L-026

セキュア・プラットフォームの研究開発（３） リソース構成情報管理
Research and development of Secure Platform (3) Resource information management for integrated access control

但野 紅美子† 町田 文雄† 川戸 正裕† 前野 義晴†
Kumiko TADANO Fumio MACHIDA Masahiro KAWATO Yoshiharu MAENO

１． はじめに

サーバ統合環境のセキュリティ管理において、統合的なアクセス制御管理機能の重要性が高まっている。セキュア・プラットフォーム(SPF)の研究開発では、サーバ統合された企業システムをロールベースのアクセス制御ポリシーによって統合管理する統合アクセス制御情報管理機能(IAM)を開発している。アクセス制御ポリシーを一元的に生成し、個々のプラットフォーム個別の形式に適切に変換するためには、アクセス制御の対象となる多数の異なる資源情報の効率的な収集・管理が必要である。本稿では、多数の異なる資源情報を一元的に提供するリソース構成情報管理機能について述べる。多様な資源情報を統一的に管理するため、資源情報モデルの標準に準拠した形で資源情報モデルを拡張し、また Web サービスの標準に準拠する。Web サービス標準への準拠は相互運用性を確保する反面、SOAP/HTTP による通信処理や XML 形式のデータの解析処理を含むため処理コストが高い。本稿では、リソース構成情報管理機能の性能を向上させるために、軽量なエージェント実装、および、資源情報キャッシュとキャッシュされた資源情報を選択的に更新する機能を導入し、基礎的な性能評価を行う。

2. リソース構成情報管理機能

2.1 資源情報モデル

SPF では、異なるレイヤの資源（仮想化機能、OS、DB 等）に対してアクセス制御を行うために、複数のリファレンスモニタ（アクセス制御モジュール）を用いる。各リファレンスモニタは、それぞれ固有のアクセス制御の粒度情報を持つ。各リファレンスモニタにアクセス制御ポリシーを適用するためには、固有のアクセス制御の粒度情報としてアクセス制御の主体（ロール等）、アクセス制御対象の資源（ファイル、VM 等）、アクセス制御するアクション(read、shutdown 等)の情報が必要となる。異なるリファレンスモニタによるアクセス制御に必要となる異種資源の情報を共通の形式で統合管理するためには、資源情報モデルの標準が必要である。

複数のプラットフォームで利用可能な資源情報モデルの標準として、Common Information Model(CIM)[2]が広く利用されている。

SPF では、統合的なアクセス制御情報の管理を実現するために、CIM 情報モデルを拡張してリファレンスモニタとアクセス制御の粒度情報を表すモデルを定義した(図１)[8]。リファレンスモニタは、CIM で定義されているソフトウェアモジュールを表す CIM_SoftwareElement クラスを拡張して、SPF_ReferenceMonitor クラスとして定義する。

† 日本電気株式会社 サービスプラットフォーム研究所
Service Platforms Research Laboratories, NEC Corporation

一方、リファレンスモニタ固有のアクセス制御の粒度情報は、設定情報を表す CIM_SettingData クラスを拡張して、SPF_RMTargetSettindData クラスとして定義する。SPF_RMTargetSettindData のプロパティには、アクセス制御の主体の粒度情報を格納する PrincipalType、アクセス制御対象の粒度情報を格納する ResourceType を定義する。例えば ResourceType に格納されたアクセス制御の対象資源がファイルだった場合は、ファイルシステムに付随するアクション(read,write,execute 等)とアクセス制御対象であるファイル・ディレクトリの情報を参照することでアクセス制御ポリシーの記述に必要な情報を集める。一方、アクセス制御対象が VM だった場合は、VM に対するアクション(start,shutdown 等)とアクセス制御対象である VM の情報を収集する。以上のように異なるリファレンスモニタに対応する為に抽象化されたリファレンスモニタのクラスを定義し、SPF_RMTargetSettingData に個々のリファレンスモニタが扱うアクセス制御の粒度情報を格納することで、異なるリファレンスモニタに対して統一的な手順・形式でアクセス制御に必要な情報を提供可能となる。

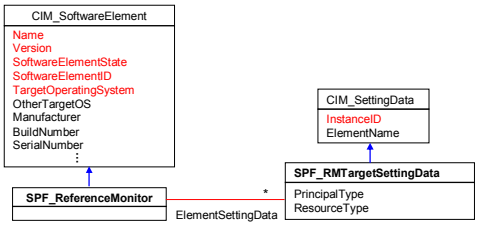


図１ 拡張した CIM モデル

2.2. 設計と実装

2.1 節で述べたアクセス制御に必要な資源情報を収集するために、リソース構成情報管理機能を実装した。図２に示すように、リソース構成情報管理機能は、検索処理機能、資源情報収集機能、資源情報キャッシュ、資源情報更新管理機能の四つのコンポーネントから構成される。検索処理機能は、ポリシー管理機能から資源情報の検索要求を受信し、要求された資源情報を返却する。CIM 情報の検索処理言語として、CQL[3]を利用する。検索処理機能は有効な資源情報がキャッシュに格納されていればそれを返却し(キャッシュヒット)、格納されていなければ資源情報収集機能を用いて収集した資源情報を返却する(キャッシュミス)。資源情報収集機能は、WS-Management[1]プロトコルを用いて、リモートの管理対象ホスト上のエージェントから資源情報を収集する。エージェントのプロトコルハンドラは WS-Management 形式のリクエストを資源情報収集機能から受信し、CIM 情報管理機能である CIMOM(CIM Object Manager)を介して要求された資源情報を返却する。CIMOM は CMPI 標準[4]に準拠

したプロバイダを用いて取得した資源情報を提供する機能である。プロバイダは、仮想化機能および OS に標準的に用意されているコマンド(virsh list、ls 等)や、2.1 で述べた統合アクセス制御に必要な資源情報を収集するために用意されたアダプタ(SPF 独自コマンド)を利用して、資源情報を取得する。資源情報更新管理機能は、定期的にキャッシュ内の資源情報から有効期限が近く検索される確率が高いものを更新対象として抽出し、管理対象ホストから更新対象の資源情報を収集して、キャッシュ内の資源情報を更新する。

次に、リソース構成情報管理機能の実装について述べる。資源情報キャッシュは、MySQL[7]を用いて実装され、資源情報とその有効期限を格納する。資源情報収集機能では、WS-Management の実装として Openwsman[5]を用いた。エージェントは、軽量な CIMOM である Small-FootPrint CIM Broker[6]と Openwsman を用いて実装した。

リソース構成情報管理機能の効果について述べる。リソース構成情報管理機能では、資源情報を一時的にキャッシュすることにより、検索応答時間を削減する。また、軽量なエージェントの実装により、キャッシュミス時の情報取得に要する時間を削減する。管理対象資源が多数存在するシステムにおいても、選択的に資源情報キャッシュを更新することで管理ホストの負荷を抑えながらキャッシュヒット率を向上できる。一方、相互運用性の観点では、WS-Management に準拠したことにより、サードパーティ製の運用管理ソフトウェアとの相互運用性を確保した。また、CQL に対応したことで資源情報の検索処理を統一的な手順で行うことができる。CMPI 準拠により、サードパーティ製プロバイダとの相互接続を容易化した。

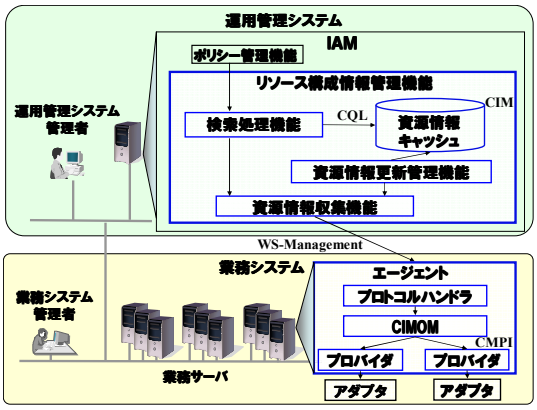
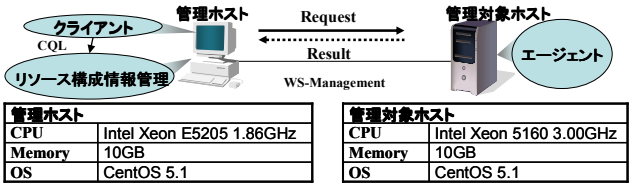


図２　リソース構成情報管理の構成

3. 性能評価

リソース構成情報管理機能の性能評価のための実験環境を、図３に示す。リソース構成情報管理機能とクライアントは管理ホストにインストールされる。本実験では、管理ホスト内のクライアントが送信する CQL の検索応答時間を測定する。各 CQL は、それぞれ特定の CIM クラスの１つのインスタンスを要求する。有効な資源情報がキャッシュ内に格納されていれば、リソース構成情報管理機能はキャッシュ内の資源情報をクライアントに返却する(キャッシュヒット)。そうでなければ、リソース構成情報管理機能は管理対象ホスト上のエージェントから資源情報を取得する（キャッシュミス）。

各 CQL について検索応答時間を 10 回ずつ測定した際の、平均検索応答時間の結果を表１に示す。全 CQL の平均検索応答時間は、キャッシュミス時 1.0686 秒、キャッシュヒット時 0.0046 秒であった。



管理ホスト	
CPU	Intel Xeon E5205 1.86GHz
Memory	10GB
OS	CentOS 5.1

管理対象ホスト	
CPU	Intel Xeon 5160 3.00GHz
Memory	10GB
OS	CentOS 5.1

図３　実験システム構成

表１　実験結果

検索対象 CIM クラス	CQL	応答時間 [秒]	
		キャッシュミス	キャッシュヒット
CIM_ComputerSystem	SELECT * FROM CIM_ComputerSystem WHERE Name='hostA'	3.0691	0.0043
CIM_FileSystem	SELECT * FROM CIM_FileSystem WHERE Name='/'	1.0686	0.0042
SPF_Directory	SELECT * FROM SPF_Directory WHERE Name='/etc/'	0.0705	0.0046
CIM_LogicalFile	SELECT * FROM CIM_LogicalFile WHERE Name='/etc/yum.conf'	0.0411	0.0053
CIM_EnabledLogicalElementCapabilities	SELECT * FROM CIM_EnabledLogicalElementCapabilities	1.0939	0.0044
平均		1.0686	0.0046

4. おわりに

本稿では、WS-Management に準拠し、かつ、CIM に準拠した形式で資源情報モデルを拡張することで、相互運用性が高くアクセス制御の管理に必要な異種資源情報を統一的に管理可能な資源情報管理機能を実現した。また、軽量なエージェント実装、および、資源情報キャッシュとキャッシュされた資源情報を選択的に更新する機能を導入することで、検索応答時間を削減する機能を実装し、その基礎的な性能評価を行った。

謝辞

本研究は、経済産業省から技術研究組合 超先端電子技術開発機構(ASET)へ委託されている「平成 19 年度セキュア・プラットフォームプロジェクト」の成果である。

参考文献

[1] WS-Management http://www.dmtf.org/standards/wsman
[2] Common Information Model (CIM) Standards, http://www.dmtf.org/standards/cim/
[3] CIM Query Language Specification (DSP0202), http://www.dmtf.org/standards/published_documents/DSP0202.pdf
[4] Common Manageability Programming Interface (CMPI), http://www.opengroup.org/pubs/catalog/c051.htm
[5] Openwsman, http://www.openwsman.org/
[6] SBLIM, http://sblim.wiki.sourceforge.net/
[7] MySQL, http://www.mysql.com/
[8] F. Machida, et al., "CIM-based Resource Information Management for Integrated Access Control Manager", SVM'08