脆弱性対策のためのセキュリティ保護システムにおける IT産管理機構の拡張

20G475 西岡大助（最所研究室）

1. はじめに

近年，脆弱性を利用したサイバー攻撃による被害が深刻になっている．一方で，リモートワークの普及に伴い，BYOD と呼ばれる個人の機器を組織に持ち込み業務利用する形態を導入する組織が増えている．BYOD が脆弱性を持つ場合，その BYOD だけでなく組織内ネットワークで繋がっている機器全てが攻撃される可能性がある．

これらの背景から，我々の研究室では脆弱性対策を行うためのセキュリティ保護システム”BEYOND(Bring Enhancement Your Own New-vulnerable Device)”の開発を行なっている．先行研究では BEYOND の機構の一つとして，脆弱性対策を行うために，エージェントを用いて組織の IT 資産を収集・DB 化し，一元管理する IT 資産管理機構が開発されてきた．しかし，先行研究の機構には以下に示す課題があった．

課題①：これまで IT 資産管理 DB で管理してきた情報では，管理している機器がクライアントかサーバか判断できず，サーバである場合にアクセスを遮断しても良いか判断できなかった．また，BEYOND において脆弱性が検出された機器についての情報を保持できていなかった．

課題②：エージェントで収集できない情報を，利用者が登録するための手段がなかった．また，登録されている情報の管理が IT 資産管理機構のシステム管理者しかできなかった．

課題③：ラップトップ PC では Windows ユーザが大半を占めていることから，Linux のみの対応では不十分であった．

本研究ではこれら①～③の課題を解決することを目的として IT 資産管理機構の拡張を行う．本稿では，IT 資産管理 DB を拡張することによる管理情報の追加と，利用者に管理情報操作を提供するための WebAPI，Windows のソフトウェア情報収集のためのエージェントの拡張について述べる．

2. BEYOND

BEYOND のシステム構成を図 1 に示す．BEYOND は，本研究の対象である IT 資産管理部の他に脆弱性情報収集部，影響算出部，ネットワーク制御部から構成されている．また，利用者が情報操作を行うための窓口となる Web インタフェースを実装しているフロント部も存在する．

BEYOND で行う処理の流れについて述べる．まず，脆弱性情報収集部でインターネット上に公開されている脆弱性情報を収集し，DB 化する．同時に IT 資産管理部にて組織内ネットワークを利用する機器の情報を収集し，DB 化する．それらの収集した脆弱性情報と IT 資産情報から影響算出部にて組織内の脆弱性を診断し，脆弱性が検出された場合，対策方針を算出する．算出された対策方針や検出された脆弱性に関する情報は機器の所有者やシステム管理者に通知され，通知を受けた者はその内容に応じた対策を取る．攻撃のリスクが高い脆弱性の場合は所有者の対応を待つ間に攻撃を受けることを考慮し，ネットワーク制御部にてアクセス制御を行う．

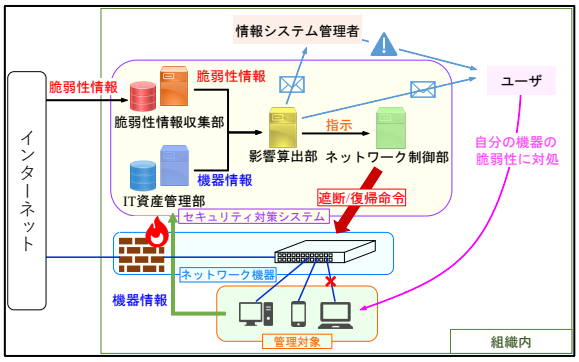

図 1. BEYOND のシステム構成

表 1. 追加で管理する情報

サービス情報	脆弱性の状況情報
コンテンツ内容，種別，提供範囲，停止可能期間，IP アドレス，ポート番号	脆弱性 ID，対応状況 アクセス制御状況

3. IT 資産管理機構の拡張

1 章の課題①～③を解決するために IT 資産管理機構の機能を拡張する．

3.1. 課題①の解決:IT 資産管理 DB の拡張

先行研究の DB では機器のネットワーク情報や OS，重要度をユーザ情報と紐付けて管理していた．他にも機器にインストールされているソフトウェアの情報を登録済みの機器と紐付けて管理していた．それらの情報を用いて BEYOND では脆弱性の検出や機器のアクセス制御を行うことが出来た．しかしながら先行研究で管理していた情報では，サービスを提供している機器を停止させることによる外部への影響がわからないことが課題となっていた．また，BEYOND において脆弱性を検出した際の制御方針や脆弱性への対応状況を管理できていなかった．

そこで，当機構の DB を拡張することで，サーバが提供しているサービスや脆弱性が見つかった機器の状況を管理できるようにした．追加で管理する情報を表 1 に示す．サービス情報を用いることで，機器の利用実態に応じた柔軟な対応が可能になる．脆弱性の状況情報から利用者に自身の機器の脆弱性情報を提供することや，継続したアクセス制御を行うことができる．

3.2. 課題②の解決:WebAPI の開発

これまで利用者に対して IT 資産管理 DB の情報登録，取得，更新，削除の操作(DB 操作)を提供する際には，BEYOND のフロント部と連携し，IT 資産管理 DB の内容をやり取りすることで実現していた．しかし，利用者ごとに適切なデータへのアクセス権限を

割り振る必要があることや，DB 操作によっては複雑な SQL を要求されることが課題となっていた．そこで，本研究では上記の問題を解決するために，フロント部に向けた WebAPI を開発した．

開発した API では，利用者に対して認証・認可の機能と DB 操作の機能を提供する．認証・認可には JWT(JSON Web Token)と呼ばれる電子署名や暗号化の技術を用いて，認証を行い，利用者ごとにアクセス権を適用することで実現した．そして，情報の登録と削除の際に，DB の更新作業が複雑になる課題は，複数の処理を一つの URI として API 化することで解決した．例えば，機器の中には複数NIC を搭載しているものもあり，MAC アドレスや IP アドレスを複数持っている可能性がある．その場合，機器と NIC が一対多となるため，DB を正規化すると複数回の登録操作が必要となってしまう．また，機器情報を削除した場合，その機器の NIC 情報，提供しているサービス情報，インストール情報，その機器が持っている脆弱性情報など多岐に渡って削除操作を行う必要がある．このように複数の DB 操作を一つの URI にまとめることで IT 資産管理部と連携する他の機構の負担を軽減できる．開発した API のうち，一般利用者に向けた API を表 2 に示す．ソフトウェア情報の登録，更新はエージェントでのみ行い，手動で操作することは想定していない．また，ユーザとソフトウェアの削除は一般利用者が行う必要がないと考え，管理者専用 API として実装している．

3.3. 課題③の解決:Windows への対応

先行研究で開発された，ソフトウェア情報を収集するためのエージェントは Linux に対応していた．しかしながら，ラップトップ PC は Windows が多く，Linux のみの対応では不十分であったため，本研究では Windows に対応するように機能拡張した．

表 2. 利用者向け API

API	概要
SignIn	サインイン
SignUp	ユーザ作成
GetUser	ユーザ情報取得
GetHard	機器情報取得
GetSoft	ソフトウェア情報取得
GetService	サービス情報取得
GetVulnerability	機器の脆弱性情報取得
RegHard	機器情報登録
RegSoft	エージェントによる情報登録
RegService	サービス情報登録
UpdateUser	ユーザ情報更新
UpdateHard	機器情報更新
UpdateService	サービス情報更新
DeleteHard	機器情報削除
DeleteService	サービス情報削除

Windows には機器のシステム情報やソフトウェア情報などを格納するためのレジストリと呼ばれるデータベースがある．この中から，インストールされているソフトウェアの情報を“reg”コマンドを用いて取得する．取得したデータ内には余分なデータも含まれており，必要なデータだけを抽出する必要がある．本研究では，“DisplayName”で表される，ソフトウェア名と，“DisplayVersion”で表されるバージョン情報を抽出し，JSON 形式に加工して IT 資産管理サーバに送信する．

4. 評価

開発した API と拡張したエージェントの機能評価について述べる．API を用いて機器を登録した後の DB の内容を図 2 に示す．HTTP リクエストは curl を用いて作成し，送信した．ハードウェアテーブルに登録した機器が追加されていることが確認できる．複数の NIC を持つ機器を登録したため，一つの機器情報に対して，複数のネットワーク情報が登録されており，それぞれの情報が NIC テーブルにて機器 ID と紐付いている．

エージェントを用いて Windows のソフトウェア情報を収集した結果を図 3 に示す．取得したソフトウェア情報が正しく整形されて DB 内に格納されていることが確認できた．

拡張したエージェントの性能評価について述べる．エージェント実行時の CPU 使用率は 0.2%であった．CPU のプロファイル結果からソフトウェア情報取得のための外部コマンド実行が CPU 利用のほぼ全てを占めていることから，CPU 使用率はソフトウェア数に比例しているといえる．ソフトウェア数が多い機器で，実験環境の 10 倍ほどを想定しているが，その場合でも 2.0%の CPU 使用率に収まっていることから，他のシステムへの影響は小さいと考える．また，メモリ使用量についても，ソフトウェア数に比例しており，計測値は 1,536kB であった．この 10 倍で考えても 15MB ほどであり，現代のスペックの PC には影響がないと考える．情報取得時のネットワーク帯域使用量は 21.3kB であった．研究室のネットワーク帯域を計測したところ，524Mbps であり，このネットワークを圧迫することはないと考えられる．

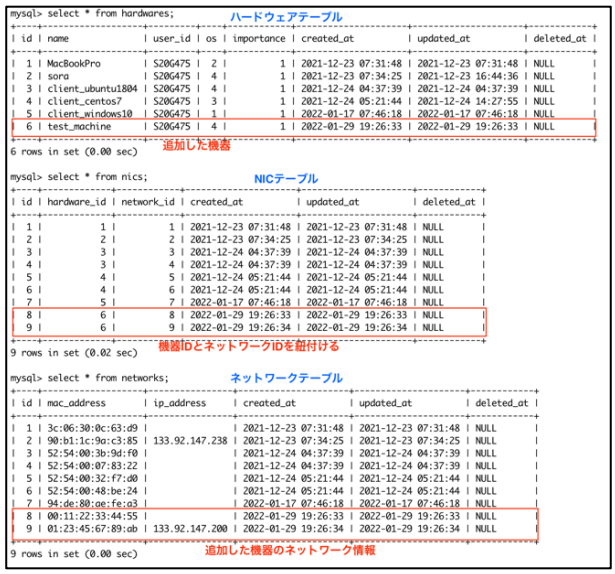

図 2. 開発した API を用いた機器情報登録結果