脆弱性情報データベースシステム

Vulnerability Database System

横地　裕[*1]　　大野　浩之[*2]
YOKOCHI Yutaka　　OHNO Hiroyuki

インターネットのセキュリティに対する関心はここ数年の間に大きく高まり，インターネット上の脆弱性に関する技術的な対策の情報や，さまざまなセキュリティ関連製品/サービスが数多く出てきている。

脆弱性情報データベースはセキュリティ情報のうち大きな要素であり，対策を考える上では，網羅的であると共に情報の詳細さも必要である。同時に情報の急速な増加に対処できることも重要である。また新規の攻撃が発生した場合，既存の脆弱性データに類似の攻撃の有無を迅速に検索する必要があり，これに耐える検索性能が必要となる。そして実際の攻撃の際，あるいはそれに備える際に，これらの情報がどこにでも持ち運べることが組織的，組織間にまたがる防御に役立つ。本稿では，このような背景を踏まえて構築した，新しい脆弱性情報データベースシステムを紹介する。

Internet security has been highlighted recently. The information of technical countermeasures against the vulnerability on the Internet and various securities associated products as well as services have been emerging.

The Vulnerability Database is a core element among security information. Considered countermeasures, it is required that the information should be detailed as well as comprised, while following up the rapid information growth. Moreover, since quick reference whether there has been any similarity attack in the existing Vulnerability data should be performed when a new attack occurs, the reference performance that fills this is required. Also, in the case of an actual attack or when preparing for it, it is useful for the organizational and inter-organizational defense that these information can be available anywhere.

This paper introduces new Vulnerability Data System that built up on these backgrounds.

1. はじめに

脆弱性とは，情報や資産やサービスに損害を与えようとする脅威に対するセキュリティ上の弱点のことである。脆弱性情報データベースシステムは，インターネット上の脆弱性を整理格納して利用するためのシステムである。**図1**にその一例を示す。

インターネットにおける脆弱性に関しては，常に最新の情報に基づいて対策を考える必要がある。そのためには，情報収集およびその整理が重要になってくる。

CERT/CC(The CERT® Coordination Center)は，米国連邦予算で運営されているインターネットセキュリティの研究開発機関である。CERT/CCは収集した情報を整理して勧告を発し，セキュリティ対策を行う人の便宜を図る活動を行っている。但し，この組織の情報だけ

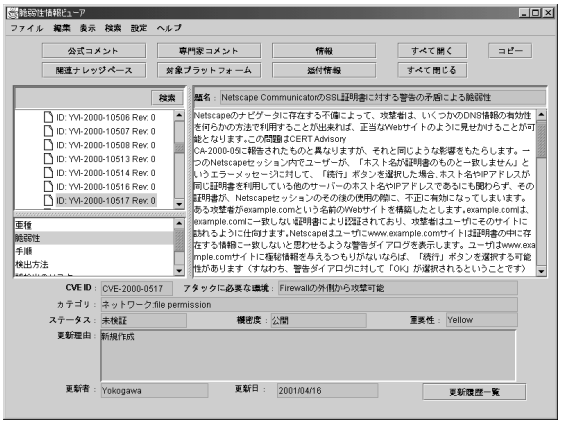

図1　脆弱性情報データベースシステム

*1 IT事業部　NSビジネスセンター
*2 独立行政法人通信総合研究所　非常時通信研究室

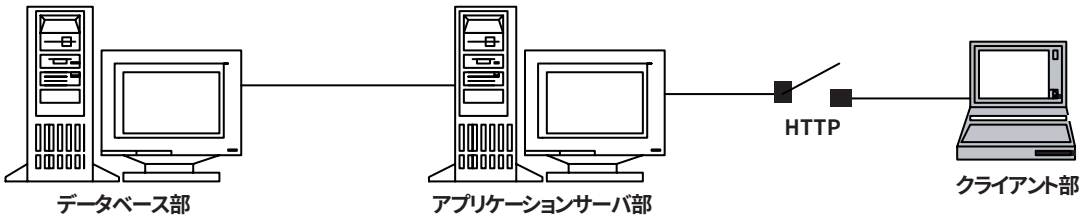

図 2　典型的なハードウェア構成

で全てのセキュリティ対策をカバーできる訳ではない。一方他のいくつかの組織もまた，情報を収集して公開しており，これらもセキュリティ対策に役に立っている。しかし，これら各組織の間では情報の互換性が乏しく，セキュリティ対策を考える上で障害になっている。

CVE(Common Vulnerabilities and Exposures) は，セキュリティ関連組織によるコラボレーション組織である。CVEにより脆弱性情報のデータの相互関連性を維持管理していく活動が始まっているが，あくまで脆弱性情報のレコードの関連性を図っているだけである。インターネット上の脆弱性に関連する情報は膨大であり，組織間で協力しなければ，今後対応が難しくなっていくと予想される。

また，これら互換性や協力の問題とは別に，利用する脆弱性情報自体にも問題がある。一般に公開されている情報は，セキュリティの性格上敢えて重要な部分を隠していることが多く，対策を考える上では不十分である。また日本語で記述されたものや，日本版のソフトウェアに関する情報は少なく，緊急な対応が必要な場合には問題である。

以上のような現状の問題点を踏まえて，脆弱性情報を維持管理し，各組織での対策のために必要な情報を提供でき，組織間で協力することも可能なデータベースシステムを開発した。

2.　本システムの特長

脆弱性情報データベースシステムは，作成／利用／更新／破棄といったライフサイクルに亘って脆弱性情報を管理する。

脆弱性情報を検索した結果は，XML(Extensible Markup Language 1.0)で出力される。この情報をファイルとして保存することで，脆弱性情報をどこにでも持ち運ぶことができる。このため，脆弱性情報データベースサーバと切り離されたような環境(オフライン)でも，情報を閲覧することができる。また，オフラインで編集した脆弱性情報を後で脆弱性情報データベースサーバに登録することもできる。この仕組みは，そのまま組織間での脆弱性情報のやり取り，そして組織間協力に役立つ。

一方，格納する脆弱性情報は日本語化しつつ，日本版のソフトウェアの情報も含む。また，脆弱性のソースコードと攻撃のソースコードが含まれているため，攻撃の兆候，足跡を調査して対策の確立に役立つ。

3.　システムの構成

システムは，脆弱性情報データベースサーバ側がデータベース部とアプリケーションサーバ部から成り，クライアント部はオフラインでも脆弱性情報を扱うことができるように構成した。

3.1 ハードウェア構成

図 2 に典型的なハードウェア構成を示す。

データベース部とアプリケーションサーバ部を物理的に分離しても，１台のサーバにまとめても良く，さらにアプリケーションサーバについては複数配置しても良い。これにより，障害隔離と負荷分散が可能となる。

3.2 処理とデータのフロー

3.2.1　データベース部

データベース部は，運用方法が確立されていて実績も多いRDB(Relational Database)を利用して，データの完全性と可用性を確保している。また脆弱性情報の高速な検索が可能になるように，各種インデックスを利用している。この中には，あいまい検索を可能にする全文検索用インデックスも含まれる。

脆弱性情報は，新規作成時には情報が非常に少なく，誤報の場合もある。また脆弱性が発見されてから，

・脆弱性の影響などが調査された段階
・一時回避策が確立された段階
・パッチのような恒久的な対応策が出た段階

などの各段階で情報が更新される。一方，誤報であると判った場合に脆弱性情報を破棄してしまうか，恒久的な対応策が出た場合に一時回避策を削除してしまうか，といったことについては，脆弱性情報の利用方法を考えて決定する必要がある。

脆弱性情報データベースシステムは，過去の事例を元に，既知のまたは新たな脆弱性への対策を確立するためのシステムである。これは一種のナレッジベースであると考えられる。例えば，誤報と判った情報にも次回の誤報に備えるといった利用価値がある。またパッチが出て不要になったと考えがちな一時回避策にも，パッチに副作用や条件があった場合に利用価値がある。

そこで，脆弱性情報データベースシステムでは，情報にバージョンを付与して永久保存している。情報を破棄する場合でも，物理的にはレコードに“破棄”マーキングをするだけで，削除はしない。この方法では容量が一方的に増えてゆくことになるが，脆弱性情報の情報量の増加が激しいとはいえ，製造業の操業データのような急激な増加は考えられない。将来，容量が厳しくなった頃に，機器を増強してデータを移行することは可能である。データ移行の際にも，情報がXML形式で出力できることは役に立つ。

3.2.2 アプリケーションサーバ部

アプリケーションサーバ部の主な役割は，データベース部への情報の入出力である。データベース部にある情報は

・全件出力
・項目キーワード検索
・全文あいまい検索
・全文正規表現検索

といった方式で取得可能であり，出力フォーマットはXML形式になっている。更新に関しては，クライアント部であらかじめ編集したXML形式の脆弱性情報をHTTP（Hyper Text Transfer Protocol 1.0）で受け取り，データベース部に格納する。

脆弱性情報は攻撃コードまで含むため，使い方を誤ると危険である。アプリケーションサーバ部では，認証とアクセスコントロールを行っており，権限のない人間が情報にアクセスできないようになっている。

アプリケーションサーバ部から出力した脆弱性情報や，クライアント部で編集してこれからアプリケーションサーバ部に入力しようとしている脆弱性情報は，XMLファイルとしてクライアント部に存在する。クライアント部にはアプリケーションサーバのアクセスコントロールが及ばない。そのため，クライアント部の取り扱いによっては，危険な情報が漏洩する可能性もある。このような危険を低減するために，アプリケーションサーバ部では脆弱性情報の暗号化をサポートしている。

3.2.3 クライアント部

クライアント部は脆弱性情報データベースサーバ側とオンラインで接続されていても，オフラインであっても良い。オンラインで接続されている場合は，アプリケーションサーバ部の機能で脆弱性情報の各種検索を行うことができる。

脆弱性情報は，サイトが正に今攻撃を受けているような状況で利用するということも考えられる。また緊急サポートのために，全件出力した脆弱性情報とクライアント部を，モバイル機器にインストールして持ち運んで利用することも考えられる。このような場合に備え，クライアント部単独で情報を閲覧できるだけでなく，検索絞り込みもできるようになっている。

脆弱性情報の編集もオフラインで行うことができ，攻撃の行われている現場で編集した情報を，後から脆弱性情報データベースサーバ側に入力することができる。

他の組織が作成した脆弱性情報も同様で，クライアント部で利用することも，脆弱性情報データベースサーバ側に入力することも可能である。

アプリケーションサーバ部やクライアント部におけるＸＭＬ形式の脆弱性情報へのアクセスにはＤＯＭ（Document Object Model Level 1）を利用している。またクライアント部における検索機能では，XSL Processorを利用しているため，XPath（XML Path Language Version 1.0）式を検索式として指定することにより，条件に合う脆弱性情報を抽出することができる。

4. おわりに

他の組織と組織間で協力するための仕組みとして，今回XMLを利用するという方式を採用した。これで情報を相互に交換するための準備は整ったが，現状では人間の手を介する必要があるなど，不十分な点もある。

・脆弱性情報を相互に交換するための手続き
・XMLの型定義が異なる場合に，自動的に変換するための仕組み
・型定義の中に現れる用語が同じで，意味が違うような場合の調整方法
・脆弱性の情報を別々に収集してしている組織同士で，情報を共有するときに生ずる冗長性を排除するためのノウハウ

などの研究を行い，脆弱性情報データベースシステムをより運用面で発展させてゆく必要がある。

脆弱性情報という危険な情報を取り扱うための仕組みとして，今回XML文書全体の暗号化という方式を採用した。これで情報の機密性を保つことは可能になったが，

・暗号化アルゴリズムが固定である。
・暗号化してしまうと，もはやXMLではなくなってしまう。

といった運用上の課題は残っている。

現在W3C（World Wide Web Consortium）で“XML Encryption”を制定中である。これが利用可能になれば，より運用性も高く機密性を保つことができる。

一方，これもW3Cで“ XML-Signature ”が制定中である。これが利用可能になれば，脆弱性情報の要素毎に署名を付けることができる。組織間で情報を持ち寄って協力をする場合，情報の部分部分に署名して文責を明らかにできる方が良い。

W3Cで既に勧告になったXML Schemaの型定義における表現力は強力である。今後XML Schema対応のDOMが利用可能になれば，これを利用することにより，さらに脆弱性情報の一貫性を高めることができる。

参 考 文 献

(1) CERT/CC Advisories
http://www.cert.org/advisories/
(2) CVE(Common Vulnerabilities and Exposures)
http://cve.mitre.org/
(3) CVE-Compatible
http://cve.mitre.org/compatible/
(4) 大野浩之, 武智洋, 永島秀行, “ インターネットの脅威に対抗しうる脆弱性データベースと検証システムの構築 ”, (社)情報処理学会DSM研究会主催DSMシンポジウム'2001