(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2013-143146
(P2013-143146A)
(43)公開日　平成25年7月22日(2013.7.22)

(51)Int.Cl.		F I		テーマコード（参考）
G06F 21/64	(2013.01)	G06F 21/24	167B	

審査請求 未請求　請求項の数 15　OL　外国語出願　（全 16 頁）

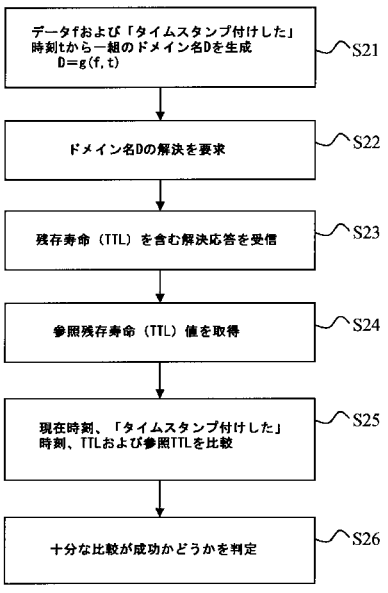
(21)出願番号　特願2013-1747(P2013-1747)
(22)出願日　平成25年1月9日(2013.1.9)
(31)優先権主張番号　12305037.9
(32)優先日　平成24年1月10日(2012.1.10)
(33)優先権主張国　欧州特許庁(EP)
(31)優先権主張番号　12305152.6
(32)優先日　平成24年2月13日(2012.2.13)
(33)優先権主張国　欧州特許庁(EP)
(31)優先権主張番号　12305180.7
(32)優先日　平成24年2月16日(2012.2.16)
(33)優先権主張国　欧州特許庁(EP)

(71)出願人　501263810
トムソン　ライセンシング
Ｔｈｏｍｓｏｎ　Ｌｉｃｅｎｓｉｎｇ
フランス国，　９２１３０　イツシー　レ　ムーリノー，　ル　ジヤンヌ　ダルク，　１－５
１－５，　ｒｕｅ　Ｊｅａｎｎｅ　ｄ’Ａｒｃ，　９２１３０　ＩＳＳＹ　ＬＥＳ　ＭＯＵＬＩＮＥＡＵＸ，　Ｆｒａｎｃｅ
(74)代理人　100107766
弁理士　伊東　忠重
(74)代理人　100070150
弁理士　伊東　忠彦
(74)代理人　100091214
弁理士　大貫　進介

最終頁に続く

(54)【発明の名称】データにタイムスタンプする方法および装置ならびにタイムスタンプ検証のための方法および装置

(57)【要約】　（修正有）
【課題】データにタイムスタンプする方法、装置並びにタイムスタンプ検証の為の方法、装置を提供する。
【解決手段】所収者がデータfおよび「タイムスタンプ付け」時刻tから一組のドメイン名Dを生成する事によってデータfにタイムスタンプ付けする。所有者はドメイン名Dについての解決要求を一つ以上のDNSサーバーに送る。タイムスタンプを検証するには、検証者はデータfおよび「タイムスタンプ付けした」時刻tから一組のドメイン名Dを生成し（Ｓ２１）、ドメイン名Dについての解決要求を所有者と同じDNSサーバーに送り（Ｓ２２）、TTL値を含む解決応答を受け取り（Ｓ２３）、前記DNSサーバーについての参照TTLを取得し（Ｓ２４）、現在時刻を「タイムスタンプ付けした」時刻tおよび各解決応答についてTTLおよび参照TTLと比較する（Ｓ２５）。比較のうちの所定の割合が一致すれば（Ｓ２６）、タイムスタンプは検証される。
【選択図】図２



【特許請求の範囲】
【請求項１】
データfにタイムスタンプ付けする方法であって、装置において：
処理手段によって、前記データfから少なくとも一つのドメイン名Dを生成する段階と、
前記処理手段によって、生成されたドメイン名Dのそれぞれについて DNS解決要求を送る段階と、
前記処理手段によって、前記データfに基づく値を含むタイムスタンプを出力する段階とを含む、
方法。
【請求項２】
前記少なくとも一つのドメイン名Dはさらにタイムスタンプ付け時刻tから生成される、請求項１記載の方法。
【請求項３】
データfのタイムスタンプを検証する方法であって、装置において：
処理手段によって、タイムスタンプ付け時刻tおよび前記データfに基づく値から少なくとも一つのドメイン名Dを生成する段階と；
前記処理手段によって、生成されたドメイン名Dのそれぞれについて DNS解決要求を送る段階と；
前記処理手段によって、送られた各DNS解決要求について解決応答を受け取る段階であって、各解決応答は残存寿命値を含む、段階と；
前記処理手段によって、DNS解決要求が送られた各DNSサーバーから参照残存寿命値を取得する段階と；
前記処理手段によって、各解決応答について、現在時刻、前記タイムスタンプ付け時刻t、受領された前記残存寿命値および前記解決応答が受領された元のDNSサーバーについての前記参照残存寿命値を比較する段階と；
前記処理手段によって、少なくとも所定数の比較が正しい場合に前記タイムスタンプが正しいと判定する段階とを含む、
方法。
【請求項４】
データfにタイムスタンプ付けする装置であって：
前記データfから少なくとも一つのドメイン名Dを生成し、
生成されたドメイン名Dのそれぞれについて DNS解決要求を送り、
前記データfに基づく値を含むタイムスタンプを出力するよう構成されたプロセッサを有する、
装置。
【請求項５】
前記プロセッサは、前記少なくとも一つのドメイン名Dをさらにタイムスタンプ付け時刻tから生成するよう構成される、請求項４記載の装置。
【請求項６】
前記データfに基づく前記値が前記データfと同一である、請求項５記載の装置。
【請求項７】
前記プロセッサがさらに、前記少なくとも一つのドメイン名を生成するのを：
前記データfおよび前記タイムスタンプ付け時刻tに対して一方向性関数を使って中間値を得て；
前記中間値から少なくとも一つのIPアドレスを生成し；
前記少なくとも一つのIPアドレスのそれぞれについて DNS逆解決要求を送り；
各DNS逆解決要求に応答してドメイン名を受け取ることによって行うよう構成されている、
請求項６記載の装置。
【請求項８】

前記プロセッサはさらに、少なくとも前記データfから数値を得て、その数値を使ってDNSサーバーのリストからDNSサーバーを選択することによって、DNSサーバーを選択するよう構成されており、前記DNS解決要求の少なくとも一つがその選択されたDNSサーバーに送られる、請求項４記載の装置。

【請求項９】

前記プロセッサが再帰的解決を強制するよう構成される、請求項４記載の装置。

【請求項１０】

データfのタイムスタンプを検証する装置であって：

タイムスタンプ付け時刻tおよび前記データfに基づく値から少なくとも一つのドメイン名を生成し；

生成されたドメイン名Dのそれぞれについて DNS解決要求を送り；

送られた各DNS解決要求について、残存寿命値を含む解決応答を受け取り；

DNS解決要求が送られた各DNSサーバーから参照残存寿命値を取得し；

各解決応答について、現在時刻、前記タイムスタンプ付け時刻t、受領された前記残存寿命値および前記解決応答が受領された元のDNSサーバーについての前記参照残存寿命値を比較し；

少なくとも所定数の比較が正しい場合に前記タイムスタンプが正しいと判定するよう構成されたプロセッサを有する、

装置。

【請求項１１】

前記プロセッサは、前記少なくとも一つのドメイン名を生成するのを：

中間値を得て；

前記中間値から少なくとも一つのIPアドレスを生成し；

前記少なくとも一つのIPアドレスのそれぞれについてDNS逆解決要求を送り；

各DNS逆解決要求に応答してドメイン名を受け取ることによって行うよう構成される、請求項１０記載の装置。

【請求項１２】

前記中間値が前記データfおよび前記時刻tに対して一方向性関数を使うことによって得られる、請求項１１記載の装置。

【請求項１３】

前記プロセッサはさらに、少なくとも前記データfから数値を得て、その数値を使ってDNSサーバーのリストからDNSサーバーを選択することによって、DNSサーバーを選択し、前記DNS解決要求の少なくとも一つをその選択されたDNSサーバーに送るよう構成される、請求項１０記載の装置。

【請求項１４】

プロセッサによって実行されたときに請求項１または２記載の方法を実行する記憶された命令を有するコンピュータ可読記憶媒体。

【請求項１５】

プロセッサによって実行されたときに請求項３記載の方法を実行する記憶された命令を有するコンピュータ可読記憶媒体。

【発明の詳細な説明】

【技術分野】

【０００１】

本発明はデジタル文書のタイムスタンプ付けに関する。

【背景技術】

【０００２】

このセクションは、下記で記述および／または特許請求される本発明のさまざまな側面に関係しうる技術のさまざまな側面を読者に紹介することを意図したものである。この議論は読者が本発明のさまざまな側面をよりよく理解するのを容易にするための背景情報を提供する助けとなると考えられる。よって、これらの陳述はこの観点で読まれるものであ

って、従来技術の自認として読まれるものではないことは理解しておくべきである。
【０００３】
デジタル・タイムスタンプ付けは、当事者pが所与の時点tにおいてデジタル・データfを知っていたことを事後に証明できるようにするすべであると定義できる。
【０００４】
デジタル世界では、信頼できるタイムスタンプ付け、すなわちあるファイルまたは文書がある時刻に存在していたという証明を必要とする多くの状況がある。そのような状況の例は次のようなものを含む。
・オンライン・オークション。ビッドの正しい順番を保証するために。
・電子投票。票が許されうる時間に投じられたことを保証するために。
・公表物。文書が所与の時に公開されたことを証明するために。
・オンラインの賭け。賭けがなされたのがイベントより前であることを確実にするために。
・デジタル権利管理。コンテンツ項目が許可されるときにのみ使用されることを保証するために。
【０００５】
従来技術は時間に関係した多くの異なる解決策を提供する。
【０００６】
ネットワーク時間プロトコル（NTP: Network Time Protocol）はイベントについての信頼できる時間指標を提供する一般的なプロトコルである。これは当事者がイベントの同時性やイベント間の時間差を推定することを許容する。しかしながら、これは保持の証明と時間指示とを強く結び付ける能力を提供するものではない。
【０００７】
信用されるタイムスタンプ付けに関するウィキペディアの記事http://en.wikipedia.org/wiki/Trusted_timestampingはタイムスタンプ機関に基づく種々の解決策を記載している。これらの解決策は、タイムスタンプを提供する信用されるサードパーティーに依拠する。ある解決策では、サードパーティーが文書を受け取り、要求があったときにその文書がいつタイムスタンプ付けされたかを証明するために、その文書をたとえばその文書を現在時刻と一緒にハッシュすることによって処理する。もう一つの解決策では、ユーザーは文書のハッシュをサードパーティーに送り、そのサードパーティーがそのハッシュをハッシュ木に含める。これがタイムスタンプ付けを可能にする。「マスター」ハッシュ値が公表された時点ではその文書が明らかに存在していたことになる。
【０００８】
BitCoin（"A Peer-to-Peer Electronic Cash System", http://www.bitcoin.org/bitcoin.pdf参照）が提供するタイムスタンプ付けサーバーは、タイムスタンプ付けされるべき各項目にタイムスタンプを加え、受領されたすべての文書をハッシュし、そのハッシュを新聞またはユースネット投稿において広く公開する。タイムスタンプは、そのデータが、ハッシュに含まれるためには明らかにその時点で存在していたに違いないことを証明する。各タイムスタンプはそのハッシュ内に前のタイムスタンプを含み、チェーンを形成し、タイムスタンプが一つ加わるごとにそれ以前のタイムスタンプを補強する。
【０００９】
エフェメラル公表（"Ephemeral Publishing", C. Castelluccia et al., http://code.google.com/p/ephpub/参照）システムは暗号鍵を保存するためにドメイン名システム（Domain Name System）を使う。鍵は、DNSキャッシュ内のドメインの挿入を強制することによって記憶される。ひとたびTTLが期限切れになったらエントリーはDNSサーバーによって自動的にキャッシュから除去される。こうして、しばらくしたら暗号鍵は自動的に忘れられる。このシステムはある時刻における情報の存在を証明するという問題に対処するものではない。これは鍵の消去に焦点を当てるのみである。
【００１０】
従来技術のタイムスタンプ付けの一つの既知の欠点は、信頼できる時間ならびにタイム

スタンプを作成および検証するプロセスの両方を提供する信用されるサードパーティーが必要だということである。サードパーティーが利用不能になったらシステムは破綻するので、これは、信用（trust）問題、また信頼性（reliability）問題を引き起こすことがありうる。

【発明の概要】

【発明が解決しようとする課題】

【００１１】

よって、従来技術の問題点を克服するタイムスタンプ付け解決策が必要とされていることが理解されるであろう。本願のシステムはそのような解決策を提供する。

【課題を解決するための手段】

【００１２】

第一の側面では、本発明は、データにタイムスタンプ付けする方法に向けられる。装置が前記データから少なくとも一つのドメイン名を生成し、生成されたドメイン名のそれぞれについてDNS解決要求を送り、前記データに基づく値を含むタイムスタンプを出力する。

【００１３】

第一の好ましい実施形態では、少なくとも一つのドメイン名はさらにタイムスタンプ付け時刻から生成される。

【００１４】

第二の側面では、本発明はデータのタイムスタンプを検証する方法に向けられる。装置がタイムスタンプ付け時刻および前記データに基づく値から少なくとも一つのドメイン名を生成し；生成されたドメイン名のそれぞれについてDNS解決要求を送り；送られた各DNS解決要求について解決応答を受け取り、各解決応答は残存寿命（Time-to-Live）値を含み；DNS解決要求が送られた各DNSサーバーから参照残存寿命値を取得し；各解決応答について、現在時刻、前記タイムスタンプ付け時刻、受領された前記残存寿命値および前記解決応答が受領された元のDNSサーバーについての前記参照残存寿命値を比較し；少なくとも所定数の比較が正しい場合に前記タイムスタンプが正しいと判定する。

【００１５】

第三の側面では、本発明は、データにタイムスタンプ付けする装置に向けられる。本装置は、前記データから少なくとも一つのドメイン名を生成し、生成されたドメイン名のそれぞれについてDNS解決要求を送り、前記データfに基づく値を含むタイムスタンプを出力するよう構成されたプロセッサを有する。

【００１６】

第一の好ましい実施形態では、前記プロセッサは、前記少なくとも一つのドメイン名をさらにタイムスタンプ付け時刻から生成するよう構成される。前記データに基づく値が前記データと同一であることが有利である。さらに、前記プロセッサがさらに、前記少なくとも一つのドメイン名を生成するのを：前記データおよび前記タイムスタンプ付け時刻に対して一方向性関数を使って中間値を得て；前記中間値から少なくとも一つのIPアドレスを生成し；前記少なくとも一つのIPアドレスのそれぞれについてDNS逆解決要求（DNS reverse-resolution request）を送り；各DNS逆解決要求に応答してドメイン名を受け取ることによって行うよう構成されていることが有利である。

【００１７】

第二の好ましい実施形態では、前記プロセッサはさらに、少なくとも前記データから数値を得て、その数値を使ってDNSサーバーのリストからDNSサーバーを選択することによって、DNSサーバーを選択するよう構成される。ここで、前記DNS解決要求の少なくとも一つが前記選択されたDNSサーバーに送られる。

【００１８】

第三の好ましい実施形態では、前記プロセッサは再帰的解決を強制するよう構成される。

【００１９】

第四の側面では、本発明はデータのタイムスタンプを検証する装置に向けられる。本装置は：タイムスタンプ付け時刻および前記データに基づく値から少なくとも一つのドメイン名を生成し；生成されたドメイン名のそれぞれについてDNS解決要求を送り；送られた各DNS解決要求について解決応答を受け取り、各解決応答は残存寿命（Time-to-Live）値を含み；DNS解決要求が送られた各DNSサーバーから参照残存寿命値を取得し；各解決応答について、現在時刻、前記タイムスタンプ付け時刻、受領された前記残存寿命値および前記解決応答が受領された元のDNSサーバーについての前記参照残存寿命値を比較し；少なくとも所定数の比較が正しい場合に前記タイムスタンプが正しいと判定するよう構成されたプロセッサを有する。

【００２０】

第一の好ましい実施形態では、前記プロセッサは、前記少なくとも一つのドメイン名を生成するのを：中間値を得て；前記中間値から少なくとも一つのIPアドレスを生成し；前記少なくとも一つのIPアドレスのそれぞれについてDNS逆解決要求（DNS reverse-resolution request）を送り；各DNS逆解決要求に応答してドメイン名を受け取ることによって行うよう構成される。前記中間値が前記データおよび前記時刻に対して一方向性関数を使うことによって得られることが有利である。

【００２１】

第二の好ましい実施形態では、前記プロセッサはさらに、少なくとも前記データから数値を得て、その数値を使ってDNSサーバーのリストからDNSサーバーを選択することによって、DNSサーバーを選択し、前記DNS解決要求の少なくとも一つを前記選択されたDNSサーバーに送るよう構成される。

【００２２】

第五の側面では、本発明は、プロセッサによって実行されたときに前記第一の側面の方法を実行する記憶された命令を有するコンピュータ可読記憶媒体に向けられる。

【００２３】

第六の側面では、本発明は、プロセッサによって実行されたときに前記第二の側面の方法を実行する記憶された命令を有するコンピュータ可読記憶媒体に向けられる。

【図面の簡単な説明】

【００２４】

本発明の好ましい特徴についてこれから限定しない例として、付属の図面を参照しつつ説明する。

【図１】本発明のある好ましい実施形態に基づくデジタル・データのタイムスタンプ付けを示す図である。

【図２】本発明のある好ましい実施形態に基づくタイムスタンプの検証を示す図である。

【図３】本発明のある好ましい実施形態に基づくタイムスタンプ付けシステムを示す図である。

【発明を実施するための形態】

【００２５】

本発明は、デジタル・データの信頼できるタイムスタンプ付けに関する。タイムスタンプを作成するエンティティは「所有者」と呼ばれ、タイムスタンプを検査するエンティティは「検証者」と呼ばれる。

【００２６】

本発明はドメイン名システム（DNS: Domain Name System）を利用するので、ここで必要な概念について説明しておく。

【００２７】

DNS解決：DNSは完全修飾ドメイン名（FQDN: Fully Qualified Domain Name）、たとえばwww.example.comをその対応するIPアドレス、たとえば192.0.43.10に変換する世界規模のデータベースである。このプロセスは「解決」と呼ばれる。一つの例は次のシェル・セッションに示される：

```
C:\>nslookup
```

>www.example.com
Non-authoritative answer:
Name:192.0.43.10。

【００２８】

DNS逆解決：DNSは逆の作用も行う。すなわち、IPアドレスを対応するFQDNに変換するのである。このプロセスは「逆解決」と呼ばれる。一つの例は次のシェル・セッションに示される：

C:\>nslookup
>192.0.43.10
Name:www.example.com。

【００２９】

トップレベル・ドメイン（TLD: top level domain）：トップレベル・ドメインは、FQDNのルート（最後の部分）である。www.example.comのTLDは「.com」である。最も人気のあるTLDのいくつかは「.com」および「.org」である。本発明のある実施形態は「.com」を使う。

【００３０】

規範的DNS（authoritative DNS）：DNSについての決定的な制約条件は、高い可用性および一貫性である。したがって、DNSは世界中に高度に分散していて、多くの局所的または地域的なレプリカがある。DNSサーバーがあるドメインについて規範的であると言われるのは、対応するIPアドレスおよび残存寿命（後述）のような他の属性を定義できる場合である。他のすべてのDNSサーバーは非規範的（non-authoritative）と言われる。あるドメインにおける非規範的DNSサーバーは、そのドメインの規範的サーバーからの情報を、自らのローカルなDNSキャッシュにおいて一定期間にわたり複製することしかできない。

【００３１】

DNSキャッシュ：各DNSサーバーはDNSエントリーのキャッシュを維持している。解決要求がサーバーによって受信されるとき、まず要求されたドメインがそのキャッシュに保存されているかどうかを検査する。もしそうであれば、サーバーはただちに応答する。しかしながら、そのドメインがキャッシュ内にない場合には、サーバーは通常、該要求をその階層的サーバーに転送し、該階層的サーバーが上記の検査プロセスを繰り返す。解決に成功した階層的サーバーが応答を階層構造を下ってもとのサーバーまで送る。下位のDNSサーバーは、該要求のドメインをそのキャッシュに追加する。

【００３２】

再帰的解決：クライアントは、DNS解決要求において、サーバーがその階層的サーバーに要求を転送すべきか否かを指定することが可能である。サーバーキャッシュが要求されたドメインを保存しておらず、サーバーが要求を転送しない場合、サーバーは解決要求に対して空の応答を返す。

【００３３】

DNS残存寿命（Time-To-Live）：DNSサーバーはDNSエントリーをそのキャッシュに無期限に保存しておくわけではない。各エントリーは、規範的サーバーのみが定義できる残存寿命（TTL）期間に関連付けられる。このTTL期間は本稿では「参照TTL」と呼ばれる。典型的な値は86400秒（１日）であるが、7日までの値がサポートされる。非規範的さーバーは、TTLの期限が切れるまで、解決情報を複製することが許容される。解決要求に答えるとき、非規範的サーバーも残っているTTL値を返す。あるエントリーについてTTL値が切れると、非規範的さーバーはそのエントリーについてそれ以上要求を解決しない。その後の要求に応答するためには、非規範的さーバーは規範的サーバーから再び解決を得る必要がある。

【００３４】

本発明は、上記のDNS機能を利用する。本発明は、相異なるが関係した部分を有する：デジタル・データのタイムスタンプ付けと、既存のタイムスタンプの検証である。

【００３５】

〈デジタル・データのタイムスタンプ付け〉

図１は、本発明のある好ましい実施形態に基づくデジタル・データのタイムスタンプ付けを示している。所有者pは、時刻tをもってタイムスタンプ付けされるべきデータfを有している。所有者は、決定論的な一方向性関数gを使って一組のドメイン名Dを生成する（S11）：D＝g(f,t)　（詳細は後述）。一方向性関数gは生成されたドメイン名Dをデータfおよび時刻tに結び付ける。一方向性関数gは有利には暗号学的ハッシュ関数であるが、暗号ブロック連鎖（CBC: Cipher Block Chaining）を使ったブロック暗号であることもできる。

【００３６】

当業者は、時刻tを使うことなく一組のドメイン名Dを生成することが可能であることを認識するであろう。これは伝統的な意味でのタイムスタンプ付けを許容しないが、それでも所有者が、当該データの以前の所持を証明することを許容できる（特に、DNS解決を定期的に更新する場合）。

【００３７】

あるいはまた、上記生成に、秘密の値のようなさらなるデータを含めることも可能である。

【００３８】

所有者pは、一つまたは複数のDNSサーバーにDNS解決要求を送ることによって、各ドメイン名を解決する（ステップS12）。所有者は好ましくは、再帰的解決のオプションをDNS解決要求に含めることによって再帰的解決を強制する。その効果として、要求されたドメイン名をキャッシュしていないDNSサーバーは（規範的DNSサーバーから応答が返ってくるとき）そのドメイン名を自らのキャッシュに加え、TTL値を設定する（ステップS13）。DNSサーバーは、そのドメイン名を、DNS規格において定義されているように、TTLが0に達するまで自らのキャッシュに保存する。ドメイン名をすでにキャッシュしているDNSサーバーはTTL値を更新しないことを注意しておくべきである。

【００３９】

非常に人気のあるDNSドメイン（たとえばあらゆるところにキャッシュしている可能性が高いgoogle.com）の影響を避けるとともに、いくつかのDNSタイムスタンプ間の衝突を避けるとともに、以下のことが推奨される。

【００４０】

・要求されたDNSドメインは擬似ランダム関数または暗号学的関数を使って生成されるべきである。

【００４１】

・一つのタイムスタンプについて複数の解決要求が送られるべきである。当業者は、解決要求の数とともにタイムスタンプの信頼性が高まることを理解するであろう。他方、要求の数とともに資源の使用も増す。典型的なトレードオフの結果は、たとえば64個、128個、256個または512個の要求である。

【００４２】

・要求は種々のDNSサーバーに拡散されるべきである。これは、単一の（または少数の）DNSサーバーに頼ることを減らすためである。所有者p（または別の者）がシステムを「ハックする」難しさはサーバーの数とともに増すことは理解されるであろう。諸DNSサーバーはしばしば独立して運営されているので、あるDNSサーバーを制御することが他のDNSサーバーの制御を与えることにはならない可能性が高い。

【００４３】

〈既存のタイムスタンプの検証〉

図２は、本発明のある好ましい実施形態に基づくタイムスタンプの検証を示している。検証者は、データfおよび「タイムスタンプ付け」時刻tを取得しており、一方向性関数gを知っていると想定される。検証者はまず所有者と同じ仕方で（すなわち、時刻tなしでもよく、さらなるデータを含めてもよい）D＝g(f,t)を計算する（ステップS21）。次いで、検証者は計算されたドメイン名についての解決要求を、所有者pと同じDNSサーバーに送

る（ステップS22）。ドメイン名計算のためには、検証者は所有者によって使用されたアルゴリズムと等価な（好ましくは同じ）アルゴリズムを使う。DNSインフラストラクチャーが公に利用可能であることが好ましい。そうすればタイムスタンプ付け方法（タイムスタンプ付けおよび検証の両方）が公のものになるからである。検証者が解決応答を受け取ると、検証者はD内の各ドメイン名について残っているTTLを取得する（ステップS23）。

【００４４】

さらに、検証者はD内の各ドメイン名について参照TTL値を取得する（ステップS24）。これは、各ドメインについての規範的サーバーに問い合わせることによって達成される。

【００４５】

検証者は次いで、ステップS25において、現在時刻を、「タイムスタンプ付け」時刻tおよび各ドメイン名について受領された参照TTLおよび残っているTTLと比較する。すなわち、検証者は

【数１】

$$\text{現在時刻} \approx t + (\ \text{参照}\ TTL - \text{残っている}TTL)$$

であるかどうかを検査する。

【００４６】

検証者は最後に、ステップS26において、Dからのドメイン名の少なくとも(1－ε)の割合について、参照TTL値と残っているTTL値との差が証明と一緒に提供された「タイムスタンプ付け」時刻tと整合する場合に、検証成功と判断する。他の場合には検証は不成功と判断される。

【００４７】

タイムスタンプが時刻tを含まない場合には、Dからのドメイン名の少なくとも(1－ε)の割合について、残っているTTL値（または参照TTL－残っているTTL）が少なくともほぼ同一であることを検査することによって検証が行われることが好ましい。

【００４８】

〈解説例〉

次の例は、タイムスタンプの検証を例解する。PERLコマンドライン結果は次のとおり：

「You lost the game〔あたなはゲームに負けた〕」というテキストについてのタイムスタンプを生成する（generate）：

Mon 06/20/2011 14h52:00>perl dnstamp.pl -text "You lost the game" -time “Mon 06/20/2011 14h52:00” -generate

この例では、解決要求は示されていないが、二つのドメイン名が生成され、二つの解決要求が所有者によって送られ、検証者が各検証試行について二つの解決要求を送ることは理解されるであろう。

【００４９】

各検証についての秒数は二つの異なるDNSサーバーから来て、タイムスタンプの「齢（エージ（age））」を与える。ネットワーク遅延または軽微な同期差からわずかな変動（すなわち、4秒から19秒の間の範囲の、二つの値の間の差の変動）が生じることがありうる。しかしながら、参照TTLはずっとずっと大きい86400秒なので、そのような軽微な差はシステムの信頼度には影響しないことは理解されるであろう。

【００５０】

「You lost the game」というテキストおよび時刻についてのタイムスタンプを検証する（verify）：

Mon 06/20/2011 14h53:00>perl dnstamp.pl -text "You lost the game" -time “Mon 06/20/2011 14h52:00” -verify

56 seconds〔秒〕

37 seconds

Mon 06/20/2011 14h53:30>perl dnstamp.pl -text "You lost the game" -time “Mon 06/20/2011 14h52:00” -verify
93 seconds
80 seconds

Mon 06/20/2011 16h07:00>perl dnstamp.pl -text "You lost the game" -time “Mon 06/20/2011 14h52:00” -verify
4453 seconds
4449 seconds

Mon 06/27/2011 11h02:42> perl dnstamp.pl -text "You lost the game" -time “Mon 06/20/2011 14h52:00” -verify
Expired〔期限切れ〕
Expired。

【００５１】
タイムスタンプは14:52:00に実行され、14:53:00、すなわちタイムスタンプ後1分における最初の検証がそれぞれ56秒および37秒という「齢」を与えていることが見て取れる。さらに、14:53:30、すなわちタイムスタンプ後1.5分における検証はそれぞれ93秒および80秒という「齢」を与え、16:07:00、すなわちタイムスタンプ後1時間15分（4500秒）における最初の検証はそれぞれ4453秒および4449秒という「齢」を与えている。最後の例は、タイムスタンプが期限切れになっていることを示している。検証はタイムスタンプ生成後7日近くたってからなされており、参照TTLは86400秒、すなわち1日であった。

【００５２】
〈好ましい実施形態〉
タイムスタンプ付け
以下では本願のタイムスタンプ付け方法のさらなる詳細を述べる。抽象的な点の例解のために例が用いられる。その例では、タイムスタンプは時刻tにおいてデジタル・データfに対して生成され、単一のDNSサーバーが解決のために使われる。むろん、複数のDNSサーバーが使われることもできる。

【００５３】
ステップ１：ランダムなドメイン名を生成する
ステップ１．１：α＝h(f.t)を計算する。ここで、「.」は連結を表し、hは暗号学的ハッシュ関数である。今の例では、256ビット幅のハッシュ関数sha256が使用された。よって、αは256ビットの数である。
ステップ１．２：αから８個のIPアドレスA_1～A_8が生成される：A_1＝α[1..32]、A_2＝α[33..64]、……、A_8＝α[224..256]。ここで、α[X..Y]はαのビットXからビットYを表す。
ステップ１．３：上記８個のIPアドレスA_1～A_8について逆解決要求を送る。これは一般に結果として、所有者に返される８個のFQDN D_1、……、D_8を与える。

【００５４】
ステップ２：FQDNを解決する
再帰的解決を強制することによって上記８個のFQDN D_1、……、D_8に対する解決を要求する。DNSサーバーがすでに問題のドメインをキャッシュしているのでない限り、問い合わせされたドメインは自動的にDNSサーバーのキャッシュに追加され、最大TTL値$TTLMaxD_1$、……、$TTLMaxD_8$がDNSサーバーによって設定される。

【００５５】
どれに解決要求が送られるべきかを決定するには種々の方法がある。

【００５６】
第一のオプションは、すべての要求を一つまたは複数のあらかじめ決定されたDNSサー

バーに送ることである。これらのサーバーは、常に使用される「標準的」サーバーであることができる。二つ以上のサーバーが使用される場合には、各サーバーにあらかじめ決定された（多様であってもよい）数の解決要求を送ることが可能である。ここで、前記数は、1であってもよいし、より大きな数であってもよい。

【００５７】

第二のオプションは、所有者がその決定をして、必要な情報をタイムスタンプに添付するというものである。

【００５８】

第三の、好ましいオプションは、どのDNSサーバーを使うかを決定するためにデータfに関係した情報を使うというものである。データfはいくつかの値を得るために、たとえば、一回または複数回（時刻tと一緒にまたは時刻tなしに）ハッシュされることができる。それらの値が次いで、タイムスタンプ付けのために使用されることになるDNSサーバーのリストにマッピングされることができる。リストは、タイムスタンプを実行するソフトウェア・プログラムおよびその検証を実行するソフトウェア・プログラムに含まれることができるが、DNSサーバーの公に利用可能なリストにマッピングされることもできる。値αを直接におよび／またはシードとして使うことが特に有利である。データfが、そのようなリストにマッピングされる一連の数として解釈されることもできる。

【００５９】

例として、リストが128個の異なるDNSサーバーを含んでいるとする。値α（今の例では256ビット）が使われる場合、これはたとえば6ビットのグループ42個に分割されてもよく、それらのうちの最初の8個（または他の所望される数）がリストからDNSサーバーを選択するために使用されてもよい。

【００６０】

当業者は、シードからの値の（擬似）ランダムな選択のための他の多くのよく知られた代替法が利用可能であることを理解するであろう。

【００６１】

〈証明の送付〉

所有者はデータfおよび時刻tを公開する。fの代わりにα（または結果として得られるIPアドレスA_1～A_8）を公開することも可能であることを注意しておく。この証明は世界に向けて（新聞でまたはウェブ上で）、あるグループ（たとえばソーシャル・ネットワーク）に向けて、あるいは検証者に対して直接に（たとえば電子メールによって）公開されることができる。公開の詳細は本発明の範囲を越える。

【００６２】

〈検証〉

検証者はタイムスタンプ付けの際に所有者がしたのと同じDNS解決要求を生成する。それには、たとえばα＝h(f.t)およびそれからのIPアドレスの生成を使う。検証者は生成された要求を所有者と同じDNSサーバーに（または場合によっては同じ複数のDNSサーバーに）送る。そうするためには、検証者はあらかじめ決定されたDNSサーバーを使う、タイムスタンプから情報を取得する、あるいは所有者と同じ仕方でDNSサーバーを選択する。応答して、検証者は残っているTTL値を受け取る。検証者は参照TTL値を得るために対応する規範的サーバーに問い合わせもする。残っているTTL値と参照TTL値との間の差がタイムスタンプ以降実際に経過した時間と少なくとも概括的に整合すれば、検証は成功する。現在時刻とタイムスタンプ付け時刻tとの間の差より小さい参照TTL値に対応する残っているTTL値は無視することが好ましいことは理解されるであろう。より正確には次のようになる。

【００６３】

ステップ１：検証者がfおよびtを得る。検証者はタイムスタンプ付け方法のステップ１のようにして８個のFQDNドメインD_1、……、D_8を計算する。

【００６４】

ステップ２：検証者は前記ドメイン（または各ドメイン）の規範的サーバーを使って前

記８個のドメインを解決する。検証者は、各ドメインについて返された最大TTL値TTLMaxD$_1$、……、TTLMaxD$_8$を記憶する。

【００６５】

ステップ３：検証者は標準的なDNSサーバー、すなわち所有者と同じものを使って前記８個のドメインを解決する。DNSサーバーは各エントリーについてデクリメントされたTTL TTLD$_1$、……、TTLD$_8$を返す。TTLが期限切れになったのでない限り、DNSサーバーは、これらのエントリーすべてを記憶することを注意しておくべきである。これらは所有者の要求に応答してキャッシュされたものであるが、先述したように、キャッシュ・エントリーは所有者の要求の時点で既に存在していた場合には更新されないから。このためDNSサーバーが期待されるより小さなTTLD値を返すことがありうる。これが、複数の要求を使うことが重要である理由の一つである。

【００６６】

ステップ４：ここで検証者は各ドメインについて差TTLMaxD$_1$－TTLD$_1$、……、TTLMaxD$_8$－TTLD$_8$を計算する。結果として得られる値の大半が等しく（わずかな差は許容される）、経過時間が宣言された時間tと整合していれば、証明は成功となる。時間についての許容差および要求される整合する時間の割合を決めるのは検証者次第である。たとえば、10秒または600秒（10分）の時間差が許容されてもよく、時間のうちの少なくとも四分の一、三分の一または二分の一が整合していることがタイムスタンプを信用するために要求されてもよい。

【００６７】

〈変形実施形態〉

専用ドメインの使用

タイムスタンプ付けの際にランダムにIPアドレスを選ぶ代わりに、「example.com」のような一つまたは複数の所与のドメインを使うことができる。これにより、前もって当該ドメインの規範的ドメインがどれであるかを知ることが可能になる。

【００６８】

これを達成するために、ランダム関数h(f.t)はやはりn個のランダム・ストリングs$_1$、……、s$_n$の組を返し、s$_1$.example.com、……、s$_n$.example.comというFQDNを使って生成される。

【００６９】

検証者にとっての追加的情報

好ましい実施形態では、検証者は結局、次の情報をもつことになる：「その証明は有効である」または「その証明は有効ではない」。これは、たとえばタイムスタンプ付けプロセスにおける最短TTL、最長TTL、TTLの平均値およびそれらの組み合わせについての精度を与えるより饒舌なモードにおいて拡張されることができる。

【００７０】

図３は、本発明のある好ましい実施形態に基づくタイムスタンプ付けシステム３００を示している。システム３００は所有者装置３１０および検証者装置３４０を有する。所有者装置３１０および検証者装置３４０はそれぞれ、少なくとも一つのプロセッサ３１１、３４１、メモリ３１２、３４２、好ましくはユーザー・インターフェース３１３、３４３および少なくとも一つの入出力ユニット３１４、３４４を有する。所有者装置３１０および検証者装置３４０はいずれもたとえばパーソナル・コンピュータ、ワークステーション、テレビジョン受信機、タブレット、スマートフォンおよびセットトップボックスでありうる。

【００７１】

所有者装置３１０は本稿に記載されるタイムスタンプ付け方法の任意の実施形態に従ってデータにタイムスタンプ付けするよう構成され、検証者装置３４０は本稿に記載される検証方法の任意の実施形態に従ってタイムスタンプを検証するよう構成される。

【００７２】

システム３００は非規範的DNSサーバー３２０および規範的DNSサーバー３３０をも有する。しかしながら、これらのサーバーは前記方法の一部を実行するが、その機能は本発明によって修正されないことは理解されるであろう。

【００７３】

第一のコンピュータ可読記憶媒体３６０は、所有者装置３１０のプロセッサ３１１によって実行されたときに、記載されたようにデータにタイムスタンプ付けする記憶された命令を有する。第二のコンピュータ可読記憶媒体３７０は、検証者装置３４０のプロセッサ３４１によって実行されたときに、記載されたようにタイムスタンプを検証する記憶された命令を有する。

【００７４】

本願の解決策が所有者と検証者がタイムスタンプ付け動作に先立って情報を共有することを必要としないことが理解されるであろう。特に、所有者および検証者は一つの共通のタイムスタンプ付け機関、一つの共通の証明機関、一つの共通のネットワーク時間プロトコル・サーバーなどを信用する必要がない。

【００７５】

また、本願の解決策は大規模な公共的な検証を許容できることも理解されるであろう。まず、所有者が証明要素を例えばウェブ・サイト上で公開する。すると、インターネットに接続されている誰でも、タイムスタンプを検証しうる（「期限切れ」になるまで）。DNSシステムは非常に高い可用性をもち、本願の解決策は多くの異なるDNSサーバーを使用できるので、多くの検証者が同時に進むことができる。

【００７６】

さらに、本願の解決策に基づく要求はDNSサーバーに対しては通常の要求のように見えることが理解されるであろう。さらに、DNSサーバーは好ましくはランダムに選択される。よって、所有者の主要なサーバーとは異なる悪意のあるDNSサーバーはタイムスタンプを攻撃する際に何らの利点をももたない。

【００７７】

本発明のタイムスタンプは典型的には何十ものDNS要求を必要とする。これは通常のウェブ・サーフィンによって生成される要求の数に匹敵する。たとえ大規模に使われたとしても、本発明は全体的なDNSサーバー３２０および規範的システムについて無視できるほどのオーバーヘッドしかもたらさない。

【００７８】

さらに、本発明は、信用されるサードパーティーの不在のもとでも信頼できるタイムスタンプ付けを許容できる。唯一の要件は、すべての参加者がインターネットのドメイン名システム（DNS）へのアクセスをもつということである。単一障害点（SPOF: Single Point Of Failure）がなく、あらかじめ存在している信用が必要ないので、本発明は、世界中の非常に多数の場所での非常に多数のユーザーに関わる大規模なタイムスタンプ付け使用事例を許容する。

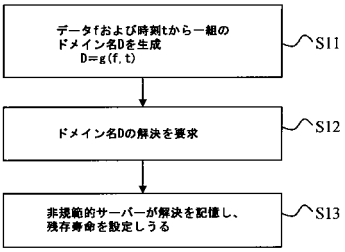
【００７９】

しかしながら、DNSの現在の限界のため、本発明は今のところ、タイムスタンプの高々7日の有効性に限定される。ただし、この制限は、タイムスタンプが時刻tを含まない変形には当てはまらない。

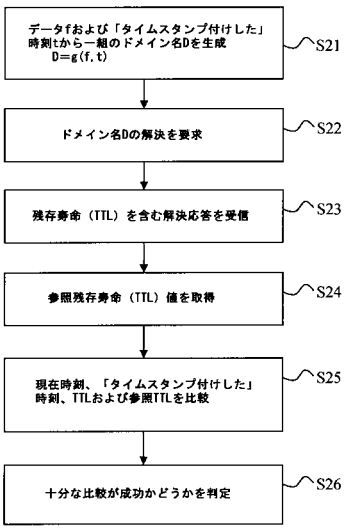
【００８０】

本稿および（該当する場合には）請求項および図面に開示される各特徴は独立して、あるいは任意の適切な組み合わせにおいて提供されうる。ハードウェアで実装されると記載されている特徴はソフトウェアで実装されてもよく、逆にソフトウェアで実装されると記載されている特徴がハードウェアで実装されてもよい。請求項に現れる参照符号は例解のためであって、請求項の範囲に対して何ら限定効果をもつものではない。

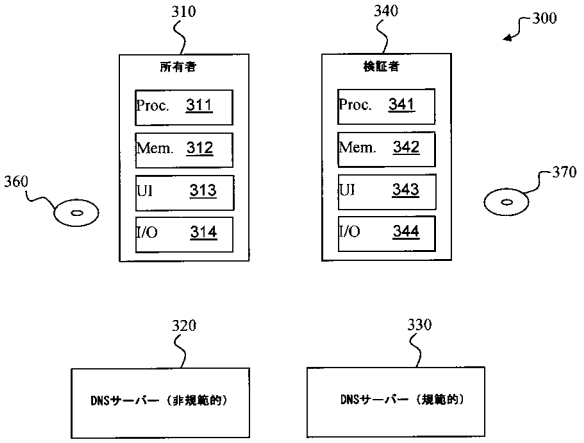
【図１】


データfおよび時刻tから一組の
ドメイン名Dを生成
D=g(f, t)
S11
ドメイン名Dの解決を要求
S12
非規範的サーバーが解決を記憶し、
残存寿命を設定しうる
S13

【図２】


データfおよび「タイムスタンプ付けした」
時刻tから一組のドメイン名Dを生成
D=g(f, t)
S21
ドメイン名Dの解決を要求
S22
残存寿命（TTL）を含む解決応答を受信
S23
参照残存寿命（TTL）値を取得
S24
現在時刻、「タイムスタンプ付けした」
時刻、TTLおよび参照TTLを比較
S25
十分な比較が成功かどうかを判定
S26

【図３】


310
所有者
Proc. 311
Mem. 312
UI 313
I/O 314
340
検証者
Proc. 341
Mem. 342
UI 343
I/O 344
300
360
370

320
DNSサーバー（非規範的）
330
DNSサーバー（規範的）

フロントページの続き

(72)発明者　クリストフ　ノイマン
フランス国　１７６　１６　－３５　５７６　セソン・セヴィニェ　アヴェニュ・ド・シャン・ブラン　ザック・ド・シャン・ブラン　シー・エス　９７５　テクニカラー・アールアンドディー・フランス
(72)発明者　ステファーヌ　オノ
フランス国　１７６　１６　－３５　５７６　セソン・セヴィニェ　アヴェニュ・ド・シャン・ブラン　ザック・ド・シャン・ブラン　シー・エス　９７５　テクニカラー・アールアンドディー・フランス
(72)発明者　オリヴィエ　イーン
フランス国　１７６　１６　－３５　５７６　セソン・セヴィニェ　アヴェニュ・ド・シャン・ブラン　ザック・ド・シャン・ブラン　シー・エス　９７５　テクニカラー・アールアンドディー・フランス

【外国語明細書】
2013143146000001.pdf