秘密計算技術の基本方式の安全性に関する基準および
その応用方式・応用システムの安全性の宣言に関する基準

2022 年 3 月 10 日

秘密計算研究会

用語定義

用語	説明
秘密計算	データを秘密分散・準同型暗号などの暗号技術で保護しながら復号・復元をせずに処理を行う技術
秘密計算方式	秘密計算を実現するためのアルゴリズムやプロトコル
秘密計算処理	実際の秘密計算の動作
秘密計算システム	秘密計算と、DB 等秘密計算以外のものを結合させたシステム全体
導入企業	秘密計算を導入し、その秘密計算を利用者に提供する企業
提供企業	導入企業に、秘密計算を提供する企業
ユーザ	秘密計算を利用する利用者
基本方式	3 章に記述。特段の前提なく暗号化したまま処理している秘密計算方式
応用方式・応用システム	4 章に記述。基本方式に当てはまらない秘密計算方式。特に方式そのものを応用方式、他のサブシステムと結合しているものを応用システムと呼ぶ
パーティ	秘密計算でデータのやりとりを行う主体
安全な基本方式の定義	全て暗号化したまま処理している基本方式が満たすべき条件
秘密分散	暗号化の一種。秘密計算の構成要素として用いられる
準同型暗号	暗号化の一種。秘密計算の構成要素として用いられる
ガーブルドサーキット	秘密計算の構成方法の 1 種
完全準同型暗号	暗号化の一種。秘密計算の構成要素として用いられる

1 はじめに

本文書は秘密計算技術の普及を目指し、どのような安全性を持つ方式・実装およびシステムがユーザにとって安全に利用できる秘密計算と言えるのかを議論し、秘密計算の基本的な方式における安全性基準や、それを応用した応用方式・応用システムにおける安全性の宣言に関する基準を議論・提案する。本文書は主に秘密計算技術のベンダ・開発者など、秘密計算の実用を推進する専門家向けに理解と協力を求めるものである。本文章で示している基準を参考にすることで、例えば、秘密計算を利用したシステムを提供する企業と導入する企業間の議論において、秘密計算の安全性についての相互理解が深まることで、秘密計算の普及が促進されることを目指している。本文章が示す安全性の基準は、より実効的な基準を目指して今後継続的に拡充・具体化を図っていく。

2 秘密計算の普及に向けた課題と安全性基準

2.1 秘密計算とは

本文書で考える秘密計算は、データを秘密分散・準同型暗号などの暗号技術で保護しながら復号・復元をせずに処理を行う技術を指す。特に主眼におくのは、統計分析や AI などのデータ分析が可能な秘密計算である。これは分析の前提となる、データベース処理や四則演算・論理演算などの基本演算を行う秘密計算を含む。文脈によっては秘匿計算やプライバシー保護技術などと呼ばれていることもある。

2.2 秘密計算の普及に向けた課題

秘密計算は古くから処理速度が課題であったが、近年研究の発展により性能は著しく向上し、実用的な処理が現実的な時間で可能な方式・実装も現れ、普及が期待される。しかし、新たな課題が顕在化しつつあり、普及の妨げになっている。それは、秘密計算にはさまざまな方式や安全性が存在し、ユーザにとって理解が困難で、自身にとって適切なものを選択することができず、秘密計算の導入を見送ったり、導入したとしても不適切な秘密計算を導入してしまうリスクがある。以降に、この課題の要因を示す。

2.2.1 要因 1：複数の基本的な方式の存在と俯瞰的議論の不在

一つの要因は、加算・乗算等を実現する基本的な方式のレベルでも秘密分散ベースの秘密計算（もしくはマルチパーティ計算)、準同型暗号、ガーブルドサーキットなど複数の方式が存在し、それらの間で異なる安全性の議論がなされていることである。これら複数の方式は全く構成方法が異なるため、複数の方式を理解している者は多くはない。そのため、方式を俯瞰した安全性の議論は限られており、方式毎に異なる安全性を主張してきた。これは、暗号理論を理解しないユーザからすればどの方式が本当に安全なのか判断できず、秘密計算を導入する際の障壁となってしまう。

2.2.2 要因２：拡張性の高さに起因する安全性のバリエーションの多さ

もう一つの要因は、秘密計算を色々な場面に応用した際に、その場面に合わせて様々な前提を置くことによって生じる安全性のバリエーションの多さである。秘密計算は、加算・乗算などの基本的な演算から論理回路等を構成してより実用的なアルゴリズムや実装・システムに拡張してゆく。そのような拡張を性能よく実現していくために、例えば「このパーティはこのデータを見る権限を持っている」といった前提などをおき、一部のデータを復号したり最初から平文で扱ったりする場合がある。これは技術的には妥当な判断である。

しかし、暗号理論を理解しないユーザにとってみれば、そのような個別の各応用例が本当に安全なのかどうか判断できず、応用例の提案を信じることしかできない。特に、ユーザの期待と採用した秘密計算の安全性との間に齟齬があることで、結果として「実は守りたいデータが漏れていた」場合、ユーザや社会から見れば「秘密計算は安全でない技術だ」という認識になり、普及の妨げとなる恐れもある。

2.3 秘密計算の安全性基準の目的

本文書の目的は、上記の秘密計算の安全性の理解の難しさに起因する懸念を解消し、秘密計算技術を社会的に安全な技術としていくことに貢献することである。前述の２つの要因を鑑みると、秘密計算の普及のために２つの観点の安全性基準が必要であると考えられる。

一つは良く知られた基本的な方式を俯瞰し、方式に寄らない統一的な要件を明確にする観点である。本文書では、まず秘密計算とは暗号化したまま処理する技術であると定義する。その上で、秘密計算の安全性を「暗号化したまま処理する」という共通点でシンプルに表現し、ユーザに理解しやすいように方式に寄らず統一的に定義する。

二つ目は基本方式を拡張した応用方式や実装・システムが、どのような条件を満たせば安全と言えるのか明確にする観点である。本文書では、秘密計算が「暗号化したまま処理する」技術であるとの定義のもと、その範囲、すなわち、暗号化されている範囲、復号したり平文のまま扱う範囲を、明確に公開することでベンダとユーザの齟齬を低減する。

本文書ではこれらをそれぞれ「秘密計算技術の安全な基本方式の安全性に関する基準」、「秘密計算技術の応用方式・応用システムの安全性の宣言に関する基準」と呼び、この二つをまとめて「秘密計算の安全性基準」と呼ぶこととする。

2.4 安全性基準の策定方針

本文書で提案する秘密計算の安全性基準では、重要な考え方が２点ある。

１点目は、理想の安全性を目指すのではなく、安全性として一定の価値がある秘密計算であれば基準内とすることである。セキュリティの専門家コミュニティでは、しばしば効率等を犠牲にしてもより安全であるほうが良いこととして捉えられるが、ユーザから見て一定の価値があるのであれば、秘密計算としても価値があるからである。

2 点目は、秘密計算を提供する側とユーザの間での安全性の理解の齟齬を可能な限り無くすことである。現状では技術的に正しくても、保護する範囲の理解の齟齬が原因で、ユーザにとって保護してほしかったはずのデータが守られておらず情報漏洩等のインシデントとなる可能性がある。これにより「秘密計算は安全ではない」という解釈が広がってしまうと、秘密計算の技術の普及には大きなマイナスである。

本文書では、まず複数の基本的な秘密計算の実現方式を俯瞰し、詳細な条件を付与せずとも明らかに安全と言えるために秘密計算の範囲について明示し、改めて「基本方式」として定義する(3 章)。その上で、基本方式には当てはまらない、秘密計算の方式・実装や従来のシステムと結合したもの（これを本文書では応用方式・システムと呼ぶ）は、ユーザに齟齬なく安全性を伝えられることが重要であるため、そのような安全性の宣言に関する基準を定義する(4 章)。

3 秘密計算の基本方式の定義と基準

本章では、最も基本的な秘密計算である基本方式の定義と、その安全性の統一的な基準を与える。

3.1 秘密計算の安全性の観点

秘密計算では複数の主体がデータのやりとりを行う。本文書ではこの主体をパーティと呼ぶ。大まかに言えば、秘密計算が安全であるとは、1 つないしは複数のパーティが攻撃者としてふるまった場合に元データが漏れないことが必要である。その中で、攻撃の前提や攻撃者の能力等によりいくつかの安全性の観点が存在する。以下では、秘密計算の安全性を評価するための主な評価軸を挙げる。

3.1.1 入出力

秘密計算の入力および出力の形式が暗号文であるか、もしくは暗号化されていない平文であるか。

3.1.2 復号に必要なパーティ数と許容される攻撃者数

復号に必要なパーティ数とは、パーティのうち最低何人集まれば復号できるのかを表す数であり、攻撃者数とは、パーティのうち許容される攻撃者数である。なお、完全準同型暗号においても復号には鍵の管理者が必要なので、計算するパーティと合わせ、復号に必要なパーティ数は最低でも 2 となることに注意されたい。許容される攻撃者の人数においては特に、許容される攻撃者数が復号に必要なパーティ数の半数未満の場合を honest majority、半数以上の場合を dishonest majority などと呼ぶ。Dishonest majority の方が許容される攻撃者の数が多く、そのような攻撃者の数に耐える秘密計算は安全性が高い。

3.1.3 攻撃者のふるまい

攻撃者がどのような攻撃を行うかという評価軸である。プロトコルに従うが自身が得た情報から元データの情報を得ようとする攻撃者のふるまいを passive（semi-honest, honest-but-curious とも呼ばれる）、改ざんを含む任意の行動を行う攻撃者のふるまいを active（malicious とも呼ばれる）と呼ぶ。Active である方が攻撃者の能力が高い。

3.1.4 攻撃者の計算能力

攻撃者の計算能力による評価軸である。攻撃者は現実的な計算リソース（メモリ・計算量）を利用すると仮定するものが計算量的安全性であり、制限を設けず攻撃者が無限の計算リソースを利用してもよいのが情報理論的安全性である。当然、情報理論的安全の方が攻撃者の能力は高い。

3.1.5 その他（入力の独立性・改ざん検知・公平性・出力保証）

攻撃者のふるまいが active である場合に幾つかの観点がある。入力の独立性とは、パーティは他のパーティの入力に依存して自身の入力を選択できないという性質であり、改ざん検知とは攻撃者がプロトコルにおいて改ざんした場合にそれを検知できる性質であり、公平性とは一人のパーティが出力を得たならばその他のパーティ全員も出力を得ることが出来るという性質であり、出力保証とは、攻撃者がどのようなふるまいをしたとしても全てのパーティは必ず出力を得ることができるという性質である。

3.2 基本方式が満たすべき安全性

2.2 節で議論した通り、秘密計算の普及に向けた課題は、複数の方式毎に異なる安全性議論、および安全性のバリエーションの多さである。これらの課題を解決するためには、秘密計算の統一的な安全性基準は「秘密計算 = 方式に拠らない単一の安全性を満たすもの」となるのが最もシンプルで分かりやすく望ましい。そのため、本文書では、安全性基準は方式によらず定めることとする。

次に、前述の安全性の評価軸において、どのレベルまで満たせば安全とすればよいだろうか。 2.4 節で議論したように、ユーザに対して齟齬なく明快に、一定の価値を与えられるかどうかが重要なため、まずは想定する秘密計算の価値を定義する必要がある。

まず、秘密計算の価値の定義は「暗号化したまま処理ができる」が適切だと考えられるであろう。価値をよく表しており且つ自明に安全でない方法を含まないものである。ここで暗号化とは旧来の暗号だけでなく、秘密分散や、一様乱数でマスクされたものも含んでいる。

そのため、各評価軸においては「暗号化したまま処理ができる」という価値を提供できれば安全とする。この考えのもと、無条件で「暗号化したまま処理している」と見なせる基本方式と言えるために、具体的に各々の観点で満たすべき要件は下記となる。

3.2.1 入出力

入力と出力のどちらかが平文だと「暗号化したまま」ではないので、どちらも暗号文である必要がある。

3.2.2 復元に必要なパーティ数と許容される攻撃者数

秘密計算であるため、復元に必要なパーティ数は 2 以上である。3.1 節で触れた通り完全準同型暗号の場合であっても復号には鍵の管理者が必要なので、パーティ数は 2 となることに注意されたい。また、2 パーティ以上であれば、各パーティは暗号化したまま処理ができる。そのため、パーティ数は 2 以上、許容される攻撃者数は 1 が必要である。

3.2.3 攻撃者のふるまい

Passive, active どちらであっても暗号化したまま処理ができる。また、データやプロトコルの改ざんを行う技術力はないが盗み見はするという攻撃者も現実的であるため、passive でも価値がある。

3.2.4 攻撃者の計算能力

計算量的安全性、情報理論的安全性どちらであっても暗号化したまま処理ができる。また、計算量的安全性は十分な理論的根拠と実績を併せ持つ実用的な安全性であるため、計算量的安全性でよい。

3.2.5 その他（改ざん検知・公平性・出力保証・入力の独立性）

これらはどの安全性も攻撃者が active である場合の安全性観点であり、攻撃者が passive であることを認めているため、どの観点も必須ではない。

3.3 基本方式の定義

以上のことから、無条件で「暗号化したまま処理している」と見なせる基本方式の満たすべき必要条件を列挙すると下記となる。

安全な基本方式は、以下の必要条件をすべて満たすものである。

条件 1. 入出力が暗号文であること

条件 2. 入力に依存したデータは常に暗号化したまま処理されていること

条件 3. その暗号化は少なくとも 1 人以上の、passive な攻撃者に対し計算量的安全であること

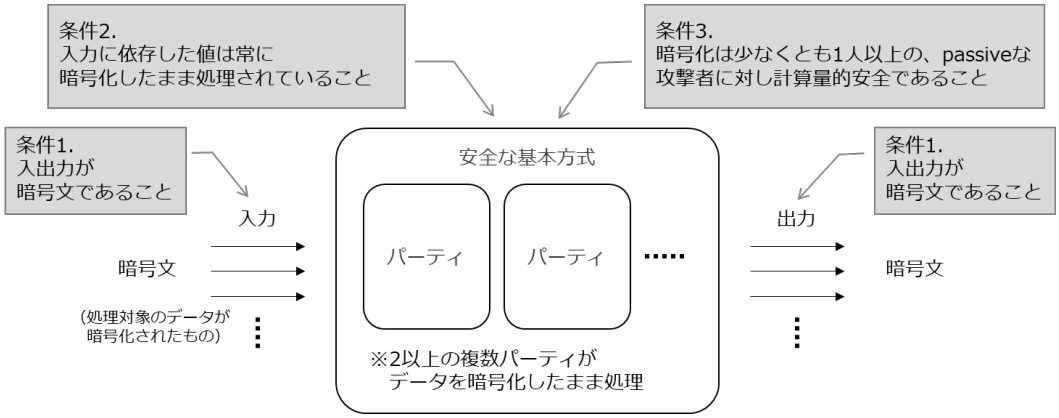

図 1 基本方式

この定義に当てはまるものとして、秘密計算の主な構成方法である、秘密分散ベース、準同型暗号を用いたものを例示する。

秘密分散ベースでは、データは秘密分散という技術を用いてシェアと呼ばれる状態で、分離して保存・管理される。それぞれのシェアからは元データの情報がわからないよう、暗号化が為された状態である。統計分析等の計算を行う際には、そのようなシェアの状態を保ったままで（元のデータに戻さずに）計算を行う。前述の安全な基本方式の定義に照らし合わせると、秘密分散ベースの秘密計算においては、入力・出力が共にシェアであり、その秘密計算の中で秘密情報の復元が行われないものが基本方式に当てはまる。

準同型暗号ベースでは、データは準同型暗号を用いて暗号化されて保存される。計算を行う際には、準同型演算と呼ばれる手続きを行うことで、暗号文のままで（元のデータに戻さずに）計算を行う。そのため、やはり入力・出力が共に暗号文であり、秘密計算の中で暗号文の復号が行われないものが基本方式に当てはまる。

4 秘密計算の応用方式・応用システムの安全性の宣言に関する基準

秘密計算には、3 章で定義した基本方式に属さないものも存在する。そのような基本方式の範囲外の秘密計算として、本章では、基本方式に当てはまらない秘密計算を応用方式、秘密計算を他の（平文で処理をする）実装と組み合わせたシステムのことを応用システムと呼ぶこととする。

これら応用方式・システムは無条件で「暗号化したまま処理している」と見なすことはできず、安全であるためには何かしらの特別な条件が必要である。そのような条件に対して技術提供者とユーザとの理解に齟齬が発生すると、結果として「実は守りたいデータが漏れていた」などといったことが発生する。

このような安全性の理解の齟齬を防ぐため、応用方式・システムでは、「ユーザに理解できるように安全性を宣言する」ことを基準の原則とする。これを達成するため、宣言すべきだと考えられる項目を抽出し、これらが宣言されていることを現時点の具体的基準と

する。さらに、理解の助けのために、どのようなケースがあるのか例も挙げる。

4.1　応用方式

応用方式は基本方式に当てはまらない秘密計算であるので、安全な基本方式の定義の否定を取ることにより、以下のうちどれかである。なお、条件 3 の否定は自明に安全でなくなるため含めていない。

A）一部の入力が暗号文でない

B）一部の出力が暗号文でない

C）暗号化されていたデータの一部が処理中に復号されている

A,B,C のどれかに該当する応用方式には、例えば以下のようなものが考えられる。

1. 秘密でない平文と暗号文が混在する応用方式
   - 属性の値は暗号化されているが、レコード ID は平文として扱う秘密計算
2. あるパーティにとっては秘密だが別のあるパーティにとって秘密ではない入力があるという応用方式
   - 2 者間でお互いのデータを統合・分析する場合（入力のうち自分のデータの部分は秘密ではないと扱う応用方式）
3. 元データに由来するが秘密にしなくてよいと判断したデータを復号する方式
   - 対象のデータベースのレコード数を復号する秘密計算 DB
   - 機械学習の収束判定結果を復元する秘密計算 AI
4. 汎用的な分析が可能で、分析結果を都度復号する応用方式
   - 分析者が結果を見て試行錯誤しながら探索的に分析する統計分析システム

例えば古くから知られるガーブルドサーキットを用いた秘密計算である Yao の 2 パーティ計算では、すべての入力の平文について、2 パーティのうちどちらかが知っていることが前提となっているため、2 の前提が必要である。

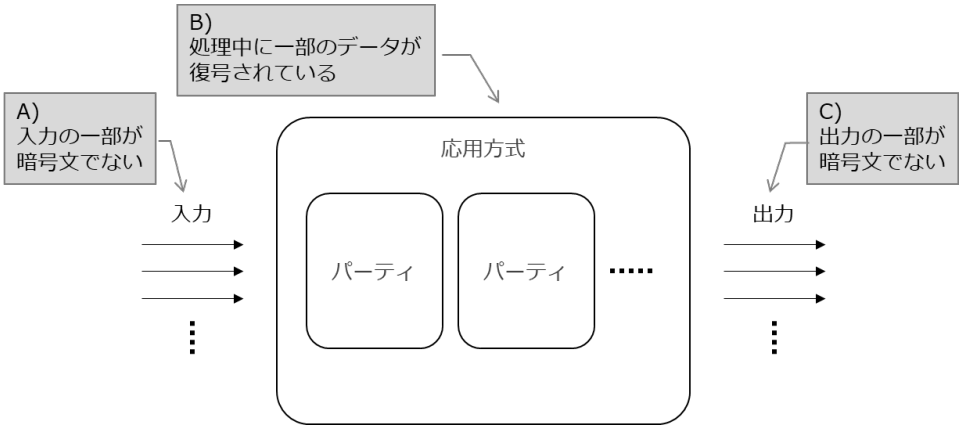

図 2 応用方式

4.2　応用システム

応用システムとは、一つや複数の基本方式・応用方式の実装を、通常の（平文で処理をする）サブシステムと結合したものである。例えば、秘密計算で統計値を計算する実装と、既存の統計分析システムを組み合わせて、分析者のインターフェースは統計分析システムを用い、実際に統計値を計算するときは秘密計算の実装を用いて計算する、といったものが挙げられる。

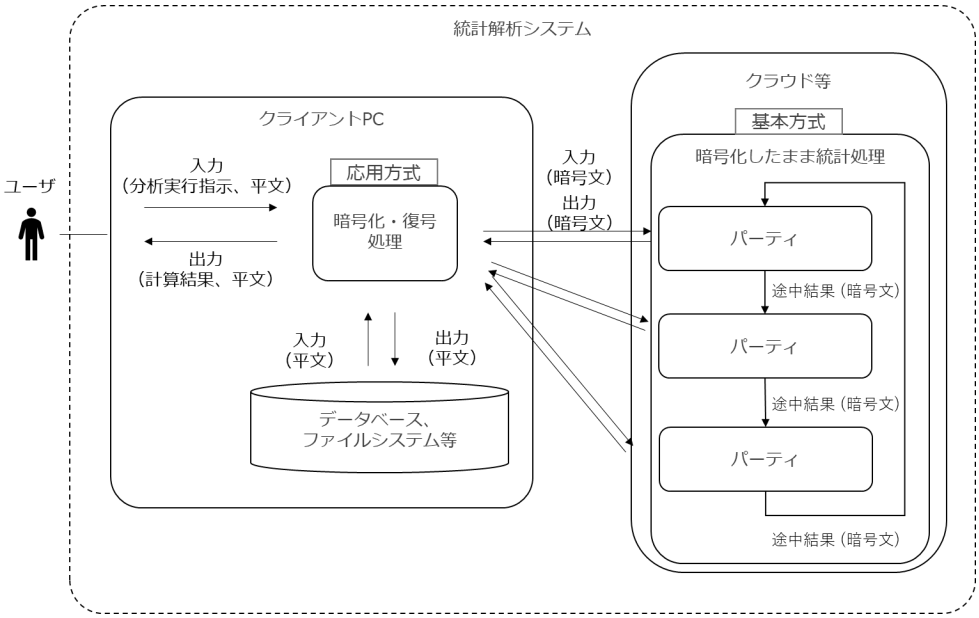

図 3　応用システムの一例

4.3　安全性の宣言の必要性

応用方式・応用システムは基本方式と異なり、全てを「暗号化したまま処理」しているわけではなく、何かしら復号される情報が存在する、もしくは基本方式では存在していなかった前提が必要となる。そのため、安全性基準を一律に設けることは原理的に困難である。一方で、復号される情報・および前提を正しく理解せずに利用すると、ユーザが期待している安全性を満たすことができない場合がある。そのため、応用方式・応用システムではそのような情報を、ユーザが理解できるレベルで適切に提示する必要がある。すなわち、安全な応用方式・応用システムに求めるものは以下の 2 点である。

1. 「暗号化したまま処理できる範囲」および「必要な前提」を宣言し公開する
2. 1 で宣言する内容は、ユーザに理解できる表現であること

また、これらの情報はユーザがサービスの導入を検討する際に必要な情報であるので、ユーザが簡便にアクセスできる必要がある。例えば、web 上で公開されているなどが望ましいと考えられる。

現段階では、2 については更なる議論が必要なため将来的な目標とし、継続的に議論を進めていくこととするが、2 も重要な事項であることには再度言及しておく。2.2.2 節に述べ

た通り、もし 1 の説明に暗号の専門家が用いる専門用語を用いて複雑な情報を提示すれば、暗号理論を理解しないユーザにとれば理解不能である。そしてユーザが安全性を理解できないまま結果としてインシデントが起きれば、秘密計算全体が安全でない技術という認識に波及する恐れがあるため、ユーザに理解できる範囲を適切に設定することは非常に重要である。

4.4 安全性の宣言に関する基準

本節では、応用方式・応用システムが宣言すべきと考えられる情報について述べる

4.4.1 応用方式における安全性の宣言

応用方式では一部の入出力が安全でない、もしくは何らかの前提の上で復号が為されているため、それらの情報を宣言すべきである。具体的には以下の情報を宣言すべきである。

応用方式に関しては、以下の情報を宣言しなくてはならない。

1. 一部の入力（出力）が暗号文でない場合
   - 暗号文として扱わない入力（出力）の明示
2. あるパーティにとっては暗号化されているが他のパーティにとっては復号される入出力がある場合
   - 特定のパーティに復号される情報の明示
3. 元データに由来するが秘密にしなくてよいデータがある何かが復号される場合
   - 復号される情報の明示
4. 平文と暗号文が混在する方式・実装
   - 平文として扱う，入出力以外の値の明示

4.4.2 応用システムにおける安全性の宣言

応用システムに関しては、以下の情報を宣言しなくてはならない。

1. 応用システムのうち，どの部分が秘密計算で構成されているのか
2. 秘密計算-サブシステム間, およびサブシステム同士でのデータのやりとりで復号がされているのか
3. 応用システムのうち，どの機能に秘密計算が使われているのか、どの機能は暗号化されておらず平文で扱っている（秘密計算が使われていない）のか
4. どのデータが，誰に対し（秘密計算技術の特徴により）漏れるのか，平文で処理されるため秘匿されないのか
5. どのデータが，誰に対し（秘密計算技術の特徴により）暗号化されたままで復号されないのか，暗号化されたものが復号されるのか

応用システムでは、通常は平文で処理を行うサブシステムと秘密計算が合成されるため、サブシステムと秘密計算がどこで、どのように分割されているのかを明示する必要がある。

4.4.3 安全性の宣言の例外

宣言が煩雑となり理解しづらくなることを防ぐために、復元しても明示する必要がないなどの例外も考慮していく必要がある可能性がある。しかし現時点では一律に宣言が不要だと技術的に保証できるレベルには議論が至っておらず、例外を設けない。

5 まとめ

本文書では、秘密計算の普及を目指し、複数の方式の俯瞰的議論の不在とユーザの理解できない安全性バリエーションの多さが課題であることを指摘し、まず安全と言える基本方式の安全性基準を策定し、さらに基本方式の枠に収まらない応用方式・応用システムに対しては、暗号化されている範囲をユーザと相互に理解できるよう、その範囲の宣言に関する基準を提案した。

本文章では秘密計算の方式に着目し、暗号化したまま処理している範囲を明確化するための基準を議論・提案した。さらに秘密計算を安全に実装・普及させていくためには、具体的なアプリーケーションやアルゴリズムごとのパラメータ設定や実装方法、運用方法の設定、さらにユーザに対して秘密計算の理解を促進してもらう活動なども重要である。

本文書は秘密計算研究会で議論されたものである。

6 付録：議論参加者・参加企業

・日本電信電話株式会社 菊池 亮、五十嵐 大、高橋 克巳
・株式会社イエラエセキュリティ
・株式会社デジタルガレージ
・日本電気株式会社

以上