Person-wise Privacy Level Access Control for Personal Information Directory Services

Hyung-Jin Mun, Keon Myung Lee, and Sang-Ho Lee
School of Electrical and Computer Engineering,
Chungbuk National University, Korea**
hjmun@cbnu.ac.kr

Abstract. This paper proposes a policy-based access control mechanism for the personal information directory service systems which prevents the information users from illegally accessing the personal information and enables the information subjects to control access to their own information. In the proposed mechanism, the individuals' personal information which is encrypted with different keys is stored into the directory repository. In order to control access to her own personal information, information subject sets up the access control policy for it and the access control is practiced out by providing encryption keys to the legal users according to the subject's policy.

1 Introduction

With the advance of the information technology and the diversity of society, lots of personal information has been collected and distributed. To provide the quality services for the individuals, the organizations and the companies try to use the information of the customers, the employees, the partners, and so on. The collected personal information could be used without the permission of the information subjects. The unprotected personal information could infringe on personal rights and sometimes could financially damage the information subjects. Therefore, there have been increasing concerns to the individuals about their personal information. Now the personal information protection becomes one of important issues in the various computing and communication environments.[2-11]

In order to protect personal information, some organizations and companies employ the access control techniques. Sometimes the information subjects might want to control access to their personal information even though it is provided through the directory services. The traditional access control skills have difficulty in providing fine-grained person-wise access control to the personal information. This paper is concerned with the person-wise access control mechanism in which information subjects could determine who can access which personal information in what context.

In the proposed person-wise access control mechanism, the underlying idea is for each individual to use different encryption keys for each attribute of her/his

** This work was supported by the Regional Research Centers Program of the Ministry of Education & Human Resources Development in Korea.

own record. They write their own access policies and register them into the directory service system. The directory service system hands the encryption keys to the information users when they are allowed to access the designated personal information according to the information subjects' policy.

2 Related Works

Encryption techniques have been employed to protect the e-mail messages, stored files, on-line communication, and so on.[1-5] Once encryption is done, only the legal actors have the keys for the encrypted data and thus can access it.

The mandatory access control(MAC) is a technique that controls the users' access according to the rules enforced by the restricted number of security managers in the military or highly restricted environments. The discretionary access control(DAC) is a technique to enables the information owners to delegate or revoke the access rights to other users at their disposal. Therefore it is more flexible than MAC and can perform the distributed access control. However, both MAC and DAC is inappropriate to manage the various strategies of the organizations or the companies. Therefore, as an alternative, the role-based access control(RBAC) is proposed by R. Sandhu, et al.[6] RBAC takes into account the complicated structure of the organizations and could satisfy the different security and policy requirements. RBAC grants the access rights according to the users' role and restricts the access to the data which do not fit with the role of the users. The activity-based access control(ABAC) is an access control technique for the cooperative working environments such as workflows.[7] The task role-based access control(TRBAC) is an access control technique which combines both RBAC and TRBAC.[8,9,10] TRBAC fits with the organizations or the companies that have the complicated organizational structures, various kinds of users and information.

HP research center proposed a technique that protects a large volume of the personal information systematically stored in a database of the organizations or companies[11]. In the model, a specific piece of personal information is encrypted and stored into the database, and the Privacy Management Service module maintains the keys. The Privacy Management Service module provides the decryption keys to the users when they are allowed to access the requested data according to the enforced policy. In the model, an attribute shares the same encryption key across all records and thus it is impossible to control the access rights to the record attributes in a person-wise way. In addition, the allowed policy rules can hardly express contextual condition such as *John's current location information is available to his boss only on-duty.*

3 The Proposed Person-wise Access Control Model

The proposed person-wise access control model, named P^2MS(Person-wise Policy-based access Management System), allows the information subjects to control access to their own information according to their policy. The model takes the strategy to encrypt each attribute for each individual with different keys and to

endow the decryption keys for the allowed attributes to the information users according to the information subjects' policy.

3.1 The Application Environment

Individuals, i.e., information subjects(IS), is supposed to give their personal information to the companies or organizations. The personal information is managed by the database management system and published through the directory service system. Information users(IU) try to access the stored personal information(PI) on their own demand. The individuals write their own policy about who can access which pieces of their information in what context. The directory service system maintains the information subjects' policy and enforces them when an IU asks some pieces of PI. It is supposed that only allowed IU can access the allowed pieces of PI. Figure 1 shows the situations in which IUs access PI according to the endowed access rights by the information subjects' policy.

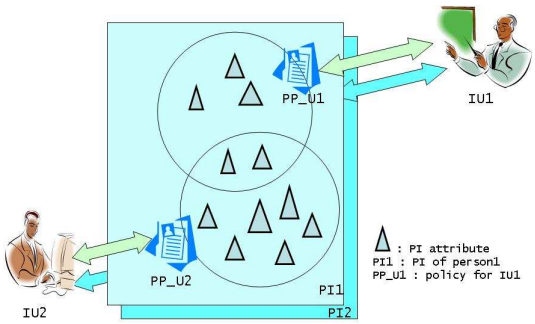

Fig. 1. Access Control of Information User for Personal Information

3.2 The Person-wise Privacy Level Access Control Model

The proposed model consists of the PI access client(PIA), the P^2MS, and the Data Repositories as shown in the Figure 2.

The PI access client An information subject provides her own information through PI access client and makes their policy to prevent illegal access of the information users. Through PI access client, IU accesses the information by receiving the authentication and requesting it. PI access client consists of Access Module, Decryption Module.

Access Module. The module accesses P^2MS, requests the information list, and then gets the encryption keys. It also plays the role of getting the encrypted information from the Data Repositories.

Decryption Module. The module is to decrypt the encrypted personal information with the received keys from P^2MS.

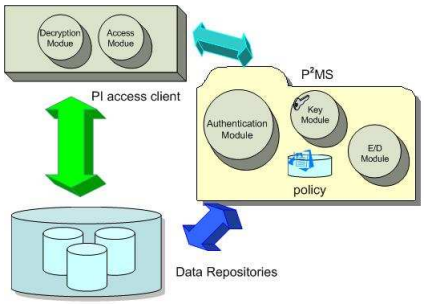

Fig. 2. The Proposed Person-wise Privacy Level Access Control Model

The P^2MS P^2MS plays the role of protecting the personal information according to personal policies written by the information subjects through PI access client. IUs and ISs can access P^2MS only through PIA. P^2MS consists of the Authentication Module, the Policy database, the Encryption/Decryption modules, and the Key module.

Authentication Module authenticates eligible information users according to information subjects' policy. There are two types of authentications one of which is for the information subjects and the other is for the information users.

Policy contains two types of policies. One is the access control policies written by the organizations, which are enforced by the customs and regulations to protect privacy. The other is the policies from information subjects to control access to their own personal information at will.

Encryption Module is the module that encrypts the information provided by the information subjects. It encrypts each information attribute using the keys generated in the Key module and then sends the encrypted information to Data Repositories.

Decryption Module is the module that decrypts the retrieved encrypted PI. When some person's policy or the organization's policy has been changed, the corresponding PI should re-encrypted with different keys. At that situation, P^2MS retrieves the encrypted PI to be re-encrypted from DR, and recovers and encrypts it with new keys, and then stores it to DR.

Key module comprises Key generation module and Key DB. In the proposed model, each field of a record is encrypted by different keys. Therefore, it is crucial to effectively maintain keys. *Key generation module* generates a master key for each person and generates attribute keys from the master key. *Key DB* is the storage that securely stores generated master keys.

Data Repositories The Data Repositories(DR) is in charge of storing the personal information. In the DR, each record of a person consists of the hashed value of her ID and the encrypted sentences for her PI. The attributes are encrypted with different keys which are maintained by P^2MS. When PI access client requests some personal information, DR sends the encrypted personal information for which decryption keys could be provided from P^2MS.

3.3 Data Format in Data Repositories

The personal information is stored into DR in an encrypted format. Each individual has a record to store her PI as shown in Figure 3(a). P^2MS encodes such records into encrypted records as shown in Figure 3(b), and then stores them into DR.

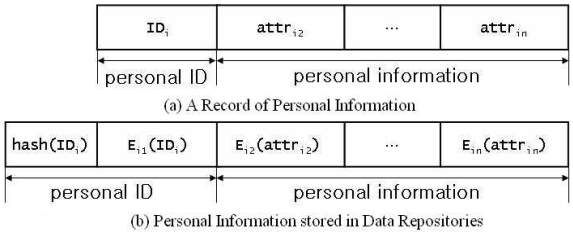

Fig. 3. Structure of Personal Information stored in the DB

Instead of IS's ID, ID_i, the pair of its hashed value ($hash(ID_i)$) and its encrypted data ($E_{i1}(ID_i)$) is stored. The hashed value plays the role of an index for the encrypted database, with which record retrieval is enabled. All the other personal information such as phone number, address and so on are encrypted using different key, respectively. IU can search the records with the hashed values of IS's ID of interest. Due to inherent properties of hashed functions, it is nearly impossible for the DB administrator to recover the real IDs from the encrypted records, and thus the information subject's privacy can be successfully protected.

4 The Protocols Employed in the Proposed Model

This section presents how the proposed method generates keys and maintains them, how information subjects register their personal information, how information users retrieve others' personal information, how information subjects modify their personal information stored in DR, the situations in which personal information should be re-encrypted, and what the policy looks like.

4.1 Key Management

To protect PI safely, the proposed model allows each attribute of PI to be encrypted with different keys. In the model, the key generation module generates a master key(K_{ID_i}) for each IS i, which is later used to build attribute keys for encrypting the attributes of IS i's PI. In the attribute key generation, both the PI's ID and random number R_j are used as follows: $key_{ij} = E_{K_{ID_i}}(ID_i|R_j)$. The generated attribute keys are symmetric cipher keys, which are advantageous in encryption time and key size, and only the master keys but attribute keys are stored in the Key DB. When an IU requests the keys for the allowed attributes, P^2MS provides the keys which are generated at the moment from the stored master key. The number of keys stored in Key DB is equal to that of subjects. The master keys are secure because they are used only in the attribute key generation. P^2MS provides the permitted IU with a ticket which contains the information about which attributes are allowed to retrieve and their valid access time period. The tickets are signed with the P^2MS's private key for their verification.

4.2 Personal Information Registration

An IS sends PIA her personal information for its registration. An IS writes her policy with which she controls access to her own PI. Figure 4 shows how an IS stores her PI in the DR and the following presents the steps:

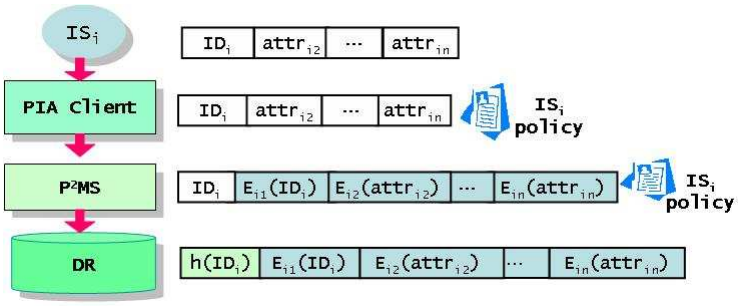

Fig. 4. Personal Information Registration

1. An IS i provides her information $DataSet_i$ for PIA.
   $ISi \rightarrow PIA : [DataSet_i]$ where $DataSet_i = (ID_i | \bigcup_{j=2}^{n} attr_{ij})$
2. IS i writes her access policy POL_i with help of PIA.
3. PIA sends both $DataSet_i$ and POL_i to P^2MS.
   $PIA \rightarrow P^2MS : [DataSet_i | POL_i]$
4. P^2MS encrypts each attribute value $attr_{ij}$ of $DataSet_i$ with a different key key_{ij} generated from the master key K_{ID_i} for IS i, computes the hashed value $h(ID_i)$ for ID_i, and in addition stores POL_i into the policy DB.
5. P^2MS stores the encrypted information into the DR.
   $P^2MS \rightarrow DR : [h(ID_i) | E_{K_{i1}}(ID_i) | \bigcup_{j=2}^{n} E_{K_{ij}}(attr_j)]$

4.3 Information Retrieval

When an IU intends to get the PI for a person ISs, the steps take place:

1. An IU u sends P^2MS the ID ID_s of the IS s and the attribute list AL_u of interest for IS s, and the access intent AI_u about how to use it, along with the her certificate $Cert_{ID_u}$ on her ID.
   $IU \rightarrow P^2MS : [Cert_{ID_u} | ID_s | AL_u | AI_u]$
2. P^2MS authenticates the ID of the IU.
3. P^2MS looks up the policies of the organization and that of ISs, and then decides which attributes fr_{AL} to be allowed to be accessible with reference to AL_u and AI_u at the moment, and the valid access time period TP_{su}. Then it encrypts fr_{AL}, TP_{su} with the P^2MS's private key KP_{P^2MS}, and issues a ticket $T = E_{KP_{P^2MS}}(fr_{AL} | TP_{us})$. Then, it encrypts the ticket T and the key list $KL_{fr_{AL}}$ for fr_{AL} with IU_u's key KS_{ID_u} which is shared with the PIA of IUu.
   $P^2MS : E_{KS_{ID_u}}(T | KL_{fr_{AL}})$
4. P^2MS sends the message $E_{KS_{ID_u}}(T | KL_{fr_{AL}})$ to PIA.
   $P^2MS \rightarrow PIA : [E_{KS_{ID_u}}(T | KL_{fr_{AL}})]$
5. PIA decrypts the received message with KS_{ID_u} and sends both IS's ID ID_s and ticket T to DR.
   $PIA \rightarrow DR : [ID_s | T]$

6. DR verifies the ticket's validity, and checks the requested attribute list AL_u and the ticket's valid period TP_{su}. And then it generates the hashed values $h(ID_s)$ of ID_s and searches the DB for the records with $h(ID_s)$.
7. DR sends PIA the retrieved records with $h(ID_s), E_{K_{s1}}(ID_s), E_{K_{sj}}(attr_j)$ for $attr_j \in fr_{AL}$ as far as the ticket T is valid.

$DR \rightarrow PIA : [h(ID_s), E_{K_{s1}}(ID_s), E_{K_{sj}}(attr_j)$ for $attr_j \in fr_{AL}]$

8. PIA decrypts the received records using the keys KL_{AL} and recovers the corresponding IS's PI.
9. PIA sends the recovered PI to the IU.

4.4 Personal Information Modification

When an information subject tries to modify her information, the following steps take place.
1. An IS s sends P^2MS her ID ID_s, her certificate $Cert_{ID_s}$ and the attribute list MR_{AL} to be modified.

$IS \rightarrow P^2MS : [Cert_{ID_s}|ID_s|MR_{AL}]$

2. P^2MS authenticates ID_s.
3. P^2MS sends PIA a ticket $T = E_{KP_{P^2MS}}(MR_{AL}|TP_{sp})$ and the keys $K_{MR_{AL}}$ for attributes in MR_{AL} which are encrypted with the ID_s's key KS_{ID_s} which is shared with P^2MS.

$P^2MS \rightarrow PIA : [E_{KS_{ID_s}}(K_{MR_{AL}}|T)]$

4. PIA sends ID_s, T, and MR_{AL} to DR.

$PIA \rightarrow DR : [ID_s|T|MR_{AL}]$

5. DR checks the validity of the ticket T. If so, retrieve the record(s) corresponding to the hashed value $h(ID_s)$.
6. DR sends PIA the retrieved record(s) corresponding to MR_{AL}.
7. PIA identifies the ID_s's record by decrypting the encrypted ID field with the key received from P^2MS.
8. PIA sends $ISID_s$ the ID_s's information record to modify.
9. IS ID_s modifies its information in the record and sends it to PIA.
10. ID_s sends P^2MS the modified record $newVMR_{AL}$ encrypted by the shared key KS_{sp}.

$IS \rightarrow P^2MS : [E_{KS_{sp}}(newVMR_{AL})]$

11. P^2MS generates a new key and encrypts the modified information from ID_S.
12. P^2MS sends the encrypted information to DR.

$P^2MS \rightarrow DR : [h(ID_s)|E_{K_{s1}}(ID_s)|\bigcup_{j=2}^{n} E_{K_{sj}}(attr_j)]$ where $attr_j$ indicates the updated one, if updated.

13. DR updates ID_s's information with the received one.

4.5 Re-encryption of Personal Information

When an IU requests some pieces of information for an individual, P^2MS provides the decryption keys for the allowed pieces. Once an IU has the keys, IU could retrieve the individual's corresponding data from DR at any time. Therefore, P^2MS takes charge of re-encrypting the PI at each expiration of issued tickets, to keep safe the PI.

4.6 Policy Registration

Each IS is supposed to write the access control policy about her own information. To write a policy effectively, the organizations or companies could provide subject with IS's group categories like Figure 5. An IS writes the group-wise access authorization policies as well as policies for some known users. When some policy for an individual user contradicts with that for a user group, the policy of individual users has a higher priority to group-wise policies.

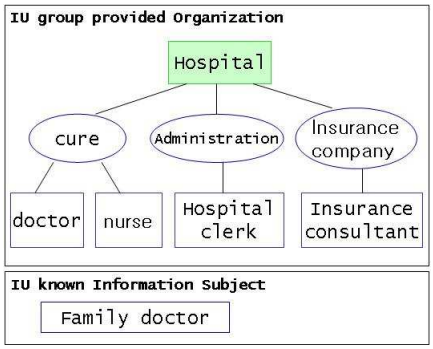

Fig. 5. IU group category

Figure 6 shows an example of person-wise policy using the group categories in a medical application. The policy tells that a family doctor can access such sensitive information as disease history, and so on.

```
- <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  - <xsd:element name="Hong's Policy">
    - <xsd:complexType>
      - <xsd:sequence>
          <xsd:element name="Info_user" type="xsd:string" />
          <xsd:element name="ID" type="xsd:string" />
          <xsd:element name="name" type="xsd:string" />
          <xsd:element name="tel" type="xsd:string" />
          <xsd:element name="health_checkup" type="xsd:string" />
          <xsd:element name="job" type="xsd:string" />
          <xsd:element name="sex" type="xsd:string" />
          <xsd:element name="medical_fee" type="xsd:string" />
          <xsd:element name="prescription" type="xsd:string" />
          <xsd:element name="disease_name" type="xsd:string" />
          <xsd:element name="disease_history" type="xsd:string" />
        </xsd:sequence>
      </xsd:complexType>
    </xsd:element>
  - <xsd:Policy Info_user="doctor">
    - <xsd:AttributeInformation>
        <xsd:Attribute ID="read" />
        <xsd:Attribute tel="read" />
        <xsd:Attribute disease_name="modify" />
        <xsd:Attribute job="read" />
        <xsd:Attribute health_checkup="read" />
      </xsd:AttributeInformation>
    - <xsd:Constraint>
        <xsd:Time>AM 9:00~PM 6:00</xsd:Time>
      </xsd:Constraint>
    </xsd:Policy>
  - <xsd:Policy Info_user="nurse">
    - <xsd:AttributeInformation>
        <xsd:Attribute ID="read" />
        <xsd:Attribute prescription="read" />
      </xsd:AttributeInformation>
    - <xsd:Constraint>
        <xsd:timeDuration>under medical treatment</xsd:timeDuration>
      </xsd:Constraint>
    </xsd:Policy>
  - <xsd:Policy Info_user="hospital clerk">
    - <xsd:AttributeInformation>
        <xsd:Attribute prescription="read" />
        <xsd:Attribute tel="read" />
        <xsd:Attribute medical_fee="write" />
      </xsd:AttributeInformation>
    </xsd:Policy>
  - <xsd:Policy Info_user="Insurance consultant">
    - <xsd:AttributeInformation>
        <xsd:Attribute name="read" />
        <xsd:Attribute tel="read" />
        <xsd:Attribute address="read" />
        <xsd:Attribute job="read" />
        <xsd:Attribute medical_fee="read" />
      </xsd:AttributeInformation>
    - <xsd:Constraint>
        <xsd:Time>AM 9:00~PM 6:00</xsd:Time>
      </xsd:Constraint>
    </xsd:Policy>
  - <xsd:Policy Info_user="family doctor">
    - <xsd:AttributeInformation>
        <xsd:Attribute ID="read" />
        <xsd:Attribute tel="read" />
        <xsd:Attribute health_checkup="modify" />
        <xsd:Attribute disease_history="modify" />
        <xsd:Attribute disease_name="write" />
      </xsd:AttributeInformation>
    - <xsd:Constraint>
        <xsd:IPaddress>192.168.0.100</xsd:IPaddress>
        <xsd:Time>AM 9:00~PM 5:00</xsd:Time>
        <xsd:timeDuration>2006.6.31</xsd:timeDuration>
      </xsd:Constraint>
    </xsd:Policy>
  </xsd:schema>
```

Fig. 6. Person-wise policy

5 Privacy Preservation Strength of the Model

In order to analyze the privacy preservation strength of the proposed model, we compare the proposed model with the HP model in the following four aspects:

First, the proposed model can fully respect the information subjects' intention in the access control of their personal information, whereas the HP model protects the personal information only according to the organization policy and thus it is very difficult to finely control the access to the personal information according to the information subjects' intention.

Second, the proposed model allows the elaborate access control to the personal information with the consideration of contextual information such as time, location, relationship, and so on. The HP model does the access control by the role and the duty, whereas the proposed model does the access control with the information subjects' policy as well as the organization policy.

Third, the proposed model can protect the personal information even from the database administrator since the key management part is separated from the database and the personal identity information is encrypted in the database. In the HP model, the identity information is recorded in a plain text in order to allow the search service.

Fourth, in the proposed model, each attribute of a record comes to have a different key even though all attribute encryption keys are generated from the master key of an individual. Therefore, even a key is exposed in public by accident or by incident, the damage incurred is so small. However, the HP model each attribute shares a key across the records of a table. Compared to the proposed model, the risk incurred by the loss of a key is severe.

6 Conclusion

The organizations and companies employ the encryption techniques and access control techniques to protect personal information maintained by themselves, yet the personal information subjects do not feel comfortable in that their information is stored somewhere. The privacy protection guidelines and regulation ask the information management authorities to get the permission from the information subjects, and to give the information subjects the access control rights to their own information.

In this paper we proposed an access control model in which the information subjects write the access control policy for their personal information and the policies are enforced in a secure way. In the model, each attribute of an information subject is encrypted with different encryption keys, respectively. The allowed information users can acquire the decryption keys for the allowed attributes according the information subject's policy. In order to improve the privacy preservation level, the key management part is separated from the database which stores the encrypted personal information. The personal identify information is also not exposed to the unintended information users, even to the database administrator. The proposed model has to maintain as many keys as the number of attributes times that of records. In order to resolve this problem, the model takes the approach to assign a single master key to each record and to generate

attribute encryption keys from the master key. This strategy avoids the potential key management problem. The proposed model is expected to be used in the personal information directory service systems such as e-government systems in which personal information need to be provided to authorized parties with the permission of the information subjects.

References

1. Stallings, W.: Cryptography and Network Security. 3rd edn. Prentice-Hall, New Jersey (2003)
2. Fischer-Hubner, S.: IT-Security and Privacy : Design and Use of Privacy Enhancing Security Mechanism. Lecture Notes in Computer Science, LNCS 1958 (May 2001)
3. Chaum, D.L.: The Dining Cryptographers Problem: Unconditional Sender and Recipient Untraceability. Journal of Cryptology. 1(1). (1988) 65-75
4. Rither, M.K., Rubin, A.D.: Crowds : Anonymity for Web Transactions. ACM Transactions on Informatino and System Security. 1(1). (1998) 66-92
5. Chaum, D.L.: Untraceble Electronic Mail Return Address, and Digital Pseudonyms. Communications of the ACM. 24(2) (1981) 84-88
6. Sandhu, R.S., Coyne, E.J., Feinstein, H.L., Youman, C.E.: Role-Based Access Control Models. IEEE Computer. 29(2) (1996) 38-47
7. Huang, W.K., Atluri, V.: SecureFlow: A secure Web-enabled Workflow Management System. Proc. Of 4th ACM Workshop on Role-based Access Control (1999)
8. Thomas, P.K., Sandhu, R.S. : Task-based Authorization Control(TBAC): A Family of Models for Active and Enterprise-oriented Authorization Management. Proc. of the IFIP WG11.3 Workshop on Database Security (1997)
9. Oh, S., Park, S.: An Integration Model of Role-based Access Control and Activity-based Access Control Using Task. Proc. of 14th Annual IFIP WG11.3 Working Conference on Database Security (Aug. 2000)
10. Oh, S., Park, S.: A Process of Abstracting T-RBAC Aspects from Enterprise Environment. Proc. DASFAA'01 (Apr. 2001)
11. Mont, M.C., Pearson, S., Bramhall. P.: An Adaptive Privacy Management System for Data Repositories. http://www.hpl.hp.com/techreports/2004/HPL-2004-211.html (2004)