Vol.2019-IOT-44 No.35
2019/3/8

# ブロックチェーンを使用したクラウド上での ソフトウェア著作権保護システムの提案

高木誠也[†1]　柿崎淑郎[†1]　廣瀬幸[†1]　猪俣敦夫[†1]

**概要**：PC 上のソフトウェアには不正コピーの対策としてクラウド化が存在し，クラウド外のユーザとの通信をプロキシで管理することでユーザの利便性を上げる研究もおこなわれている．この研究のプロキシにはユーザ情報やライセンス情報が内蔵されており，攻撃が集中した場合に情報の流出等重大な損失が生じてしまう．そのため本研究では，この問題を解決するためにクラウド内通信にブロックチェーンを使用し，ユーザ情報やライセンス情報を格納することでプロキシのリスクを分散する手法を提案する．更に提案手法の有用性を評価するためにアプリケーションを開発し，実際にブロックチェーンに情報を格納できたことでプロキシでのリスクを軽減できていることを確認した．

# Proposal of Software Digital Rights Management System on Cloud Using Block-Chain

TAKAGI SEIYA[†1]　KAKIZAKI YOSHIO[†1]
HIROSE MIYUKI[†1]　INOMATA ATSUO[†1]

## 1. はじめに

正規の方法で入手していない著作権を侵害するような不正コピーは日本においても蔓延しており，2015 年時点で PC にインストールされたソフトウェアのうち正規の方法で入手しなかったソフトウェアは 18%，このような不正ソフトウェアによる損害額は 9 億 9400 万ドルと日本円換算で 1200 億円にものぼる[1]. この不正コピーの対策として販売会社が取り入れている手法の一つにソフトウェアのシリアル番号とユーザの端末番号で認証を行うアクティベーションが存在するが，認証を回避され，新たなアクティベーションを設定しては回避される繰り返しであり有効な手段とは言えない．またもう一つの手法に「クラウド化」が存在する．クラウドとは，データをユーザの PC や端末に置かずインターネット上に保存する利用方法やサービスのことである．この手法であればソフトウェアのダウンロードやインストールを行わずに実行するため，不正コピーの心配がないという利点がある．しかし，このクラウド化されたソフトウェアは販売会社ごとに著作権保護技術や認証技術，データセンターやネットワークのインフラが異なっており，ユーザは購入したそれぞれのソフトウェアに別々の認証を行わなければならない．そのため，ユーザはソフトウェア毎にログインを行わなければならない煩雑さのために，パスワードの使いまわしを行ってしまう危険性がある[2]．このようなパスワードの使いまわしは，同じパスワードを使いまわすユーザを標的に不正ログインを行う「パスワードリスト攻撃[3]」による不正アクセスにつながる恐れもあるため結果的にセキュリティとして脆弱なものになってしまう．

このような複数ある認証方式への対策として Lee らは二つのモデルを提案している[4,5]．一つはデジタル著作権保護（以下，DRM）機能を持った DRM クラウド[4]，二つ目はユーザの為に DRM サービスの管理を行う DRM プロキシ[5]である．しかし DRM プロキシではユーザ認証のためにプロキシ内にユーザ情報やライセンス情報をすべて格納しているため，プロキシを集中的に攻撃された場合に情報の改ざん・流出が発生した際のリスクが大きいという問題点が存在する．

本研究ではクラウド化されたソフトウェアの取引を行うクラウドとユーザ間の通信をプロキシに集約することで煩雑なライセンス管理を簡潔にし，ユーザの利便性を高めることでパスワードの使いまわしを解消するモデルを提案する．更にクラウド内の通信に分散型台帳であるブロックチェーンを用いユーザ情報やライセンス情報をブロックチェーン内に格納することで，情報改ざん・流出のリスクを軽減するモデルを提案する．また提案方式のアプリケーションを実装し，評価考察を行う．

## 2. 関連研究

異なるフォーマットのコンテンツを一つにまとめた上でデジタル著作権管理（以下 DRM）機能を使用するクラウドサービスとして，Lee の DRM クラウドと DRM プロキシというモデルがある[4,5]．2 つのモデルについて以下で説明する．

### 2.1 DRM クラウド

DRM クラウドとは Lee ら[4]が提案した DRM 機能を持つ

†1 東京電機大学

サービスを提供するクラウドであり，コンテンツのパッケージング･ライセンス管理･キー管理･ドメイン管理等 DRM に必要なリソースと機能を提供することを目標としている．この研究において DRM 技術者は DRM クラウドによって提供される API を使用して DRM 技術の提供や改修を行う．これによってユーザは DRM によって保護されたコンテンツの購入や受信を行うことができる．モデル図を図 1 に示す．DRM クラウドにはライセンス管理機能等の DRM 機能が備わっており，DRM 機能の種類やコンテンツに関わらず提供することができる．このように DRM クラウドにはいかなる種類の DRM 機能にも対応して提供することのできる高い相互運用性を持っている．

### 2.2 DRM プロキシ

DRM プロキシとは Lee ら[5]が提案した DRM クラウドの境界に存在している DRM クラウドの為のプロキシサーバのことである．この DRM プロキシには DRM クラウド内で DRM サービスに関する情報一式を管理し，ユーザの為のサービスの提供等ユーザ向け DRM サービスの管理全てを処理するという役割を持っている．例えばユーザが異なる DRM 技術のコンテンツを購入することがあってもユーザがアクセスするのは DRM プロキシのみであり，DRM 技術者が新しい DRM 技術を開発した際も DRM プロキシに送信すればよい．このようにユーザや技術者がアクセスするのは DRM プロキシのみであるため DRM 技術の管理が容易になっている．DRM プロキシが異なる DRM 技術で保護されたコンテンツをユーザに送る際，それぞれのコンテンツを再生するタイミングで DRM 技術を認証するエージェントをユーザにダウンロードさせる．これによって複数ある DRM 機能を相互に運用することができる．

### 2.3 関連研究の利点と課題

Lee らの研究において，DRM クラウドはユーザやデベロッパーからの通信をすべて DRM プロキシに集中させてから各 DRM クラウド内システムに分散している．認証自体はコンテンツ利用時にダウンロードしたエージェントによって行われるが，全てのコンテンツの購入処理を DRM プロキシで行えることからユーザ認証方式を DRM プロキシに一本化することができる．これによりユーザはコンテンツを購入し利用する際の ID やパスワードを 1 つに絞ることができ，パスワード管理を簡潔にできる．また DRM 技術を技術者が DRM プロキシに登録する方式により DRM クラウドはそれぞれの DRM 技術で保護されたコンテンツを適切な認証を使用することで配信することができるため，高い相互利用性を持っている．

しかし，この DRM クラウドは通信やユーザ・ライセンス情報が DRM プロキシに集中する形となっている．そのため DRM プロキシを攻撃されてしまうと DRM プロキシ内に存在するアカウント情報・認証ツールの改ざんや流出が発生した際のリスクが非常に大きい．そのほかにもライセンス情報・購入履歴等の改ざんや DRM プロキシが機能停止したことによるサービス停止の危険があるという課題が存在する．

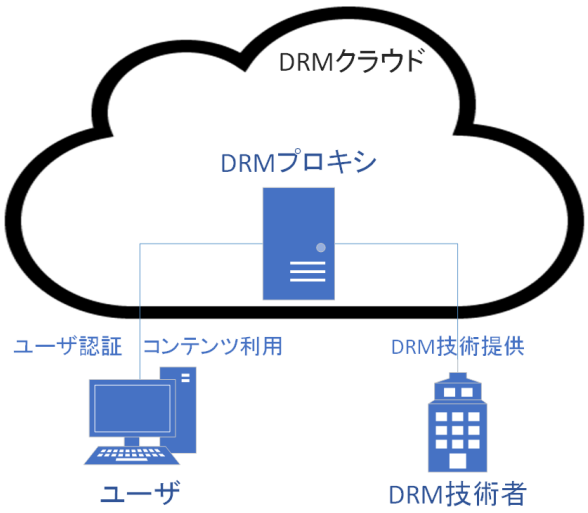


図 1　DRM クラウドモデル

## 3. 提案方式

### 3.1 要件

クラウド上で著作権保護システムを行うためには各々の販売会社がそれぞれ異なるクラウド化されたソフトウェアを販売しているため，それぞれのソフトウェアに別々の認証を行わなければならない．その煩雑さからユーザビリティの低下やパスワードの使いまわしによる不正アクセスにつながる恐れもある等，セキュリティとして脆弱なものになってしまう．それらの危険を防ぐため異なる方式をまとめて認証を一本化し，ユーザのライセンス管理を簡潔にすることでユーザの利便性を向上させる必要がある．

### 3.2 要件を満たす手法

Lee らの定義した DRM クラウドは，ユーザからの通信とコンテンツの配信，DRM 認証をすべて DRM プロキシで行うものであった．この方式であれば，ユーザはユーザ認証やソフトウェアの購入申請，ライセンス認証全てを DRM プロキシと通信することによって行うことができるため，認証の一本化が行える．これによりユーザが管理する ID とパスワードを一つに絞り込むことができるため，ユーザを悩ませた煩雑なライセンス管理を簡潔にすることができ，ユーザの利便性が向上できる．よって，要件を満たす手法は Lee らの提唱する DRM プロキシを引用した構成になる．

しかし Lee らの定義した DRM プロキシにはすべてのユーザ・ライセンス情報が格納されているため，DRM プロキシを攻撃され情報の改ざんや流出が発生した際のリスクが大きいという課題点がある．

### 3.3 ブロックチェーン技術の利用

Lee らの定義した DRM プロキシの問題点を解消するた

めにブロックチェーンの技術を利用する．ブロックチェーンとはP2Pのネットワーク上で動作する分散型台帳システムである．ネットワーク内で発生したトランザクションと呼ばれる取引が行われると，そのトランザクションの記録や付帯する情報等をブロックと呼ばれるパッケージに格納し生成する．この時ブロックにはトランザクションの情報に加えて，生成するひとつ前に生成されたブロックのヘッダ情報をハッシュ値と呼ばれる一定量の情報にまとめたデータも同時に格納する．このハッシュ値を時系列に沿って参照され続けるデータ構造が1本のチェーンのように連なっているため，ブロックチェーンと呼ばれる．またブロックチェーンで参照されるハッシュ値はデータの改変がなされていないかを確認するものである．ハッシュ値から元のデータを復元することはできないが，同じデータをハッシュ値に生成すると必ず同じハッシュ値が生成され，違うデータから同じハッシュ値は生成されない．この性質から生成したハッシュ値の比較を行うことでデータに改変が起こっていないかを確認できる．このように内容の改ざんが難しいハッシュ値をデータ構造として利用しているためにブロックチェーンは改ざん耐性が高いとされている．図2にブロックチェーンのデータ構造を示す．

このブロックチェーンがネットワーク上で分散型台帳としての役割を持つことで，ブロックチェーンの参加者はブロックチェーンに格納されているデータの全てを自身で保持せずとも運用ができる．そのためDRMプロキシがユーザ・ライセンス情報をブロックチェーンに格納し，認証処理の際にDRMプロキシがブロックチェーンに格納されているデータを検証することでDRMプロキシ自身がデータを保持せずとも認証処理を行うことができる．またブロックチェーンは参加者であればだれでも利用することができる分散型台帳であるため，DRMプロキシが多数配置されていたとしてもすべてのDRMプロキシで同じ処理を行うことができる．そのため一つのDRMプロキシが機能を停止しても他のDRMプロキシで処理を容易に行える．図3にDRMクラウド内でブロックチェーンを用いた通信について示す．更に高い改ざん耐性からブロックチェーンによる取引処理において取引の改ざんを防ぐことができ，悪意を持ったブロックチェーンの参加者による取引の改ざんや不正なユーザによる否認・なりすまし防止に繋がることができる．

また，使用するブロックチェーンはコンソーシアムチェーンである必要がある．DRMクラウド内の通信に使用するブロックチェーンがBitcoinのようなパブリックチェーンでは，不特定多数の参加者が入り取引履歴が全世界に公開されてしまう等プライバシーに問題がある．そのため参加者を特定複数に制限したコンソーシアムチェーンをクラウド内通信に用いることでプライバシーへの対策を行う．これにはブロックチェーンへの参加者を限定することで，悪意あるユーザが不正に侵入し改ざんや悪意あるトランザクションを行うことを防ぐ意味もある．またコンソーシアムチェーンではトランザクションを許可する合意形成の仕組みが参加者間の承認によって決定されるため，誤ったトランザクションや悪意のある参加者による不正なトランザクションが発生してもトランザクションをチェックした参加者により未然に防ぐことができる．

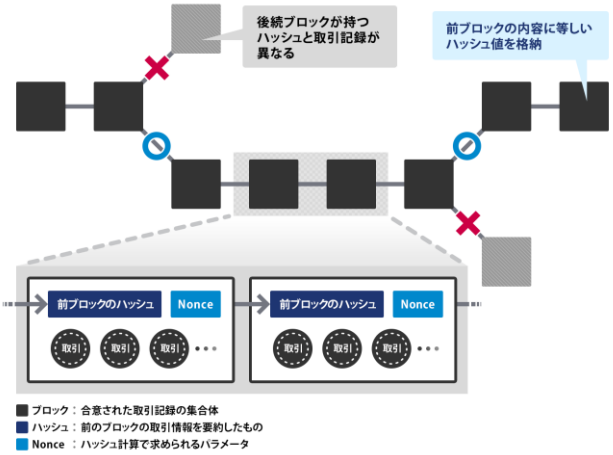


図 2　ブロックチェーンのデータ構造図[6]

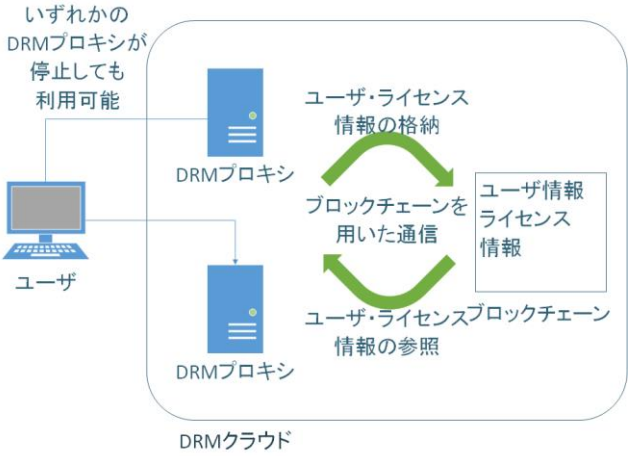


図 3　DRM クラウド内におけるブロックチェーン通信

### 3.4 提案方式

ユーザが複数の認証方式による煩雑さを嫌ってパスワードの使いまわしを行ってしまうと，パスワードの使いまわしを狙った攻撃による不正ログインの危険性が高まる．また複数の認証方式を解消するために Lee らの定義した DRM クラウドのモデルを使用すると，ユーザ・ライセンス情報がすべて格納されている DRM プロキシに攻撃を受け情報の改ざんや流出が発生した場合のリスクが高い．このため提案方式では DRM クラウド内通信にブロックチェーンを利用してユーザ・ライセンス情報を格納し，ブロックチェーンに格納された情報とユーザから DRM プロキシに送られてきた情報で認証を行う．これによって DRM プロキシは常時ユーザ・ライセンス情報を格納する必要が無

くなり，攻撃を受けた場合情報の改ざん・流出によるリスクを軽減することができる．提案方式を図 4 に示す．

### 3.5 提案方式の手順

まずユーザはユーザ登録を行うため DRM プロキシに ID・パスワード・メールアドレス等のユーザ情報を送信し，DRM プロキシは送られてきたユーザ情報を元にユーザ毎のブロックチェーンを作成しユーザ情報を格納する．その後ユーザがソフトウェアを購入する場合，自身の端末から購入したいソフトウェアの情報を DRM プロキシに送信し，ユーザが購入処理を完了する際に，DRM プロキシは当該ユーザのブロックチェーンに購入したソフトウェアの商品番号とソフトウェアの使用期限を格納する．同時に販売会社はソフトウェアのリリース時に生成したソフトウェア毎のブロックチェーンに購入したユーザの ID とソフトウェアの使用期限を格納する．

ユーザがソフトウェアを実行する際に，DRM プロキシは生成したユーザ毎のブロックチェーンとソフトウェア毎のブロックチェーンを元にユーザの検証を行う．検証が完了すると初回認証であった場合 DRM プロキシはユーザ毎のブロックチェーンに当該ユーザの使用する端末情報を記録し，ユーザは購入したソフトウェアを使用することができる．またユーザ情報とユーザごとのブロックチェーンを検証した際に端末情報が一致しなかった場合，端末情報を更新するトランザクションを行うことで以前までの端末では利用できなくなる．

## 4. アプリケーションの実装と評価

ブロックチェーンを用いた通信の有用性を評価するために，ブロックチェーンを使用する通信の部分においてアプリケーションを実装し，悪意あるユーザによる改ざんや不正ログインの防止について評価を行う．

### 4.1 ビジネスネットワークの構築

本研究では仮想環境を構築し，ブロックチェーン基盤として Hyperledger Fabric[7]を，開発ツールとして Hyperledger Composer[8]を使用した．仮想環境上にブロックチェーン環境の構築を行い，開発ツールを用いることでブロックチェーンによるビジネスネットワークを構築した．以下で構築したビジネスネットワークについて説明する．

構築したビジネスネットワークの参加者(Participant)として DRM プロキシが参加する Proxy，販売会社が参加する Company，一部の管理担当が参加する Manager の 3 組を設定した．これらの参加者が扱う資産(Asset)として DRM プロキシがユーザ毎に生成する User，販売会社がソフトウェア毎に生成するブロックチェーンである Software の 2 つを設定した．上記の参加者がユーザの購入処理によって各情報をブロックチェーン内に記録するために取引(Transaction)として DRM プロキシがユーザ毎のブロックチェーンに書き込む UserTrade，販売会社がソフトウェア毎のブロックチェーンに書き込む SoftTrade の 2 つを設定した．

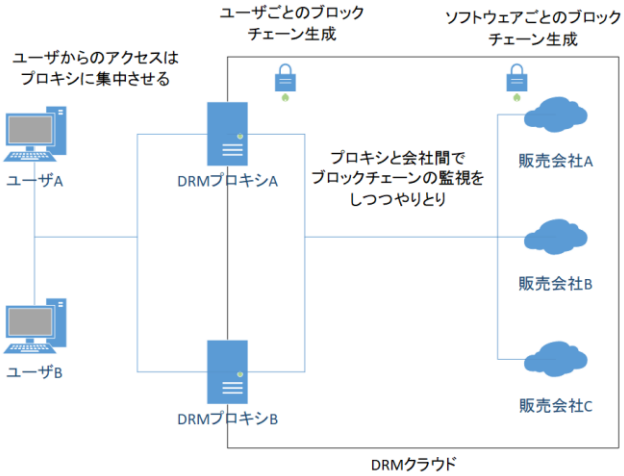


図 4 提案方式

提案方式において DRM プロキシはユーザ毎のブロックチェーンの生成・更新を行い，ソフトウェア毎のブロックチェーンには内容を検証する行為のみである．同様に販売会社はソフトウェア毎のブロックチェーンを生成・更新し，ユーザ毎のブロックチェーンでは内容の検証のみ行う．それ故に参加者の役割を超えた操作や誤った処理を行わないよう，参加者グループ毎にアクセス権を設定した．Asset におけるアクセス権の設定を表 1 に，Transaction におけるアクセス権の設定を表 2 に示す．表記はすべて許可設定であり新しく生成する CREATE，内容を閲覧できる READ，内容を更新できる UPDATE，削除を行える DELETE の 4 種類が存在する．また全て行える際には ALL と表記する．

構築したビジネスネットワークをデプロイし，ビジネスネットワークの参加者が http メソッドでアクセスすることによって操作が行えるよう，REST サーバを構築した．ビジネスネットワークの参加者である DRM プロキシや参加者が REST サーバにアクセスを行った際，まず Github による認証を行う．認証を行うことによって得られた Cookie や REST サーバの Web ページに記載されるアクセストークンを用いて通信を行うことで取引が許可されるため，部外者からの操作を防ぐことができる．またブロックチェーンの参加者である DRM プロキシや販売会社は前もってブロックチェーン環境の証明書の役割を果たすカードファイルが配布されている．そのためカードファイルを使用しなければ取引を行うことはできず，また参加者の権限を越えた操作が禁止されているため参加者側も不正に処理を行うことはできない．

### 4.2 アプリケーションの評価

実装したアプリケーションに http 通信を用いてデータの入出力とトランザクションの実行を行った結果設計通りに稼働したため，提案手法通りの情報をブロックチェーン内に格納できた．また，ブロックチェーンからハッシュ値を確認し格納されたデータの内容が調べられず，ハッシュ値

が 1 つ前のブロックのハッシュ値を継承していることを確認した．このことからブロックチェーンに格納されたデータが改ざん，盗聴される可能性は低く安全性は高いと言える．ただ，今回のアプリケーションは REST サーバへの通信に http 通信を利用して操作を行っている．これによって暗号化処理を施さないと通信内容から取引情報等の機密情報が簡単に盗聴されてしまう．そのため，実際に稼働させる場合には https 通信が必須となる．

通信速度の面において速度実装したアプリケーションに対し連続でトランザクション実行を行い，連続してトランザクションを実行した時やチェーンが連なったときのトランザクション処理速度において実験を行った．結果トランザクション実行時間はトランザクションの連なりに関わらず 2600 ミリ秒前後だったものの, トランザクション処理履歴取得に関しては処理数がトランザクション処理 10 回ごとに 75 ミリ秒程度処理時間が長くなることが判明した. このことから大規模なソフトウェアの販売の際には処理時間が秒数単位で延長されることが予想され，取引処理の遅延に繋がることが懸念される．

## 5. 考察

はじめに, 3.1 節で取り上げた要件について検証を行う．3.1 節では著作権保護として脆弱なものとなる要因であるパスワードの使いまわしを防ぐためにユーザ認証の一本化を要件としていた．本研究では提案方式に Lee らの定義した DRM クラウドのモデルを利用し, DRM プロキシがユーザからの通信を集約し認証処理を行うことでユーザ認証の一本化を行うことができている. そのためユーザは煩雑な, 要件を満たせている．次に 3.3 節で取り上げたブロックチェーン技術が DRM プロキシの問題点を解決できているかについて検証を行う．DRM プロキシを攻撃された際，情報の改ざん・流出が発生した場合のリスクが大きいことが DRM プロキシの問題点であった．本研究では DRM クラウド内通信にブロックチェーンを利用しユーザ・ライセンス情報を格納することができており，DRM プロキシ本体にユーザ・ライセンス情報を格納する必要が無くなるため情報の流出が発生するリスクを軽減することができている．

次に安全性と利便性の面から考察していく．まず安全性において，DRM クラウドを用いることでユーザが行わなければならない認証処理を一括でまとめることができている．これにより煩雑なライセンス管理が簡略化され不正アクセスに繋がる危険性は軽減されており，ユーザの認証の部分において安全性が高まっていると言える．DRM クラウド内の取引は改ざん耐性が高いブロックチェーンを利用して行われるため内容の改ざんは難しく，悪意のある取引を行うことも参加者の合意形成によって取引が成立するため困難であり悪意あるユーザからの攻撃を防ぐことができる. また REST サーバを操作する際にも REST サーバの Web

表 1　Participant のアクセス権設定_Asset

| | Software | User |
|---|---|---|
| Proxy | READ | CREATE<br>READ UPDATE |
| Company | CREATE<br>READ UPDATE | READ |
| Manager | ALL | |

表 2　Participant のアクセス権設定_Transaction

| | SoftTrade | UserTrade |
|---|---|---|
| Proxy | READ | CREATE READ |
| Company | CREATE READ | READ |
| Manager | ALL | |

ページによる認証やカードファイルによる権限の設定が行われていることから，DRM クラウド内の通信においても改ざんやなりすまし等が発生しづらく安全性の高い状態であると考えられる．ただし REST サーバも DRM プロキシと同様攻撃を受けた場合に情報の流出等リスクが存在するため，セキュリティレベルを強化する必要がある．

利便性においてこのシステムを利用してソフトウェアの購入を行うユーザについては実際に購入処理や起動処理等を行えるツールを使用してみてわかる内容であるため利便性の評価はできないが，ログイン処理を 1 つに絞ることができるという点においては利便性が向上できていると言えるだろう．処理時間においても以前までユーザ・販売会社が利用していたシステムと同値であるか短ければ利便性が向上していると言える．しかし 4.2 節での評価では大規模なシステムとなった際に処理時間が延長することが懸念されているため，対策を考える必要がある．

また REST サーバにおいて同じ処理を重複して送信してしまった場合や誤った処理を送ってしまった場合において, アプリケーションの都合上消去処理等はできても復元処理は困難であるため更新履歴等は残ってしまう．これは間違った取引が発生した際にその取引履歴を利用されて不正にソフトウェアを利用されてしまうほか取引を行っていないユーザに請求が届く等悪意あるユーザや参加者に悪用されてしまう可能性が高い．そのため合意形成による取引処理で誤った処理や重複した処理を防ぐ必要があるが，合意形成のために時間を消費してしまいその分利便性は低下してしまう．そこである程度のバランスを考えた上で折り合いをつけていく必要がある．

## 6. おわりに

本研究ではソフトウェアでの著作権保護におけるユーザ側でのセキュリティレベルの向上を行うために，ユーザからの通信をクラウドのプロキシにまとめる手法を提案した．

また悪意あるユーザによるライセンス情報の流出や不正なログイン・取引のリスクを軽減する方法として，クラウド内の通信を分散型台帳システムであるブロックチェーンを用いて行う手法を提案した．これら提案した手法を元にアプリケーションを用いて実験を行い，ライセンス情報やユーザ情報等を高い改ざん耐性を持つブロックチェーンに格納し実際に通信を行うことで，通信の内容が保護できまた正常に取引処理が行えることを確認した．今後は，ユーザが使用期限より前にソフトウェアを破棄した場合のトランザクション処理や，通信速度を考慮したブロックチェーンの扱いを考えていきたい．

## 参考文献

[1] ”BSA グローバルソフトウェア調査 2016 年版調査報告書”http://bsa.or.jp/wp-content/uploads/Global_Software_Survey2016_J.pdf (参照 2019-02-03)
[2] ”パスワードの利用実態調査 2017” https://www.trendmicro.com/ja_jp/about/press-release/2017/pr-20171005-01.html (参照 2019-02-03)
[3] ”プレス発表　パスワードリスト攻撃による不正ログイン防止に向けた呼びかけ” https://www.ipa.go.jp/about/press/20140917.html (参照 2019-02-03)
[4] H. Lee, C. Seo, and S. U. Shin, DRM Cloud Architecture and Service Scenario for Content Protection. JISIS. 2013, vol. 3, no. 3/4, p. 94-105.
[5] H. Lee, S. Park, C. Seo, and S. U. Shin, DRM Cloud Framework to Support Heterogeneous Digital Rights Management Systems. Multimed Tools Appl. 2016 vol. 75, no. 22, p. 14089–14109.
[6] ”ブロックチェーンの仕組み” http://www.nttdata.com/jp/ja/services/sp/blockchain/mechanism/ (参照 2019-02-03)
[7] ”HyperledgerFablic” https://hyperledger-fabric.readthedocs.io/en/release-1.4/ (参照 2019-02-03)
[8] ”Hyperledger Composer” https://hyperledger.github.io/composer/latest/ (参照 2019-02-03)