コンピュータセキュリティ　14−20
（2001．7．25）

タイムスタンプ方式における10の安全性クラス

宇根　正志[†*]
une@mlab.jks.ynu.ac.jp

松本　勉[†‡]
tsutomu@mlab.jks.ynu.ac.jp

†横浜国立大学大学院
工学研究科
〒240-8501 横浜市保土ヶ谷区
常盤台 79-5

‡横浜国立大学大学院
環境情報研究院
〒240-8501　横浜市保土ヶ谷区
常盤台 79-7

*日本銀行金融研究所
〒103-8660　東京都中央区
日本橋本石町 2-1-1

あらまし

タイムスタンプ技術は，特定のデータが特定の日時・時刻に存在したことを証明する技術である．近年の電子商取引の拡大等に伴い，データを長期間保管する技術として，その重要性に対する認識が高まっている．本稿では，宇根・松本[9, 10]が提案したタイムスタンプ方式の枠組みに基づき，タイムスタンプの改ざんを検出するための十分条件について検討する．まず，各タイムスタンプ方式の十分条件を明らかにし，10通りの十分条件が存在することを示す．次に，同じ十分条件をもつタイムスタンプ方式の集合をクラスと定義し，各クラスに対応する十分条件の相互関係を明らかにする．最後に，本検討結果を6つの主要なタイムスタンプ方式に適用する．

キーワード　タイムスタンプ，安全性クラス，安全性評価

Ten Security Classes of Time Stamping Schemes

Masashi Une[†*]
une@mlab.jks.ynu.ac.jp

Tsutomu Matsumoto[†‡]
tsutomu@mlab.jks.ynu.ac.jp

† Graduate School of Engineering,
Yokohama National University
79-5 Tokiwadai, Hodogaya,
Yokohama, Kanagawa, 240-8501

‡ Graduate School of Environment
and Information Sciences,
Yokohama National University
79-7 Tokiwadai, Hodogaya,
Yokohama, Kanagawa, 240-8501

*Institute for Monetary and
Economic Studies,
Bank of Japan
2-1-1, Nihonbashi-
Hongokucho, Chuo, Tokyo,
103-8660

Abstract

Time stamping is a technique to prove the existence of certain digital data prior to specific point in time. Under recent expansion of electronic commerce, it has been widely realized to be a crucial technique to ensure integrity of digital data for a long time. This paper discusses sufficient conditions to detect alteration of time stamps on the basis of the framework for time stamping schemes proposed by Une and Matsumoto [9, 10]. First, we clarify the sufficient condition in each time stamping scheme and show that there exist ten varieties of the conditions. Secondly, we define a "class" as a set of the schemes having the same sufficient condition and clarify relationships between the conditions. Finally, we apply our result to six existent time stamping schemes.

key words　time stamp, security class, security evaluation

1 はじめに

タイムスタンプ技術は特定のデータが特定の日時・時刻に存在したことを証明する技術である．近年電子商取引や電子文書管理を進める動きが活発化する中，電子文書やその取扱履歴を長期間保管する技術として注目されている．こうしたニーズを背景に，わが国では法務省が電子公証制度[4]の検討を進めているほか，時刻署名分散システム[8]，Cuculus[1, 2]，PKITS[3]，TIMESEC[6]等のタイムスタンプ方式が提案されている．また，Digital Notary[7]/SecureSeal[5]等の商用サービスも開始されている．

様々なタイムスタンプ方式が提案される中，それらの安全性確保は重要な課題であり，タイムスタンプ方式の安全性評価に関する研究が必要となっている．こうした問題意識から，宇根・松本[9]は，タイムスタンプ方式の体系的な評価を行う際の土台となる概念整理を行い，タイムスタンプの定義やエンティティを定め，タイムスタンプ方式の分類方法を提案した．また，宇根・松本[10]は，宇根・松本[9]をベースとしてタイムスタンプの検証手続を定義し，タイムスタンプ方式を 108 種類に分類したほか，各方式におけるタイムスタンプの改ざんに対する安全性について検討を行った．

本稿では，これらの研究成果を土台として，タイムスタンプの改ざんに対する安全性について更なる考察を行う．宇根・松本[10]では，各タイムスタンプ方式において，検証者がタイムスタンプの改ざんを検出するための十分条件を示した．しかし，それらを整理し，十分条件の相互関係や，各十分条件に対応するタイムスタンプ方式を明確にするという課題が残されていた．本稿では，これらの課題を検討し，その検討結果をタイムスタンプ方式の安全性評価にどのように適用するかについて検討する．

本稿の構成は以下のとおりである．まず 2 において，宇根・松本[9, 10]が提案したタイムスタンプ方式の枠組みを説明する．3 では，宇根・松本[10]に基づき，各タイムスタンプ方式におけるタイムスタンプの改ざんを検出するための十分条件を検討・整理し，十分条件の相互関係や，各十分条件に対応するタイムスタンプ方式を明らかにする．4 では，6 つの主なタイムスタンプ方式に対して 3 の検討結果を適用し，タイムスタンプ方式の安全性評価を行う際に本検討結果をどのように活用するかを説明する．5 では，検討結果を整理し，今後の課題を示す．

2 タイムスタンプ方式の枠組み

2.1 エンティティ

- **発行者**（time stamp issuer）：タイムスタンプを発行し，タイムスタンプやその発行・検証に用いられるすべてのデータを保管するエンティティ．複数のエンティティが協力して 1 つのタイムスタンプを発行する場合，それらを 1 つの発行者とみなす．また，タイムスタンプの検証に利用される 2 種類のデータ E_{TSI} と E_{ORE} を生成する場合がある．E_{TSI} は，タイムスタンプを構成するデータと発行者のデータベースのデータとの整合性を確認するデータであり，“TSI” は “Time Stamp Issuer” を意味する．E_{AMP} は E_{TSI} の一貫性を確認するデータであり，発行者は後述の証拠補強者に E_{AMP} を送る．“AMP” は “AMPlifier” を意味する．
- **検証者**（verifier）：タイムスタンプや他のエンティティから得たデータを用いて，あるデータ M が T の時点に存在したことを確認するエンティティ．
- **証拠補強者**（evidence amplifier）：E_{AMP} を発行者から入手・保管し，検証時に検証者に提供するエンティティ．E_{AMP} を安全に保管することによって E_{TSI} の証拠性を補強する役割をもつ．例えば，E_{AMP} を新聞に掲載する場合，新聞が証拠補強者となる．
- **発行依頼者**（time stamp requester）：あるデータ M に対するタイムスタンプの発行を発行者に依頼するエンティティ．また，タイムスタンプに E_{ORE} が含まれる場合，別のタイムスタンプの検証時に E_{ORE} を検証者に送る場合がある．E_{ORE} は，E_{TSI} の一貫性を確認するデータであり，検証対象のタイムスタンプとは別のタイムスタンプに含まれる．“ORE” は “Other REquester” を意味し，検証対象のタイムスタンプをもつ発行依頼者とは別の発行依頼者が保管することを示す．
- **証明者**（prover）：あるデータ M が T の時点に存在したことを証明するために，M に対するタイムスタンプを検証者に送るエンティティ．一般に，証明者は発行依頼者と同一となる．

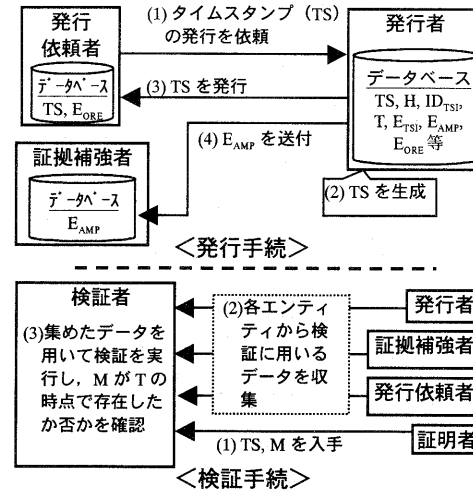

図 1 タイムスタンプの発行・検証手続

2.2 タイムスタンプを構成するデータ

タイムスタンプを，「特定のデータが特定の日時・時刻に存在したことを証明する目的で生成され，少なくとも H と ID_{TSI} を含むデータ」と定義する．タイムスタンプの対象となるデータを M，そのハッシュ値を H，発行者の識別データを ID_{TSI} とする．また，以下で説明する T，$Info_{INT}$，E_{TSI}，E_{ORE}，ID_{REQ}，ID_{AMP}，ID_{ORE} の各データもタイムスタンプを構成する場合がある．

- T：発行者が発行依頼者からタイムスタンプ発行要求データ REQ（H や発行依頼者の識別データ ID_{REQ} 等から

構成）を受信する日時・時刻のデータ．
- $Info_{INT}$：タイムスタンプを構成するデータのうち，$Info_{INT}$ を除くデータを用いて生成され，それらの一貫性を確認するデータ．例えば，$Info_{INT}$ としてディジタル署名が考えられる．
- ID_{REQ}：発行依頼者の識別データ．
- ID_{AMP}：証拠補強者の識別データ．
- ID_{ORE}：E_{ORE} を有する発行依頼者の識別データ．

2.3 発行手続（図 1 上参照）

(1)発行依頼者は REQ を発行者に送付．
(2)発行者はタイムスタンプを生成．
(3)発行者は発行依頼者にタイムスタンプを送付．
(4)発行者は E_{AMP} を証拠補強者に送る場合もある．

2.4 検証手続（図 1 下参照）

(1)証明者は，少なくともデータ M と M に対応するタイムスタンプを検証者に送付．
(2)検証者は，各エンティティから検証に用いるデータを収集．
(3)検証者は，一定の検証手続を実行し，その結果からデータ M が T の時点に存在したか否かを確認．

2.5 検証手続を構成する 6 つの処理

検証手続を構成する処理として，次の 6 つの処理 **a**～**f** を定義する．

- 処理 **a**：検証者がタイムスタンプを構成するハッシュ値 H とデータ M のハッシュ値を比較する．
- 処理 **b**：検証者が $Info_{INT}$ を用いてタイムスタンプを構成するデータ（$Info_{INT}$ を除く）の一貫性を確認する．
- 処理 **c**：検証者はタイムスタンプを発行者に送り，発行者が，そのタイムスタンプと自分が保管するデータとの整合性を確認し（例えば，発行者が保管するタイムスタンプとの比較を行う），その結果を検証者に通知する．T がタイムスタンプを構成するデータでない場合には，発行者は T も検証者に送る．
- 処理 **d**：検証者は，発行者から得る E_{TSI} を用いて，タイムスタンプと発行者のデータベースのデータとの整合性を確認する．
- 処理 **e**：検証者は，証拠補強者から得る E_{AMP} を用いて E_{TSI} の一貫性を確認する．
- 処理 **f**：検証者は，発行依頼者から得る E_{ORE} を用いて E_{TSI} の一貫性を確認する．

これらのうち，処理 **a** は，M とタイムスタンプの対応関係を確認するものであり，必須の処理となる．したがって，5 種類の処理 **b**～**f** の組み合わせによって 32 通りの検証手続が想定される．以下では，処理を表すアルファベットを並べて検証手続を示す．例えば，**ade** は処理 **a**，**d**，**e** を実行する検証手続を表す．

2.6 タイムスタンプ方式の分類

タイムスタンプを構成するデータの種類，検証者による

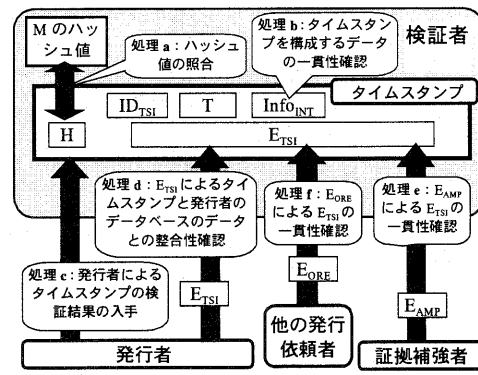

図 2　検証手続を構成する 6 つの処理

E_{TSI} の入手可能性，タイムスタンプの生成方法の観点から，タイムスタンプ方式を 10 のグループに分類する．

まず，タイムスタンプが T と E_{TSI} を含まない時刻・証拠無方式（NN，No time information and No evidence），T を含むが E_{TSI} を含まない時刻付・証拠無方式（TN，Time information and No evidence），T と E_{TSI} の両方を含む時刻・証拠付方式（TE，Time information and Evidence）の 3 つに分類する．第二に，検証者が E_{TSI} を取得可能な取得型方式（A，evidence-Available scheme）と，E_{TSI} を取得不可能な非取得型方式（U，evidence-Unavailable scheme）に分類する．第三に，他のタイムスタンプを構成するデータを用いて生成する連鎖型方式（L，Linked time stamp scheme）と，他のタイムスタンプを構成するデータを用いずに生成する個別型方式（I，Isolated time stamp scheme）に分類する．

このようにすると，タイムスタンプ方式は NN-U-L，NN-U-I，NN-A-L，NN-A-I，TN-U-L，TN-U-I，TN-A-L，TN-A-I，TE-A-L，TE-A-I の 10 のグループに分類される（表 1 参照）．例えば，NN-U-L は，時刻・証拠無－非取得型－連鎖型のタイムスタンプ方式のグループを表す．

次に，各グループに対して適用可能な検証手続を調べる．処理 **a** は，すべてのグループの検証手続に必須である．処理 **b** は，すべてのグループの検証手続に適用可能である．処理 **c** は，時刻・証拠無方式のグループ（NN-U-I，NN-U-L，NN-A-I，NN-A-L）において，T を入手する必要があることから必須であり，それ以外のグループでは必須でなく適用可能である．処理 **d**，**e**，**f** は，時刻・証拠無－非取得型方式のグループ（NN-U-I，NN-U-L）と時刻付・証拠無－非取得型方式のグループ（TN-U-I，TN-U-L）で，検証者が E_{TSI} を入手不可能なため，適用不可能である．さらに，処理 **f** は，連鎖型方式のグループ（NN-U-L，NN-A-L，TN-U-L，TN-A-L，TE-A-L）にのみ適用可能である．これらを基に各グループで適用可能な検証手続を整理すると，タイムスタンプ方式は 108 に分類される（表 1 参照）．

3 タイムスタンプ改ざん検出の十分条件

本節では，宇根・松本[10]に基づき，各タイムスタンプ方式を対象に，検証者がタイムスタンプの改ざんを検出するための十分条件を検討・整理する．

表1　10のグループと各グループに適用可能な検証手続

グループ	分類の観点			各グループに適用可能な検証手続
	ﾀｲﾑｽﾀﾝﾌﾟに含まれるﾃﾞｰﾀによる分類	E_{TSI}の取得可能性による分類	ﾀｲﾑｽﾀﾝﾌﾟの生成方法による分類	
NN-U-I	NN	U	I	**ac, abc**
NN-U-L			L	**ac, abc**
NN-A-I		A	I	**ac, abc, acd, abcd, acde, abcde**
NN-A-L			L	**ac, abc, acd, abcd, acde, acdf, abcde, abcdf, acdef, abcdef**
TN-U-I	TN	U	I	**a, ab, ac, abc**
TN-U-L			L	**a, ab, ac, abc**
TN-A-I		A	I	**a, ab, ac, ad, abc, abd, acd, ade, abcd, abde, acde, abcde**
TN-A-L			L	**a, ab, ac, ad, abc, abd, acd, ade, adf, abcd, abde, abdf, acde, acdf, adef, abcde, abcdf, abdef, acdef, abcdef**
TE-A-I	TE	A	I	**a, ab, ac, ad, ae, abc, abd, abe, acd, ace, ade, abcd, abce, abde, acde, abcde**
TE-A-L			L	**a, ab, ac, ad, ae, af, abc, abd, abe, abf, acd, ace, acf, ade, adf, aef, abcd, abce, abcf, abde, abdf, abef, acde, acdf, acef, adef, abcde, abcdf, abcef, abdef, acdef, abcdef**

3.1 前提

次の4つの前提を置く．第一に，攻撃には様々なものが想定されるが，本稿では最も基本的な攻撃であるタイムスタンプの改ざんを前提とする．つまり，攻撃者は，自分のタイムスタンプのハッシュ値Hを，別のデータM'のハッシュ値H'に改ざんし，M'に対応するタイムスタンプTS'を偽造するとする．

第二に，タイムスタンプの生成時に用いられるハッシュ関数のsecond-preimageの探索が計算量的に困難とする．

第三に，処理 **c**，**d**，**e**，**f** で用いられる暗号技術には，攻撃および検証の時点でセキュリティ上の問題が存在しないとする．

第四に，各エンティティ間で交信されるデータは守秘性と一貫性が確保されるとする．

3.2 攻撃の条件

第一に，$Info_{INT}$を生成する暗号技術（以下，$Info_{INT}$生成技術と呼ぶ）の安全性が低下する場合と低下しない場合を想定する．$Info_{INT}$生成技術の安全性が低下するとは，$Info_{INT}$生成技術に致命的な欠陥が存在し，発行者が有する秘密のデータ（例えば署名生成鍵）を知らなくても$Info_{INT}$を効率的に偽造できる状態にあるとともに，攻撃者だけがその方法に気づいている場合を意味する．一方，$Info_{INT}$生成技術の安全性が低下しないとは，$Info_{INT}$生成技術に致命的な欠陥が存在しない状態にある場合，または，欠陥が存在したとしても，攻撃者がその方法に気づかない場合を意味する．

第二に，攻撃者は，発行者，証拠補強者，他の発行依頼者と結託する場合としない場合を想定する．結託する可能性は，各エンティティの属性，エンティティが複数の場合にはその数等に依存する．本稿では，各エンティティとの結託可能性を同一とみなす．

最後に，攻撃者が，検証者に対して，各エンティティになりすます場合となりすまさない場合を想定する．処理bを実行しない場合や，処理bを実行するものの$Info_{INT}$生成技術の安全性が低下する場合，攻撃者は，ID_{TSI}，ID_{AMP}，ID_{ORE}の改ざんによって各エンティティになりすますケースが考えられる．この場合，検証者は，なりすます攻撃者に対して処理cの実行やE_{TSI}の送付を要求し，攻撃者に都合のよい検証結果やデータを入手することになる．なお，攻撃者による各エンティティへのなりすましの可能性は同一とみなす．

3.3 各方式における十分条件の検討

宇根・松本[10]では，108のタイムスタンプ方式における十分条件を逐次検討している．ただし，その方法では十分条件の導出の手間が膨大となるため，ここでは，各方式の十分条件が検証手続に依存するという宇根・松本[10]の検討結果を利用し，比較的シンプルな検証手続を用いる方式の十分条件を最初に検討した上で，より複雑な検証手続を用いる方式の十分条件を検討する．具体的には，必須である処理 **a** のみを用いる方式，および，処理 **a** に他の1つの処理を加える検証手続 **ab**，**ac**，**ad**，**ae**，**af** をそれぞれ用いるタイムスタンプ方式を最初の検討対象とする．

以下では，同一の検証手続を用いるタイムスタンプ方式の集合をタイプと定義する．例えば，検証手続 **abde** を用いる方式をまとめて「タイプ **abde**」と呼ぶ．

3.3.1 タイプ a の場合

検証者は，処理 **a** において，M'のハッシュ値H'とTS'を構成するハッシュ値を比較する．TS'にはH'が含まれており，2つのハッシュ値は必ず一致する．このため，いかなる条件の下においても，検証者は処理 **a** によって改ざんを検出不可能である．

3.3.2 タイプ ab の場合

3.3.1の結果から，処理 **a** においては，検証者はタイムスタンプの改ざんを検出不可能である．したがって，検証者が処理 **b** において改ざんを検出する十分条件を検討すればよい．以下で検討するタイプ **ac**，**ad**，**ae**，**af** についても同様である．

まず，$Info_{INT}$生成技術の安全性が低下しないケースを検討する．攻撃者が発行者と結託しない場合，攻撃者がTS'と整合的な$Info'_{INT}$を偽造困難であり，検証者は処理 **b** において改ざんを検出する．攻撃者が発行者と結託する場合，

その発行者が TS'と整合的な $Info'_{INT}$ を生成するため，検証者は改ざんを検出不可能である．

次に，$Info_{INT}$ 生成技術の安全性が低下する場合，攻撃者は TS'と整合的な $Info'_{INT}$ を偽造するため，検証者は処理 **b** においても改ざんを検出不可能である．

以上より，$Info_{INT}$ 生成技術の安全性が低下しない，かつ，攻撃者が発行者と結託しないならば，検証者は改ざんを検出する．

3.3.3 タイプ ac の場合

攻撃者が発行者と結託する場合，発行者は，処理 **c** における検証者からの TS'に対する検証依頼に対して，TS'が自分のデータベースと整合的であるとの検証結果や T を返す．このため，検証者は処理 **c** において改ざんを検出不可能である．また，攻撃者が発行者になりすます（ID'_{TSI} を改ざんする）場合も，検証者は，発行者になりすました攻撃者から同様の検証結果を得るため，改ざんを検出不可能である．

一方，攻撃者が発行者と結託しない，かつ，発行者になりすまさない場合，検証者は，発行者から TS'が自分のデータベースと整合的でないとの検証結果を得るため，改ざんを検出する．

この結果，攻撃者が発行者と結託しない，かつ，発行者になりすまさない場合，検証者は改ざんを検出する．

3.3.4 タイプ ad の場合

攻撃者が発行者と結託する場合，発行者は処理 **d** において TS'と整合的な E'_{TSI} を生成して検証者に送るため，検証者は改ざんを検出不可能である．また，攻撃者が発行者になりすます場合も，検証者は，攻撃者から E'_{TSI} を得るため，改ざんを検出不可能である．

一方，攻撃者が発行者と結託しない，かつ，発行者になりすまさない場合，検証者は発行者から E'_{TSI} を得ることはなく，検証者は改ざんを検出する．

この結果，攻撃者が発行者と結託しない，かつ，発行者になりすまさない場合，検証者は改ざんを検出する．

3.3.5 タイプ ae, af の場合

処理 **e**，**f** では，検証者が E_{TSI} の一貫性をそれぞれ E_{AMP} と E_{ORE} を用いて確認する．攻撃者は，H を H'に改ざんするものの，検証者が E_{TSI} を用いた検証（処理 **d**）を行わないため，E_{TSI} を改ざんすることはない．

このため，いかなる条件の下でも，検証者は処理 **e**, **f** によって改ざんを検出不可能である．

3.3.6 検討結果の整理

検討結果を整理する前に，条件を表わす記号 J，K，N，O，P，Q，R をそれぞれ次のように定義する．

- J：$Info_{INT}$ 生成技術の安全性が低下しない．
- K：攻撃者が発行者と結託しない．
- N：攻撃者は発行者になりすまさない．
- O：攻撃者は証拠補強者と結託しない．
- P：攻撃者は証拠補強者になりすまさない．
- Q：攻撃者は E_{ORE} を有する発行依頼者と結託しない．
- R：攻撃者は E_{ORE} を有する発行依頼者になりすまさない．

なお，JK，KN，OP，QR は，それぞれ「J かつ K」，「K かつ N」，「O かつ P」，「Q かつ R」という条件を意味するとものとする．

これらの記号を使うと，タイプ **ab** における改ざん検出の十分条件は JK となり，タイプ **ac**，**ad** における十分条件は KN となる．

タイプ **ae**，**af** に関する結果は，処理 **a** のほかに，処理 **e** と **f** もタイムスタンプの改ざん検出にとって意味がないことを示唆しているようにみえる．しかし，処理 **e** や **f** の検証対象となる E_{TSI} を攻撃者が改ざんする場合，すなわち，処理 **e** や **f** に加えて検証手続に処理 **d** も含まれる場合には，処理 **e** や **f** は改ざん検出の十分条件を弱める効果をもつと予想される．そこで，タイプ **ade** とタイプ **adf** の十分条件を検討する．

最初に，タイプ **ade** における十分条件を検討する．攻撃者が発行者と結託しない，かつ，発行者になりすまさない場合，検証者は，処理 **d** において改ざんを検出する．また，攻撃者が証拠補強者と結託しない，かつ，発行者になりすまさない場合，検証者は，処理 **e** において改ざんを検出する．一方，これら以外の場合には，検証者は，発行者や攻撃者から E'_{TSI}，および，証拠補強者や攻撃者から E'_{AMP} を得るため，改ざんを検出不可能である．この結果，タイプ **ade** における改ざん検出の十分条件は「KN または OP」，すなわち KN∨OP となる．

タイプ **adf** についても，タイプ **ade** と同様の検討によって，十分条件が「KN または QR」，すなわち KN∨QR となることを容易に導出できる．

このように，タイプ **ade** とタイプ **adf** における十分条件は，タイプ **ad** における十分条件である KN を含み，処理 **e** と **f** は，処理 **d** と併用すれば，改ざん検出の十分条件を弱める効果をもつことが確認された．

なお，処理 **e** や **f** を処理 **b** や **c** と併用する（ただし処理 **d** を用いない）タイムスタンプ方式も考えられる．しかし，攻撃者は，処理 **b** や **c** において改ざんを検出できないようにする目的で H や $Info_{INT}$ を改ざんしても，処理 **d** が存在しないため，E_{TSI} を改ざんしないでおくことができる．この結果，検証者は処理 **e** や **f** で改ざんを検出することができず，処理 **e** や **f** は改ざん検出の十分条件を弱める効果をもたない．

3.4 クラスの定義

3.3 を基に，各タイプにおけるタイムスタンプの改ざん検出の十分条件を検討する．例えば，タイプ **abc** の場合，タイプ **ab** と **ac** における十分条件がそれぞれ JK，KN であるため，タイプ **abc** における十分条件は，JK と KN のいずれかの事象が成立するという条件となり，「JK または KN」，すなわち JK∨KN となる．同様の検討を他のタイプに適用した結果を整理すると，以下のとおり．

表2　10のクラスの定義

クラス	定義
1	すべてのタイムスタンプ方式の集合（いかなる条件下においても改ざん検出不可能な方式も含む）
2	JK の場合に検証者がタイムスタンプの改ざんを検出するタイムスタンプ方式の集合
3	KN の場合に検証者がタイムスタンプの改ざんを検出するタイムスタンプ方式の集合
4	JK∨KN の場合に検証者がタイムスタンプの改ざんを検出するタイムスタンプ方式の集合(クラス2とクラス3の共通部分に相当する集合)
5	KN∨OP の場合に検証者がタイムスタンプの改ざんを検出するタイムスタンプ方式の集合
6	JK∨KN∨OP の場合に検証者がタイムスタンプの改ざんを検出するタイムスタンプ方式の集合（クラス4とクラス5の共通部分に相当する集合）
7	KN∨QR の場合に検証者がタイムスタンプの改ざんを検出するタイムスタンプ方式の集合
8	JK∨KN∨QR の場合に検証者がタイムスタンプの改ざんを検出するタイムスタンプ方式の集合（クラス4とクラス7の共通部分に相当する集合）
9	KN∨OP∨QR の場合に検証者がタイムスタンプの改ざんを検出するタイムスタンプ方式の集合（クラス5とクラス7の共通部分に相当する集合）
10	JK∨KN∨OP∨QR の場合に検証者がタイムスタンプの改ざんを検出するタイムスタンプ方式の集合（クラス6，8，9の共通部分に相当する集合）

- **タイプ a, ae, af, aef においては，いかなる条件下においても改ざん検出は不可能.**
- **タイプ ab, abe, abf, abef における十分条件は JK.**
- **タイプ ac, ad, acd, ace, acf, acef における十分条件は KN.**
- **タイプ abc, abd, abcd, abce, abcf, abcef における十分条件は，JK∨KN.**
- **タイプ ade, acde における十分条件は，KN∨OP.**
- **タイプ abde, abcde における十分条件は，JK∨KN∨OP.**
- **タイプ adf, acdf における十分条件は，KN∨QR.**
- **タイプ abdf, abcdf における十分条件は，JK∨KN∨QR.**
- **タイプ adef, acdef における十分条件は，KN∨OP∨QR.**
- **タイプ abdef, abcdef における十分条件は，JK∨KN∨OP∨QR.**

これらの十分条件に基づき，タイムスタンプ方式の集合として，クラスを定義する．クラスは10通り存在し，各クラスを表2のように定義する．

3.5 各クラスに対応する十分条件の相互関係

次に，各クラスに対応するタイムスタンプの改ざん検出の十分条件における相互関係を考察する．例えば，2つの十分条件AとBを比較し，「BならばA」が成立するケースを「B→A」と表す．このとき，AはBよりも弱い十分条件となる．以下では，クラスj（j = 1,..., 10）に対応する十分条件をCjする．各十分条件の相互関係は，以下のとおり（図3参照）．なお，クラス1の十分条件C1については，クラス1がすべてのタイムスタンプ方式によって構成されており，他のクラスを構成する方式をすべて含むことから，C1は他のクラスにおける十分条件よりも強いと考えることとする．

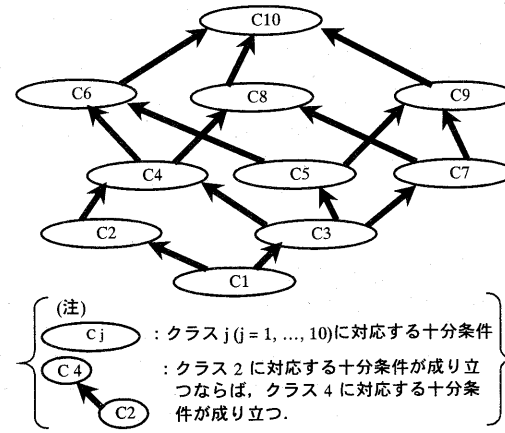

図3　十分条件の相互関係

- C1 → C2, C3, C4, C5, C6, C7, C8, C9, C10
- C2 → C4, C6, C8, C10
- C3 → C4, C5, C6, C7, C8, C9, C10
- C4 → C6, C8, C10
- C5 → C6, C9, C10
- C6 → C10
- C7 → C8, C9, C10
- C8 → C10
- C9 → C10

この結果，クラス10に対応する十分条件が最も弱いことがわかる．タイムスタンプの改ざんに対する安全性の観点からみると，十分条件が弱い方が望ましく，クラス10に属するタイムスタンプ方式が最も望ましいこととなる．

3.6 各クラスを構成するタイムスタンプ方式

各クラスがどのようなタイムスタンプ方式によって構成されるかを検討する．

- **クラス** 10：C10が最も弱い十分条件であるため，クラス10は，タイプ **abdef** とタイプ **abcdef** の和集合となる．
- **クラス** 9：C9よりも弱い十分条件はC10であるから，クラス9は，タイプ **adef**，タイプ **acdef**，クラス10の和集合となる．
- **クラス** 8：C8よりも弱い十分条件はC10であるから，クラス8は，タイプ **abdf**，タイプ **abcdf**，クラス10の和集合となる．
- **クラス** 7：C7よりも弱い十分条件は，C8，C9，C10である．一方，C10はC8および9よりも弱い．このため，クラス7は，タイプ **adf**，タイプ **acdf**，クラス8，クラス9の和集合となる．
- **クラス** 6：C6よりも弱い十分条件はC10であるため，クラス6は，タイプ **abde**，タイプ **abcde**，クラス10の

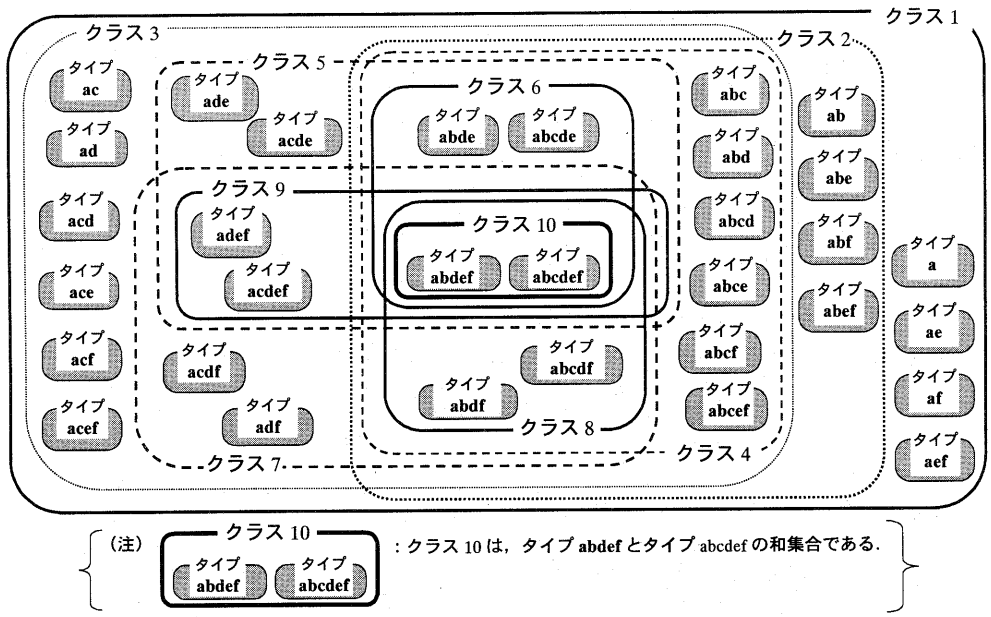

図 4　クラスとタイプの関係

和集合となる．

- **クラス** 5：C5 よりも弱い十分条件は，C6，C9，C10 である．一方，C10 は，C6 および C9 よりも弱い．このため，クラス 5 は，タイプ **ade**，タイプ **acde**，クラス 6，クラス 9 の和集合となる．
- **クラス** 4：C4 よりも弱い十分条件は，C6，C8，C10 である．一方，C10 は，C6 および C8 よりも弱い．このため，クラス 4 は，タイプ **abc**，タイプ **abd**，タイプ **abcd**，タイプ **abce**，タイプ **abcf**，タイプ **abcef**，クラス 6，クラス 8 の和集合となる．
- **クラス** 3：C3 よりも弱い十分条件は，C4，C5，C6，C7，C8，C9，C10 である．一方，C4 よりも弱い十分条件は，C6，C8，C10 であり，C5 よりも弱い十分条件は，C6，C9，C10 である．このため，クラス 3 は，タイプ **ac**，タイプ **ad**，タイプ **acd**，タイプ **ace**，タイプ **acf**，タイプ **acef**，クラス 4，クラス 5，クラス 7 の和集合となる．
- **クラス** 2：C2 よりも弱い十分条件は，C4，C6，C8，C10 である．一方，C4 よりも弱い十分条件は，C6，C8，C10 である．このため，クラス 2 は，タイプ **ab**，タイプ **abe**，タイプ **abf**，タイプ **abef**，クラス 4 の和集合となる．
- **クラス** 1：すべてのタイプの和集合となる．

クラスとタイプの関係を整理すると，図 4 のとおり．なお，同一のクラスに含まれるタイプの中では，検証手続の数が最小となるものがコストの観点から望ましいと考えられる．このようなタイプは，クラス 1 ではタイプ **a**，クラス 2 ではタイプ **ab** となる．また，クラス 3 ではタイプ **ac** もしくは **ab**，クラス 4 ではタイプ **abc** もしくは **abd**，クラス 5 ではタイプ **ade**，クラス 6 ではタイプ **abde**，クラス 7 ではタイプ **adf**，クラス 8 ではタイプ **abdf**，クラス 9 ではタイプ **adef**，クラス 10 ではタイプ **abdef** となる．

本検討結果から，個々のタイムスタンプ方式においてタイムスタンプの改ざんに対する安全性を評価する際に，方式の詳細な内容に立ち入ることなく，どのような点に留意して評価すべきかを比較的容易に把握することができる．まず検証手続に着目して，検討対象の方式がどのタイプに該当するかを確認し，その方式がどのクラスに該当するかを確認する．次に，そのクラスに対応する十分条件を手掛かりに，実際の方式に採用されている暗号技術（例えば $\mathrm{Info}_{\mathrm{INT}}$ 生成技術）やエンティティの属性を検討し，十分条件が満足されているか否かを吟味する．

4　既存のタイムスタンプ方式への適用

3 の結果を主要なタイムスタンプ方式に適用する．例として，法務省の電子公証制度[4]，NTT の時刻署名分散システム[8]，米 Surety.com 社と NTT データの Digital Notary[7]/SecureSeal[5]，スペインの PKITS [3]，ベルギーの TIMESEC[6]，エストニアの Cuculus[1, 2]を取り上げる．

まず，各方式の特徴を抽出し，各方式が属するタイプやクラスを調べる（表 3 参照）．この結果，電子公証制度と時刻署名分散システムは，タイプ **ab** に属し，したがってクラス 2 に属することがわかる．また，Digital Notary/SecureSeal は，タイプ **ac** に属し，したがってクラス 3 に属することがわかる．PKITS，TIMESEC，Cuculus は，タイプ **abde** に属し，したがってクラス 6 に属することがわかる．

次に，各方式が属するクラスからタイムスタンプの改ざん検出の十分条件（表 2 参照）を調べ，各方式の安全性を左右する要因を明らかにする．クラス 2 に対応する電子公証制度と時刻署名分散システムでは，タイムスタンプを構

表 3　主要なタイムスタンプ方式の特徴と安全性のクラス

方式	ﾀｲﾑｽﾀﾝﾌﾟ構成データ	エンティティ	検証手続	グループ	タイプ	クラス
電子公証制度 時刻署名分散ｼｽﾃﾑ	H, ID_{TSI}, T, ID_{REQ}, $Info_{INT}$	発行者, 発行依頼者	(1)ハッシュ値の比較 (2)ディジタル署名を検証	TN-U-I	**ab**	2
Digital Notary/ SecureSeal	H, ID_{TSI}, T, ID_{REQ}		(1)ハッシュ値の比較 (2)発行者にタイムスタンプを送り検証を依頼	TN-U-L	**ac**	3
PKITS	H, ID_{TSI}, T, ID_{REQ}, ID_{AMP}, $Info_{INT}$	発行者, 発行依頼者, 証拠補強者（他の発行者）	(1)ハッシュ値を比較 (2)ディジタル署名を検証 (3)発行者から得たデータから連鎖データを再生し，その整合性を確認 (4)他の発行者から入手した連鎖データと再生した連鎖データを比較	TN-A-L	**abde**	6
TIMESEC		発行者, 発行依頼者, 証拠補強者（インターネット上のサイト）	(1)ハッシュ値を比較 (2)ディジタル署名を検証 (3)発行者から得たデータから連鎖データを再生し，その整合性を確認 (4)インターネット上のサイトから入手した連鎖データと再生した連鎖データを比較			
Cuculus	H, ID_{TSI}, T, ID_{REQ}, ID_{AMP}, E_{TSI}, $Info_{INT}$	発行者, 発行依頼者, 証拠補強者（新聞）	(1)ハッシュ値を比較 (2)ディジタル署名を検証 (3)タイムスタンプを構成するデータから連鎖データを再生し，その整合性を確認 (4)連鎖データと新聞掲載の連鎖データを比較	TE-A-L		

成するディジタル署名の安全性，および，攻撃者と発行者の結託可能性が，タイムスタンプの改ざんに対する安全性を評価する際のポイントとなることがわかる．クラス 3 に属する Digital Notary/SecureSeal では，攻撃者と発行者の結託やなりすましの可能性が評価のポイントとなることがわかる．また，クラス 6 に属する PKITS，TIMESEC，Cuculus では，ディジタル署名の安全性に加え，攻撃者と発行者もしくは証拠補強者との結託やなりすましの可能性が安全性評価のポイントとなることがわかる．

さらに，クラス 6 に対応する十分条件は，クラス 2 やクラス 3 に対応する十分条件よりも弱い．このため，攻撃者が各エンティティと結託する可能性，および，攻撃者によるなりすましの可能性を同一とすれば，クラス 6 に該当する PKITS，TIMESEC，Cuculus が安全性上比較的望ましいと考えられる．

5 おわりに

本稿では，まず，108 種類のタイムスタンプ方式におけるタイムスタンプの改ざんを検出するための十分条件を検討・整理し，十分条件が 10 通り存在することを示した．その上で，それらの相互関係や各十分条件に対応するタイムスタンプ方式を明らかにした．さらに，本検討結果を用いることによって，タイムスタンプの改ざんに対する安全性を評価する際のポイントを比較的容易に把握できることを，6 つのタイムスタンプ方式への適用例を示しながら説明した．

今後は，タイムスタンプの改ざんを検出するための必要条件を検討するほか，タイムスタンプの改ざん以外の攻撃法に対する安全性について検討を行う方針である．

参考文献

[1] Buldas, A., H. Lipmaa, B. Schoenmakers, "Optimally efficient accountable time-stamping," *Proceedings of PKC2000*, LNCS 1751, pp. 293-305, 2000.

[2] Cybernetica, "Cuculus: How does it work?" (http://www.cyber.ee/research/cuc-work.html, 2001 年 7 月 4 日アクセス)

[3] Fabrica Nacional de Moneda y Timbre, *PKITS: Deliverable D4a Description and Results of the Unstructured Data Time-Stamping Protocol Implementation*, Revision Number 16, July 30, 1998. (http://www.fnmt.es/pkits/)

[4] 法務省民事局，"電子取引法制に関する研究会報告書"，1998 年 3 月

[5] NTT データ，"SecureSeal<テクニカル情報>" (http://210.144.76.11/technical/tech01.html, 2001 年 7 月 4 日アクセス)

[6] Preneel, B., B. V. Rompay, J.-J. Quisquater, H. Massias and J. S. Avila, "Design of a Timestamping System," *TIMESEC Technical Report WP3*, 1998. (http://www.dice.ucl.ac.be/crypto/TIMESEC/TR3.ps.gz)

[7] Surety.com, "Secure Time/Data Stamping in a Public Key Infrastructure," (http://www.surety.com/home/pki.pdf, 2001 年 7 月 4 日アクセス)

[8] Takura, A., S. Ono and S. Naito, "Secure and Trusted Time Stamping Authority," *Proceedings of IWS'99*, pp. 123-128, Springer-Verlag, 1999.

[9] 宇根正志，松本勉，"連鎖型タイムスタンプの検証に用いられる情報の管理，" コンピュータセキュリティシンポジウム 2000 予稿集，情報処理学会，pp. 25-30，2000 年 10 月．

[10] 宇根正志，松本勉，"タイムスタンプの安全性と検証手続との関連性，" 2001 年暗号と情報セキュリティシンポジウム予稿集，電子情報通信学会，pp. 629-634，2001 年 1 月．