International Journal of Computer Science and Engineering (IJCSE)
ISSN(P): 2278-9960; ISSN(E): 2278-9979
Vol. 3, Issue 4, July 2014, 49-60

International Academy of Science, Engineering and Technology
Connecting Researchers; Nurturing Innovations

NOVELBASED METHOD FOR PATIENT HEALTH RECORDS SECURING AND SHARING IN CLOUD COMPUTING USING ATTRIBUTE BASED ENCRYPTION

S B SHIVAKUMAR[1], K M NIRANJAN[2], KUMARSWAMY H[3] & AVINASH G M[4]

[1]Principal & HOD, Department of Computer Science & E, S.J.M.I.T, Chitradurga, Karnataka, India

[2]Associate Professor, Department of Mathematics, U.B.D.T, Davanagere, Karnataka, India

[3]Associate Professor, Department of Information Science & E, S.J.M.I.T, Chitradurga, Karnataka, India

[4]M.Tech Student, Computer Science & E, S.J.M.I.T, Chitradurga, Karnataka, India

ABSTRACT

Cloud Computing servers provides hopeful platform for storage of data. Sharing of patient healthrecords (PHR) is an emerging patient centric model of well-being information exchange, which is often outsourced to store at third party, such as cloud providers. It allows patients to create, manage, control and share their patient well-being information from one place through the web, with other users as well as healthcare providers. The patient's records maintained with full security and privacy in. the centralized server. To achieve fine grained and scalable data access control for health records stored in semi trusted servers, we make use of attribute based encryption (ABE) to encrypt the each patient's health information. In this paper, we discover key-policy attribute based encryption (KP-ABE) and multi-authority attribute based encryption (MA-ABE) to enforce patient access control policy such that everyone can download the data, but only authorize user can view the medical records. It also supports multiple owner scenarios and divides the users in the system into multiple security domains that greatly reduce the key management complexity for owners and users. A high degree of patient privacy is guaranteed by exploitingma-abein public domain and (KP-ABE) in personal domain.

KEYWORDS: Patient Health Records, Cloud Computing, Data Privacy, Fine-Grained Access Control and Multi-Authority Attribute Based Encryption

INTRODUCTION

Cloud computing is the delivery of computing services over the Internet. Cloud services allow individuals and businesses to use software and hardware that are managed by third parties at remote locations.

The characteristics of cloud computing include on-demand self-service, broad network access, resource pooling, rapid elasticity and measured service. The cloud computing service models are Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS). There are different types of clouds are Public Cloud, PrivateCloud and Hybrid Cloud. In current Year's Patient Health Record (PHR) has emerged as a patient-centric model of health information exchange. A system examine allows a user to create, manage, and control their patient health data in one place through the web, which has made the loading, recapture and sharing of their health information more secure. Each patient is promised the full control of their patient health records and can share their health data with an extensive range of users, including healthcare providers, family unitor friends. Due to the high cost of creation and maintaining committed data centers, many patient health record services are outsourced to or provided by third-party service providers.

Due to the high value of the sensitive Patient Health Information (PHI), the third-party storage servers are often

the targets of various hostile actions which may lead to exposure of the PHI. To make certain patient-centric privacy control over their own medical health records, it is necessary to have fine-grained data access control mechanisms that work with semi-trusted servers, and records are stored on semi-trusted servers, and focus on addressing the doubtful and challenging key management issues has been done.

In order to protect the patient health information stored on a semi-trusted server, Attribute-Based Encryption (ABE) V. Goyal, Pandey, A.Sahai,[3] as the main encryption primitive has been adopted to attain fine grained access control. In order to estimate system proposal, a library that implement primitive of Key Policy Attribute-based Encryption (KP-ABE) algorithms and library that realize primitive of Multi-Authority Attribute-based Encryption (MA-ABE)M. Chase and S. S. Chow[5], algorithms are created. A patient medical record sharing system that allow patient to encrypt and submit their medical records to servers using KP-ABE and MA-ABE and users from public and personal domain allows to decrypt the medical records is implemented.

Key Offerings

We propose a novel ABE-based framework for patient-centric secure sharing of PHRs in cloud computing environments, under the multi owner settings. To address the key management challenges, we conceptually divide the users in the system into two types of domains, namely public and personal domains (PSDs).

In the public domain, we use Multi Authority ABE (MA-ABE)to improve the security and avoid key escrow problem. Each Attribute Authority (AA) in it governs a displace subset of user task attributes, while none of them alone is able to control the security of the whole system.

In the personal domain, owners directly assign access privileges for patient users and encrypt a PHR file under its data attributes. Furthermore, we enhance MA-ABE by putting forward an efficient and on-demand user/attribute revocation scheme, and prove its security under standard security assumptions.

Problem Definition

Patient's Health Record Sharing (PHRS) system is a system there are multiple PHRS owners and PHRS users. The owners refer to patients who have full control over their own PHRS data, i.e., they can create, manage and delete it. There is a central server belonging to the PHR service provider that stores all the owners' PHRSs. The users may come from various aspects; for example, a friend, a caregiver or a researcher. Users access the PHRS documents through the server in order to read or write to someone's PHRS, and a user can concurrently have access to multiple owners' data. To achieve "patient-centric" medical record sharing, a core requirement is that each patient can control who are authorized to access to their own PMRS documents. Especially, user controlled access is core security objectives for the proposed PHRS system. The security and performance requirements are summarized as follows:

Data Confidentiality

Unauthorized users who do not have enough attributes satisfying the access policy or do not have proper key access privileges should be prevented from decrypting a PHRS document, even under user collusion.

Fine-Grained Access Control

Fine grained access control should be enforced, meaning different users are authorized to read different sets of documents.

On-Demand Revocation

Whenever a user's attribute is no longer valid, the user should not be able to access future PHRS files using that attribute. This is usually called attribute revocation, and the corresponding security property is forward secrecy. There is also user revocation, where all of a user's access privileges are revoked.

Scalability and Usability

The PHRS system should support users from both the patient domain and public domains. Since the set of users from the public domain may be large in size and unpredictable, the system should be highly scalable, in terms of complexity in key management, communication, and storage. Additionally, the owners' efforts in managing users and keys should be minimized to enjoy usability.

Objective

To propose and develop proof of concept framework for patient centric fine grained data access control and secure sharing of medical records with multi-owner environment on a semi trusted server and try to minimize the complexity of key management using Attribute based encryption techniques which also support modification of access policies or file attributes, supports efficient user/attribute revocation. First implement the Key-Policy Attribute Based Encryption (KP-ABE) Library API. Second implement the Multi-Authority Attribute Based Encryption (MA-ABE) Library API.

RELATED WORK

This paper is mostly related to work in cryptographically enforced access control for outsourced data and attribute based encryption. To improve upon the scalability of the above solutions, one-to-many encryption methods such as ABE can be used.

Symmetric Key Cryptography (SKC) Based Solutions

Symmetric-key algorithms are a class of algorithms for cryptography that use the same cryptographic keys for both encryption of plaintext and decryption of cipher text. The keys may be identical or there may be a simple transformation to go between the two keys. The keys, in practice, represent a shared secret between two or more parties that can be used to maintain a private information link.

The SKC-based solutions have several key limitations. First, key management overhead is high when there are a large number of users and owners, which is the case in a system. Second, key distribution can be very inconvenient when there are multiple owners, since it requires each owner to always be online.

Public Key Cryptography (PKC) Based Solutions

PKC based solutions were proposed due to its ability to separate write and read privileges."Patient controlled encryption**:** ensuring privacy of electronic patient records", they purpose the solution scenario and shows how public and symmetric based encryption used, disadvantage of their solution is either suffer high key management overhead, or require

encrypting multiple copies of a file using different users' keys. Access control can be enforced if every write and read operation involves a proxy server. However, it does not support fine grained access control, and is also not collusion-safe Implementation.

Attribute-Based Encryption (ABE) Solutions

The SKC and traditional PKC based solutions all suffer from low scalability in a large patient health record system, since file encryption is done in one-to-one manner, while each patient health record may have an impulsive large number of users. To avoid such inconveniences, novel one-to-many encryption methods such as attribute-based encryption can be used.

In Attribute-Based Encryption **(**ABE) system, users' private keys and cipher text are labeled with sets of descriptive attributes and access policies respectively, and a particular key can decrypt a particular cipher text only if associated attributes and policy are matched. There are two kinds of ABE having been proposed: Key-Policy Attribute-Based Encryption (KP-ABE). Cipher text-Policy Attribute-Based Encryption (CP-ABE).

SYSTEM DESIGN

Software requirements and specification find their reflection in design system. The following section outlines proposed solution, i.e. software architecture, modularization, data flow between applications and specification of employed protocols. ABE as a building block is used in proposed system. That is because ABE not only offers fine-grained access control similar to RBAC or ABAC, but also enforces data protection against semi-trusted server.

Architecture

Architecture is an overall structure of a system. It deals with the overall working of the system. The design process for identifying the sub-systems making up a system and the framework for sub-system control and communication is architectural design. Figure shows depict the architecture of proposed PMRS system for secure sharing of the medical records.

The system is split into two security domains namely, public domains (PUDs) and personal domains (PSDs) according to the different users' data access requirements. The PUDs consist of users who make access based on their professional roles, such as doctors, nurses, medical researchers and insurance agents. For each PSD, its users are personally associated with a data owner (such as family members or close friends), and they make accesses to medical records based on access rights assigned by the owner.

In personal domains the owner used key-policy attributed based encryption and generates secret key for their PSDs user and in PUDs the multi-authority attribute based encryption is preferred. Secret Key for PUDs users are generated by multiple authorities depending on their specialization and profession in combine. The whole system is composed of three software parts which run as stand-alone applications.

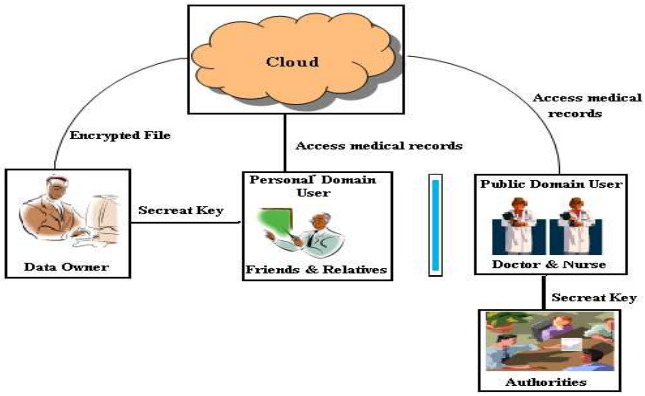

Figure 1: Architecture of Scalable and Secure Sharing of Medical Records in Cloud Computing Using Attribute-Based Encryption

Modules of the System are

- System Setup and Secret Key Generation
- Encryption of Health Records and upload.
- View Encrypted Medical Records (Decryption).
- Revocation of Public domain User / attributes.
- Policy Update.
- Login and Upload of public parameters, Encrypted Records.
- Implement the various libraries.

Existing System

Due to the high cost of building and maintaining specialized data centers, many PHR services are outsourced to or provided by third-party service providers, for example, Microsoft Health Vault.

While it is exciting to have convenient PHR services for everyone, there are many security and privacy risks. Which could impede its wide adoption. The main anxiety is about whether the patients could actually control the sharing of their sensitive patient health information (PHI), especially when they are stored on a third-party server which people may not fully trust.

Disadvantages of Existing System

- There have been wide privacy concerns as patient health information could be exposed to those third party servers and to unauthorized parties.
- They usually assume the use of a single trusted authority (TA) in the system. This not only may create a load bottleneck, but also suffers from the key escrow problem. In addition, it is not practical to delegate all attribute management tasks to one trusted authority including certifying all users attributes or roles and generating secret keys.

- There still lacks an efficient and on-demand user revocation mechanism for ABE with the support for dynamic policy updates/changes, which are essential parts of secure PHR sharing.

Proposed System

To assure the patients' control over access to their own PHRs, it is a promising method to encrypt the PHRs before outsourcing. In this paper, we propose a novel patient-centric framework and a suite of mechanisms for data access control to PHRs stored in semi-trusted servers. To achieve fine-grained and scalable data access control for PHRs, we leverage attribute based encryption (ABE) techniques to encrypt each patient's PHR file.

To ensure patient-centric privacy control over their own PHRs, it is essential to have fine-grained data access control mechanisms that work with semi-trusted servers. In order to protect the patient health data stored on a semi-trusted server, we adopt attribute-based encryption (ABE) as the main encryption primitive. Using ABE, access policies are expressed based on the attributes of users or data, which enables a patient to selectively share her/he PHR among a set of users by encrypting the file under a set of attributes, without the need to know a complete list of users.

Advantages of Proposed System

- We focus on the multiple data owner scenario, and divide the users in the PHR system into multiple security domains that greatly reduces the key management complexity for owners and users.
- We bridge the above gaps by proposing a unified security framework for patient-centric sharing of PHRs in a multi-domain, multi-authority PHR system with many users.

Proposed Solution

A secure ABE-based framework for patient-centric secure sharing of medical records in cloud computing environments, under the multi-owner settings is proposed.

To address the key management challenges, the users in the system are divided into two types of domains, namely public and personal domains. In particular, the majority professional users are managed distributive by attribute authorities in the former, while each owner only needs to manage the keys of a small number of users in their personal domain. In this way, our framework can simultaneously handle different types of sharing applications' requirements, while incurring minimal key management overhead for both owners and users in the system.

A Public Domain

In the public domain, Multi-authority ABE (MA-ABE) with enhancement to improve the security and avoid key escrow problem is proposed. Each Attribute Authority (AA) in it governs a disjoint subset of user role attributes, while none of them alone is able to control the security of the whole system. In public domain two authorities medical and Specialization authority are considered who manages the professional attribute and specialization attributes respectively.

The system defines role attributes, and a reader in a public domain obtains secret key from Attribute Authorities (AAs), which binds the user to their claimed attributes/roles.

Attribute Authorities in combine generates global public parameter and attributes specific public and master parameter of their respective attributes using MA-ABE Setup algorithm. And publish public parameters with help of service provider. All Attribute Authority in combine generates the secret key for the public domain user of their claimed

role attributes and send via secure email to obtain the secret key.

Algorithm Forma-Abe

- **Setup ():** This algorithm's input is each AA's attribute universe {Uk}k∈ {1….N}, and outputs a master key for each AA and the public key. It is cooperatively executed by all NAAs. It defines bilinear groups G1, G2 with prime order p and generators g1, g2 respectively, and an admissible bilinear map e: G1 × G2 → GT.

  The PK and AAk's master key MKk are PK = ⟨Y =e (g_1, g_2) ^Σkvk,

  {$T_{k;i}$= g^$t_{k;i}$$Ver_{k,i}$} i∈$U_k$,k∈{1…N}⟩

  MK_k= ⟨v_k, {$t_{k;i}$,$Ver_{k;i}$}i∈Uk⟩

  Where vk is the master secret of each AA, and tk; i∈Zp and Tk; i∈G2 are attribute private/public key components for attribute i.

- **Key Issue (Attributes, MK, PK):** This algorithm, the AAs collectively actively generates a secret key for a user. For a user with (secret) IDu, the secret key is in the form:

  SK_u= ⟨Du =g^R_u, {$D_{k;i}$= g^(q_k(i)/$t_{k,i}$) , $Ver_{k,i}$}k∈{1…N} ⟩

  Where *Ru* is a global ID for user *u*, and

  $q_k(0) = \Sigma_k v_k - R_u$.

- **Encryption (M, PK, Attributes []):** This algorithm takes a message *M*, *PK* and a set of attributes and outputs the cipher-text *E* as follows

  The encryptor first chooses an *s*∈Z*p*, and then returns.

  CT=[E0=M·Y ^ s,E1=g_2^s,{$C_{k;i}$=$T_{k;i}$^s,$Ver_{k;i}$};k∈{1..N}],

  Where i=no of attributes form authority k.

- **Decryption (*CT, Sku*):** This algorithm takes as input a cipher text *CT* and a user secret key *SKu*. If for each AA*k*, If the version of attribute in SK and CT matches, algorithm pairs up $D_{k;\ I}$ and $C_{k;I}$ and reconstructs $e(g1,g2)^{sq_k(0)}$.After multiplying all these values together with $e(Du,E_1)$, *u*recovers the blind factorY^s and thus gets*M*.

- **Update Parameter:** This algorithm updates an attribute to a new version by redefining its system master key and public key component. It also outputs a proxy re-encryption key between the old version and the new version of the attribute.

  Pseudocode is as:

  For each i in attribute

  - randomly pick t_{inew} from Zp;
  - compute T_i=g_1^ t_{inew}

–reEncKey$_i$= t_{inew}/ t_i

–reSecKey$_i$= t_i/ t_{inew}

–Ver$_i$=Ver$_i$+ 1

–Ti = t_{inew}

- **Upadte Secret Key:** This algorithm translates the secret key component of attribute i in the user secret key SK from an old version into the latest version.

  Pseudo code is as Input: Secret Key and reSecKeyfile:

- For each matching i in attribute of secret key and reSecfile
- $D_{k,inew}=(D_{k,i})^{reSecKey_i}$
- Update Version ofsecretkey attribute
- **ReEncry pt File:** This algorithm translates the cipher text component of an attribute I of a file from an old version into the latest version. Pseudo code is Input **:** Cipher-Text File and re Enc Key$_i$ file.
- For each matching i in attribute of secret key and re Secfile.
  - $C_{k,inew}=(C_{k,i})^{reEncKey_i}$.
  - Update Version of attribute in Encrypted File.
- **Update Cipher-Text:** Forgivens value and list of attributes and encrypted file, Algorithm modifies the cipher text attribute policy Part of the encrypted file. First check For Valids value, Ifs is valid then it modifies using Encryption() Algorithm is used above.

B Personal Domain

In the personal domain using Key Policy – Attribute Based Encryption (KP-ABE), each patient manages their secret keys and access rights of users their personal domain users. Propose mechanisms for Encryptions o that patient can specify personalized fine – grained role – based access policies during file encryption.

KP-ABE is a cryptography system built upon byline arm apand Linear Secret Sharing Schemes

Algorithm for kp-ABE

- **Kp-Abe Setup (A):** Outputs public key PK and Master key MK for A asset of attributes.-It defines a bilinear group G1 of prime order p with a generator g, a bilinear map e:G1× G1->G2 which has the properties of bi-linearity, computability, and non-degeneracy.

A as set of attributes, Associate for each attribute in A with attributes universe as U= {1, 2, ..., n}. Associate each attribute i ∈U with a number ti and choose y uniformly at random in Zp*.

Outputs are PK and MK calculated as:

- The public key is **:**PK =($T_1= g^{t1}$, ..., $T_1= g^{t|U|}$,Y = $e(g,g)^y$)

- The master key is-MK= (t_1,....., $t_{|U|}$,y)
- **KP-A be Encryption (M, Γ, PK):** Choose a random values in Zp. Encrypt a secret message M in G_T,with a set of attributes γ.

  The cipher-text is:

  $E = (\gamma, E' = MY^S, \{E_i = T_i^S\}_{i \in \gamma})$
- **KP-A be Key Generation (A, MK):** This algorithm output a secret key SK embedded with a access structure T. The access structure A is realized by the following three steps:

  For root noder, set value secret = y.mark all node un-assigned and mark root node assigned.
- Recursively, For Each assigned Non-Leaf Node, A. If the operator is 'And', And its child no des are marked un-Assigned, Let n be the number of child nodes, Set the value of each childnode, Except the last one, To be $s_i \in Zp$, And The Value Of The Last Node To Be $S_n = S - \sum S_i$.Mark This Node Assigned. B. If the operator is or, Set the values of its child nodes to be s. Mark this node Assigned.
- For each Leaf attribute $a_{j, I} \in T$, Compute $D_{j; I} = T_j^{\Lambda} s_i$, Secret Keys $k = \{D_{j, I}\}$.
- **KP-A be Decryption (E,D):**This algorithm takes as input the cipher text e encrypted Under the attributes etu, The user's secret key sk for accesstreet, And the public key PK. Finally it Output the message m If and Only if u satisfies T.

In The personal domain, Owners Directly assign access privileges for personal users and encrypt a medical record file Under Its Data Attributes. Who can access the patient's medical record are referred as data reader The Date Owners refer to patients who have full control over their own phrdata, I.E., They can Create, Manage and delete it. The data readers download medical record files from the Server, And They can Decrypt The Files Only if They have suitable Attribute based keys. System will support revocation of one or more role attributes of a public domain user.

Revocation of a public domain user which is equivalent to revoking all attributes of Public Domain User. These Operations are done By the AA That The User belongs. A medical record owner can update sharing policy for an existing medical record Document By updating The Attributes (Access Policy) In The cipher-Text.

Encryption and Upload

The Owner Encrypt the health records under a certain fine grained and role-based access policy for users from the Public domain to access, and under a selected set of data attributes that allows access from users in the personal. And Uploads Encrypted File to the server.

View Encrypted Health Record File/Decryption

User from the personal or public domain can request the file form the server. Only user can view the records, provided the secret key policy matches with the attributes attached with the files.

Revocations

There vocation public domain users/attributes are consider. Revocation of user is similar to revocation of all

attributes of the user. The Revocation of user attribute is done using following steps:

Attribute Authority (AA) redefines the MK and PK of the attributes of there voked user and also generates Proxy Re-encryption keys for files and secrets key.

Attribute Authority sends the Proxy Reencryption keys for secret key to unrevoked user via secure email to medical professional and Medical profession updates the secret key using PRE Secret Keys. Authority re-encrypts the encrypted file with the help of pre attribute keyson server.

Policy Update

Our scheme should support the dynamic add/modify/delete of part of the document access policies or data attributes by the owner. For example, if a patient does not want doctors to view their medical record after they finishes a visit to a hospital, patient cans imply delete the cipher-text components corresponding to attribute “doctor” in their medical records files. To make the computation more efficient, each owner could store the random numbers used in encrypting the symmetric key of each document on their own computer, and construct new cipher-text components corresponding to added/changed attributes based on used during encryption.

CONCLUSIONS

A novel framework of secure sharing of patient medical records in cloud computing is proposed. Considering partially trustworthy cloud servers, to fully realize the patient-centric concept, patients shall have complete control of their own privacy through encrypting their medical record files to allow fine-grained access. The framework addresses the unique challenges brought by multiple owner sand users, in that greatly reduce the complexity of key management while ensured the privacy. Different ABE encryption techniques are utilized to encrypt the medical record files, so that patients can allow access not only by personal users, but also various users from public domains with different professional roles, qualifications and affiliations. System supports there vocation and update attribute policy of the encrypted file.

Future Enhancement

One fundamental assumption was that plan is a proof-of-concept, and consequently its functionality is highly limited. On the other hand, internal complexity and need for external libraries made the mission quite a big undertaking. Due to limited time for the work, some features were not included into initial specification, but could be added in subsequent releases of software.

Current version of the plan does not provide any public directory server which would store public keys and also implement system which sends secret keys corresponding to given identities. This kind of application would be a fundamental factor of transparent data encryption.

Future Version of Web application will include write access control and other dynamic features required to complete the sharing of medical record system.

Current version of the paper supports policy in the form of ‘or’& ‘and ‘node’, In future version need to Explore more dynamic and efficient way to illustrate access structures for policy with support of more operators which is required for Key Generation.

REFERENCES

1. M. Li, S. Yu, K. Ren, and W. Lou, "Securing Personal Health Records in Cloud Computing: Patient-Centric and Fine-Grained Data Access Control in Multi-Owner Settings," Proc. Sixth Int'lICST Conf. Security and Privacy in Comm. Networks (SecureComm '10), pp. 89-106, Sept. 2010.
2. S.Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing,"inIEEEINFOCOM'10, 2010.
3. V.Goyal, Pandey, A. Sahai, and B. Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in CCS'06, 2006, pp. 89–98.
4. M. Li, S. Yu, N. Cao, and W. Lou, "Authorized Private Keyword Search over Encrypted Personal Health Records in Cloud Computing," Proc. 31st Int'l Conf. Distributed Computing Systems (ICDCS '11), June 2011.
5. M. Chaseand S. S. Chow, "Improving privacy and security in multi-authority attribute-based encryption," in CCS'09, 2009, pp. 121–130.
6. A. Akinyele, C. U. Lehmann, M. D. Green, M. W. Pagano, Z. N. J. Peterson, and A. D. Rubin. Self-protecting electronic medical records using attribute-based encryption on mobile device. Technical report, Cryptologye Print Archive, Report 2010/565, 2010.http://eprint. iacr. org/2010/565.
7. L. Ibraimi, M. Asim, and M. Petkovic, "Secure management of personal health records by applying attribute-based encryption," Technical Report, University of Twente, 2009.