PRINDA: Architecture and Design of Non-Disclosure Agreements in Privacy Policy Framework

Vikram Goyal
Dept. of Comp. Sci. and Engg.
I.I.T. Delhi
Hauz Khas, New Delhi-16
vkgoyal@cse.iitd.ernet.in

Shyam K. Gupta
Dept. of Comp. Sci. and Engg.
I.I.T. Delhi
Hauz Khas, New Delhi-16
skg@cse.iitd.ernet.in

Indira Meshram
Dept. of Comp. Sci. and Engg.
I.I.T. Delhi
Hauz Khas, New Delhi-16
indira@cse.iitd.ernet.in

Anand Gupta
Dept. of Comp. Sci. and Engg.
N.S.I.T. Delhi
Sector-3, Dwarka, New Delhi
anand@coe.nsit.ac.in

Abstract

Non Disclosure Agreements(NDAs) in real life are typically used whenever there is transfer of private or confidential information from one organization to another. The provider organization can not have any control over privacy mechanisms of the receiving organization. The privacy policy work so far has addressed itself for preventing privacy violation within an organization. We aim to give an architecture of PRINDA(PRIvacy NDA) system which incorporates NDA's in privacy policy framework. Advantages of PRINDA system will be the following: i) As there can be traces of malicious activity(invasion of privacy) either at provider-end or at recipient-end, if detected/reported, NDA can help to relegate the responsibility to the organization violating the agreement, and will strengthen control in the privacy arena. ii) detailed information of usage of data can be provided to the owner of data e.g. information of all accesses at every level, and hence strengthening the principle of providing the individual data usage information.

1. Introduction

Management of data in the information systems is becoming a crucial need for many organizations and users due to high data generation rate and the value of information embedded in data. An individual is often required to submit her/his personal information to obtain services. On the other hand, personal data of a large communities of users can be a valuable resource for the organizations. This clearly raises the issue of privacy. Intrusion of personal information may lead to misuse of private information[13, 11, 5].

In order to build trust, the organizations publish the privacy policies stating what personal information can be used for what purposes[2, 9, 6, 22]. Languages like EPAL, P3P have been introduced to specify the privacy policies[20, 2] of an organization and for an individual. However privacy policies alone are not sufficient to convince the potential users to disclose their personal information. Rakesh Agrawal et. al.[6] proposed the concept of Hippocratic Databases where they suggest that the databases should take the responsibility for the privacy of the data as a fundamental tenet. Hendrik et. al. in[13] further extend the work in privacy arena by introducing the concept of contract between the user and the organization, stating that privacy policy alone will not be sufficient for building the trust. According to [13, 16, 7, 8, 19], privacy policies are mere promises and a promise as such has no legal grounds on which to contest a privacy breach, should the data controller not keep the promise.

The number of privacy intrusions are increasing[13, 11, 5, 4, 16] and the trust of individuals is decreasing[22]. To increase the trust level, there is a need to strengthen the privacy infrastructure. Privacy policies as advocated by Rakesh Agrawal et. al.[6] put a check on such intrusions within an organization. But in real life, individual's information may not remain in the first organization but may have to be delivered to other organizations for delivery of service. As the information crosses the boundary of first organization, there will be no control of the first organization over the other organization for individual's information pri-

Proceedings of the 22nd International Conference on Data Engineering Workshops (ICDEW'06)
0-7695-2571-7/06 $20.00 © 2006 IEEE

vacy. These second level organizations may require services from other organizations and consequently the chain of information flow gets formed. To put some control over usage of information along these chains of information flows, we propose PRINDA(PRIvacy NDA). PRINDA system can bind every information transfer with an NDA(Non Disclosure Agreement) between the provider and the receiver of information. If any privacy breach gets detected at any level in the chain of information flow, the responsibility can be relegated on the organization breaching the privacy. NDA contents may be designed to provide the flexibility to an individual e.g. the individual can know her/his information usage details at any time. In the concept of hippocratic databases[6], ten principles have been identified which include openness as one of the principles. It authorizes the individual to ask for the usage details of her/his information. PRINDA system can authorize the individual to have the usage details information from every organization having her/his data directly or indirectly, and hence clears the limitation of earlier systems i.e. only usage details of first organization could be provided.

The structure of the paper is as follows. Section 2 gives the background and motivation for NDAs in PRINDA system. In section 3, we introduce PRINDA system in privacy framework. Subsection 3.1 describes the NDA agreement structure stored in PRINDA system and subsection 3.2 defines a workflow to populate the NDA structure. Section 4 discusses the PRINDA architecture in detail. Finally in section 5, we give conclusions and future scope of work.

2. Background and Motivation

Non-Disclosure-Agreement (NDA) is defined as an agreement between the receiver and provider of the information[3]. It has been a typical vehicle for effectively delegating responsibility for confidentiality of information in most of the situations in non-automated domains. They only impose responsibility on the part of the recipient organization without any control of the provider (of the data). In case of agreement violation at the recipient end, the provision of the NDA can be legally enforced(subject to confirmation of violation at that end).

Earlier work on NDA(in non-automated domains) was concerned about the legal issues [18, 14, 15, 10, 1, 3]. Emphasis had been put on the need and contents of NDA to make it legal and hence what to be included/not-included in the agreement. Issues about the meaning of those NDA agreements have been discussed by focusing on the information, for which the NDA will be legally tenable. Obligations and responsibilities have also been discussed.

Several types of NDAs have been defined depending upon the directions of information transfer[14, 1, 3], i.e. bidirectional and one way NDAs. Mutual agreements are used when the information exchange occurs from both sides and one way agreements are used if the information flows from one side only. In the paper, we consider one way NDAs. NDA types are non-disclosure agreements for employment , business negotiations, visitor agreements etc on the basis of functional characteristic[1, 3].

As stated earlier, emphasis had been put on legal issues hence on contents of the agreements in non-automated domains, therefore properties have been identified that should be considered while defining the NDA[3]. These properties include definition of confidential information, exclusions from confidential information, obligations of receiving party, time period for the validity of the agreement, relationship between the party for any purpose, severability, integration and waiver [3]. Definition of confidential information actually describes the characteristics of the information to be protected by NDA. It segregates between the confidential and non-confidential information. Exclusion from confidential information further refines the information which will not come under the category of confidential information i.e. the information which was known to the receiving party earlier or was public. The obligations of receiving party specifies the responsibilities of the receiving party after receiving the information. Relationship points out that nothing contained in the agreement shall be deemed to constitute either party, a partner, joint venturer or employee of the other party for any purpose. In last the severability defines about the portion of the agreement to be valid according to the court as there may be some provisions which are not enforceable. The importance of NDA has been discussed in [18], where the author gives the litigated issues between the two companies and shows the value of NDA. The questions like why to use the NDA, when to use the NDA, what does the NDA cover, what can the recipient can do with information, how long does the NDA lasts, and what happens when the NDA terminates, are discussed in [14].

To show the advantages of using NDA, consider the example of an individual purchasing a commodity say from ABC. He/She provides personal information (i.e. name, address, credit card number etc.) to purchase some commodity (e.g. a specific book) and expects ABC to provide the service and maintain the confidentiality of the information (and be in legal position to take action if the situation so warrants). ABC has a privacy enforcement which it deploys to satisfy its clients. It needs (has) to pass data to a payment gateway and courier company for functional purposes. ABC does keep the privacy preferences of its customers, and tries to respect them, however it neither can pass user's privacy preferences, nor does it have any control of privacy at the payment gateway and the courier company. Further, these service providers, would be servicing not only ABC, but offering their services to many organi-

zation like ABC. It may further be possible (in fact more likely) that the payment gateway or the courier company may also seek services from channel partners, to whom they will supply data, thereby effectively losing control over privacy. We may note that the ABC customers deal only with ABC, paying for all the services of all intermediate service providers, however, without any knowledge of their actual identities.

Total control of privacy (even at level 1, i.e. at ABC) is a herculean task[6, 13] and is almost impossible and malicious attempts of intrusions can occur from within inside and outside of the organization. Realizing NDA's importance in the privacy policy framework, we propose the automation of NDA in privacy framework and give an architecture and integration of PRINDA in privacy policy framework. The intention of using PRINDA is to relegate the responsibilities of violation, wherever it occurs, subject to legal framework provided by NDAs. Legal issues and the actual contents of the agreements are not considered, but the need of NDA for privacy related issues in information systems is emphasized. The system, at any point in time, can assist to audit along the whole chain of organizations, who have been provided a particular piece of information.

We propose that each service provider,(i) provides an authenticated and numbered NDA to the organization from which it receives data,(ii) seeks an authenticated and numbered NDA from the organization to whom it needs to supply data (before actually supplying the data), and (iii) have effective privacy control at its own end. Every organization maintains a database of NDA (both inward and outward) which are authenticated and numbered.

3. PRINDA in Privacy Policy Framework

The functionality of the PRINDA system is to create and maintain the NDA agreements. It achieves this by its two main components i.e. inward NDA engine and outward NDA engine. As the data transfer takes place in both the directions(both inward and outward) in an organization, two NDA repositories are maintained by PRINDA in each organization. One NDA repository is maintained by the outward NDA engine and the other is maintained by the inward NDA engine.

Abstract view of PRINDA is shown in figure 1. It has three main components i.e. NDA coordinator, inward NDA engine and outward NDA engine. It interacts with privacy policy framework, other organization's PRINDA system and the service manager.

Privacy policy framework maintains and enforces individual and organization privacy policies. We assume the privacy policy framework design, as given in[6]. Service manager gets activated, if any service has to be provided to the individual. It finds out the service provider organization for delivery of service, selects the user id (to whom service has to be provided), the data(to be transmitted to the service provider organization) and determines the purpose of data transfer. It interacts with the PRINDA system to have the NDA agreements with all the selected service providers .

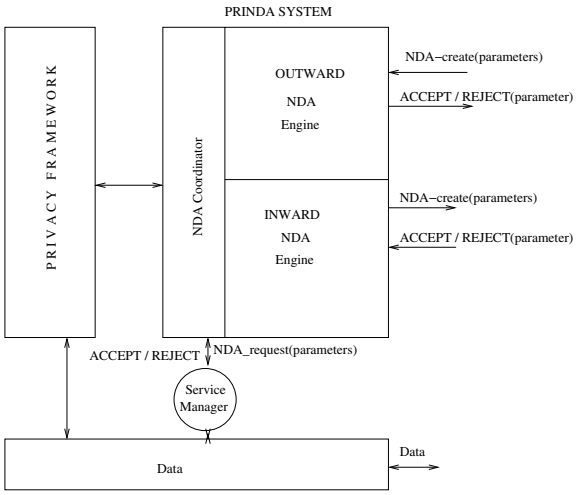

Figure 1. PRINDA(in Privacy Framework)

Whenever data is to be transmitted, service manager generates an NDA-request trigger with values user id, organization id, data id and purpose. The NDA system coordinator receives the NDA-request trigger and activates inward NDA engine. Inward NDA engine communicates with outward NDA engine of other organization(data receiver) by generating the trigger Create-NDA. The NDA system coordinator of data-receiver organization activates outward NDA engine after getting Create-NDA trigger. If an NDA agreement can be created between the two organizations then the data transfer takes place. The approved, authenticated and numbered NDA agreement is kept by both the organizations i.e. data-provider as well as the data-receiver.

The functionality of inward NDA engine is to bind the data-receiver organization with the NDA agreement. The data-receiver organization has to oblige the accepted NDA agreement to avoid itself from the penalties. This NDA agreement contents reflect the semantics of privacy policies of the organization and individuals in data-provider organization. In the binding activity, the inward NDA engine selects the particular NDA agreement from the set of NDA types, populates it with some contents and sends it to the data-receiver organization with NDA-create trigger. If the data-receiver organization approves this NDA agreement, inward NDA engine of data-provider stores the NDA agreement in its inward NDA repository.

The outward NDA engine functionality is to either accept or reject the NDA agreement, it receives from data-provider organization. If it accepts the NDA agreement, then it authenticates the agreement and populates the agreement with required values. It then sends back its acceptance with the generated values to the data-provider. Finally, after receiving the data from the data-provider, it stores the NDA agreement in outward NDA repository and maps the NDA to the privacy policies of the its privacy framework, so that its privacy framework can control the data according to the agreement signed.

It may be noted that for each data transfer, two NDA agreement records are generated, one in the data-provider side by inward NDA engine and one in data-receiver side by outward NDA engine. NDA agreement is kept in inward NDA repository at data-provider side and in outward NDA repository at data-receiver side. Both of the NDA agreements repository e.g. inward NDA repository as well as outward NDA repository have the similar structure figure 2, described in the following subsection.

3.1. NDA Structure

NDA structure is shown in figure 2. Description of the fields in the NDA structure is as follows:

NDA_t ID	NDA_r ID	Agree ID	Time stamp	Auth $sign_t$	Auth $sign_r$	Info ID	Org ID	Validity Period	Accept Bit
P	R	P	P,R	P	R	P,R	P,R	P	R

Figure 2. NDA Agreement Repository Structure

The values of the fields marked P are generated by the data-provider organization and fields marked R are generated by the data-receiver organization. Fields marked P,R indicates local values generated and stored by both organizations.

- NDA_t **ID** This is the NDA sequence number generated by the inward NDA engine of data-provider for each data transfer from it to data-receiver. This NDA sequence number value will be stored by both the data-receiver and the data-provider in their repositories.
- NDA_r **ID** This is NDA sequence number value generated at the data-receiver side for each data received. This value will also be stored by both the data-receiver and the data-provider in their repositories.
- **NDA Type ID** For each transaction, an agreement of a particular type is to be agreed by both data-receiver and data-provider organizations. Inward NDA engine of data-provider selects the NDA type and outward NDA engine of data-receiver accepts the NDA type. This set of agreement types are kept by both the organizations in their NDA agreement type repository and uniquely identified by their type. The agreement type value is stored for this field in both the repositories for each transaction.
- **Time Stamp** This field value is the actual time of data transfer. The data-provider side will store the time of transmitting data and the data-receiver side will store the time of receiving data.
- **Auth** $Sign_t$ The value of this field is the digital signature of the individual authorized by data-provider organization for the delivery of data. This value will also be stored by both the data-receiver and the data-provider in their repositories.
- **Auth** $Sign_r$ The value of this field is the digital signature of the individual authorized by data-receiver organization for receiving the data. This value will also be stored by both the data-receiver and the data-provider in their repositories.
- **Info ID** This is data id value of the data which is really transferred from the data-provider to the data-receiver. This value is local id to the data-provider in case of inward NDA repository and local id to the data-receiver in case of outward NDA repository.
- **Organization ID** This field contains the id value of the organization with which the agreement has been entered into. In case of inward NDA repository this id refers to the data-receiver organization and in case of outward NDA repository it refers to data-provider organization.
- **Validity** Validity specifies the time period for which the agreement is valid for the transaction data. This period value will also be stored in both the repository.
- **Accept Bit** This 1/0 bit indicates the acceptance/non-acceptance of the NDA by the target recipient organization.

The workflow for NDA structure population figure 3, is described in the next subsection.

3.2. Workflow for NDA creation

Workflow for NDA agreements creation is shown in figure 3. The workflow shows sequence of activities for a data transfer from one data-provider organization to data-receiver organization.

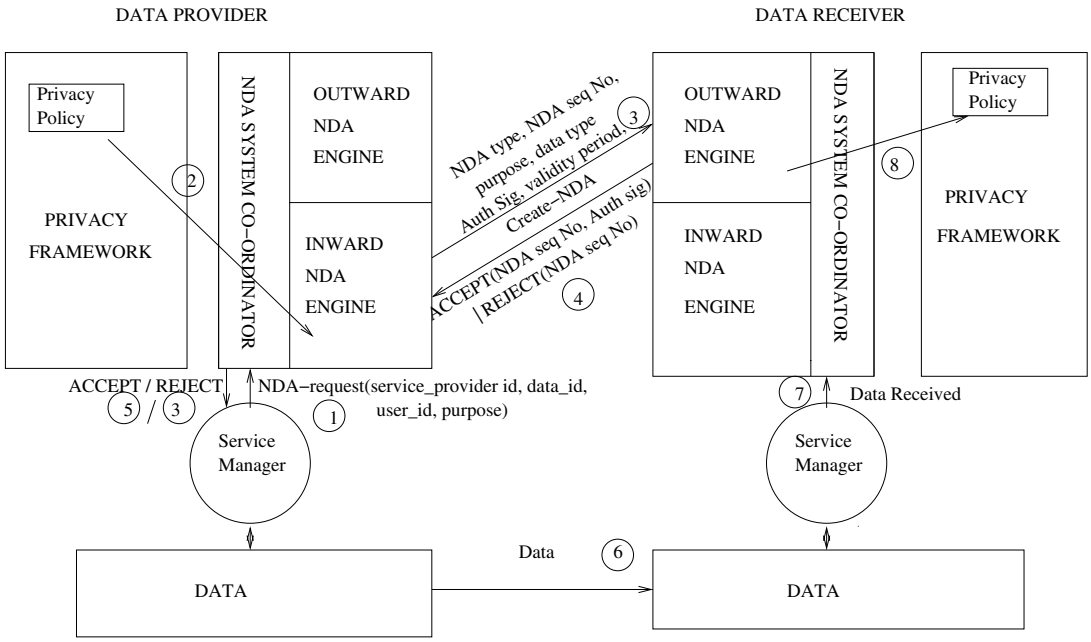

Figure 3. Workflow of Data Transfer from Data-provider to Data-receiver Organization

1. The service manager of data-provider organization generates the trigger NDA-request. NDA coordinator receives this trigger and initiates the NDA agreement creation process. The NDA coordinator receives the values organization id (organization to whom data has to be transmitted), user id(individual id, whose data is to be transmitted), purpose of usage and data type. NDA coordinator activates the inward NDA engine of data-provider with these values.

2. Data-provider's inward NDA engine interacts with its organization's privacy framework and gets the privacy policies of the individual and the organization.

3. Inward NDA engine validates all the inputs and check whether the information of the individual can be transferred. If information cannot be transferred it sends REJECT to service manager through NDA coordinator with the proper code specifying the reason of rejection. Otherwise, it selects the NDA type from the NDA type repository on the basis of all these inputs. It fills in NDA type code in field *AgreeID*, validity period value in field *ValidityPeriod* and digital signature in field $AuthSign_t$ in the NDA record. It also fills external recipients, other purposes of sharing, retention period etc in the NDA agreement template. It then generates and sets NDA sequence number in the field NDA_tID of this agreement record. Next it sends Create-NDA message with this partially created record to data-receiver.

4. The outward NDA engine of the data-receiver accepts Create-NDA message with the partially created agreement record, template. It generates NDA sequence number, digital signature and record these values in the received agreement record's fields NDA_rID and $AuthSign_r$ respectively. It analyzes the agreement record for acceptance/non-acceptance in accordance with its organization policies. If it cannot accept the NDA agreement then it sets field ACCEPT Bit to value REJECT(0) and field *Timestamp* to its current time, in the agreement record. It stores this record in it's outward NDA repository and sends it to the data-provider. If it can accept the NDA agreement then it marks ACCEPT bit to value ACCEPT(1) and sends it to data-provider.

5. Inward NDA engine of the data-provider receives either ACCEPT or REJECT signal from the receiver with the agreement record. If it receives REJECT signal it stores the agreement record in the inward NDA repository and sends the REJECT signal to service manager through its NDA coordinator. If it receives ACCEPT signal, it sets field *Timestamp*e to its current time value, stores the agreement record in its inward NDA repository and initiates data transfer to the data-receiver by sending ACCEPT signal to its service manager through NDA coordinator.

6. Service manager of data-provider sends a data trans-

mission request for data transfer.

7. The data is received in the database of data receipient organization. It is saved and a local data id is generated. It is passed to NDA coordinator.

8. Outward NDA engine receives the data id, sets field $Infoid$ to value data id, field $Timestamp$ to its current time in the system. It stores this record in its outward NDA repository and maps this NDA record to privacy policy form in its privacy framework. Now it is the privacy framework of data-receiver, that controls the access to the data associated with data id according to the NDA agreement it has created.

4. PRINDA Architecture

As stated in Section 2 earlier, PRINDA in every organization maintains NDA agreements for each data transfer(sent/received). Architecture of PRINDA system is shown in figure 4. It's components are NDA coordinator, Privacy Policies of the organization, Individual's privacy policies , Service providers information, NDA agreement type repository, Inwards's NDA database, Outward's NDA database, NDA log, Info Change Manager, NDA Updater, Inward NDA Selector, Outward NDA Creator, Outward NDA Analyzer and Inward NDA Creator. The description about each of the component is given below.

The first two components namely organization privacy policies and individual's privacy policies described below are defined in privacy policy framework. Privacy policies framework in an organization enforces these policies so that a trust level remains maintained between the individual and organization. PRINDA system interacts with the privacy policy framework to get the policies of the organization and an individual.

- **Organization Privacy Policies** Privacy policies specified by the organization mention not only purpose of usage of individual's private information but also specifies which information can be used by whom. These policies also specifies the retention period for which the information can be stored by the organization. The sharing purposes for the information sharing with the collaborators is also specified in these policies.

- **Individual's Privacy Policies** When an individual gives his/her information to the organization, he/she also describes his/her preferences about the data disclosure in addition to the consent for mandatory policies. This database maintains the privacy preferences of each user.

- **Agreement Database** A finite set of NDA agreement types is kept in PRINDA, which can be categorized, e.g. A to Z. For a particular data transfer, as per requirement of privacy policies of the data-provider, the NDA category may be chosen (by the Inward NDA selector of data-provider), relevant details pertaining to the data transfer, i.e. identity of the data, purpose, time and identity of owner may be provided (by the service manager).

- **Digital Signature Agent** The PRINDA system advantage to relegate the responsibility to the organization breaching the privacy, requires the NDA to be signed by the authorized signatory of data-provider as well as data-receiver.It is assumed that each organization nominates some employees in its organization who are authorized to sign a NDA agreement. This module takes care of signing the NDA agreement by the authorized signatories. The digital Signature module is activated by the Inward NDA creator and Outward NDA creator modules when needed. This module interacts with the NDA coordinator to get the signature of the authorized signatory. The NDA coordinator can use any electronic signature mechanism for the authentication of the signatures[17, 21, 12, 23].

- **Outward's NDA Database** This database maintains a record of all NDA agreements that the organization has entered into, for receiving the data. It's structure is same as described in the NDA structure section above.

- **Inwards's NDA Database** The database stores the NDA agreement for each data transmitted from the organization.It's structure is same as described in the NDA structure section above.

- **NDA log** The NDA log holds all the accesses to the NDA system's repositories e.g. Inwards NDA database accesses and Outwards NDA database accesses. NDA log maintained helps in auditing of NDA system for integrity and consistency check.

- **NDA Coordinator** NDA coordinator functionalities are to control the functioning and communication routines of the PRINDA system. It activates inward NDA selector, outward NDA analyzer, NDA updater, NDA information change manager, inward NDA creator and outward NDA creator. It provides the interface to the administrator for defining the NDA types and service providers. It listens all the events

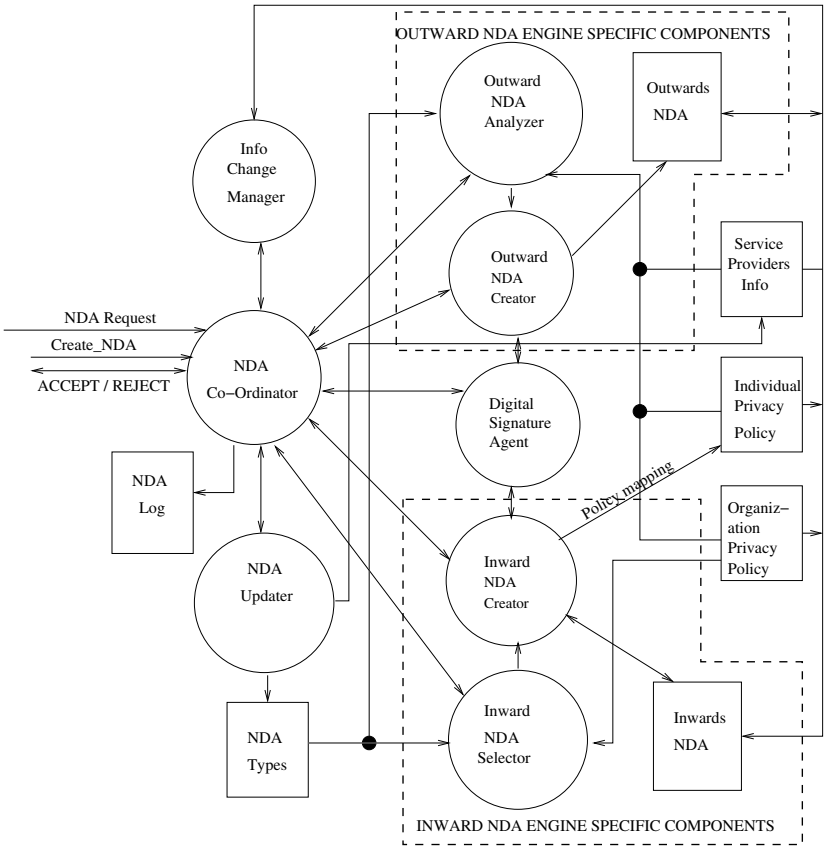

Figure 4. PRINDA Architecture

occurring in the system and accordingly invokes the modules for handling the event. It also maintains the NDA log.

- **Inward NDA Selector** The function of inward NDA selector is to select the NDA type. NDA Selector after receiving the user id, supplier id, data id and the purpose of the access from the NDA Coordinator, selects the NDA type, by also considering organization policies, user id's policies and supplier id's information. This module also checks for data transmission possibility e.g. whether the individual privacy policies allow for this data transfer ?. If not, REJECT signal is sent to service manager through NDA coordinator otherwise selected NDA type is returned to inward NDA Creator.

- **Outward NDA Analyzer** At the time of data receiving, the outward NDA analyzer gets the NDA type from the data-provider. It first generates its NDA sequence number. It checks whether it can accept NDA agreement for the respective information transfer by analyzing the information received with the organization privacy policies. If it can accept that agreement, it will forward that NDA type to the outward NDA creator. Otherwise it returns the signal for non acceptance of agreement e.g. REJECT to its NDA coordinator, which gets further transmitted with the NDA sequence number to the data-provider by the data-receiver's NDA coordinator.

- **Inward NDA Creator** Inward NDA creator has the responsibilities of populating inward NDA repository. It receives the inputs sent from the service manager by inward NDA selector, and values are extracted from the policies into its temporary buffer. This module generates the NDA sequence number, gets authorized signature through digital signature module and sends all this data in temporary buffer to the data-receiver with NDA-create message. After getting the data-receiver organization's NDA sequence number and authorized signature with the ACCEPT signal, it verifies all information and if received information is correct, it creates the complete NDA agreement by filling up the remaining values and stores agreement in inward NDA database with accept signal value equal to 1. It then returns the ACCEPT/ REJECT signal to the outside entity(service manager) through NDA coordinator after generating the trigger for data transfer. Otherwise if it gets REJECT signal from the data-receiver with the NDA sequence number, it stores the partial information in NDA agreement repository for this service provider with the accept signal value equal to 0 and sends the signal REJECT to service manager.

- **Outward NDA Creator** Outward NDA creator creates the NDA agreement records in the outward NDA repository. In case of REJECT from the outward NDA analyzer, it generates the NDA sequence number and creates the record in outward NDA repository with the information received from data-provider and NDA sequence number. It sets accept bit value equal to 0 for this record. Otherwise if it gets the ACCEPT signal from the outward NDA analyzer, it creates the NDA sequence number, gets the authorized signature and sends all this information to data-provider through NDA coordinator. When finally it receives the data from the data-provider, it stores the data in the database and creates the NDA record completely in the outward NDA repository. Then it maps the NDA agreement record to the privacy policy framework policy format so that privacy framework can control the access to the data according to the NDA signed by it.

- **NDA Updater** NDA updater functionality is to facilitate update of information in NDA types and service providers information repositories whenever the administrator likes to make changes.

- **NDA Information Change Manager** NDA information change manager looks after any changes in the policies or data repositories. It will verify the changes to maintain the integrity and consistency in the PRINDA system.

The repositories NDA types, NDA log, service provider information, and privacy policies view are shared by both the engine e.g. inward NDA engine and outward NDA engine. Inward NDA database is specific to inward NDA engine and outward NDA database is specific to outward NDA engine. Even though the structure of both the repositories are same as explained in the NDA structure subsection but the contents in outward data repository are for each data received and contents in inward NDA repository are for each data transmitted. The repository sharing view can be seen from the given table 1.

Table 1. Repositories in PRINDA

Inward NDA engine	**Outward NDA engine**	**Privacy** Framework
Inward NDA database	Outward NDA database	Individual Privacy Policies
NDA log	NDA log	Organization Privacy policies
NDA types	NDA types	-
Service Provider Info	Service Provider Info	-

5. Conclusion

In this paper, PRINDA(PRIvacy NDA) system has been proposed for incorporating the NDA in privacy policy framework. Detailed architecture of PRINDA, its integration with privacy policies framework is discussed by explaining each repositories and modules in PRINDA system. It is believed that by using PRINDA, the advantages like relegating the responsibility to the organization breaching the privacy policy and providing the detailed information of usage accesses to the individual can be achieved.

We leave optimization of PRINDA for future work.

References

[1] Employment non disclosure agreement. www.legaldocs.com/htsgif.d/xempltrs.mv.

[2] Epal 1.0 specifi cation, the enterprise privacy authorization language (epal 1.1). www.zurich.ibm.com/security/enterprise-privacy/epal/.

[3] Non disclosure agreements. www.nolo.com/article.cfm/Object1D/2ECF62E6-B334-4E83-9A94FA20A3FAFD38/cat1D/1FBE2D95-203C-4D38-90A2A9A60C6FD618/310/119/ART/.

[4] News target network. senators want to investigate privacy breach by jetblue airlines. http://www.NewsTarget.com/000251.html, october 26 2004.

[5] Stolen data: Yet another fi nancial privacy breach. Union-Tribunal Editorial, The San Diego Union-Tribune, April 17 2004.

[6] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu. Hippocratic databases. In *Proceeding of the 28th VLDB conference*, Hong Kong, China, 2002.

[7] P. Ashley, C. Powers, and M. Schunter. From privacy promises to privacy management- a new approach for enforcing privacy throughout an enterprise. In *Proceeding of the 2002 Workshop on New Security Paradigms*, pages 43–50, Virginia Beach, Virginia, September 23-26 2002.

[8] P. Ashley, S.Hada, G. Karjoth, and M. Schunter. E-p3p privacy policies and privacy authorization. In *Proceeding of the ACM Workshop on Privacy in the Electronic Society*, pages 103–109, Washington, DC, USA, November 21 2002.

[9] A.Westin. Privacy concerns and consumer choice. Technical report, Louis Harris and Associates, June 1998.

[10] B. S. S. Berger. Us intellectual property protection for business. Registered patent atorney 3990, south tropical trail, merridd isliand, fbrida-32952.

[11] F. Bowers. Dept admits privacy breach. IrishHealth.com, http://www.irishhealth.com/?level=4&id=5992, June 06 2004.

[12] P. Gutman. Simplifying public key management. *IEEE*, 37(2), 2004.

[13] M. S. O. Hendrik. J. Oberholzer. Privacy contracts as an extension of privacy policies. In *Proceedings of the International Workshop on Privacy Data Management*, pages 11–19, Tokoyo, Japan, April 2005.

[14] V. Irish. How to read an nda. *IEE Engineering management journal*, 2001.

[15] D. C. H. James C. Bruno. Enforcement of non-disclosure agreement. *Business Problems and Planning, Michigan Bar Journal*, Jan 2002.

[16] G. Karjoth, M. Schunter, and M. Waidner. Platform for enterprise privacy practices: Privacy enabled management of customer data. In L. N. in Computer Science, editor, *2nd Workshop on Privacy Enhancing Technologies*. Springer Verlag, 2002.

[17] P. Kitsos, N. Sklavos, and O. Koufopavlou. An effi cient implementation of the digital signature algorithm. *9th International Conference on Electronics, Circuits and Systems*, 3:1151–1154, 15-18 Sept 2002.

[18] M. M. Klee. The importance of having a nondisclosure agreement. IEEE Engineering in medicine and biology patent, 2000.

[19] F. Lategan and M. Olivier. On granting limited access to private information. In *Proceedings of the tenth International Conference on the World Wide Web*, pages 21–25, Hong Kong, 2001.

[20] M. Marchiori. The platform of privacy preferences 1.0 (p3p 1.0) specifi cation. W3C proposed Recommendation, January 2002.

[21] U. Maurer. New approaches to digital evidence. In *Proceeding of IEEE*, volume 92, June 2004.

[22] T. Vila, R. Greenstadt, and D. Molner. Why we can't be bothered to read privacy policies: Models of privacy economics as a lemnos market. In *Proceeding of the 3rd International Conference on Electronic Commerce*, pages 403–407, Pitsburg, PA, 2003.

[23] P. Wohlmacher. Digital certifi cates: a survey of revocation methods. In *Proceedings of the ACM workshops on Multimedia*, pages 111–114, Los Angeles, California, United States, 2000.