Layered Attribute-Based Encryption (LABE)

概要

本文書では、大規模な医療データを取り扱う際の効率性とセキュリティを両立させるため、新たな暗号化方式「IPFSおよびNFTメタデータを活用した階層化属性ベース暗号 (Layered ABE with IPFS and NFT Metadata)」を提案します。この方式は、従来の属性ベース暗号 (Attribute-Based Encryption, ABE) を基盤としつつ、IPFS (InterPlanetary File System) をデータの保管に、NFT (Non-Fungible Token) のメタデータをアクセス制御情報の一部として活用します。具体的には、医療データをABEで暗号化した後、そのデータのIPFSにおける所在情報（CID: Content Identifier）を、NFTの保有者にアクセスが制限されたLit Protocolの分散鍵で暗号化し、その結果をNFTのメタデータに格納します。これにより、巨大な医療データに対する暗号化・復号化の計算負荷を軽減し、かつ多層的で柔軟なアクセス制御を実現します。本文書を読むことで、この新しい暗号化方式の仕組み、利点、およびセキュリティに関する考察を、前提知識がない読者でも一意に理解できるよう詳細に解説します。

1. 準備：理解に必要な基礎概念

この章では、本文書を理解するために必要な基礎的な概念を、集合と写像を用いて厳密に定義します。

1.1. 集合の定義

- $\mathcal{U}$ **(全属性の有限集合):** システムで利用可能な全ての属性（例：役職、資格、所属機関など）の集まりです。
- $\mathcal{P}$ **(アクセス構造の集合):** 誰がデータにアクセスできるかの条件を定義する構造の集まりです。各アクセス構造 P は、属性の集合の部分集合の集まりとして表されます。例えば、$P = \{\{\text{研究者}\}, \{\text{医師}, \text{特定の疾患}\}\}$ は、「研究者である」または「医師であり、かつ特定の疾患に関する属性を持つ」場合にアクセス可能であることを意味します。数学的には、$P \subseteq 2^{\mathcal{U}}$ と記述されます。ここで、$2^{\mathcal{U}}$ は $\mathcal{U}$ の全ての部分集合からなる冪集合です。
- $\mathbb{A}$ **(属性値の集合):** 各属性 $a \in \mathcal{U}$ が取りうる値の集まりです。例えば、「役職」属性であれば「研究者」「医師」「看護師」などがその値となります。
- $\mathcal{K}$ **(鍵空間):** 復号鍵（データを復号するために使用する秘密の情報）が属する集合です。

- $\mathcal{C}_1$ **(一次暗号文空間):** ABEによって暗号化されたデータ（一次暗号文）が属する集合です。
- $\mathcal{M}$ **(平文空間):** 暗号化される前の元のデータ（平文）が属する集合です。
- $\mathcal{N}$ **(NFT識別子の集合):** 各NFTを一意に識別するためのIDの集まりです。
- $\mathcal{L}$ **(Lit Protocolの分散鍵空間):** Lit Protocolによって生成・管理される分散鍵が属する集合です。この鍵へのアクセスは、特定の条件（例：NFTの保有）を満たすユーザーに制限されます。
- $\mathcal{CID}$ **(IPFSのコンテンツ識別子の集合):** IPFSに保存されたデータの内容に基づいて生成される一意な識別子です。同じ内容のデータであれば、常に同じCIDを持ちます。
- $\mathcal{C}_2$ **(二次暗号文空間):** IPFSのCIDがLit Protocolによって暗号化されたデータ（二次暗号文）が属する集合です。

1.2. 写像の定義

ここでは、それぞれの要素間の関係性を明確にするために、写像（ある集合の要素を別の集合の要素に対応付けるルール）を定義します。

- $\text{attr} : \mathcal{K} \to 2^{\mathcal{U}}$**:** 復号鍵 $k \in \mathcal{K}$ を入力とし、その鍵に関連付けられた属性の集合 $\text{attr}(k) \subseteq \mathcal{U}$ を出力する写像です。
- $\text{policy} : \mathcal{C}_1 \to \mathcal{P}$**:** 一次暗号文 $c_1 \in \mathcal{C}_1$ を入力とし、その暗号文がどのような属性条件（アクセス構造）の下で復号可能かを定義するアクセス構造 $\text{policy}(c_1) \in \mathcal{P}$ を出力する写像です。
- $\text{encrypt}_{ABE} : \mathcal{P} \times \mathcal{M} \times PP \to \mathcal{C}_1$**:** アクセス構造 $P \in \mathcal{P}$、平文 $m \in \mathcal{M}$、公開パラメータ PP を入力とし、ABEアルゴリズムを用いて生成された一次暗号文 $c_1 \in \mathcal{C}_1$ を出力する写像です。公開パラメータ PP は、ABE暗号化に必要な公開情報です。
- $\text{decrypt}_{ABE} : \mathcal{C}_1 \times \mathcal{K} \times PP \to \mathcal{M} \cup \{\perp\}$**:** 一次暗号文 $c_1 \in \mathcal{C}_1$、復号鍵 $k \in \mathcal{K}$、公開パラメータ PP を入力とします。もし、復号鍵 k に関連付けられた属性集合 $\text{attr}(k)$ が、暗号文 c_1 のアクセス構造 $\text{policy}(c_1)$ を満たすならば、元の平文 $m \in \mathcal{M}$ を出力します。そうでなければ、復号失敗を示す記号 $\perp$ を出力します。
- $\text{store}_{IPFS} : \mathcal{C}_1 \to \mathcal{CID}$**:** 一次暗号文 $c_1 \in \mathcal{C}_1$ を入力とし、その内容をIPFSに保存した際に得られるコンテンツ識別子 $\text{store}_{IPFS}(c_1) \in \mathcal{CID}$ を出力する写像です。
- $\text{retrieve}_{IPFS} : \mathcal{CID} \to \mathcal{C}_1$**:** IPFSのコンテンツ識別子 $cid \in \mathcal{CID}$ を入力とし、そのCIDに対応する一次暗号文 $\text{retrieve}_{IPFS}(cid) \in \mathcal{C}_1$ をIPFSから取得する写像

です。

- $\text{encrypt}_{Lit\_CID} : \mathcal{L} \times \mathcal{CID} \to \mathcal{C}_2$: Lit Protocolの分散鍵 $l \in \mathcal{L}$ とIPFSのコンテンツ識別子 $cid \in \mathcal{CID}$ を入力とし、Lit Protocolの暗号化アルゴリズムを用いて生成された二次暗号文 $c_2 \in \mathcal{C}_2$ を出力する写像です。この分散鍵 l へのアクセスは、特定のNFT保有者に紐付けられています。
- $\text{decrypt}_{Lit\_CID} : \mathcal{C}_2 \times \mathcal{N}_{held} \to \mathcal{CID} \cup \{\perp\}$: 二次暗号文 $c_2 \in \mathcal{C}_2$ と、復号を試みるユーザーが現在保有しているNFTの集合 $\mathcal{N}_{held}$ を入力とします。もし、$\mathcal{N}_{held}$ が c_2 に関連付けられたLit分散鍵へのアクセス条件を満たすならば、元のCIDを出力します。そうでなければ、復号失敗を示す記号 $\perp$ を出力します。
- $\text{keygen}_{ABE} : MSK \times 2^{\mathcal{U}} \to \mathcal{K}$: システムの管理者（Trusted Authority, TA）が保持するマスター秘密鍵 MSK と、ユーザーの属性の集合を入力とし、その属性集合に対応するABE復号鍵 $k \in \mathcal{K}$ を生成する写像です。
- $\text{setup}_{ABE} :\to (PP, MSK)$: ABE暗号化方式の初期設定アルゴリズムです。システム全体で公開される公開パラメータ PP と、TAが秘密に保持するマスター秘密鍵 MSK を生成します。
- $\text{metadata} : \mathcal{N} \to \mathcal{C}_2$: NFTの識別子 $n \in \mathcal{N}$ を入力とし、そのNFTのメタデータとして記録されている二次暗号文 $\text{metadata}(n) \in \mathcal{C}_2$ を取得する写像です。
- $\text{link} : \mathcal{C}_2 \to n \in \mathcal{N}$: 二次暗号文 $c_2 \in \mathcal{C}_2$ に対して、その暗号文がどのNFTの保有者に関連付けられているかを示すNFT識別子 $\text{link}(c_2) \in \mathcal{N}$ を出力する写像です。

2. IPFSおよびNFTメタデータを活用した階層化属性ベース暗号の構成

この章では、提案する暗号化方式の具体的な手順を説明します。

2.1. 暗号化の手順

平文である医療データ $m \in \mathcal{M}$ を、特定の属性条件 $P \in \mathcal{P}$ を満たすユーザーであり、かつ特定のNFT $n \in \mathcal{N}$ を保有するユーザーのみが復号できるように暗号化する手順は以下の通りです。

1. **一次暗号化 (ABE):** まず、ABE暗号化アルゴリズム encrypt_{ABE} を用いて、医療データ m を、アクセス構造 P と公開パラメータ PP を用いて暗号化します。これにより、一次暗号文 $c_1 = \text{encrypt}_{ABE}(P, m, PP)$ が得られます。この c_1 は、属性条件 P を満たすABE復号鍵を持つユーザーであれば復号できる形式になっています。

2. **IPFSへの保存:** 次に、この一次暗号文 c_1 をIPFSに保存します。IPFSは、データの内容に基づいて一意の識別子 CID を生成し、データを分散的に保存するシステムです。保存後、CID $= \text{store}_{IPFS}(c_1)$ を取得します。

3. **二次暗号化 (Lit Protocol):** 取得したCIDを、特定のNFT n の保有者のみがアクセス可能なLit Protocolの分散鍵 l を用いて暗号化します。これにより、二次暗号文 $c_2 = \text{encrypt}_{Lit\_CID}(l, cid)$ が得られます。この c_2 は、NFT n の保有者でなければLit Protocolの仕組みで復号することができません。また、この二次暗号文 c_2 がどのNFTに関連付けられているかを記録しておきます ($\text{link}(c_2) = n$).

4. **NFTメタデータへの格納:** 最後に、この二次暗号文 c_2 を、NFT n のメタデータとして記録します ($\text{metadata}(n) = c_2$). NFTのメタデータは、NFTに関する様々な情報を記録するために使用される領域です。

以上の手順により、医療データはABEで暗号化され、その所在情報であるCIDがNFT保有者によってのみ復号可能な形でNFTに記録されます。

2.2. 復号鍵の生成手順

医療データにアクセスするための復号鍵は、システムを管理するTAによって、ユーザーの持つ属性に基づいて生成されます。

1. **鍵生成 (ABE):** TAは、ユーザーが持つ属性の集合 S を確認し、マスター秘密鍵 MSK を用いて、その属性集合に対応するABE復号鍵 $k = \text{keygen}_{ABE}(MSK, S)$ を生成します。この鍵 k は、$\text{attr}(k) = S$ という性質を持ちます。

2. **鍵の配布:** 生成された復号鍵 k は、安全な方法で該当するユーザーに配布されます。

2.3. 復号の手順

NFT n を保有しており、かつ医療データが要求する属性条件を満たすABE復号鍵 k を持つユーザーが、NFT n に関連付けられた医療データを復号する手順は以下の通りです。

1. **NFT保有の確認:** まず、ユーザーは自身が保有するNFTの集合 $\mathcal{N}_{held}$ に、アクセスしたい医療データに関連付けられたNFTの識別子 n が含まれているかを確認します ($n \in \mathcal{N}_{held}$)。

2. **二次暗号文の取得:** NFT n を保有している場合、NFT n のメタデータから二次暗号文 $c_2 = \text{metadata}(n)$ を取得します。

3. **CIDの復号 (Lit Protocol):** 次に、Lit Protocolの復号アルゴリズム decrypt_{Lit_CID} を用いて、取得した二次暗号文 c_2 を復号します。この復号には、ユーザーがNFT n を保有していることの証明が必要です。復号に成功すると、IPFSのコンテンツ識別子 $cid = \text{decrypt}_{Lit\_CID}(c_2, \mathcal{N}_{held})$ が得られます。NFTを保有していない場合、このステップで復号は失敗し、以降の処理は行えません。

4. **一次暗号文の取得 (IPFS):** CIDが得られたら、retrieve_{IPFS} 関数を用いて、そのCIDに対応する一次暗号文 $c_1 = \text{retrieve}_{IPFS}(cid)$ をIPFSから取得します。

5. **平文の復号 (ABE):** 最後に、ABE復号アルゴリズム decrypt_{ABE} を用いて、取得した一次暗号文 c_1 を、自身の持つ復号鍵 k と公開パラメータ PP を用いて復号します。この復号が成功するのは、自身の属性集合 $\text{attr}(k) = S$ が、一次暗号文 c_1 に設定されたアクセス構造 $\text{policy}(c_1)$ を満たす場合に限ります。復号に成功すれば、元の平文である医療データ m が得られます。

3. セキュリティと効率性に関する考察

この提案方式は、従来のABEにIPFSとNFTメタデータを組み合わせることで、セキュリティと効率性の両面で以下の利点をもたらします。

- **計算効率の向上:** 巨大な医療データに対する暗号化・復号化処理は、ABEによって一度だけ行われ、その結果はIPFSに保存されます。二次暗号化は、データの所在情報であるCIDに対してのみ行われるため、計算コストが大幅に削減されます。
- **ストレージの効率化:** 医療データ自体は、分散型のストレージシステムであるIPFSに保存されるため、単一のサーバーに負荷が集中するのを防ぎ、スケーラビリティが向上します。
- **多層的なアクセス制御:** 医療データへのアクセスは、ABEによる属性ベースの条件と、NFTの保有という所有権ベースの条件の両方を満たす必要があります。これに

より、よりきめ細かいアクセス制御が可能となり、不正アクセスのリスクを低減します。

- **柔軟な鍵管理と失効:** NFTの移転や無効化は、ABEの鍵管理とは独立して行うことができるため、アクセス権の変更や失効を迅速かつ柔軟に行うことができます。例えば、ある研究者の所属機関が変わった場合、その研究者のNFTのアクセス権を更新するだけで、過去のデータへのアクセスを制御できます。

ただし、この方式のセキュリティは、以下の要素に依存することに注意が必要です。

- **基盤となる暗号技術の安全性:** ABEおよびLit Protocolの暗号化アルゴリズムが、既知の攻撃に対して安全であることが前提となります。
- **NFTの秘密鍵の管理:** ユーザーがNFTの秘密鍵を適切に管理し、紛失や盗難を防ぐ必要があります。秘密鍵が漏洩した場合、第三者がデータに不正にアクセスする可能性があります。
- **TAの信頼性:** ABEのマスター秘密鍵を管理するTAが信頼できる主体であることが重要です。ただし、NFTによるアクセス制御が追加されているため、TAの不正行為が直接的にデータ漏洩に繋がるリスクは軽減されています。
- **IPFSの可用性とセキュリティ:** IPFSネットワークの可用性が低い場合、データの取得に遅延が生じる可能性があります。また、IPFSに保存されたデータの改ざんリスクについても、適切な対策を講じる必要があります。

4. 結論と今後の展望

本文書では、大規模な医療データ管理における効率性とセキュリティの向上を目指し、ABE、IPFS、およびNFTメタデータを組み合わせた新たな階層化属性ベース暗号方式を提案しました。この方式は、計算コストとストレージ効率を改善しつつ、多層的で柔軟なアクセス制御を実現する可能性を示唆しています。

今後の研究では、この提案方式の具体的な実装と性能評価を行うとともに、IPFSの可用性とセキュリティに関するより詳細な分析、および実際の医療データ管理システムへの適用可能性について検討を進める予定です。また、NFTのアクセス制御条件のより柔軟な定義や、ABEの属性管理との連携についても研究を進めていきます。