Rao, H.R. and Upadhyaya, S. (2008). Emergency Response Information System Interoperability: Development of Chemical Incident Response Data Model, Journal of the Association of Information Systems, Volume 9, Issue 3/4, pp. 200-230, Special Issue 2008
- [4]. Cohen, R. J., Forsberg, R. A., Kallfelz, P. A., Meckstroth, J. R., Pascoe, C. J., Snow-Weaver, A. L. (1998) Coordinating user target logons in a single sign-on (SSO) environment, Patent number: 6178511, Filing date: Apr 30, 1998, Issue date: Jan 23, 2001, Assignee: International Business Machines Corporation
- [5]. Connolly, P. (2000, september 29). *Single Sign-on dangles prospect of lower help desk costs*. Retrieved march 21, 2009, from infoworld: <http://www.infoworld.com/articles/es/xml/00/10/02/001002esnso.html>
- [6]. Engestrom, Y. (1987). Learning by expanding: An activity-theoretical approach to developmental research. Helsinki: Orienta-Konsultit Oy.
- [7]. Engestrom, Y. (1999). Activity theory and individual and social transformation. In Y. Engestrom, R. Miettinen, & R.-L. Punamaki (Eds.), Perspectives on activity theory (pp. 19–38). New York: Cambridge University Press.
- [8]. Engestrom, Y. and Miettinen, R. (1999). Introduction, in: Y. Engestr "om, R. Miettinen, R. Punam" aki (Eds.), Perspectives on Activity Theory, Cambridge University Press, Cambridge, 1999, pp. 1–16.
- [9]. Federated Identity Management. (2009). Retrieved March 22, 2009, from Tech-faq: <http://www.tech-faq.com/federated-identity-management.shtml>
- [10]. Fleury, T., Basney J., and Welch, V. (2006) Single sign-on for java web start applications using myproxy, Workshop On Secure Web Services archive, Proceedings of the 3rd ACM workshop on Secure web services, Alexandria, Virginia, USA, SESSION: Security architecture, pp. 95 – 102, 2006
- [11]. Gordon, T. (2004, Jan 24). *Quantifiable Benefits of Implementing Identity management systems*. Retrieved 3/22/09, from University of Salford: www.ils.salford.ac.uk/about/projects/idm/docs/Stageand%203/ISDSQuantBenefits.pdf
- [12]. Group, O. (2001, August 1). *Introduction to Single Sign-On*. Retrieved March 22, 2009, from The Open Group : http://www.opengroup.org/security/sso_intro.htm
- [13]. Gupta, M. (2008). Activity Theory Guided Role Engineering. Proceedings of 14th Americas Conference on Information Systems (AMCIS 2008), Toronto, Canada, August 14-17, 2008.
- [14]. Gupta, M. and Sharman, Raj. (2006) "Social Network Theoretic Framework for Organizational Social Engineering Susceptibility Index", Proc. of 12th Americas Conference on Information Systems, Acapulco, Mexico, Aug 4-6, 2006.
- [15]. Hulme, G. (2008, july 7). *Identity management as a service*. Retrieved march 22, 2009, from Information week: http://www.informationweek.com/blog/main/archives/2008/07/identity_manage.html
- [16]. Imprivata. (n.d.). *Benefits of Single Sign on*. Retrieved March 22, 2009, from Imprivata: <http://www.imprivata.com/contentmgr/showdetails.php?id=1170>
- [17]. *Incommon policies and practices*. (n.d.). Retrieved March 22, 2009, from Incommon federation: <http://www.inco-mmunionfederation.org/policies.cfm>
- [18]. Jerphanion, L. d. (2008, october). Enterprise single sign on. *evidian white paper* . EvidianInc. *Liberty Alliance specifications*. (n.d.). Retrieved March 22, 2009, from project liberty: http://www.projectliberty.org/liberty/resource_center/specifications
- [19]. Korpela, M., Soriyan, H. A. and Olufokunbi, K. C. (2000). "Activity Analysis as a Method for Information System Development." Scandinavian Journal of Information Systems(12): 191-210.
- [20]. Kreizman, Gregg. (2006) "Enterprise Single Sign-On Provides Value for Complex Environments", Gartner Research Publication, ID Number: G00138179, 22 March 2006.
- [21]. Mwanza, D. (2001). Where theory meets practice: A case for an Activity Theory based methodology to guide computer system design. INTERACT'2001, Oxford, UK, IOS Press.
- [22]. Orrell, D. & Edusery. (2005). Authentication systems and single sign on. *EuroCAMP*. Porto, portugal.
- [23]. Perkins, E. P. C. (2008). *Magic Quadrant for User Provisioning*. Gartner.
- [24]. Pfiztmann, B., Waidner, M. (2003) Analysis of liberty single-sign-on with enabled clients, IBM Zurich Res. Lab., Ruschlikon, Switzerland; IEEE Internet Computing, Nov.-Dec. 2003, Volume: 7, Issue: 6, pp. 38- 44
- [25]. Rogers, Y. (2004) New theoretical approaches for HCI. ARIST: Annual Review of Information Science and Tech., 38.
- [26]. Stetsenko, A. (2005). Activity as object-related: Resolving the dichotomy of individual and collective planes of activity. *Mind, Culture, and Activity*, 12(1), 70–88.
- [27]. Symplified. (n.d.). Retrieved March 22, 2009, from symplified: <http://www.symplified.com/>
- [28]. Ting, D. (2005) Biometrics and single sign-on, *Biometric Technology Today*, Volume 13, Issue8, September-2005, Pages 8-9.
- [29]. Wiki.ihe. (2007, october 8). *Federated Identity Management Profile*. Retrieved March 22, 2009, from Wiki.ihe: http://wiki.ihe.net/index.php?title=FEDidMGT_-_Federated_Identity_Management_Profile
- [30]. Woo, Renee. (2001). Password Reset Software Can Reduce Help Desk Costs, Forrester Research IdeaByte, March 30, 2001. Retrieved from <http://www.forrester.com/Research/LegacyIT/Excerpt/0,7208,18845,00.html>
- [31]. Wood, D., Weschler, P., Norton, D., Ferris, C., Wilson, Y., Soley, W. (2003) Log-on service providing credential level change without loss of session continuity, Patent number: 6609198, Filing date: Aug 5, 1999, Issue date: Aug 19, 2003, Assignee: Sun Microsystems, Inc.
- [32]. Zhao, G., Zheng, D., Chen, K. (2004) Design of single sign-on, IEEE International Conference on E-Commerce Technology for Dynamic E-Business, 2004, pp. 253 – 256
- [33]. Kaptelinin and Nardi B.A. (1997). Activity Theory: Basic Concepts and Applications, CHI tutorial, 1997