XML データベースへの推論攻撃による機密情報特定可能性の形式化とある前提条件のもとでの特定可能性検証法の提案

A Formalization of the Identifiability of Secret Information by Inference Attacks on XML Databases and a Proposal of an Identifiability Verification Method under Some Conditions

高須賀 史和♦　橋本 健二♣
石原 靖哲♠　藤原 融♠

Fumikazu TAKASUKA　Kenji HASHIMOTO
Yasunori ISHIHARA　Toru FUJIWARA

推論攻撃とは，ユーザが許可されている問合せとその実行結果から，許可されていない問合せの実行結果（機密情報）を推論しようとすることである．本論文では，ユーザが許可されている複数の問合せとその結果から，機密情報の候補を絞り込む攻撃と，その攻撃により機密情報が特定される可能性を考える．そして，XML データベースへの推論攻撃による機密情報の特定可能性を形式的に検証する手法を提案する．

Inference attacks mean that an attacker tries to infer the execution result of a query unauthorized to the attacker (i.e., secret information) from the execution results of queries authorized to the attacker. In this paper, we consider attacks such that the attacker narrows the candidates for the secret information using more than one authorized queries and their execution results, and the possibility that the secret information can be identified by the attacks. Then, a method of verifying the identifiability by the inference attacks on XML databases is proposed.

1. まえがき

企業や組織は少なからず機密性の高い情報を保持している．そのような機密情報が格納されているデータベースシステムにおいて，セキュリティ面での管理が必ずしも正確に行われていないことが問題となっている．たとえば，複数のセキュリティレベルのユーザをもつデータベースシステムにおいて，アクセス権の付与が適切に行われておらず，本来権限のないユーザによる機密情報へのアクセスを許可してしまっている場合がある．また，機密情報への直接のアクセスを許可していない場合でも，アクセスを許可されている情報やドメイン知識などを基に，本来権限の無いユーザがその機密情報を推論できてしまう場合がある[1]．そのような推論を行うことを推論攻撃と呼ぶ．推論攻撃によりどのような情報が得られるかは一般に自明ではないため，データベース管理者としては，データベースの運用を開始する前，あるいはユーザからの問合せに対して実行結果を返す前に，機密情報に対して推論攻撃が成功する可能性をあらかじめ把握し，その推論を制御できることが重要である．

[例1] XMLデータベースにおける推論攻撃の例を示す．Dは学生名とその成績の対応を表すXML文書であり，以下のスキーマにしたがっているとする．

科目 → 学生*
学生 → 氏名, 点数
氏名 → PCDATA
点数 → PCDATA

すなわち，科目要素の子には学生要素が0個以上並び，各学生要素の子には氏名要素と点数要素が１つずつ並ぶ．氏名要素と点数要素は文字列（PCDATA）を値としてとる．

T_1を，もとの文書における出現順序を保存したまま，科目の子の学生の子の氏名だけを取り出す問合せとする．また，T_2を，もとの文書における出現順序を保存したまま，科目の子の学生の子の点数だけを取り出す問合せとする．

今，T_1およびT_2の実行が許可されており，実行結果$T_1(D)$, $T_2(D)$がそれぞれ図1, 2に示すとおりであったとする．T_1の定義および$T_1(D)$の値より，Dは科目の子の学生の子の氏名として“高須賀”という値の要素と“橋本”という値の要素をその順にもつということがわかる．同様に，T_2の定義および$T_2(D)$の値より，Dは科目の子の学生の子の点数として“90”という値の要素と“80”という値の要素をその順にもつということがわかる．そして，Dが上のスキーマにしたがっているということより，Dは図3に示す文書しかありえないと結論できる．□

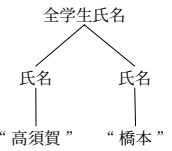

図1 実行結果$T_1(D)$
Fig.1 Execution Result $T_1(D)$

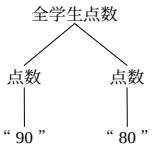

図2 実行結果$T_2(D)$
Fig.2 Execution Result $T_2(D)$

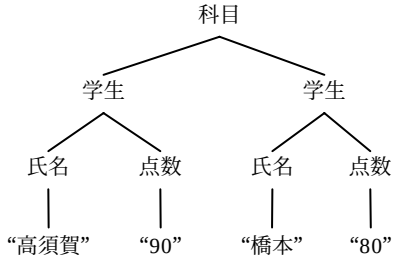

図3 Dとして推論される文書
Fig.3 Inferred Document for D

♦学生会員　大阪大学大学院情報科学研究科　博士前期課程
fmkz-tak@ist.osaka-u.ac.jp
♣大阪大学大学院情報科学研究科　博士後期課程
k-hasimt@ist.osaka-u.ac.jp
♠大阪大学大学院情報科学研究科
{ishihara,fujiwara}@ist.osaka-u.ac.jp

次に，本論文で扱う，機密情報の候補を絞り込む推論攻撃の概略を示す（図4参照）．ユーザはXML文書Dのある部分$T_S(D)$（ここでT_Sは問合せ）の値を知りたい．しかし，ユーザはD全体を直接見ることができない．ユーザが知ることが出来るのは，実行を許可されている問合せ$T_1,\ldots,T_p$の定義と，これらの問合せのDにおける実行結果$T_1(D),\ldots,T_p(D)$，およびDがしたがうスキーマGである．このとき，各i ($1 \leqq i \leqq p$)について，ユーザはT_iと$T_i(D)$とから，問合せT_iの実行結果として$T_i(D)$が得られるようなDを推論することにより，Dの候補を絞り込むことができる．すべてのiについてDの候補となっている文書のうち，Gにしたがう文書からなる集合が，Dの候補集合I_Dである．そして，どの候補$D' \in I_D$に対しても$T_S(D')$が同じ値になるのであれば，ユーザは$T_S(D)$の値を特定できる．

本論文では，まず，機密情報の特定可能性を形式的に定義する．問合せのクラスとしては，文献[2]で提案されているトップダウン木変換器を用いる．この木変換器はXSLTの部分クラスに相当する．次に，ある前提の下で機密情報の特定可能性を検証する手法を与える．本検証法の時間計算量は，実行が許可された問合せの個数pを定数とみなすと，決定性多項式時間である．

関連研究として，文献[3]では，各ユーザに security view を提供するアプローチを提案している．security viewは，ユーザがアクセス権をもつ情報をちょうど含んだ文書と，その文書のスキーマとから成る．問合せ言語としてはXPathを前提としている．元のスキーマからのビュースキーマの生成に関しては，推論攻撃の手がかりとなりうる要素名を付け替えたり削除したりするアルゴリズムを提案している．しかし，安全性の定義が不明確であるという問題点がある．一方，文献[4]は，推論攻撃を受けたとしても情報の漏洩なく公開できる部分文書を算出するアルゴリズムを提案している．攻撃者が推論に用いることが出来る情報は，スキーマ情報と関数従属性に関する情報である．本論文では関数従属性に関する情報を扱ってはいないが，[4]では考慮していない，複数の問合せ結果から機密情報の候補を絞り込む攻撃を扱うことができる．

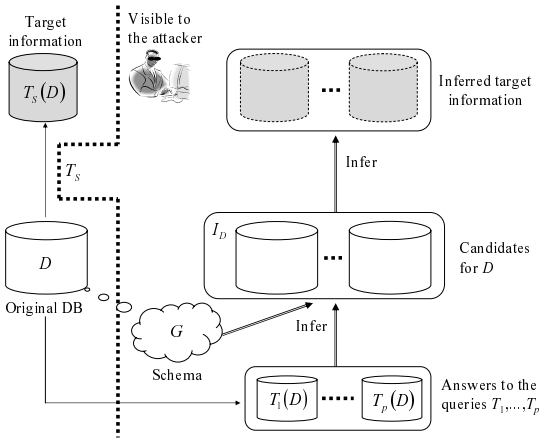

図 4　機密情報の候補を絞り込む推論攻撃
Fig.4　The Inference Attack Trying to Narrow the Candidates for Secret Information

2. 準備

木，生垣，非決定性トップダウン木オートマトン，決定性トップダウン木変換器に関する諸定義を行う．

2.1 木，生垣

あるアルファベットΣについて，Σ上の（ランクなし）木の集合T_Σと生垣の集合H_Σを，以下を満たす最小集合と定義する:

- $H_\Sigma = T_\Sigma^*$,
- $a \in \Sigma$,　$h \in H_\Sigma$であるならば，$a(h) \in T_\Sigma$.

ここでT_Σ^*はT_ΣのKleene閉包を表す．以下，空生垣（木の空系列）をεで表す．木において，空生垣を子としてもつ頂点をその木の葉頂点と呼ぶ．木t（あるいは生垣h）の大きさ$|t|$（あるいは$|h|$）はその木（あるいは生垣）の頂点の数である．

2.2 非決定性トップダウン木オートマトン

[定義1]　非決定性トップダウン木オートマトンは次の4つ組$A=(Q, \Sigma, q_0, R)$である:

- Q: 状態の有限集合,
- Σ: アルファベット,
- $q_0 \in Q$: 初期状態,
- R: 遷移規則の集合.

ただし$q \in Q$, $a \in \Sigma$とし，eをQ上の言語を受理する非決定性有限オートマトンとすると，遷移規則は$(q, a) \to e$の形式である．□

以下，木オートマトン$A=(Q, \Sigma, q_0, R)$の動作について述べる．$t=a(t_1 \cdots t_n)$とする．ここで，$t_1, \ldots, t_n \in T_\Sigma$である．木$t$の根頂点に状態$q$が割り当てられたとき$q(t)$と書く．$q_1 \cdots q_n \in L(e)$であるような$((q, a) \to e) \in R$が存在するならば，$A$は$q(t)$から$a(q_1(t_1) \cdots q_n(t_n))$へ遷移可能であると定義する．ここで$L(e)$は非決定性有限オートマトン$e$が受理する文字列言語である．$q_0(t)$から始めて，以上のような遷移をトップダウンに各部分木について繰り返すことによって，最終的にtへ遷移可能であるとき，Aはtを受理するという．Aによって受理されるすべての木の集合を$L(A)$と書く．

Aの大きさは$|Q|+|\Sigma|+\Sigma_{q \in Q,\, a \in \Sigma} |\mathrm{rhs}(q, a)|$である．ここで，$|\mathrm{rhs}(q, a)|$は$(q, a) \to e$について非決定性有限オートマトン$e$の大きさである．非決定性有限オートマトンの大きさも，同様に，状態数と記号数，そして各遷移規則における遷移先の状態集合の大きさの総和である．

木オートマトンはXMLスキーマの形式的モデルの１つであり，XML文書の“型”を表現するのに用いられる．ある木変換器Tと木オートマトンA_{in}について，$L(A'_{out}) = \{T(t) \mid t \in L(A_{in})\}$であるような木オートマトン$A'_{out}$を求めることを型推論という．逆に，ある木変換器$T$と木オートマトン$A_{out}$ について，$L(A'_{in}) = \{\, t \mid T(t) \in L(A_{out})\}$であるような木オートマトン$A'_{in}$を求めることを逆型推論という．

2.3 決定性トップダウン木変換器

Q中の記号が葉頂点のみに現れるようなΣ∪Q上の木集合を$T_\Sigma(Q)$と書く．$H_\Sigma(Q)$も同様に定義する．

[定義2]　決定性トップダウン木変換器は次の4つ組$T=(Q, \Sigma, q_0, R)$である:

- Q: 状態の有限集合,
- Σ: アルファベット,
- $q_0 \in Q$: 初期状態,
- R: 変換規則の集合.

ただし，$q \in Q$,∪{λ}, $h \in H_\Sigma(Q)$とすると，変換規則は$(q, a) \to h$の形式である．また$q=q_0$のときhは$T_\Sigma(Q) \setminus Q$の要素でなければならない．ここでλは空記号である．また，決定性トップダウン木変換器では，同じ左辺を持つ変換規則はちょうど1つである．また，$((q, \lambda) \to h) \in R$であるすべての$q \in Q$とすべての$a \in \Sigma$について，$((q, a) \to h) \notin R$であり，$((q, \lambda) \to h) \notin R$であるすべ

てのq∈Qとすべてのa∈Σについて, ((q, a)→h)∈Rである. □
決定性トップダウン木変換器T=(Q, Σ, q0, R)の動作について述べる. 木t = a(t1…tn)は, 状態q∈QでTによって次のようなTq(t)へ変換される:

- あるhに対して((q, λ)→h) ∈Rならば, Tq(t)は, h中のすべての状態記号pをTp(t)に置換して得られる生垣と等しい.
- あるhに対して((q, a)→h) ∈R (ただし, a∈Σ)ならば, Tq(t)は, h中のすべての状態記号pを生垣Tp(t1)…Tp(tn)に置換して得られる生垣と等しい. 特にt=a(ε)のときは, Tq(t)は, h中のすべての状態記号pをεに置換して得られる生垣と等しい.

tのTによる変換T(t)をT(t)= Tq0(t)と定義する. Tの大きさは |Q|+|Σ|+Σ q∈Q, a∈Σ |rhs(q, a)|である. ここで|rhs(q, a)| は((q, a)→h) ∈Rについて| h |である. 初期状態に関する変換規則の制約により, この変換器の出力が木であることが保証される.
また変換規則の中で, 右辺に状態が2つ以上存在する規則は, 規則を適用している頂点以下の部分木をコピーするのでcopying rule と呼び, 右辺にアルファベットを含まない規則は, 規則を適用している頂点を除去する変換を行うのでdeletion ruleと呼ぶ. そして, copying rule を持たない木変換器をnon-copying, deletion ruleを持たない木変換器をnon-deletingと呼ぶ.

3. 機密情報の特定可能性

本節では, 機密情報の特定可能性を定義する. ユーザに与えられる情報は以下の通りである.

- T1,...,Tp: ユーザが実行を許可されている問合せ.
- T1(D),...,Tp(D): Dに対する問合せT1,...,Tp の実行結果.
- AG: Dがしたがうスキーマ（を表す木オートマトン）.
- Ts : D中の機密情報を返す問合せ.

そしてこれらから定まる以下の集合

$I_S = \{T_S(D') \mid D' \in L(A_G), T_1(D') = T_1(D), \ldots, T_p(D') = T_p(D)\}$

が要素を 1 つしかもたないとき, Dの機密情報TS(D)は特定可能であるという.
たとえば[例 1]では, スキーマAGにしたがうXML文書の中で問合せT1の実行結果として図 1 に示すT1(D), T2の実行結果として図 2 に示すT2(D)が得られるようなDの候補D′は, T1, T2の定義より図 3 に示すものただ 1 つであると結論付けられる. そして, その文書における機密情報（この場合, 学生の氏名と点数の組）を返す問合せTS（この場合はTS(D′) = D′）を実行した結果, 得られる集合IS の要素もただ 1 つであるため, Dの機密情報TS(D)も特定可能である.

4. 特定可能性検証法

本節では, 本論文において提案する機密情報の特定可能性検証法について述べる.
機密情報の特定可能性の検証は次のように行う. まず, ユーザが実行を許可されている問合せの実行結果と D がしたがうスキーマから D の候補を求める. 次に, 得られた D の候補と, D の機密情報を返す問合せから D 中の機密情報の候補を求め, そして, その候補が 1 つであるかどうか, すなわち, 機密情報が特定可能であるかを判定する.
以降 4.1 節では本検証を行うための前提条件について, 4.2 節では本検証の手順, そして 4.3 節では本検証の時間計算量の評価について述べる.

4.1 特定可能性検証のための前提条件

ここでは検証を行うための前提条件について述べる. 本論文では, まずユーザが実行を許可されている各問合せTiとして, non-copying かつ non-deleting な木変換器Ti1とnon-copyingな木変換器Ti2の合成により定義されている問合せを考える. Ti1は, コピーや削除以外の変換を行いつつ, 後にTi2によって削除されるべき頂点のラベルを新たなラベル#に変換する機能を持つ. そしてTi2は#とラベル付けされた頂点を削除するだけの機能を持つ木変換器である. また, D中の機密情報を求める問合せTSとして, non-copyingな木変換器により定義されている問合せを考える.

4.2 特定可能性検証の手順

ここでは特定可能性検証の手順について述べる. 本検証法は以下の 5 つのステップにより構成される（図 5 参照）.

(1) 各問合せの実行結果Ti(D)から, Ti2(D)の出力としてTi(D)が得られるようなTi2への入力木の候補集合を受理するオートマトンAid を求める.
(2) ステップ(1)で得られた各木オートマトンAid と木変換器Ti1から, 逆型推論を用いて, 元のXML文書Dの候補を受理する木オートマトンAiD を構成する.
(3) ステップ(2)までで求められたすべてのAiD とDがしたがっているスキーマを表す木オートマトンAGから, Dの候補を受理する木オートマトンADを求める.
(4) ステップ(3)で求められたADと機密情報を求める木変換器TSから型推論を用いて, XML文書D中の機密情報TS(D)の候補を受理する木オートマトンASを求める.
(5) ステップ(4)で求められたASが受理する木がただ 1 つであるかを判定する.

これにより機密情報TS(D)が特定可能であるかどうかを判定する.

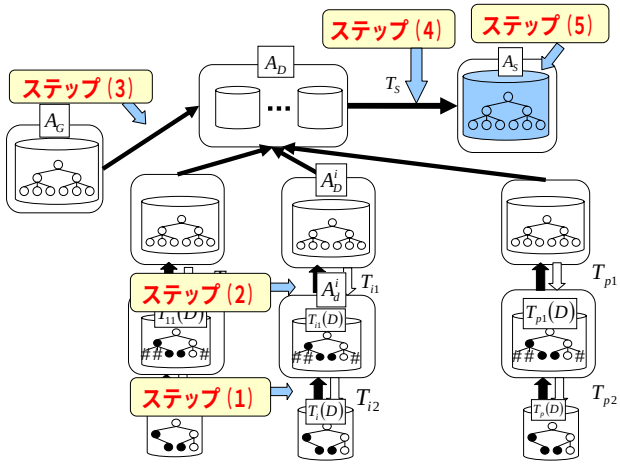

図 5 機密情報の特定可能性検証
Fig.5 Identifiability Verification of Secret Information

4.3 時間計算量の評価

ここでは, 本検証法の時間計算量について述べる.
まずステップ(1)の計算については, この計算量は|Ti(D)|についてのLOGSPACEである[6]. またここで求められるAid の大きさは, |Ti(D)|の多項式の大きさとなる.
次に, ステップ(2)の計算は, non-copying かつ non-deletingの木変換器Ti1の各変換規則につき, その右辺の記号を左辺に持つAid の遷移規則を選び, それらから新たに

遷移規則を作るという作業を繰り返すことから，時間計算量は$O(|T_{i1}|\times|A^{i}_{d}|)$となる．

ステップ(3)の計算の時間計算量は，$O(|A^{1}_{D}|\times\cdots\times|A^{p}_{D}|)$となる．$p$を定数とすると多項式時間となる．$|A_D|$についても，状態集合の大きさが各オートマトンの状態集合の大きさの積となるので，これも時間計算量と同様，pを定数とすると多項式の大きさとなる．

ステップ(4)の計算は，ステップ(2)を求める計算と同様，各変換規則と遷移規則から新たな遷移規則を作るという作業になるため，この時間計算量は$O(|T_S|\times|A_D|)$となる．

最後にステップ(5)の計算について考える．このステップでは，まずトップダウンの木オートマトンを等価なボトムアップの木オートマトンに書き換える．この時間計算量は木オートマトンのサイズについて線形時間である．次に，いま得られたオートマトンを非決定性（ボトムアップ，ランクなし）のものから，二分木オートマトンへの変換を行う．この実行時間は多項式時間であることがわかっている[7]．そしてこのボトムアップ二分木（ランクあり）オートマトンが受理する木の集合が要素をただ 1 つだけもつかどうかを判定するが，この計算も実行時間は多項式時間である[5]．よってステップ(5)の計算は$|A_S|$についての多項式時間である．

以上より，本検証法の時間計算量はA_Dを求める際に交わりを求める木オートマトンの数（すなわちユーザが実行を許可されている問合せの個数）pを定数とみなしたとき，多項式時間である．

5．あとがき

本論文では XML データベースにおける推論攻撃による機密情報の特定可能性を形式的に定義し，ある前提条件の下でそれを検証する手法を提案した．本検証法の時間計算量は，ユーザが実行を許可されている問合せの個数を定数とみなしたとき，決定性多項式時間である．

今後の課題としては，まず，検証のための前提条件の緩和が挙げられる．本検証のための前提条件では変換規則への制限が多い．たとえば，変換規則の右辺に生垣や枝分かれのある木を許していないため，問合せに対して新たな追加情報を加えて返すような動作ができない．そこで，この制限を緩めることについて検討をはじめている．

また，問合せクラスの拡張も課題として挙げられる．たとえば「A という子要素をもつときかつそのときのみ，親要素を削除する」ような動作ができない．そこで，子や孫への先読み機能をもつ木変換器に拡張することについて検討をはじめている．

次に，本論文で取り上げている機密情報の特定可能性とデータベースの安全性についての議論があげられる．機密情報が1つに絞り込まれる（特定される）可能性を本研究では考えているが，一方，機密情報が1つに絞り込まれない場合においても，以下のような両極端な状況が存在しうる．

- 機密情報の候補の個数が数個にしかならない(特定されないまでも攻撃の効果が非常に高い)
- 機密情報の候補の個数が非常に多い(攻撃の効果がない，あるいは非常に低い)

実用上，これら両極端な状況は区別されるべきケースが多いと考えられ，今後，この間に安全かそうでないかの線引きを行うことが課題となる．そこで，以下のような問題について検討をはじめている．

- 機密情報の候補が有限個になるかを検証する．
- ある整数のパラメータkを与え，機密情報の候補がk個以下になるかどうかを検証する．

他に，推論に用いることのできる情報として，[4]のような関数従属性を考慮する必要があると考えられる．また，本論文は XML 文書Dが具体的に与えられている場合での安全性検証を扱っているが，静的な特定可能性検証，すなわち，ユーザに実行を許可する問合せと，スキーマ情報から，機密情報が特定可能であるような文書が存在するか否かを検証する問題も，実用上極めて重要な課題である．

[文献]

[1] C. P. Pfleeger and S. L. Pfleeger: “Security in Computing,” 3rd Ed., Prentice Hall (2002).

[2] W. Martens and F. Neven: “Frontiers of Tractability for Typechecking Top-Down XML Transformations,” Proceedings of the 23rd ACM SIGMOD SIGACT SIGART symposium on Principles of database systems, pp. 23-34 (2005).

[3] W. Fan, C.-Y. Chan, and M. Garofalakis: “Secure XML Querying with Security Views,” Proceedings of the 23rd SIGMOD International Conference on Management of Data, pp. 587-598 (2004).

[4] X. Yang and C. Li: “Secure XML Publishing without Information Leakage in the Presence of Data Inference,” Proceedings of the 30th VLDB Conference, pp. 96-107 (2004).

[5] H. Comon, M. Dauchet, R. Gilleron, F. Jacquemard, D.Lugiez, S. Tison, and M. Tommasi: “Tree Automata Techniques and Applications,” http://www.grappa.univ-lille3.fr/tata/.

[6] W. Martens and F. Neven: “On the Complexity of Typechecking Top-Down XML Transformations,” Theoretical Computer Science, Volume 336, Issue 1, pp. 153-180 (2005).

[7] F. Neven: “Automata theory for XML researchers,” ACM SIGMOD Record Volume 31, Issue 3, pp. 39-46 (2002).

高須賀 史和 Fumikazu TAKASUKA

大阪大学大学院情報科学研究科博士前期課程在学中．2005 神戸大学工学部情報知能工学科卒業．XML データベースやデータベースセキュリティに関心をもつ．

橋本 健二 Kenji HASHIMOTO

大阪大学大学院情報科学研究科博士後期課程在学中．2006 大阪大学大学大学院情報科学研究科博士前期課程修了．XML データベースやデータベースセキュリティに関心をもつ．

石原 靖哲 Yasunori ISHIHARA

大阪大学大学院情報科学研究科助教授．1990 大阪大学基礎工学部情報工学科卒．1992 同大学院基礎工学研究科博士前期課程了． 1994 同大学院基礎工学研究科博士後期課程退学．同年より奈良先端科学技術大学院大学情報科学研究科助手．博士(工学)．XML データベースやデータベースセキュリティに関心をもつ．ACM, IEEE, 情報処理学会, 電子情報通信学会各会員．

藤原 融 Toru FUJIWARA

大阪大学大学院情報科学研究科教授．

1981 大阪大学基礎工学部情報工学科卒．1986 同大学院博士課程了．博士(工学)．情報セキュリティ，符号理論等の研究に従事．ACM， IEEE, 情報処理学会, 電子情報通信学会各会員．