Vol.2010-ARC-190 No.11
2010/8/3

サービス指向型ルータに利用可能なプライバシー保護アーキテクチャ

井上恒一[†/††]　石田慎一[†]　西　宏章[†/††]

通信機器の新しい機能として注目されている Deep Packet Inspection 技術は，通信パケットのペイロード情報を解析し，高度な通信制御を可能にしている．こうしたアプリケーション層の情報を活用した通信機器の高度化はサービス基盤化を指向する上で期待が大きい反面，アプリケーション層に含まれる利用者のプライバシー情報の保護に関する検討が欠かせない．本稿では，著者等が提案しているサービス指向型ルータに利用可能なプライバシー保護アルゴリズムを提案し，評価を行うとともに，ハードウェアアーキテクチャの提案を行った．匿名化処理を用いて処理されたデータテーブルでは *l*-多様性が満たされ，プライバシーが保護される．また，匿名化されたデータにおいては，ユーザの指定に基づき，注目属性については加工を施すことなく，推測による個人の特定の識別が防止される．

Privacy Preserving Architecture for the Service Oriented Router

Koichi Inoue[†] Shinichi Ishida[†] and Hiroaki Nishi[††]

Deep Packet Inspection technologies, receiving great attention as a new functionality of network equipments, inspect the payload portion of a packet, and enables advanced network management. These advancements of the network equipments dealing with the application layer enhance the progress to be a service platform. However, privacy information contained in the application later should be concerned to make it happen. In this paper, we proposed and evaluated a privacy preserving algorithm, and proposed hardware architecture. By the proposed method, the data to be anonymized are extracted by association rule mining. the data table processed by the method is guaranteed to have property, called *l*-diversity, and privacy is protected. In data table that anonymized by proposed method, the extracted data from network packet payload matching the criteria specified are not processed or generalized, and personal identification by data estimation is prevented.

1. はじめに

近年，OSI 参照モデルにおける最上位層にあたるアプリケーション層までの情報を識別する通信機器の Deep Packet Inspection（以下，DPI）技術の研究が盛んである．DPI 技術は通信されるパケットのペイロード情報を含めて分析することにより，さらに高度な制御を行うファイヤウォールや特定のコンテンツに対する帯域の制御，そしてコンテンツフィルタリングなどを可能にする．著者等はルータが通信トラフィックから獲得できるユニークなデータに着目し，これらをコンテンツとして活用可能にするサービス指向型ルータを提案し[1]，DPI の未来，そしてアプリケーション基盤として進化する次世代の通信ネットワークの研究を行っている．

こうした DPI 技術の活用，そして通信トラフィック情報のデータ活用においては，プライバシー情報の保護の議論は欠かせない．本稿では，著者等が提案するサービス指向型ルータでの利用を想定し，通信トラフィック情報のプライバシー保護アーキテクチャを提案する．本稿ではプライバシー保護のアルゴリズムを提案し，評価するとともに，処理速度を向上するためのハードウェアアーキテクチャを提案する．

2. 関連研究

近年，インターネット上に存在するあらゆる情報は，マッシュアップなど複数発信源からの情報を組み合わせる手法により，多角的な価値とコンテンツとしての意義を有するようになった．例えば Google や Amazon が次々と生み出す新技術が Web アプリケーション･サービスの高度化に寄与し，新たなビジネスを掘り起こし広げている．現在，Web アプリケーション･サービスのさらなる高度化のための 1 つの方法として，インターネット・インフラストラクチャであるルータやゲートウェイが取得可能な情報を積極的に活用する，あるいはルータが担うスイッチングをサービスに追加する研究が進められている[2][3][4]．例えば Cisco ISR (Integrated Services Router)向けに提供されている AXP (Application eXtension Platform) はルータ上で Linux アプリケーションを実行するための API を提供している．また，Active Network では，ネットワークノードがキャッシュを持ち，株式市況やオンラインオークションのサーバの負荷を軽減するために，トラフィックを解析してコンテンツに応じてパケット処理を最適化することなどが検討されてきた[2]．また，リコンフィギュラブルなハードウェアを用いることでアプリケーション層に及ぶパケット解析を高速に行い，IP ベースではなくコンテンツベースのルーティングを行う研究も行われている[3]．これらの研究はルータ

† 慶應義塾大学大学院理工学研究科
Graduate School of Science and Technology, Keio University
†† 国立情報学研究所
National Institute of Informatics

が単なる通信基盤に留まらず，次世代のインターネットにおけるサービスの中核になりうることを示している．

一方で近年，情報システムと人間とのインタラクションや，情報システム間の通信が随所で行われている．これらのシステムは，その活動の所産である膨大なデータを絶えず生成している．それと同時に，これらの膨大なデータを活用したいという需要が生じた．この需要に応え，大量のデータを分析する理論や技術が 1990 年代に開発され，それらは今日データマイニング(Data Mining)と呼ばれている．例えば，オンラインバンクなどの金融サービスでは，各個人や組織の金融資産の取引履歴が蓄積され，また電子カルテを用いた医療サービスでは，病歴や投薬履歴などが蓄積される．さらに今後はセンサネットワーク，RFID (Radio Frequency IDentification)などのユビキタスデバイスの普及による個人の地理情報や行動履歴の蓄積と活用が盛んになると予測される．その一方で，蓄積されたこの様な情報が漏洩し悪用された場合の被害は深刻であり，サービス提供者には収集した情報に対して慎重な扱いが要求される．しかしながら，収集したデータのプロファイルに基づくデータマイニングはサービスのクオリティ向上に大きく貢献するため，プライバシー保護に万全を期するために価値ある収集情報が利用できない場合の損失も大きい．

プライバシー保護データマイニングは，秘密情報を含むデータが複数ノードに分散している場合に，自身以外のノードやデータ集約サーバには情報を開示せずに，集約されたデータ集合から計算可能な有用な知識を発見するための技術である．プライバシー保護データマイニングの目標は，秘密情報の保護と活用のバランスを適切に管理し，利用価値の高い秘密情報を安全かつ有効に活用することである．プライバシー保護データマイニングの具体的な手法としては様々なアプローチが提案されており，匿名化アプローチ，ランダム化アプローチ，暗号学的アプローチ，データベース問い合わせ制限などがある．

本稿では，匿名化により通信トラフィック情報のプライバシー保護達成を目指す．匿名化を用いる利点としては，元データにおける個別データ間の関係性を保ちながら匿名化されたデータを出力するため，出力データに対して更にデータマイニングが行える点，またデータの特性に関わらず自由な匿名化の水準を設定可能である点が挙げられる．

3. サービス指向型ルータとプライバシー保護アーキテクチャ

3.1 サービス指向型ルータ

従来のインターネット・ルータがパケット交換に専念する単なる伝送媒体としてスループットや帯域保証などの観点で評価されてきたことに対して，著者等の提案するサービス指向型ルータは，ルータが単なる通信基盤に留まらず，次世代のインターネットにおける情報技術とサービスの中核となりうることを示す積極的な提案である．つまり，インターネット上に流通するあらゆるデータが公平あるいは公正な競争のもとで，各企業，各ユーザに提供される通信ネットワークアーキテクチャである．情報をコンテンツへと加工し，サービスに活用しようとする要求，さらにそこで生まれる創意工夫は，新しい情報社会システム実現の大きな牽引力であるとともに，さまざまな価値を産み出し，結果として経済活動を伴うことで豊かな社会を創造することができる．実際，様々な制限が存在するインターネットは，その制限の中にあっても，この様な要求・創意工夫により必要不可欠なインフラへと進化し，常に新しい価値を生み出してきた．著者等はこの様な考えに立ち，ルータをパケットデータの管理基盤として考え，この管理基盤から有用な情報を正規表現により抽出し，その情報をルーティングや新しいサービスの提供に生かすという観点からサービス指向型ルータを提案してきた．このルータは，既存のルータに高速な情報抽出を効率的に行うための正規表現プロセッサとオンメモリデータベースへの高速なデータインサーションを行うハードウェアを加えた構成を持つ．この 2 つの核となるハードウェアのアーキテクチャの提案が行われており，5Gbps 以上のスループット(データベースへの書き込み)を達成できることを示した[5][6]．トラフィック情報は，「ある URL にアクセスしたユーザは，他のどんな URL にアクセスするのか」，「ユーザがある URL にどのくらいの時間滞在していたのか」，「その情報がいつ，どこからネットワーク上に現れたのか」といった，検索サービスや人気調査にとって重要な情報を含んでいる．これらの情報は従来のネットワークシステムでは利用が困難であったが，我々が提案を行っているサービス指向型ルータアーキテクチャではリアルタイムでサービスに必要な情報をネットワークトラフィックから抽出することができる．

3.2 プライバシー保護

本稿で使用する用語を以下に定義する．また，本章の定義は文献[7]を参考とした．

(1) データテーブル

本稿で考慮するデータテーブルは，行に「タプル」，列に「フィールド」をとるテーブルとする．各々のフィールドを「属性」と呼び，実際に保持するデータ値の意味カテゴリを表す．

(2) 属性

一般にデータテーブルを構成する属性の値には個々に意味がある．その属性が持つ値が直接プライバシー情報を特定できると考えられる属性，例えば氏名や電話番号などを「識別子」と呼ぶ．ただし，識別子ではない属性であっても，他の属性と組合せることによって識別子と同じ働きをする可能性がある．この様な属性を「準識別子(Quasi-identifier: QI)」と呼ぶ．

準識別子とした属性の中で，データを解析する者にとって重要な項目であるため一般化を行わない準識別子を総称して「注目属性」と呼ぶ．それに対し，一般化処理を

行っても構わない属性を総称して「非注目属性」と呼ぶ．一般化とは，値を部分的に隠すことによりデータに秘匿性を与える処理である．また，複数タプルにおける非注目属性の値において，一般化された後のデータが同一であり，そのデータが仮に q*となる場合の複数タプルのグループを q*-ブロックと呼ぶ．

(3)　値一般化階層

提案手法では，各々の属性について設定された値一般化階層に基づいて一般化処理を行う．数値データなら下位桁から「*」で置き換える操作が考えられる．また文字列データであれば任意の位置の文字を「*」で置き換えることもできる．また意味を考慮して自由に設定することもできる．

(4)　k-匿名性

k-匿名性という性質は以下のように定義される．

「データテーブル中の各タプルにおいて，そのタプルの持つデータ値情報（各属性値の組合せ）と同じデータ値情報を持つタルが自分自身を含め k 個以上存在する状態」

準識別子の値が一致するタプルが 2 個以上存在する場合，データテーブルは 2-匿名性を満たす．k-匿名性を満たすテーブルではどのタプルも準識別子により一意に対応しないので，複数準識別子組合せによるデータ識別が防止される．また，少なくとも 3 つのタプルで QI 値が一致する場合には，このデータテーブルは 3-匿名性を満たすとされ，一般化処理により匿名性の k の値を増加させることができる．

(5)　*l*-多様性

l-多様性という性質は直感的には以下のように表現される．

「テーブル中の全ての q*-ブロックにおいて，出現する注目属性の値が少なくとも *l* 個の多様性を有する状態」

本稿では *l*-多様性の中でも，次のように定義される再帰的 *l*-多様性を適用する．まずデータテーブル中の各 q*-ブロックにおいて出現する注目属性の値それぞれの出現確率を算出する．その値の中で最も高い出現確率の値を r_1 と置き，次に高い確率値を r_2 と順に置く．仮にその q*-ブロック中には n 種類の注目属性が存在するとし，定数 c(c>0)を任意に定めた時次式が成り立つならば，その q*-ブロックは再帰的(c,*l*)-多様性を満たす．

$$r_1 \le c(r_l + r_{l+1} + ... + r_n) \ , \ 1 < l < n \qquad (1)$$

同様に残りの全ての q*-ブロックが再帰的(c,*l*)-多様性を満たすならば，このデータテーブルは再帰的(c,*l*)-多様性を満たす．

データテーブルに k-匿名性を保持させることにより，プライバシー保護が可能である．しかし，k-匿名性だけでは防げない，プライバシー保護に対する攻撃が指摘されている．それが同種攻撃と背景知識攻撃である．同種攻撃では，仮に攻撃者が特定の個人の準識別子の値を知っているとするならば，当該の個人がどのタプルグループ に属しているか判明する．背景知識攻撃では，もし攻撃者が非注目属性の値に関して背景知識を持っている場合，q*-ブロックに含まれる注目属性の値について推測範囲を狭めることが可能となる．

データテーブルが *l*-多様性を満足する場合には，同種攻撃や背景知識攻撃は不可能となる．よって本稿では，提供するデータテーブルが再帰的(c,*l*)-多様性において定数 c=1 とした再帰的(1,*l*)-多様性が保証されている状態をもって，データテーブルのプライバシーが保護されていると定義する．

(6)　データフォーマット

本稿で扱うデータテーブルのフォーマットを表 1 に示す．各属性のうち，id はデータ処理のために利用する属性であり，個人を識別するものではない．timestmap，src_ip，dst_ip，src_port，dst_port はネットワークトラフィックの主要な情報であると共に，データを送信したマシンや個人を識別する手がかりとなり得るため，特に指定しない限りは準識別子とする．ネットワークトラフィックから抽出したデータは extracted 属性の値として格納される．extracted 属性情報はデータ解析者にとって有用な情報であるため，一般化による加工処理を行わないことが望ましい．

属性	説明
id	識別子
timestamp	データ取得時刻
src_ip	送信元 IP アドレス
dst_ip	宛先 IP アドレス
src_port	送信元ポート番号
dst_port	受信側ポート番号
extracted	ネットワークトラフィックから抽出されたデータ

表 1 :　データテーブルのフォーマット

さらに，上記のデータフォーマットを有する 3 種類のデータテーブルを新たに定義する．

PT (Private Table) :　ネットワーク中のトラフィクストリームから抽出されたそのままのデータを格納した状態の秘密テーブル．

GT (Generated Table) :　PT に対し任意のタプルについて属性値の一般化処理を行った一般化テーブル．特に，任意の整数 k について k-匿名性を保証する GT を k-GT とする．

AT (Anonymized Table) :　PT に対し任意の QI と extracted 属性について *l*-多様性を保証するように一般化処理を行った匿名化テーブル．特に，任意の整数 *l* について *l*-多様性を保証するデータテーブルを *l*-AT とする．

4. ソフトウェア処理の検討

提案するプライバシー保護アーキテクチャはルータに搭載するものとし，下記に示す通り，外部データベースへのデータ提供の際のプライバシー保護処理を行うものとする．

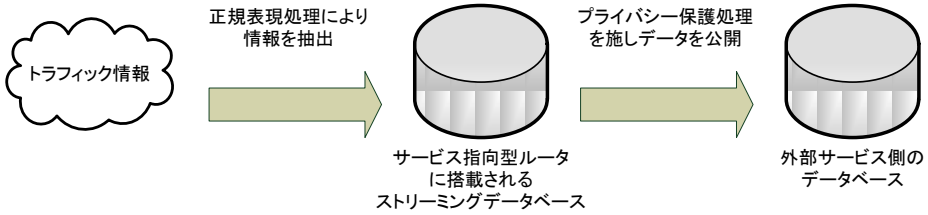

図 2： サービス指向型ルータにおけるプライバシー保護処理

本稿では，処理すべきタプルを逐次探索する手法と，クラスタリングを用いて処理コストを軽減した手法を提案する．クラスタリングを用いる手法は，階層的クラスタリングによる手法と非階層的クラスタリングによる手法をそれぞれ提案する．以下，それぞれの手法を逐次探索法，階層的クラスタリング法，非階層的クラスタリング法と呼ぶ．

4.1 逐次探索法

逐次探索法の処理手順を図 3 に示す．

逐次探索法ではデータテーブルに *l*-多様性を保証させるために，データテーブルに対して匿名化すべきタプルの逐次探索を行う．具体的には，任意のタプルグループにおけるある準識別子 QI の値 v と注目属性の値 s について，条件付き出現確率 p(s|v)が 1/*l* を超えるタプルが存在する場合には *l*-多様性が保証できないため，当該タプルについて一般化処理を行う．これにより，属性値は値一般化階層においてより上の階層に向かうため，QI の値が root である最大一般化値に向かい，一般化の過程で別の注目属性の値を有するタプルの QI と集約される可能性がある．仮に，タプル t1 における QI と注目属性の値が v1 と s1，別のタプル t2 における QI と注目属性が v2 と s2 であり，データテーブルについて p(s1|v1) = p(s2|v2) = 1，かつ，一般化関数 f について f(v1) = f(v2) = v0 であるとする．このとき，t1 と t2 から成るタプルグループにおいて準識別子の値を一般化した場合，データテーブルについて p(s1|v0) = p(s2|v0) =1/2 となり，2-多様性が保証された状態となる．ただしこれは一般化を行う前のデータテーブル全体において p(s1|v0) = p(s2|v0) = 0 であった場合の仮定であり，そうでない場合は必ずしも 1 回の一般化処理で多様性が保証できないため，再びデータテーブル全体を検査するステップを繰り返す．こうして処理すべきタプルを逐次探索し一般化処理を繰り返すことにより，最終的にデータテーブル全体の *l*-多様性を保証する．

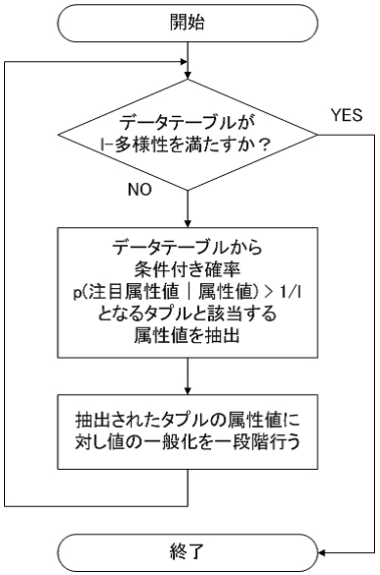

図 3： 逐次探索法

4.2 階層的クラスタリング法

階層的クラスタリング法の処理手順を図 4 に示す．

逐次探索法の場合，タプルに含まれる属性値の出現確率のみに注目し，値同士の類似度を考慮しないため，一般化すべきタプルの選択が適切に行われない可能性がある．仮に，逐次探索法において値の類似度が高いタプル同士を優先的に選んで一般化処理を行うならば，一般化処理後に同じ値となる確率が高い．理想的には，タプルグループを選択する際，一般化すべき非注目属性値について，値一般化階層における親ノードが同一となる子ノード同士を選択するのが望ましい．そうすれば一段階の一般化処理でタプルグループの k-匿名性が保証される．逆に，タプル選択が適切に行われない場合，一段階の一般化処理ではタプルグループの k-匿名性が保証されず，複数回の一般化処理が必要となる．

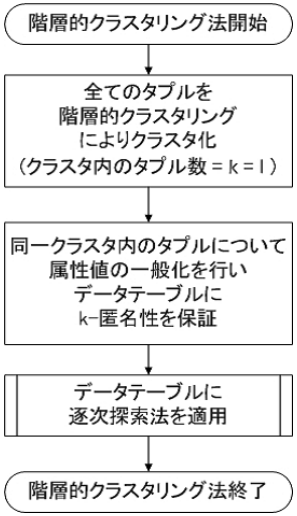

図 4： 階層的クラスタリング法

また逐次探索法では，抽出したルールに該当するタプルを一般化する処理の繰り返しによって，*l*-多様性を満たしていた q*-ブロック中の一部タプルの属性値が変化し *l*-多様性を満たさなくなる可能性がある．これが起こると，*l*-多様性を満足していた q*-ブロックが次の繰り返し処理では *l*-多様性を満たさないと判定される．つまり逐次探索法では, 目指す局所最適解に達するまでの一般化処理回数が増大する可能性がある．階層的クラスタリング法では，匿名化を行う前にあらかじめ階層的クラスタリングを用いて，一般化を行うべきタプルの候補を絞り込む．さらに，候補となったタプル中の属性値に対して一般化処理を行い，与えられた任意の整数 k について k-匿名性を保証した k-GT を出力する．そして，出力された k-GT に対して逐次探索法を適用し，最終的に任意の整数 *l* について *l*-多様性を保証された *l*-AT を得る．なお階層的クラスタリング法においては，常に k = *l* とする．一度 k-GT を出力するのは，あるタプルグループについて一般化を行った結果そのタプルグループが q*-ブロックとなったとき，その q*-ブロックが *l*-多様性を満たす可能性があるためである．理想的には，k-GT を出力した段階でその k-GT におけるすべての q*-ブロックが *l*-多様性を満たし，k-GT が *l*-多様性を満たすのが望ましい．

出力される k-GT は，逐次探索法において値一般化階層を高さ 1 ずつ上昇する過程を省略し，求める *l*-多様性を満たすデータテーブルの状態に近づく．よって，逐次探索法のプロセスにおいて PT を入力とする場合と k-GT を入力とする場合では後者の方が処理すべきタプル数が減り，処理にかかるコストも減少すると考えられる．

4.3 非階層的クラスタリング法

階層的クラスタリング法において階層的クラスタリングを行う場合，全てのクラスタ同士の距離関数を評価し，かつその計算がクラスタを併合する度に行われるためクラスタリングに係る計算コストが大きく，データ数(タプル数)の増加に伴い計算量が増大する．さらに，クラスタリングによって得られるクラスタは，*l*-多様性を満たす q*-ブロックを形成するためには最低でも *l* 個のタプルが必要となるため，整数 *l* の値が大きい場合には，個別タプル同士の類似度は必ずしも重要ではない．また，逐次探索法においても同様に，考慮すべき準識別子の数が増えるほど探索すべき連関規則の組合せ数が発散し，タプルの選択に必要な計算量が増大する．

非階層的クラスタリング法では，階層的クラスタリング法において階層的クラスタリングを行うプロセスを非階層的クラスタリングで置き換える．これによりクラスタリング処理に要する計算量のオーダーを $O(N^2)$から O(N)に削減する．さらにクラスタリング処理は考慮する準識別子の数によって計算量が変化しない特長があるため，データ量が考慮すべき準識別子が増加した場合でもより少ない計算時間内での処理が期待できる．また階層的クラスタリング法の場合と同様に，クラスタリング処理を行った後であれば逐次探索法のプロセスにおいて探索すべきタプル数が減少する．

非階層的クラスタリング法の処理手順を図 5 に示す．

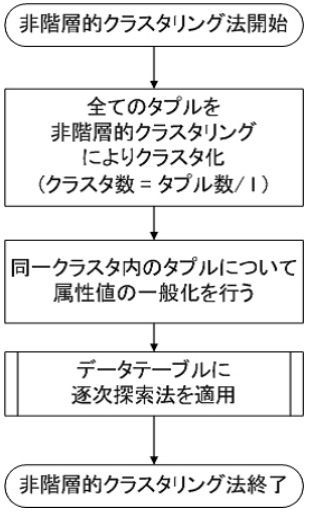

図 5： 非階層的クラスタリング法

Vol.2010-ARC-190 No.11
2010/8/3

5. ソフトウェア処理の実験と評価

各提案手法を計算機上で構築し，シミュレーション実験を行い評価した．用いたシミュレーション環境としては，CPU 構成はキャッシュサイズ 4096KB の Intel Xeon(R) CPU X5365 Quad-Core: 3.00GHz が 2 個, メモリ構成が DDR3 SDRAM: 8192 MB である．開発環境としては統計解析向け開発言語 R を用いた．実データについては当研究室で開発したソフトウェアベースルータシミュレータを用い，2009 年 7 月に当研究室ネットワークゲートウェイサーバのトラフィックを 3 日間に渡って取得したデータから条件を指定し抽出した．抽出条件は行動履歴解析を想定し，extracted 属性値は，閲覧されたウェブページタイトルとした．上記のデータに各手法を適用した場合の計算時間を，開発言語 R の提供する system.time()関数により評価した．評価は各処理を 5 回ずつ行い出力されたプロセス時間の平均値を算出した．

5.1 逐次探索法の評価

図 6,7 に，逐次探索法においてタプル数，*l* 値をそれぞれ変化させた場合の，タプルに対して一般化処理が行われた回数と計算時間を，属性（準識別子）数|QI|ごとに示す．

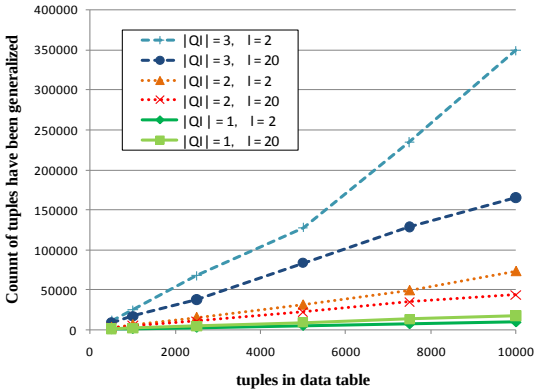

図 6： タプルの一般化処理回数

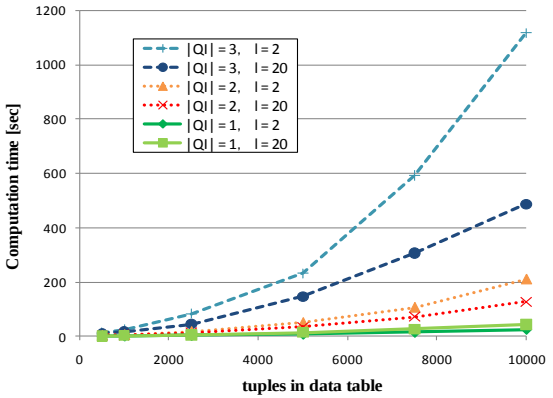

図 7： 計算時間

傾向として，計算時間はタプル数に大きく影響を受けるが，*l* 値の変化による影響は限定的である．計算時間と処理ルール数を比較すると相関関係があるため，逐次探索法の計算時間に支配的な影響を及ぼすのは処理タプル数である．計算時間，処理タプル数とも準識別子数が大きくなるに伴い増加するのは，考慮すべき準識別子数が増えるほど探索すべき準識別子の組合せ数が増大するためである．

5.2 階層的クラスタリング法の評価

図 8,9 に階層的クラスタリング法においてタプル数，*l* 値をそれぞれ変化させた場合の計算時間を示す．

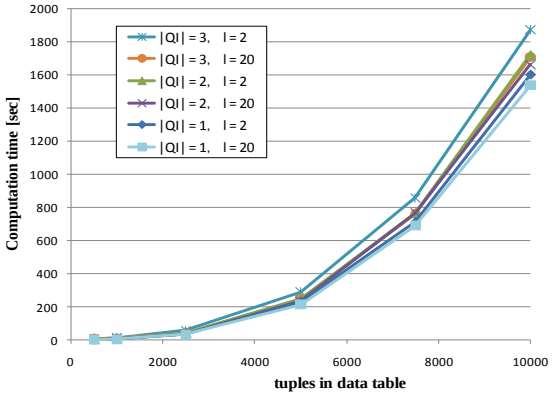

図 8： 階層的クラスタリング法全体に要する計算時間

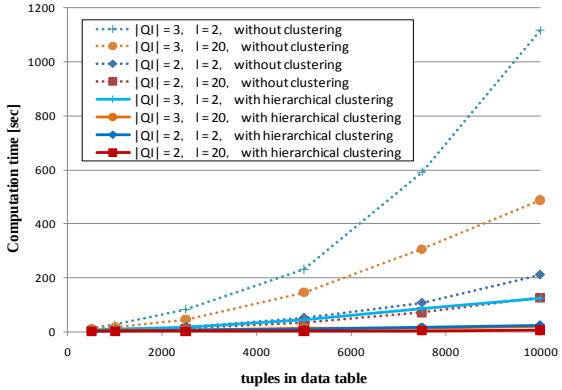

図 9： 逐次探索法のプロセス部分に要する計算時間

クラスタリング結果を用いた匿名化処理により，k-GT を入力とした逐次探索法のプロセスにおける処理タプル数は，PT を入力とした場合と比較して大幅に減少する．よって，クラスタリングの結果を用いてある程度まで一般化処理を行い，全プロセス中で逐次探索法のプロセスに係る計算コストを低減する狙いが達成されている．

一方，逐次探索法のプロセスに係る計算時間と比較して階層的クラスタリング処理に係る計算時間が支配的であり，全処理時間中で逐次探索法のプロセスに必要な計算時間は最高の場合でも全過程のうち 7%程度を占めるのみである．

支配的であるクラスタリング処理に係る計算コストが大きいため，階層的クラスタリング法は逐次探索法に対して全体の処理に係る計算時間が大きい．階層的クラスタリングの全プロセスに要する計算時間のうちクラスタリングに係る計算時間に影響を与える支配的な要因はタプル数である．これは，*l* 値または考慮する属性数によらず，全てのタプルを用いて階層的クラスタリングを行うためである．また後述するように情報損失度については，クラスタリングを用いる手法は逐次探索法と比較して大きくなる．

5.3 非階層的クラスタリングの評価

図 10,11 に非階層的クラスタリング法においてタプル数，*l* 値をそれぞれ変化させた場合の計算時間を示す．

傾向として，タプル数の増加により計算時間も増加する一方で，*l* 値の増加に伴い計算時間が減少する点が挙げられる．これは，非階層的クラスタリングを行う際に指定するクラスタ数 K を K=タプル数/*l* で与えるため，*l* 値の増加に伴いクラスタ数が減少し，クラスタリング処理に係る時間が削減されるからである．クラスタ併合のタイミングで全クラスタの中心点と全観測対象との距離を再計算するため，クラスタ数が減少すると計算回数も減少する．

全プロセス中で逐次探索法のプロセスに係る計算時間は，階層的クラスタリング法の場合と比較して 2～3 倍程度となる．この傾向はタプル数，*l* 値，|QI|が変わっても同様である．非階層的クラスタリング法全体の計算時間の中で支配的なのは原則的にはクラスタリング処理部分であるが，*l* 値や|QI|が大きい，あるいはタプル数が少ないと，クラスタリングの計算時間は，逐次探索法によるプロセスに係る計算時間と比較して同程度もしくは小さい場合もある．逐次探索法は|QI|の増加により計算時間が増加し，階層的クラスタリング法は計算時間が大きく|QI|や *l* によって計算時間がほぼ一定である．非階層的クラスタリング法の計算時間に関する傾向は他の 2 手法と比較した際に有利であるといえる．

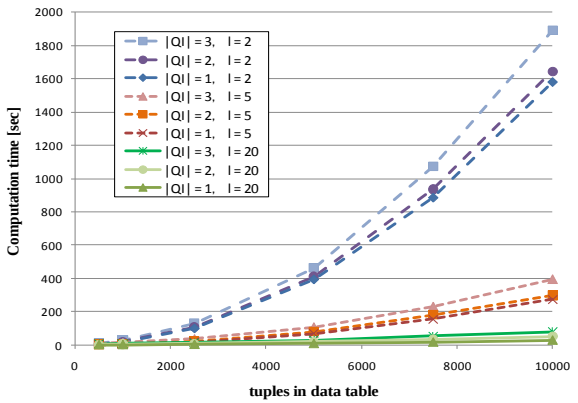

図 10： 非階層的クラスタリング法全体に要する計算時間

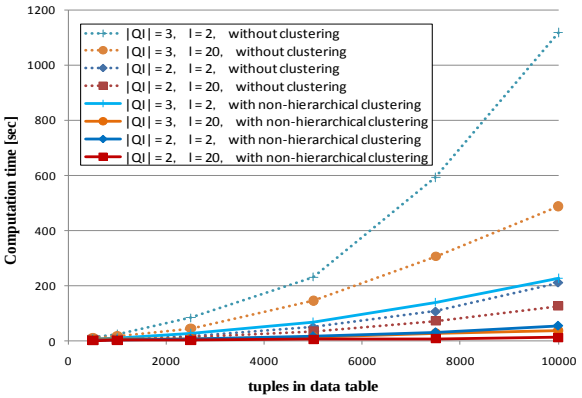

図 11： 逐次探索法のプロセス部分に要する計算時間

6. ハードウェア処理の検討

ソフトウェア処理における計算時間は，準識別子数が少ない場合や多様化指数 l の値が小さい場合には多くのケースで逐次探索法が最も小さい．しかし準識別子数が 2 以上，もしくは l の値が大きいと非階層的クラスタリング法が計算時間では最小となる．階層的クラスタリング法はタプル数が少ないいくつかの条件では計算時間が最小となるが，タプル数が多いと三手法で最大の計算時間となる．また，タプル数の増加による計算時間への影響が大きい．

このように l-多様性を満たすようにクエリの結果をソフトウェアで処理した場合コストを考えると，実システムでの利用は困難と言わざるを得ない．

そこで，専用ハードウェアによる処理を考える．一般に l-多様性の算出には膨大なメモリ空間が必要となるが，その削減方法について議論する．l-多様性をハードウェアで実現する際，トラフィックストリームに対してあるウインドウサイズを定めて，その情報をデータテーブルに蓄える．その中でのみ l-多様性を満たすような出力を行うように処理すれば，注目するウインドウサイズを限定することで情報損失度が増大するが，効果的にメモリ使用量を削減し，確実に l-多様性を満たすことができる．また，匿名化においても，もっとも情報損失度が少ない部位を限定的に匿名化する手法が望ましいが，ここでは匿名化する箇所はあらかじめ優先順位が付けられており，優先順位の高いところから順に匿名化することで，計算コストを削減することを考える．

まず，トラフィックストリームを到着時間順にデータテーブルに蓄える．その中で，注目するタプルと同じタプルを選び出し，その数 n が $n<l$ である場合は l-多様性を満たしていないため，匿名化処理を行う．匿名化処理には，優先順位に従って該当するビット(最優先順位から k 個)を全て 0 もしくは 1 に統一し，再び注目するタプルと同じタプルを選び出し，その数 n が $n>=l$ を満たすまで n と k を増大させながら繰り返す．なお，注目するタプルは，データテーブルの中ほどにあるデータであることが好ましいと考えられる．

上記を実現するために，一般にルータの経路探索で利用されている Ternary CAM を応用したハードウェアを考える．Ternary CAM は，ルーティングテーブルの検索で用いる場合，ターナリ値が示すプレフィックス長情報から比較するビット列の長さを任意に変更したうえでの可変長のビット列の比較を可能とする CAM である．まず，l-多様性実現ハードウェアとしては，比較の結果該当した数を算出すること，各比較 k を自由に変更可能であることが必要となるため，次のようなハードウェアを利用する．まず，マスク値は固定ではなく可変とする．IP アドレスの保存に利用される領域にタプルを保存する．また，プライオリティエンコーダは必要ないため削除し，代わりに比較該当数を数え上げるツリーカウンタを備える．なお，匿名化するべきデータ列が長い場合には，データ列をブロックにわけ，各ブロックについてのハッシュ値を格納することで対応することが考えられる．

このアーキテクチャにおいて，まず，トラフィックストリームが順に専用 TCAM に蓄えられているとする．l-多様性を確認するために，まず全比較回路を利用して類似タプルの数を算出する．その結果，$n<l$ であった場合，比較結果が一致したタプルのマスク値を 1 増やす．結果として，タプルの最優先順位ビットが匿名化されたことと

なり，このビットは今後の比較において常にヒットするようになる．この状態で再び比較を行う．この処理を繰り返すことで，n>=*l* とする．この状態で専用 TCAM 内部のタプルを該当するマスク値に合わせて匿名化して処理結果とする．なお，処理結果はトラフィックストリームの時系列順において最も古いタプルである．その後，新しいタプルを時系列で最も新しいタプルとして登録する．一度マスク値が修正された場合は，そのマスク値を記憶するとともに，今後その値が増加することがあっても減少はさせない．

以上の処理を行うことで，効率よく *l*-多様化を行うことが可能となる．

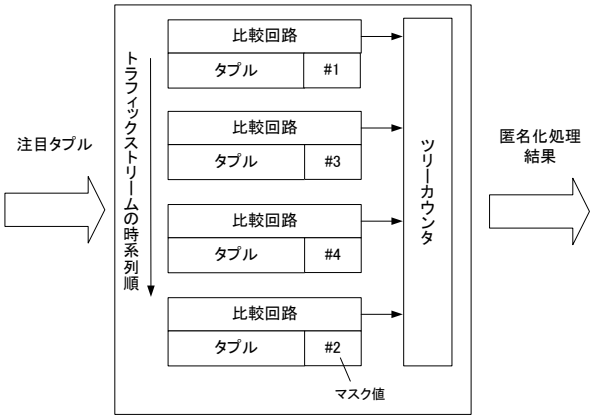

図 12： ハードウェアアーキテクチャ

7. 結論と今後の課題

本稿では，通信機器において通信パケット情報を有効活用することを目的として，通信機器に搭載することを想定してプライバシー保護のアルゴリズムを提案し，評価するとともに，処理速度を向上するためのハードウェアアーキテクチャを提案した．ソフトウェア処理での検討を踏まえ提案したハードウェアアーキテクチャにおいては，加工が望ましくない注目属性を柔軟に設定し，加工の優先度を選択できる汎用性の高いアーキテクチャに至ることができた．今後の取り組みとしては，ハードウェア実装を行い，ネットワークの実トレースを利用した評価を予定している．

謝辞 この研究は，独立行政法人情報通信研究機構の高度通信・放送研究開発委託研究「新世代ネットワーク技術戦略の実現に向けた萌芽的研究」および文部科学省科学技術研究費補助金基盤研究 C「安心・安全な情報提供を可能とするインターネット基盤の構築に関する研究」の一環として行われた．

参考文献

[1] K. Inoue, D. Akashi, M. Koibuchi, H. Kawashima and H. Nishi: “Semantic router using data stream to enrich services", Proc. CFI, pp. 20-23 (2008).
[2] David J. Wetherall and Ulana Legedza and John Guttag: “Introducing New Internet Services: Why and How”, IEEE Network Magazine (1998).
[3] J. Moscola, Y. H. Cho and J. W. Lockwood: “A reconfigurable architecture for multi-gigabit speed content-based routing", HOTI '06: Proceedings of the 14th IEEE Sympo sium on High-Performance Interconnects, Washington, DC, USA, IEEE Computer Society, pp. 61-66 (2006).
[4] Lane, J.R. and Nakao, A.: “End-Host Path Monitoring and Selection Supporting Packet Dispersion on Multipath Overlay Networks”, Proc. of the 3rd International Conference on Future Internet Technologies (CFI08) (2008).
[5] 永冨泰次，石田慎一，三野峻徳，川島英之，鯉渕道紘，西　宏章：リッチなユーザサービスを提供するセマンティックルータにおける正規表現プロセッサの提案，電子情報通信学会，ネットワークシステム研究会 (NS) (2008).
[6] 牧野友昭，辻　良繁，川島英之，鯉渕道紘，西　宏章：サービス指向型ルータにおける高速な書き込み機構の提案”，電子情報通信学会技術研究報告，電子情報通信学会，コンピュータシステム研究会 (CPSY2009-23),Vol. 109, No.168，pp.79‐84 (2009).
[7] L. Sweeney. “Achieving k-anonymity privacy protection using generalization and suppression”, Int. J. Uncertainty. Fuzziness Knowledge.-Based Systems, vol.10, No.5, pp.571‐588, 2002.
[8] R. Agrawal and R. Srikant: “Privacy-preserving data mining, ACM SIGMOD Record, Vol. 29, No. 2, pp. 439-450 (2000).