エンドポイントでポリシー強制を行うアクセス制御フレームワーク

佐々木　啓†　白石　善明†　福田　洋治‡　毛利　公美††　野口　亮司†††
名古屋工業大学†　愛知教育大学‡　岐阜大学††　(株)豊通シスコム†††

1. はじめに

近年，クラウドというサービス形態が普及し始めている．本稿では「クラウド」は多数のユーザを対象にインターネットを通じてサービスを提供する大規模なサーバ群・データセンタを指し．特に断りがなければ，企業をユーザとするエンタープライズ・クラウドの意味で使う．社内ネットワークの構築における大幅なコスト削減が見込めるなどの理由でクラウドは現在注目を浴びている．

エンタープライズ・クラウドにはデータセンタを運営するクラウドインフラ提供者，クラウド上で，様々な(複合)サービスを提供するクラウドサービス提供者，クラウドサービスを利用するクラウドサービス利用者がアクタとして存在する．クラウドインフラ及びサービス提供者をまとめてクラウド提供者と呼ぶ．

現在，クラウドに対する標準的なアクセス制御規格は定まっておらず，クラウドサービス利用者がクラウドサービス提供者やクラウドインフラ提供者を何らかの基準により信頼してサービスを利用するまでには至っていない[1]．そこで，本稿ではクラウドに対する高信頼なアクセス制御フレームワークを提案する．

2. クラウドでのアクセス制御の要件

2.1 課題

クラウド環境では情報漏えい，ウィルスや不正アクセスによる業務停止などの脅威がある．これらはデータセンタにアクセスする端末がクラウドインフラ提供者の定める基準を満たすか証明できないこと，不正な端末のデータセンタへのアクセスを拒否できないことが原因と考えられるので，この二つを解決することが課題として挙げられる．

一方，高度なサービスを提供するには，ログを残す場所が分散してしまうクラウド環境でも，フォレンジックの観点からログの網羅性を証明できること[2]，高可用性という観点から Web スケールのクラウドサービス利用者が見込まれるクラウドインフラ提供者のアクセス負荷を制御することが課題として挙げられる．

2.2 要件

以上の 4 つの課題に対応させたクラウドでのアクセス制御の要件は次のようになる．

【要件1】　データセンタにアクセスする端末がクラウドインフラ提供者の基準を満たすことを証明できること
【要件2】　基準に満たない端末のデータセンタへのアクセスを禁止できること
【要件3】　網羅性を証明できるようなログが採れること[2]
【要件4】　不正な端末からのクエリを除去するなどクラウドインフラ提供者側の負荷を減らす機構があること

このとき，【要件 1】は次のようなサブ要件に細分化する．

【要件1.1】　データセンタに接続する端末のセキュリティ対策状況が基準を満たすかクラウドインフラ提供側からチェックできること
【要件1.2】　端末にセキュリティ対策状況確認のためのプログラムを常駐させる場合，このプログラムがホストユーザに書き換えられてしまう危険性があるので，プログラムが書き換えられていないことを証明できること
【要件1.3】　端末のセキュリティレベルが変更された場合，即座にアクセス権限が更新されること
【要件1.4】　端末がウィルス感染経路となりうるサイトにアクセスできないようにクラウドインフラ提供者側から制御できること
【要件1.5】　クラウドインフラ提供者の定めるポリシーをアクセス許可基準に満たない端末に提示できること

3. 既存のアクセス制御技術

GAE(Google)や EC2(Amazon)など，既存のクラウド提供者のアクセス制御技術ではゲートウェイで認証・制御するため，【要件 1.4】を満たさない．また，端末のセキュリティ状況を確認されることもないため，【要件 1.1】【要件 1.2】【要件 1.3】を満たさない．

3.1 TNC アーキテクチャ

社内ネットワークにおけるアクセス制御技術は検疫ネットワークと呼ばれ，この実質的な標準規格が TCG（Trusted Computing Group の定める TNC アーキテクチャである[2].

TNC アーキテクチャを構成する主な要素は AR (Access Requester), PEP (Policy Enforcement Point), PDP (Policy Decision Point), 情報資産の入ったサーバ群である．サーバ群と AR(例：ホスト PC）とインターネットの間には PEP（例：スイッチ等のネットワーク機器）があり，許可のない AR のインターネットとサーバ群へのアクセスを防ぐ．正当な AR がサーバ群へアクセスする場合，まず AR 上の IMC（Integrity Measurement Collector）が AR の安全性を PDP（例：認証サーバ）へ送る．PDP は AR を安全と判断すると，PEP にアクセス許可を送る．そして，PEP の設定が変更され，AR はサーバ群へアクセス可能となる．

TNC アーキテクチャは【要件 1.1】【要件 1.2】【要件 1.3】を満たす機構を持つため，これをクラウド環境に適応することで要件が全て満たされるか確認する．

3.2 TNC アーキテクチャのクラウドへの適用

TNC アーキテクチャは組織内ネットワークでの利用を想定して設計されており，クラウドに対するものではない．しかし，2009 年に発表された新仕様 IF-T for TLS により IP ネットワーク上での TNC アーキテクチャを実現している[3].

IF-T for TLS をクラウドに適用すると，データセンタ側から IP ネットワークを介して AR の Integrity Report（AR の安全性情報）の確認が可能となる．このとき，AR に直接接続している PEP（例：スイッチ，ルータ）はデータセンタの管理下にないため，AR のアクセスを制御する点はデータセンタと IP ネットワーク間の PEP となる．

不正なアクセスをはじく場所が AR と IP ネットワーク間にないため【要件 1.4】を満たすことができず，【要件 2】についてもアクセス制御を行う点がデータセンタから遠いのでアクセス制御の安全性が低い．また，ログや負荷分散についての記述がないため【要件 3】【要件 4】を満たさない．そこで，TNC アーキテクチャを拡張し，端末上でアクセスを強制することで【要件 1.4】【要件 3】【要件 4】を満たすフレームワークを提案する．

4. 提案するアクセス制御フレームワーク

4.1 フレームワーク

ホスト PC 上の端末アクセスコントローラがサーバ側からの制御命令に従ってアクセス制御を行う．インベントリ収集プログラムと端末アクセスコントローラがユーザによって書き換えられていないか内部監査モジュールが監査する．その監査ログを受け取

Access control framework with policy enforcement function at endpoint
†Kei Sasaki, Yoshiaki Shiraishi・Nagoya Institute of Technology
‡Youji Fukuta・Aichi University of Education
††Masami Mohri・Gifu University
†††Ryoji Noguchi・Toyotsu Syscom Corporation

った端末認証局は提出者が正当な PC である場合，端末アクセスコントローラに対して接続資格証明書を配布する．これらのサイクルが定期的に行われる．

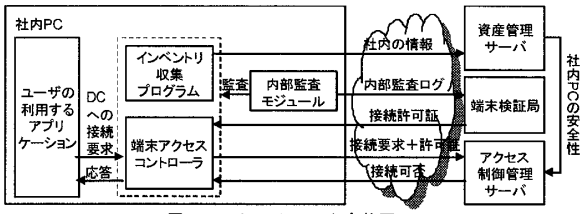

図 1 フレームワーク全体図

ユーザの使用するアプリケーション（ブラウザなど）がデータセンタへの接続を始めたとき，端末アクセスコントローラはこれを遮断し，接続要求に接続資格証明書を添えて，アクセス制御管理サーバへ送る．同サーバは資産管理サーバからホストのインテグリティ情報を取得し，ホストの安全性を確認後，アクセス許可を返信する．許可が下りると，切断していた接続を回復し，ユーザの使用するアプリケーションはデータセンタへ接続可能になる．

アクセス制御ポリシーは，拡張性のあるポリシー記述言語 XACML を使用する．アクセス元（Subject）・アクセス先（Resource）・動作（Action）・アクセス元の環境（Condition）の 4 つの組み合わせに対し，許可又は拒否を規定するルールの集合として表される．

詳しい構成図は次のようになる．

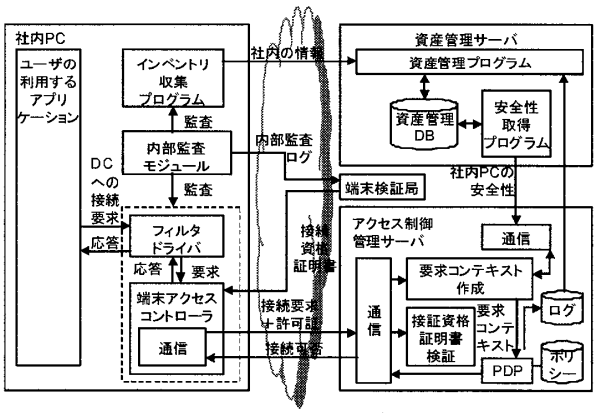

図 2 フレームワーク詳細図

図 2 の要求コンテキストとは Subject, Resource, Action, Obligation の組み合わせであり，PDP はポリシー決定ポイントである．（要求コンテキストからアクセスの可否を判断する）

4.2 各コンポーネント間のプロトコル

端末―アクセス制御管理サーバ間のプロトコルではサーバ認証と通信の暗号化を SSL/TLS を用いて行い，クライアント認証のうち接続資格証明書と TPM 内の秘密鍵により作成された署名の検証を行う．接続資格証明書によりインベントリ収集プログラム・端末アクセスコントローラが改ざんされていないことを確認し，署名の検証により，証明書の提出者と証明書の被発行者との同一性を確認する．クライアント認証は他にインテグリティの確認があり、これが済むとホストへアクセスを許可するサーバリストが送られる．これは TNC アーキテクチャの IF-T for TLS プロトコル [5] を拡張したものである．

インテグリティの確認は情報資産管理サーバ―アクセス制御管理サーバ間のプロトコルで行う．情報量が多く柔軟性が要求されるため XML 形式での通信を用いる．TNC アーキテクチャの IF-MAP プロトコル[5] を拡張し，ホストの安全性を検証するために，アクセス制御管理サーバはホスト上ソフトウェアに関する情報を収集する．収集される情報は表 1 の通りである．

表 1 アクセス制御管理サーバの収集する情報

対象とするソフトウェア	OS，アンチウィルスソフト，アンチスパイウェア，アンチマルウェア，ファイヤーウォール，インストールすべき/すべきでないソフトウェア，VPN
収集する情報	製品情報，バージョン情報（数字，文字列），動作状況，管理するポート（FW,VPN のみ），インストールされたパッケージ，ユーザ情報，MAC アドレス

端末―端末検証局間のプロトコルには，関連研究[6]でのプロトコルを使用する ．

5. 試作システム

本研究のシステムでは，端末でのアクセス制御にはフィルタドライバ（Windows を想定）を使用し，ネットワークデバイスを制御することでアクセスの制御を行う[7].

内部監査モジュールは TPM（Trusted Platform Module）[8]を用いて確実な監査をする．インベントリ収集プログラムはエージェントを用い端末の情報を取得する．IT 資産管理サーバでの資産管理 DB には MySQL を用いる．この DB に動的にアクセスするために Java2 SE5.0 に含まれる JDBC3.0API[9]を用いる．情報資産管理サーバからアクセス制御管理サーバに送られる XML 文書での通信には Java SE6 に含まれる SOAP API[9]を使用する．アクセス制御管理サーバでの XACML で記述されたポリシーの管理は，jBoss の配布している jboss-xacml-2.0.2.GA API[11]を用いる．以上を用いて試作システムを実装している．

6. 評価とまとめ

関連技術との比較を表 1 に示す．表中の記号は次のことを表している．

A：既存のクラウドインフラ提供企業
B：TNC アーキテクチャをクラウドに適用した場合
C：提案フレームワーク

表 2 提案フレームワークの評価

		要件	A	B	C
【要件 1】	1.1	セキュリティ対策の遠隔確認	△	○	○
	1.2	端末の健全性	△	○	○
	1.3	セキュリティ対策の変化への対応	△	○	○
	1.4	データセンタ以外へのアクセスの制御	×	×	○
	1.5	改善方法の提示	○	○	△
【要件 2】		データセンタへのアクセスの制御	○	○	○
【要件 3】		ログの法的証拠能力	×	×	○
【要件 4】		負荷分散	×	×	○

端末アクセスコントローラにより端末とインターネット間のアクセスを制御する点ができたので【要件 1.4】を満たす．また，ホスト上にログを蓄積することで【要件 3】を満たし，不正な端末からのクエリを端末で止めたり帯域制御することで【要件 4】を満たす．

以上，本稿ではエンタープライズ・クラウド向けのアクセス制御フレームワークを提案した．このフレームワークによりフォレンジックや高可用性が要求されるクラウドサービスを展開できるようになる．

参考文献

[1] 丸山，“クラウドの成立過程とその技術的特長について”，情報処理学会誌，vol.50，no.11，pp.1055-1061，2009 年 11 月
[2] 福田，溝渕，毛利，白石，野口，“ネットワークフォレンジックのためのホスト型のロギングについて”，電子情報通信学会総合大会 基礎・境界講演論文集，pp.S-25-S-26，AS-1-3，2009 年 3 月
[3] TCG, TNC Architecture, http://www.trustedcomputinggroup.org/files/resource_files/38BA4157-1D09-3519-AD08262A419DA3B9/Open%20Standards%20for%20Integrity-based%20Network%20Access%20Control.pdf
[4] TCG, TNC IF-T Binding TLS version1.0, http://www.trustedcomputinggroup.org/files/resource_files/51F0757E-1D09-3519-AD63B6FD099658A6/TNC_IFT_TLS_v1_0_r16.pdf
[5] TCG, TNC IF-MAP Binding SOAP Version 1.1 http://www.trustedcomputinggroup.org/files/resource_files/51F74E9B-1D09-3519-AD2DAE1472A3A846/TNC_IFMAP_v1_1_r5.pdf
[6] 脇田，白石，福田，毛利，野口，“サーバサイドネットワークを保護するための TPM を用いた接続資格保証基盤”，情報処理学会第 72 回全国大会，2010 年
[7] 大谷，毛利，白石，福田，野口，“ポリシー強制ポイントをエンドホストで実現するための通信制御機構の提案”，情報処理学会第 72 回全国大会，2010 年
[8] TCG, Trusted Platform Module(TPM), https://www.trustedcomputinggroup.org/groups/tpm/
[9] Sun Microsystems, JDBC 3.0 API, http://java.sun.com/j2se/1.5.0/ja/docs/ja/guide/jdbc/index.html
[10] Sun Microsystems, SOAP API, http://java.sun.com/javase/ja/6/docs/ja/api/javax/xml/soap/package-summary.html
[11] JBoss Community, JBossXACML 2.0.2.GA, http://www.jboss.org/jbosssecurity/download/index.html