Vol.2016-CSEC-73 No.1
Vol.2016-IOT-33 No.1
2016/5/26

教育用PCにおける電子証明書の信頼性操作と複数の証明書チェインによる柔軟なアプリケーション実行制御

関根 利一[1]　岡本 大輔[2]　山井 成良[1]　北川 直哉[1]　河野 圭太[2]

概要：大学などの教育機関において，学生が使用する教育用 PC の管理方法の一つにアプリケーション実行制御システムがある．我々は，このアプリケーション制御を柔軟かつ堅牢に行うために，電子証明書を用いて制御するシステムの開発を行ってきた．しかし，これまでの電子証明書を用いたアプリケーション実行制御システムでは，アプリケーションを複数のグループに含めることができないという問題があった．そこで本研究では，電子証明書に対し複数の証明書チェインを用いることで，アプリケーションを複数グループに含めることを可能にし，より柔軟なグループ制御を行う手法を提案する．

Flexible Application Execution Control by Reliability Management of the Digital Certificates and Multiple Certificate Chains in Educational Computer Systems

Riichi Sekine[1]　Daisuke Okamoto[2]　Nariyoshi Yamai[1]　Naoya Kitagawa[1]　Keita Kawano[2]

Abstract: Application Execution Control System is one of the management methods of educational PC that students use. To perform this application control flexible and strict, we have developed Application Execution Control System using digital certificates. However, the previous system has a problem that it is impossible to include one application in plural groups. In this paper, we propose a more flexible control method using plural certificate chains to digital certificates to solve these problems.

1. はじめに

大学などの教育機関では，教育用 PC と呼ばれる PC 群が学内の複数施設に設置されている．学生がこれらのどの PC 端末からでも共通のユーザ環境を利用することができるように，教育用 PC の環境設定方法では，学内のサーバで共通のディスクイメージを集中管理し，ネットワークを通してディスクイメージをダウンロードすることで PC を動作させる方法が一般的に利用されている．しかし，授業によって必要になるアプリケーションと禁止するアプリケーションが異なるため，PC 上では様々なユーザ環境を提供する必要がある．共通イメージを用いるこの方法では，個別のユーザ環境を提供するにはその数だけイメージディスクを作成して反映させなければならない．そのため，この方法では必要となる環境の数だけイメージは増えることになる．

そこで，我々の研究グループでは同じイメージディスクを用いて要求に応じたアプリケーション環境を構築することができるアプリケーション実行制御システムを開発してきた．このシステムでは，教育用 Windows PC 上にあるアプリケーションに対して個別に実行を許可あるいは禁止と設定することで制御を行う．

また，このアプリケーション実行制御システムをより堅牢かつ柔軟に行うために，電子証明書を使用する制御方

[1] 東京農工大学
Tokyo University of Agriculture and Technology, Koganei, Tokyo 184–8588, Japan
[2] 岡山大学
Okayama University, Okayama, Okayama 700–8530, Japan

法 [1][2] を考案してきた．電子証明書による実行制御では，アプリケーションを起動する際にアプリケーションのデータから算出されるハッシュ値と証明書のデータから算出されるハッシュ値を比較する．この 2 つの値が一致すれば情報が正当なものだと判定されてアプリケーションの実行が許可されるが，値が一致しなければデータの改ざんがされたと判定して実行を禁止にすることができる．

加えて，電子証明書の信頼性は証明書チェインと呼ばれる階層構造で保証がされており，電子証明書を用いた実行制御では，この保証関係をアプリケーショングループ制御に応用している．電子証明書が有効であるか無効であるかの信頼性は，上位証明書によって保証されている．そのため，ある証明書の信頼性が保証されなくなった場合，その証明書により信頼性が保証されている下位証明書もまた保証されなくなる．そこで，任意の証明書の信頼性を操作することで，その下位にある証明書により署名された複数のアプリケーションを一括して操作するグループ制御を行うことができる．

しかし，アプリケーショングループを作成する際，アプリケーションによっては複数のグループに含めることが必要となる場合がある．これまでの電子証明書を用いた階層構造では 1 つのアプリケーションを複数のグループに含めることができないという問題点があった．そこで本論文では，複数の証明書チェインによる階層構造を用いることで，この問題の解決を行う．複数の証明書チェインは，複数の署名を付与する方法と複数の上位証明書により署名を行う方法により構築することができる．

本論文の構成はまず 2 章において，従来の電子証明書を用いたアプリケーション実行制御システムについて述べる．次に 3 章において，提案手法である複数の証明書チェインによる制御方法について述べる．さらに 4 章で，提案手法が正しく動作することを確認するために行った実験の結果を示し，5 章で本論文のまとめと今後の課題について述べる．

2. 電子証明書を用いたアプリケーション実行制御システム

本章では，これまでに開発してきた電子証明書を用いたアプリケーション実行制御システムについて述べる．まず，システム全体の構成について説明し，その後，電子証明書を用いた実行制御方法について詳しく述べる．

2.1 システム全体の概要

アプリケーション実行制御システムとは前述したように，Windows PC 上にインストールされたアプリケーションの実行を制御するシステムである．このシステムでは，教員から送られた制御ポリシーをもとに学生の使用する PC 上でアプリケーションの実行を制御する．

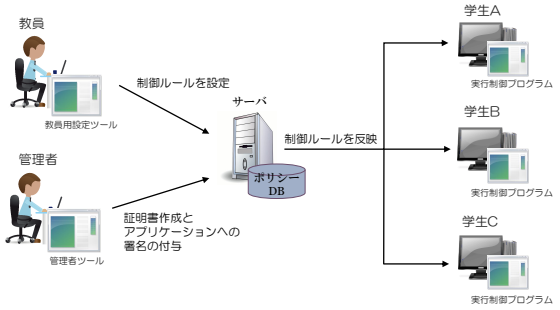

図 1　アプリケーション実行制御システムの構成図

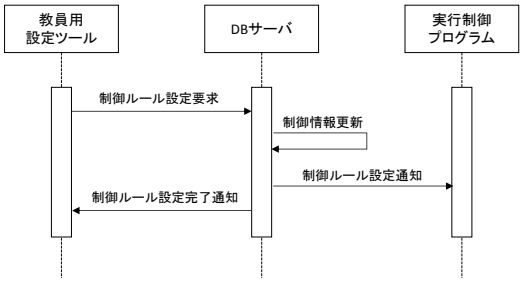

図 2　教員が制御ポリシーを設定するときのシステムの流れ

システムは図 1 のように教員用の設定ツール，管理者ツール，ポリシーデータベース（以下，ポリシー DB）を格納するサーバ，学生用 PC 上で動作する実行制御プログラムの 4 つで構成される．教員用の設定ツールは教員が学生に対し，どのアプリケーションの実行を許可し，どのアプリケーションの実行を禁止するか設定するプログラムである．次に，管理者ツールは管理者が電子証明書を作成し，アプリケーションに署名を付与するためのプログラムである．また，ポリシー DB サーバはアプリケーション制御情報を格納しており，制御情報が変更されたときには DB の内容を更新する．最後に，実行制御プログラムは設定されたポリシーをもとに教育用 PC 上で実際に制御を行うためのプログラムである．学生が教育用 PC にログインすると実行制御プログラムが起動し，あらかじめ DB に設定されている制御ポリシーを取得する．さらに，教員から制御ポリシーが送られると，それに応じたアプリケーションの制御を行う．

教員が設定ツールを用いて教育用 PC にルールを設定するときのシステムの流れを図 2 に示す．まず，教員が設定ツールで制御ルールの情報を入力すると，設定ツールから DB サーバへ制御ルールの設定要求がされる．DB サーバに教員から要求が送られると，DB サーバは DB 内に格納されている制御情報を書き換えた上で，実行制御プログラムへ制御ルールを送信する．また，DB サーバは実行制御プログラムにルール送信を行った後，教員用設定ツールに制御ルールの設定完了通知を行う．

教育用 PC 上での実行制御には Windows のグループポリシーの機能を使用している．グループポリシーの機能では各アプリケーションに対して，起動の許可，禁止を個別に設定することができる．この機能において，アプリケーションを識別するための情報として，実行ファイル名やパス，ハッシュ値，コードサイニング証明書 [3] と呼ばれるアプリケーションに付与された電子証明書を用いることができる [4][5]．電子証明書によるアプリケーション実行制御では実行を許可する場合，そのアプリケーションに付与された証明書を信頼できる証明書として登録する．一方，実行を禁止したい場合は証明書を信頼できない証明書として登録することで，その証明書を用いて署名されたアプリケーションの実行を禁止することができる．このようにしてグループポリシーでの電子証明書による制御では，証明書が信頼か不信頼かという判別により実行制御を行う．

2.2 電子証明書の信頼性操作によるアプリケーション実行制御

電子証明書による実行制御を行うために，学生が使用する教育用 PC 上では Windows のユーザ権限を適切に設定する必要がある．そのため，すべてのアプリケーションに対して，デフォルトで起動禁止にするように設定を行う．また，電子証明書のインポート権限を学生には持たせずに管理者のみができるように設定する．その上で，信頼された証明書で署名されたアプリケーションのみ実行を許可する．このように環境を設定することで，アプリケーションデータが改ざんされたような場合でも，アプリケーションのデータから算出したハッシュ値とアプリケーションに付与された証明書のデータから算出されたハッシュ値が一致しないことになり，電子証明書の署名が検証できないことからアプリケーションの実行を禁止することができる．

また，電子証明書の信頼性は証明書チェインと呼ばれる階層構造で保証がされている．この階層構造において証明書は大きく 3 種類に分けることができる．階層構造の最上層に位置するものをルート証明書，階層構造の最下層に位置するものをエンド証明書，その間に位置するものを中間証明書と呼ぶ．証明書チェインは証明書のプロパティから確認することができ，その例を**図 3** で示す．この図で Root と名前のついた証明書がルート証明書にあたり，InterA と InterB が中間証明書である．さらにその下の End がエンド証明書にあたる．この 3 種類の分類のうち，アプリケーションに付与するコードサイニング証明書は最下層のエンド証明書のことを指す．ルート証明書は下層の中間証明書の信頼性を保証し，中間証明書もまた，下層の中間証明書やエンド証明書の信頼性を保証する．この電子証明書の階層的構造による保証関係をアプリケーションのグループ制御に利用している．

ある中間証明書やルート証明書の信頼性が保証されなくなった場合，その証明書により信頼性が保証されている下位証明書は保証されず，さらにその下位証明書で保証されている証明書も同時に保証されなくなる．この保証関係が繰り返され，下層にあるすべての証明書の信頼性が保証されない状態となる．上位証明書の信頼性が保証されなくなったときの証明書チェインは**図 4** のように確認できる．これにより，任意の中間証明書やルート証明書の信頼性を操作することで，その下位にある証明書で署名されたすべてのアプリケーションを一括して操作するグループ制御を行うことができる．

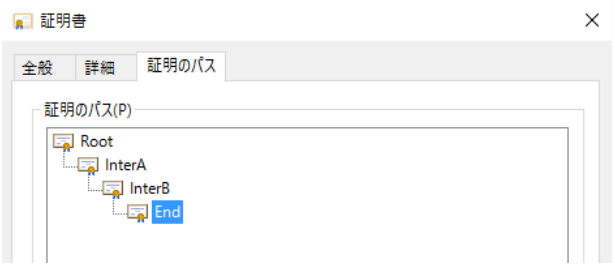

図 3 証明書チェインの例

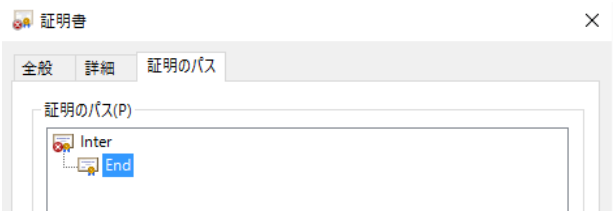

図 4 上位証明書の信頼性を無効にしたときの証明書チェイン

表 1 電子証明書のインポート先

証明書の種類	インポートする証明書ストア	
	信頼	不信頼
ルート証明書	信頼されたルート証明機関	信頼されていない証明書
中間証明書	中間証明書機関	信頼されていない証明書
エンド証明書	信頼された発行元	信頼されていない証明書

Windows での電子証明書は証明書ストア [6] と呼ばれる場所に格納される．証明書が信頼されるものか，信頼されないものかの判別は，各証明書がストア内のどの場所にインポートされているかにより決定される．証明書ストアのうち，このシステムで用いるストアを**表 1** に示す．証明書が信頼されるものであった場合，ルート証明書は「信頼されたルート証明機関」のストア，中間証明書は「中間証明書機関」のストア，エンド証明書は「信頼された発行元」のストアにそれぞれ格納される．反対に証明書が信頼されないものであった場合は，3 種類の証明書はどれも「信頼されていない証明書」のストアに格納される．教育用 PC 上の実行制御プログラムでは，それぞれの電子証明書をポリシー DB から受けた制御情報に基づいて適切な証明書ストアにインポートする．

しかし，従来の証明書の階層構造ではアプリケーショングループを作成する上で，不便な点が存在する．それは，署名を付与したアプリケーションは構造上，1 つのグループにしか属することができないという点である．前述した証明書の階層構造では，下位の証明書は上位の証明書を 1 つしかもつことはできない．そのため，もし，アプリケーションが 2 つのグループに属しており，直接チェインされていないグループを禁止にしたい場合は，グループ自体を禁止した上で，さらにそのアプリケーションを禁止にする必要があり，教員のルール設定の負担が増加する．

3. 複数の証明書チェインを用いた制御

2 章で述べた問題点を解決するために，本論文では証明書の階層構造において複数の証明書チェインを構築する手法を提案する．これにより，1 つのアプリケーションを複数のグループに含ませることができる．

複数の証明書チェインの構築には 2 つの方法がある．1 つはアプリケーションに複数の署名を付与する方法である．この方法について 3.1 節で詳しく述べる．もう 1 つは，階層的構造において証明書を複数の上位証明書にチェインさせる方法である．この方法について 3.2 節で詳しく述べる．

3.1 アプリケーションへの複数署名付与による制御

市販やフリーのソフトウェアに付与されている電子証明書は単一であることが多い．しかし，アプリケーションによっては，複数の電子証明書を用いて署名がされているものも存在する．例えば，Microsoft 社の Internet Explorer には 2 つの証明書を用いて署名が行われている．この 2 つの証明書は異なる署名情報をもつため，別々の証明書だと認識される．そこで，アプリケーションに対して複数の署名を付与することにより，1 つのアプリケーションから複数の上位証明書に対してチェインを構築することができる．これにより，アプリケーションを複数のグループに含ませることが可能になる．

アプリケーションに複数の署名を付与した場合，付与した複数の証明書のうち少なくとも 1 つの証明書の信頼性が保証されれば，アプリケーションの実行が許可される．例えば，あるアプリケーションが 2 つのグループに含まれている場合，片方のグループに対して禁止ルールが設定されていたとしても，もう片方のグループが許可されていればアプリケーションの実行は可能である．両方のグループが禁止になったときに，アプリケーションの実行は禁止される．

ただし，この方法の複数チェインはアプリケーションに対して行うものであるため，階層構造における中間証明書とエンド証明書の間でしか行うことができない．そのため，グループ同士といった中間証明書間での複数チェインの構築には用いることができない．これに対応するために，中間証明書間では 3.2 節で述べる複数の上位証明書による制御方法を用いる．

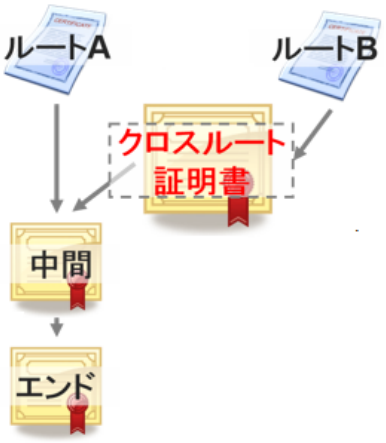

図 5 クロスルート方式

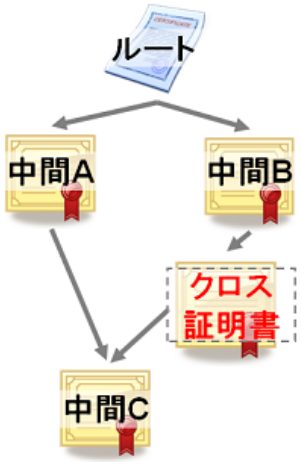

図 6 クロスルート方式の改良

3.2 複数の上位証明書による制御

一般的な証明書の階層構造において，中間証明書やエンド証明書の上位証明書は 1 つだけである．しかし，電子証明書の階層構造には複数のルート証明書を 1 つの中間証明書につなげるクロスルート方式 [7] という構造があり，これは**図 5** の構造で表される．

図 5 のクロスルート証明書はルート A の証明書と同じ秘密鍵と証明書要求ファイルを持ちながら，上位証明書をルート B 証明書とした電子証明書である．中間証明書から見ると，ルート A 証明書とクロスルート証明書の中身は同じなので，どちらの証明書も上位証明書と認識される．ゆえに，ルート A 証明書がない環境でもクロスルート証明書とルート B 証明書があれば，中間証明書はクロスルート証明書により，クロスルート証明書はルート B 証明書により署名が検証される．

このクロスルート方式を改良し，**図 6** で表される構造の証明書チェインを作成する．この図におけるクロス証明書は，中間 A 証明書と同じ秘密鍵と証明書要求ファイルを用いて，中間 B 証明書により署名がされている．

この証明書構造を用いることで，単一の証明書から複数の上位証明書へのチェインを構築し，アプリケーショングループをその上位の複数グループに含ませることができる．図 6 において，中間 A 証明書の信頼性が保証されなくなった場合でも，クロス証明書と中間 B 証明書の信頼性が保証されていれば，中間 C 証明書の信頼性は保証される．そのため，この方法も 3.1 節で述べたアプリケーションに複数の署名を付与する方法と同様に，複数のグループに含まれているアプリケーションやグループは，複数ある上位グループのすべてに禁止ルールが設定されていない限り，その実行は許可される．

4. 動作確認実験

3 章で提案したアプリケーションに複数の署名を付与する方法と複数の上位証明書を用いる方法のそれぞれについて正しく制御を行うことができるか確認するため，これに対応した証明書の階層構造を作成した．これらの証明書をそれぞれルール変更により信頼されない証明書としたとき，署名されたアプリケーションがどのように動作するか確認実験を行った．

[初期状態]

本実験を行うために作成した証明書チェインと証明書を付与したアプリケーションの構成を**図 7** で示す．test.exe に End1 証明書と End2 証明書の 2 つの証明書を付与している．End1 証明書は InterA 証明書により署名されており，InterA 証明書は Root 証明書により署名されている．また，End2 証明書は InterD 証明書により署名されており，InterD 証明書は InterB 証明書により署名されている．さらに，Cross 証明書は InterB 証明書と同じ秘密鍵と証明書要求ファイルで作成され，InterC 証明書で署名されている．最後に，InterB 証明書と InterC 証明書は Root 証明書により署名されている．Windows のグループポリシーの証明書の規則を適用し，信頼された証明書をもつアプリケーションの実行を許可している．

[実験手順]

(1) 実行制御プログラムを起動する．
(2) 設定ツールを用いて InterD グループの禁止制御を行う．
(3) test.exe が実行できることを確認する．
(4) 設定した InterD グループの制御ルールを削除する．
(5) 設定ツールを用いて InterA グループの禁止制御を行う．

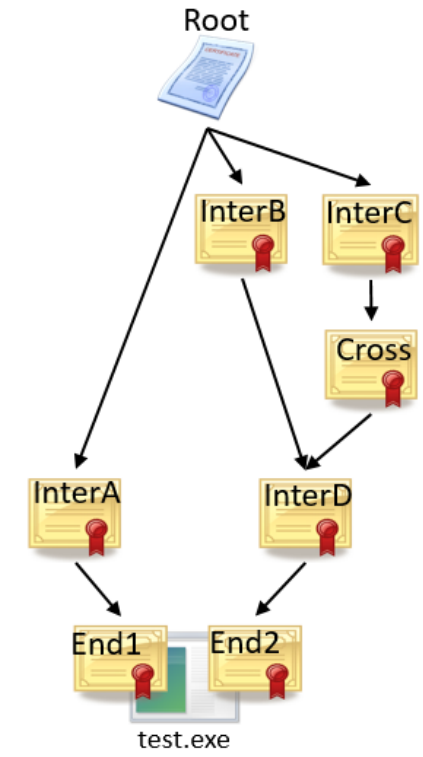

図 7 複数証明書チェインによる制御実験の証明書チェイン図

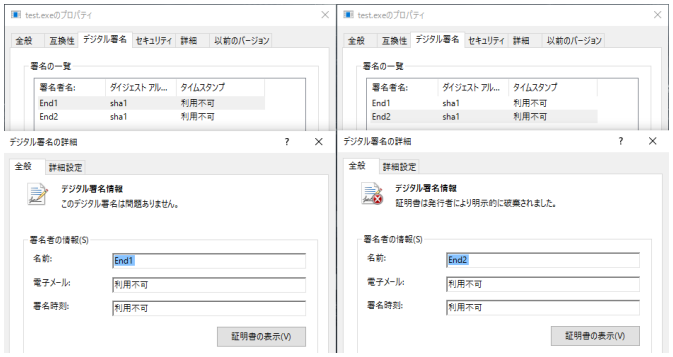

図 8 InterD グループ禁止制御時における test.exe の署名情報

(6) 証明書チェインがどのように行われているか確認し，test.exe が実行できることを確認する．
(7) 設定ツールを用いて InterB グループの禁止制御を行う．
(8) 証明書チェインがどのように行われているか確認し，test.exe が実行できることを確認する．
(9) 設定ツールを用いて InterC グループの禁止制御を行う．
(10) test.exe の実行が禁止されていることを確認する．

[実験結果]

実行制御プログラムを起動し，教員用設定ツールから InterD グループの実行禁止ルールを設定した．禁止ルールを設定した結果，InterD 証明書が信頼され

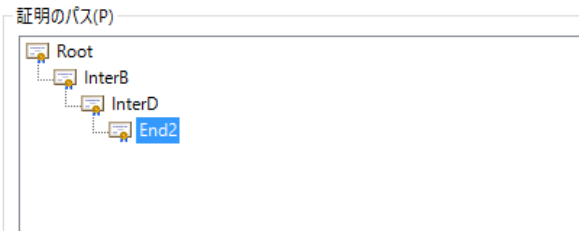

図 9　End2 証明書から Root 証明書へのチェイン

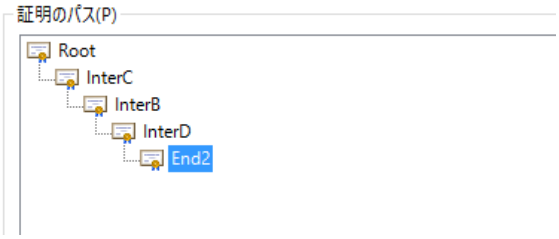

図 10　InterB グループ禁止制御時の End2 証明書から Root 証明書へのチェイン

ていない証明書のストアに格納されたことが確認できた．test.exe の実行が許可されているか確認を行ったところ，問題なく実行することができた．また，このときの test.exe の署名情報を図 8 で示す．図の左側は End1 証明書の詳細であり，その信頼性は保証されていた．対して，右側は End2 証明書の詳細であり，こちらの証明書の信頼性は保証されていなかった．

次に，InterD グループに対して設定していた実行禁止ルールを削除し，InterA グループの実行禁止ルールを設定した．信頼されていない証明書のストアから InterD 証明書は削除され，InterA 証明書がインポートされていた．この場合でも，test.exe が実行できることを確認した．また，test.exe の署名を確認すると，InterD グループ禁止時とは反対に End1 証明書の信頼性は保証されておらず，End2 証明書の信頼性は保証されていた．このときの End2 証明書から Root 証明書へのチェインを図 9 で示す．InterD 証明書は InterB 証明書により署名を検証され，InterB 証明書は Root 証明書により署名を検証されていた．

さらに，InterB グループに対して実行禁止ルールを設定した．すると，信頼されていない証明書のストアに InterB 証明書がインポートされた．test.exe が実行できるか確認したところ，この場合でも test.exe の実行は許可されていた．このときの End2 証明書から Root 証明書へのチェインを図 10 に示す．InterD 証明書は InterB 証明書により署名を検証され，さらに InterB 証明書は InterC 証明書により署名が検証されていた．ただし，この InterB 証明書は InterB 証明書と同じ秘密鍵と証明書要求ファイルで作られた Cross 証明書である．したがって，InterD 証明書は Cross 証明書，InterC 証明書を辿って Root 証明書へとチェインされていた．最後に，InterC グループに対して実行禁止ルールを設定すると，test.exe の実行が禁止されていた．

5. おわりに

従来の電子証明書を用いたアプリケーションの実行制御では，アプリケーションを複数のグループに含めることができないという問題があった．そこで，本論文では証明書の階層構造において複数の証明書チェインを用いる手法を提案した．これにより，1 つのアプリケーションが複数グループに含まれる階層構造を構築することができた．また，複数の証明書チェインをもつ階層構造を実際に作成し，その動作確認を行い，所望どおり動作することを確認できた．

今後の課題として，複数の証明書チェインでの制御方法に改善を加えることが挙げられる．今回の複数証明書チェインでは，複数ある上位証明書のうち 1 つでも信頼されていれば実行が許可された．この場合，実行を禁止するためにはすべての上位証明書の信頼性を無効にする必要がある．これに対し，複数ある上位証明書のすべてが信頼されていなければ実行が許可されない制御方法も考えられる．この場合，上位証明書のうちの 1 つの証明書の信頼性を無効にするだけで実行の禁止ができる．この制御方法を行うための証明書チェイン方法や証明書操作方法を考える必要がある．

参考文献

[1] Keita K., Daisuke O.,Masanori F., and Nariyoshi Y.: *A Flexible Execution Control Method of Application Software for Educational Windows PCs*, Journal of Information Processing, Vol.22, No.2，pp.161-174 (2014).
[2] 岡本大輔，河野圭太，山井成良，横平徳美：教育用 WindowsPC におけるデジタル証明書を用いた柔軟かつ堅牢なアプリケーション実行制御システムの設計，情報処理学会研究報告，Vol.2014-IOT-26，No.3，pp1-8 (2014).
[3] Symantec：Microsoft Authenticode 用コードサイニング証明書 (online)，入手先 ⟨http://www.symantec.com/ja/jp/code-signing/microsoft-authenticode⟩ (2016.04.10).
[4] Microsoft Developer Network：ソフトウェア制限ポリシーの概要 (online)，入手先 ⟨https://msdn.microsoft.com/ja-jp/library/cc759106⟩ (2016.04.10).
[5] Microsoft Developer Network：証明書の規則を作成する (online)，入手先 ⟨https://msdn.microsoft.com/ja-jp/library/cc757067⟩ (2016.04.10).
[6] Microsoft TechNet：証明書ストアを表示する (online)，入手先 ⟨https://technet.microsoft.com/ja-jp/library/cc725751.aspx⟩ (2016.04.10).
[7] Global Sign：[EV SSL] クロスルートとは何ですか (online)，入手先 ⟨https://jp.globalsign.com/support/faq/431.html⟩ (2016.04.10).