SECURE KNOWLEDGE MANAGEMENT: CONFIDENTIALITY, TRUST AND PRIVACY

Tubagus Mohammad Akhriza

Abstrack : Knowledge is one of the most important assets for an organization or a company. The knowledge in a company is the intellectual capital which could make the company keeps alive and competitive. As an important asset, the company's knowledge should be managed securely. In this paper, the author limits the discussion of the aspect of secure knowledge management to confidentiality, trust and privacy. For confidentiality, author discusses Role Based Access Control (RBAC) and Usage Control (UCON). Then we move to discuss the trust management and negotiation and stress the topic to trust negotiation (TN). Finally the author discusses about the privacy management. The conclusion is because the knowledge management is an important asset of organization as the intellectual capital, so security has to be incorporated into the knowledge-management lifecycle especially on integrating security strategy with knowledge management and business strategies of an organization.

I. INTRODUCTION

1.1. The Company's Knowledge

During the operational and also in order to keep living continuity, of course a company has built a lot of knowledge such as:

- Short and long term strategies
- Trade secret and plan.
- The manual book of employee recruitment
- Employee's psychology examination result
- Company's financial assessment report
- Company's information technology infrastructure.
- Etc.

1.2. The Secure Knowledge Management

Since we admit that the company's knowledge is a 'company's software or operating system' as the intellectual capital to keep company in high competitive value so we need to manage the knowledge.

We need to secure the knowledge management so only who has the authority can access the knowledge.

So even maybe if there is an employee doesn't work any longer in a company, we can still keep his/her knowledge in the company's knowledge base and make sure that he/she couldn't access the company's knowledge base management system.

II. THE ASPECTS OF SECURE KNOWLEDGE MANAGEMENT

In order to secure knowledge management, we need to conduct some computer security aspects. There are five aspects involve in this management as shown in Figure. 1

A lecture of STMIK Pradnya Paramita Malang

Studying for PhD Degree in the school of information scurity engineering shanghai Jiao Tong University P.R . of China

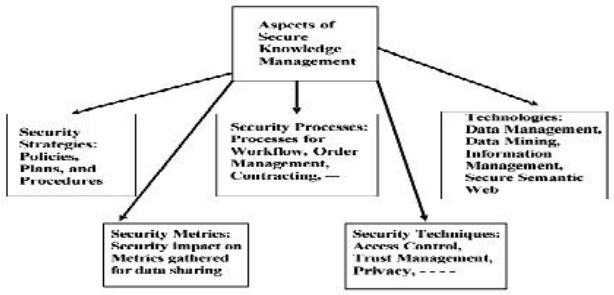

Figure 1. Aspects of
Secure Knowledge Management

1. Security strategies

Security strategies for knowledge management include the policies and procedures that an organization sets in place for secure data and information sharing as well as protecting the intellectual property.

Some of the questions that need to be answered include how often should knowledge be collected? How often should the organization conduct audit strategies? What are the protection measures that need to be enforced for secure knowledge sharing? Secure knowledge-management strategies should be tightly integrated with business strategies.

That is, if by enforcing intellectual-property protection the organization is going to be unprofitable, then the organization has to rethink its secure knowledge-management strategy.

2. Security processes

Secure processes for knowledge management include secure workflow processes as well as secure processes for contracting, purchasing, and order management. Security has to be incorporated into the business processes for workflow, contracting, and purchasing. For example, only users with certain credentials can carry out various knowledge-management processes.

3. Security metrics

Metrics for secure knowledge management should focus on the impact of security on knowledge-management metrics. Some examples of knowledge-management metrics include the number of documents published, number of conferences attended, or the number of patents obtained.

When security is incorporated, then the number of documents published may decrease as some of the documents may be classified. Organizations should carry out experiments determining the impact of security on the metrics gathered.

4. Security techniques

Security techniques include access control, UCON, trust management, as well as privacy control. These techniques are enforced at all stages of knowledge-management processes.

5. Supporting technologies

Secure knowledge-management technologies include data mining, the semantic web, as well as technologies for data and information management. The component technologies have to be secure if we are to ensure secure knowledge management.

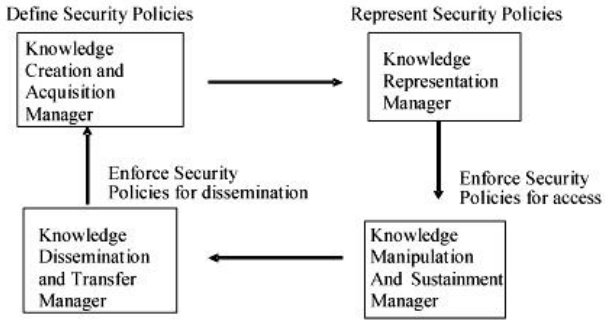

Figure 2. Secure Knowledge Management Architecture

Figure.
2 illustrates architecture for secure knowledge management. The components of the architecture are:

1. A secure knowledge-creation manager

The secure knowledge-creation task includes creating knowledge as well as specifying security policies enforced based on the knowledge.

2. A secure knowledge-representation manager

Secure knowledge representation tasks include representing the knowledge as well as policies in a machine-understandable format. Knowledge representation languages such as rules and frames as well as some of the more recent semantic-web languages such as resource descriptive framework (RDF) and ontology languages are appropriate for knowledge and policy representation.

3. A secure knowledge manipulation and sustainment manager

Secure knowledge-manipulation tasks include querying and updating the knowledge base. In addition, the knowledge gained has to be sustained as long as possible. Various processes have to be in place to sustain the knowledge securely.

4. A secure knowledge dissemination and transfer manager.

Secure knowledge dissemination and transfer task includes disseminating and transferring the knowledge to authorized individuals.

III. STRENGTHEN CONFIDENTIALITY

One of the aspects in secure knowledge management is the security techniques especially to ensure the confidentiality. We could use simple read-write policies, role based policies and UCON policies. In this paper the authors discuss RBAC and UCON.

3.1. RBAC for Knowledge Management

The simple way to describe Role Based Access Control (RBAC) is that the users and permissions are assigned to roles. User acquires permission indirectly via roles.

There are two major elaborations of the simple RBAC concept.

1. One elaboration is to have hierarchical roles, such as senior analyst and junior analyst. The senior analyst automatically inherits the permissions assigned to the junior analyst. This further simplifies administration.

2. The second elaboration is to have separation of duty and other constraints. For example, we may require the roles of analyst and mission specialist to be mutually exclusive,

so the same user cannot be assigned to both roles.

Here are some RBAC terms and concepts [2]

1. Access – a specific type of interaction between a subject and an object that results in the flow of information from one to the other
2. Access control – the process of limiting access to the resources of a system only to authorized programs, processes, or other systems (in a network)
3. Administrative role - a role that includes permission to modify the set of users, roles, or permission, or to modify the user assignment or permission assignment relations.
4. Constraint – a relationship between or among roles
5. Group – a set of users.
6. Object – a passive entity that contains or receives information
7. Permissions – a description of the type of authorized interactions a subject can have with an object.
8. Resource – anything used or consumed while performing a function. The categories of resources are time, information, objects, or processors.
9. Role – a job function within the organization that describes the authority and responsibility conferred on a user assigned to the role.
10. Role hierarchy – a partial order relationship established among roles.
11. Session – a mapping between a user and an activated subset of the set of roles the user is assigned to.
12. Subject – an active entity, generally in the form of a person, process, or device, that causes information to flow among objects or changes the system state
13. System administrator – the individual who establishes the system security policies, performs the administrative roles, and reviews the system audit trail.
14. User – any person who interacts directly with a computer system

The basic structure of RBAC is illustrated in Figure. 3.

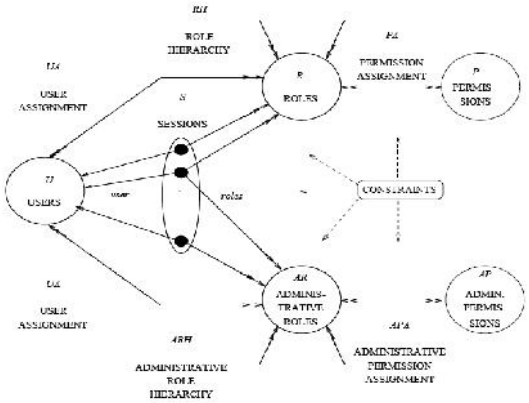

Figure 3. RBAC Administrative Model

According to [2], the bottom half of Figure 3 is a mirror image of the top half for administrative roles and administrative permissions. It is intended that administrative roles AR and administrative permissions AP be respectively disjoint from the regular roles R and permissions P.

The model shows that permissions can only be assigned to roles and administrative permissions can only be assigned to administrative roles. This is a built-in constraint.

It is also important to recognize that constraints can cut across both top and bottom halves of Figure 3. We have already asserted a built-in constraint that permissions can only be assigned to roles and administrative permissions can only be assigned to administrative roles. If administrative roles are mutually exclusive with respect to regular roles, we will have a situation where security administrators can manage RBAC but not use any of the privileges themselves

We could find the implementation of RBAC in Novell NetWare Directory Service and Microsoft Windows Server.[2]
Figure 4 shows the Novell NetWare Directory Service [3].

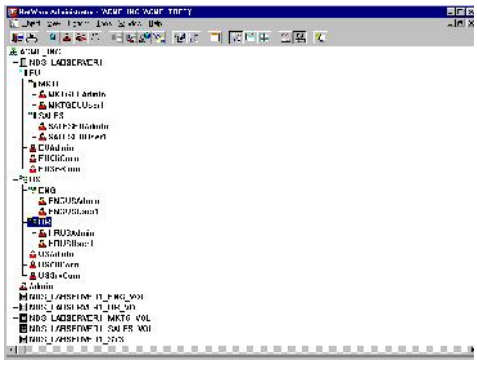

Figure 4. Novell NDS Structure

Novell NetWare manages RBAC
in the context of Directory Service [6]. Sales, Marketing (MKTG), Human and Resource (HR), and engineering (ENG) are the organizational unit (OU) of this Organization (ACME_INC) whose has branch in some continents (EU and US). Each OU could have its own role and administrator (Admin) who being the person in charge of all objects in the OU. For example, the administrator assistance in Marketing is able to create a new user only in that unit and also has responsibility to administer user access control list in that unit.

Windows Server also has built-in groups as the main role as shown in figure 5 and table 1 [4].

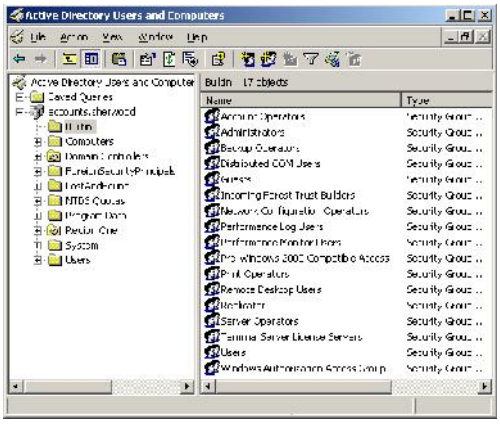

Figure 5. Windows Server's Built-In Groups

TABLE 1. The Predefined Built-in Local Security Groups

Name	Predefined Members	Abilities
Administrators	Administrator, Domain Admins, Enterprise Admins	By default, members of this group have almost total control of the domain controllers of the domain, including formatting hard drives and all the rights that the following four "operators" have. For Active Directory, this group has by default "Full Control except Delete Subtree" permission for almost all objects in the domain.
Account Operators	None	By default, members of this group can create, delete, and manage user, group, and computer objects in the Active Directory domain.
Server Operators	None	Members of this group can create, delete, and manage file shares, printers, and services in the domain controllers of the domain.
Backup Operators	None	By default, members of this group can back up and restore files and folders in the domain controllers of the domain, even if the member user doesn't have permissions for those files and folders.

A user who became a member of server operators group is able to operate some server's important task but doesn't have any permission to manage user which is a part of account operators group, and vice versa.

One of the key questions in applying RBAC to secure knowledge management is the nature of permissions in the knowledge-management context. The RBAC model is deliberately silent about the nature of permissions since this is highly dependent on the kind of system or application under consideration. Secure knowledge management requires control of access to a diverse set of information resources and services.

We can identify the following broad categories:

1) Information sources including structured and unstructured data, both within the organization and external to the organization;
2) Search engines and tools for identifying relevant pieces of this information for a specific purpose;
3) knowledge extraction, fusion, and discovery programs and services;
4) Controlled dissemination and sharing of newly produced knowledge.

Access to structured information in an organization, which resides in the organization's databases and other application repositories, is likely to already be under the purview of some form of RBAC using organizational roles. It is reasonable to assume that these organizational roles could be the foundation for role-based access to unstructured information.

However, access to unstructured information, which may reside on individual users' personal computers or small department-level servers, is fundamentally more problematic. Organizations will need to articulate and enforce access-control

policies with respect to this information.

While the initial thrust of knowledge management has been on techniques to extract useful knowledge from this scattered but extremely valuable information, challenging access-control issues must be addressed before these techniques can be applied in production systems.

We can assume a facility for distinguishing shared information from private information on a user's personal computer. Existing personal-computer platforms have notoriously weak security so it will require fundamental enhancements in personal-computer technology to give us a reasonable level of assurance in this regard.

Fortunately, there are several industry initiatives underway and, hopefully, some of these will come to fruition. The users will also need to determine how to share the public information. While reasonably fine-grained sharing techniques are available, it is unreasonable and undesirable to rely on the end users to specify these policies in detail. Moreover, current approaches are based on individual identities rather than roles.

Scalability of information-sharing policies will require that they use organizational roles as a foundation. Information sharing policies must be under control of the organization and cannot degenerate into anarchy where every user shares what they feel appropriate.

Access to specific search engines might involve access controls because of the cost or sensitivity issues. Cost comes about in terms of cost of licenses and such for accessing the search engine as well as the cost of the actual effort of performing the search.

Sensitivity comes about in terms of sensitivity of the results obtained by the search. Similar comments apply to the knowledge-extraction algorithms. Finally, the resulting knowledge itself needs to be shared and protected, thus completing the cycle.

The challenge for RBAC is to go beyond the traditional picture of human administration of authorizations as depicted by the administrative roles in Figure. 3, and move to a more seamless and less user-intrusive administrative model. More generally, we may need to invent new forms of RBAC that allow users to do some degree of exploration in the information space of an organization without allowing carte blanche access to everything.

This presents a significant research challenge to information security researchers.

3.2. UCON for Knowledge Management

The concept of Usage Control (UCON) was recently introduced in the literature by Park and Sandhu. A UCON system consists of six components:

1. Subjects and their attributes - A subject is an entity associated with attributes, and holds or exercises certain rights of objects. [5]
2. Objects and their attributes – Objects are set of entities that subjects hold rights on, whereby the subjects can access or use objects. [5]
3. Rights – privileges that a subject can hold and exercise on an object. Rights consist of a set of usage functions that enables a subject's access to objects. [5]
4. Authorizations – are functional predicates that have to be evaluated for usage decision and return whether the subject (requester) is allowed to perform the requested rights on the object. Authorizations evaluate subject

attributes, object [5]

5. Obligations – functional predicates that verify mandatory requirements a subject has to perform before or during a usage exercise. [5]
6. Conditions – environmental or system-oriented decision factors [5]

The authorizations, obligations, and conditions are the components of the UCON decisions. An authorization rule permits or denies access of a subject to an object with a specific right based on the subject and/or object attributes, such as role name, security classification or clearance, credit amount, etc. An attribute is regarded as a variable with a value assigned to it in each system state.

UCON is an attribute-based model, in which permission is authorized depending on the values of subject and object attributes. UCON extends continuity and mutability.

the traditional access-control models in one aspect that the control decision depends not only on authorizations, but also on obligations and conditions.

Obligations are activities that are performed by the subjects or by the system. For example, playing a licensed music file requires a user to click an advertisement and register in the author's web page. Such an action can be required before or during the playing process.

Conditions are system and environmental restrictions that are not directly related to subject or object attributes, such as the system clock, the location, system load, system mode, etc. Another aspect that UCON extends traditional access control models is the concepts of

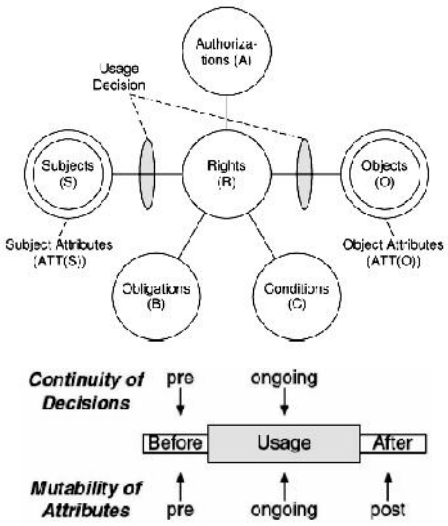

Figure. 6. UCON components.

For example, a user named Bob wants to buy an e-Book via an online book store. Then we can define the UCON components of this activity:

1. Bob is a subject whose has some attributes such as age and email address.
2. e-Book is an object whose has some attributes like label, author, the reading price, buying price, downloading price,

recommendation rank, etc.

3. Rights are privileges that subject (Bob) can hold and exercise object (e-Book), for example read only, download, modify, etc.
4. Authorizations are functional predicates that have to be evaluated for usage decision and return whether subjects are allowed to perform the requested rights to the objects. For example, the system will decide to allow or refuse when Bob requested to download e-Book.
5. Obligations are functional predicates that verify mandatory requirements a subject has to perform before or during a usage exercise. For example, to download the e-Book, Bob has to complete the registration form by filling the age, email address, user name and password. And then Bob also has to confirm by replying the email sent by the system after Bob click the 'submit' button in the registration form.
6. Conditions are environmental or system-oriented decision factors. Condition predicates evaluate current environmental or system status to check whether relevant requirements are satisfied or not and return either true or false. For example, to enable download the e-Book, Bob has to use a special kind of internet browser or download manager.

The continuity of decision is depended on the three states: pre-usage, ongoing-usage and post-usage. The mutability of attributes is also depended on the pre, ongoing and post usage. It means that the attributes of objects before its usage are could be different with ongoing usage. The attribute could be changed when the obligations and conditions are satisfied and the authorization decides to allow subjects to access objects.

For example, the price of e-Book is free of charge when Bob just wants to browse the content, but its price could became RMB 10 when Bob decided to download it. It means the attribute of e-Books is changed when Bob decided to change from only browsing to buying. The e-Book's recommendation rank could also be changed after a lot of users accessed or downloaded it. Please remember that UCON is usage control mechanism which has purpose to monitor and audit overall usage activities of subjects, objects and the usage of objects.

IV. TRUST MANAGEMENT AND NEGOTIATION

Trust management and negotiation is a key aspect of secure knowledge management. Knowledge management is essentially about corporations sharing knowledge to get a competitive advantage. This means that one needs to trust the individuals with whom he or she is prepared to share the knowledge.

Furthermore, corporations may have to negotiate contracts about knowledge sharing, and this means that a corporation has to trust another corporation before drawing up the contracts.

4.1 Definitions of Trust

What is Trust? A lot of sources gave different approach about trust, but authors adopt a more restricted notion of trust, which are the one underlying trust negotiation (TN) systems. Such notion was initially proposed by Blaze and Feigenbaum, according to whom, "Trust management problems include formulating security policies and security credentials, determining whether particular sets of credentials satisfy the relevant policies, and deferring trust to third parties"

Such a definition of trust basically refers to security policies regulating accesses to resources and credentials that are required to satisfy such

policies. TN thus refers to the process of credential exchanges that allows a party requiring a service or a resource from another party to provide the necessary credentials in order to obtain the service or the resource. Notice that, because credentials may contain sensitive information, the party requiring the service or the resource may ask to verify the other party's credentials before releasing its own credentials. This definition of trust is very natural for secure knowledge management as organizations may have to exchange credentials before sharing knowledge.

4.2 Trust Services

In the web-based transaction context, trust services are emerging as a business enabler, with the goal of delivering trust and confidence at various stages of the interaction among the parties involved in a transaction, including: establishing and maintaining trust, negotiations, contract formation, fulfillment, collaboration, through to dispute resolution.

Trust services attempt to solve problems such as: establishing the authenticity of electronic communications; ensuring that electronic signatures are fair and legally binding, and creating an electronic audit trail that can be used for dispute resolution.

The area of trust services is today a fast-moving area and it is difficult to anticipate the range of trust services that will be available in the next few years.

We can, however, reasonably expect that they include mechanisms to support trust establishment, negotiation, agreement, and fulfillment, such as identity services, authorization services, and reputation services. Knowledge-management strategies make use of the trust services.

4.3 TN Systems

TN is an emerging approach exploiting the concept of properties of the entities as a means for establishing trust, particularly in open environments such as the web, where interacting entities are usually unknown to each other.

TN is a peer-to-peer interaction, and consists of the iterative disclosure of digital credentials, representing statements certified by given entities, for verifying properties of their holders in order to establish mutual trust. In such an approach, access to resources (data and/or services) is possible only after a successful TN is completed.

A TN system typically exploits digital identity information for the purpose of providing a fine-grained access control to protected resources. However, unlike conventional access-control models, TN assumes that the interacting parties are peer and that each peer needs to be adequately protected. For instance, with respect to the peer owning the resource to be accessed, assets that need to be protected are, in addition to the resource, the access-control policies, as they may contain sensitive information, and the credentials of the resource owner.

With respect to the peer requiring access to the resource, the assets to be protected are the credentials as they often contain private information about the individual on behalf of whom the peer is negotiating.

4.4 TN Building Blocks

A TN involves two entities:

1. A client, which is the entity asking for a certain resource, and
2. A server, which is the entity owning (or more generally, managing access to) the requested resource.

The model is peer to peer: Both entities may possess sensitive resources to be protected, and thus must be equipped with a compliant negotiation system. The notion of resource comprises both sensitive information and services, whereas the notion of entity includes users, processes, roles, and servers.

The term resource is intentionally left generic to emphasize the fact that the negotiations we refer to are general purpose, that is, a resource is any sensitive object (e.g., financial information, health records, and credit card numbers) whose disclosure is protected by a set of policies.

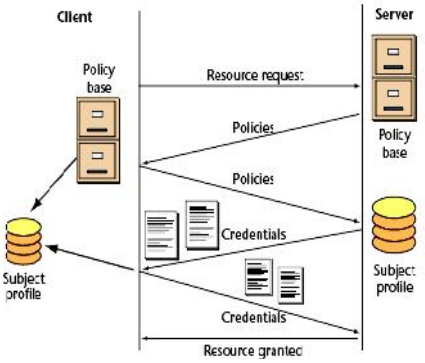

Figure 7. Organization of a TN process.

Figure 7 illustrates a typical negotiation process. During the negotiation, trust is incrementally built by iteratively disclosing digital credentials in order to verify properties of the negotiating parties.

Using the picture 7 we could describe if a client wants to access some resources in server then the trust negotiation should be:

1. Client request a resource to server
2. Server deliver the policy of accessing the resource that client must agree. The example of policy is like license agreement and the rights that client may have from the organization. Business role when we want to buy something from online store is also a kind of policy.
3. Client agree (or disagree) with the policy and reply to the server
4. Server accept the client's reply and then deliver the condition of credential that client must satisfied, for example credit card number, student ID number, passport number or other certificate
5. Client agree (or disagree) to fulfill the credential condition and send it to server
6. Finally, server accepts the overall request and grants the resource to client.

Credentials are typically collected by each party in appropriate repositories, called subject profiles. Another key component of any TN is a set of access-control policies, referred to as disclosure policies, governing access to protected resources through the specification of the credential combinations that must be submitted to obtain access to the resources.

To carry out a TN, parties usually adopt a strategy, which is implemented by an algorithm determining which credentials to disclose, when to disclose them, and whether to succeed or fail the negotiation.

Several TN strategies can be devised, each with different properties with respect to speed of negotiations and caution in releasing credentials and policies. The efficiency of a strategy depends on two factors: the communication cost and the computational cost. The communication cost includes the sizes of the messages exchanged and their number. Communication and computational costs of a negotiation strictly depend on the adopted strategy and vary from exponential, when a brute-force strategy is adopted, to more efficient strategies.

4.5 TN Requirements

The dimensions of TN requirements can be classified in two main groups, i.e.:

1. Those related to the adopted language

TN policy languages are a set of syntactic constructs (e.g., credentials, policies) and their associated semantics, encoding security information to be exchanged during negotiations. Effective TN languages should be able to simplify credential specification and also to express a wide range of protection requirements through specification of flexible disclosure policies. The main relevant dimensions for these languages are related with expressiveness and semantics

2. Those related to the system and its components.
   The development of comprehensive TN systems is quite challenging. On the one hand, such systems should be flexible, scalable, and portable. On the other, they should support advanced functions, such as support for credential chains, authentication of multiple identities, and complex compliance-checking modes whose efficient implementation is often difficult

It is important to note that these requirements are a partial list and other requirements are likely to be identified as research and deployment of negotiation systems progress, given also the increasing number of researchers actively contributing to the trust-management area.

4.6 Selected TN Systems

Because of the relevance of TN for web-based applications and knowledge management, several systems and research prototypes have been developed.

The most well-known systems include KeyNote by Blaze et al., TrustBuilder by Yu and Winslett, and Trust-X by Bertino et al., which we briefly discuss in what follows. These systems are relevant for TN in knowledgemanagement systems.

1. KeyNote has been developed to work for large- and small-scale Internet-based applications. It provides a single unified language for both local policies and credentials.
2. TrustBuilder is one of the most significant systems in the negotiation research area. It provides several negotiation strategies, as well as a strategy- and language-independent negotiation protocol ensuring the interoperability of the defined strategies.
3. Trust-X supports all aspects of negotiation, specifically developed for peer-to-peer environments. Trust-X supports an Extensible Markup Language (XML)-based language, known as XML-based Trust Negotiation Language (X-TNL), for specifying Trust-X certificates and policies.

V. PRIVACY MANAGEMENT

The security and privacy controllers will ensure that the security and privacy rules are not violated when executing the knowledge-management processes.

Since data may be mined and patterns and trends extracted, security and privacy constraints can be used to determine which patterns are private and sensitive and to what extent. For example, suppose one could extract the names of patients and their corresponding healthcare records. If a privacy constraint states that names and healthcare records taken together are private, then this information is not released to the general public. If the information is semiprivate, then it may be released to those who have a need to know, such as say a healthcare administrator.

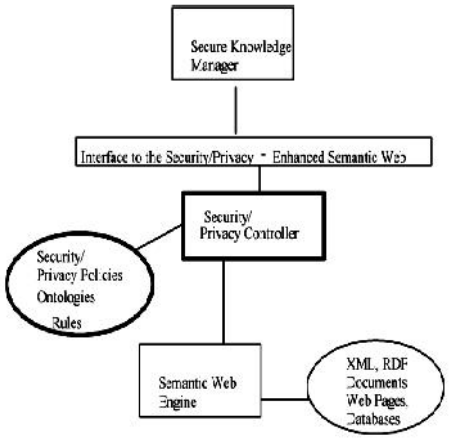

Figure 8. The privacy controller for the semantic web.

Figure 8 illustrates security/privacy controllers for the semantic web. As illustrated, there are data-mining tools on the web that mine the web databases. The privacy controller should ensure privacy-preserving data mining. Ontologies may be used by the privacy controllers. For example, there may be ontology specifications for privacy constraints and these specifications may be used by the inference controller to reason about the applications.

Furthermore, XML and resource description framework (RDF) may be extended to specify security and privacy policies. The secure knowledge manager will utilize the secure semantic web to execute its knowledge-management strategies and processes.

Now, the semantic-web community has come up with platform for privacy preferences (P3P) specifications. That is, when a user enters a web site, the site will specify its privacy policies and the user can then submit information if he/she desires. One example of P3P implementation is cookies management. We also can refer to some P3P explanation website like http://p3pbook.com/examples.html which provide syntax such as policy syntax.

VI. SUMMARY AND DIRECTION

Secure knowledge management will continue to be critical as organizations work together, share data, as well as collaborate on projects. Protecting the information and activities while sharing and collaborating will be a major consideration. This paper has discussed some key points in secure knowledge management:

1. Integrating security strategy with knowledge management and business strategies of an organization. Security has to be incorporated into the knowledge-management lifecycle.
2. The architecture for secure knowledge management.
3. For the confidentiality aspect, the paper describes briefly about RBAC and UCON as security techniques
4. For the trust aspect, the paper describes about trust management and negotiation focusing on trust negotiation
5. For the privacy aspect, the paper describes about secure knowledge management including privacy problems that arise due to data mining and inference inherent to the semantic web.

Also we found that there are many areas that need further work.

1. First, we need to develop a methodology for secure knowledge management. While we have discussed some aspects of secure knowledge management strategies, processes, and metrics, we need a comprehensive lifecycle for secure knowledge management.
2. We also need to investigate further RBAC and UCON, as well as trust management and negotiation. The definitions and rules discussed in this paper have to be formalized for RBAC, UCON, and trust.

3. The security critical components have to be identified for knowledge management.
4. Finally, privacy issues need to be investigated further.
5. In addition to enhancing and formalizing the policies discussed here, we also need to explore the incorporation of some of the real-world policy specifications into the knowledge management strategies. For example, we need to examine the P3P specified by the World Wide Web Consortium and determine how we can enforce such a policy within the framework of secure knowledge management.
6. We also need to investigate integrity aspects of knowledge management. For example, how do we ensure the integrity of the data and the activities? How can we ensure data, information, and knowledge quality? The best way to test out the policies is to carry out pilot projects for different types of organizations including those from academia, industry, and government. Based on the results obtained, we can then continue to refine the policies for knowledge management.

VII. REFERENCES

[1] Elisa Bertino, Latifur R. Khan, Ravi Sandhu, and Bhavani Thuraisingham. 2006. Secure Knowledge Management: Confidentiality, Trust, and Privacy. IEEE Explore

[2] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein and Charles E. Youmank. 1996. Role-Based Access Control Models. IEEE Computer

[3] Eric Z. Leonard and Jim Lathrop. 2010. Integrating Systems Management Server 2.0 with Novell NetWare. http://technet.microsoft.com/en-us/library/cc750160.aspx

[4] Sakari Kouti and Mika Seitsonen. 2002. Managing OUs, Users, and Groups in Active Directory. 2002. http://www.informit.com/articles/article.aspx?p=26918

[5] Jaehong Park and Ravi Sandhu. 2004. The $UCON_{ABC}$ Usage Control Model. ACM Transactions on Information and System Security

[6] Jeremy Epstein and Ravi Sandhu. 1996. NetWare 4 as an Example of Role-Based Access Control. ACM RBAC Workshop, MD, USA.