IPA Technical Watch

2011年11月17日
独立行政法人情報処理推進機構

IPA テクニカルウォッチ：『ソースコードセキュリティ検査』に関するレポート

～出荷前のソフトウェアの脆弱（ぜいじゃく）性を低減するために～

IPA（独立行政法人情報処理推進機構、理事長：藤江 一正）は、安心・安全なIT社会の実現を目指し、システムライフサイクル(*1)に沿ったセキュリティ上の弱点（脆弱性）への対策を推進しています。これらセキュリティ対策のうち、「ソースコードセキュリティ検査(*2)」の有効性と重要性についての理解を深め、実際の開発現場で活用されることを目的に、具体的に有効な場面や、実施・成功事例を紹介する技術レポート（IPA テクニカルウォッチ 第5回）を公開しました。

出荷されたソフトウェアやシステムに脆弱性が発見された場合、システム設計の見直しが必要になる場合があり、対策にかかるコストが予想外に増大する可能性があります。また、脆弱性を悪用された攻撃が原因で利用者に被害が生じた場合、ソフトウェアを開発した企業にも追加の費用負担や社会的責任が生じてきます。

ソフトウェアやシステムを保護するためには、システムライフサイクルに沿って、トータルなセキュリティ対策を行うことが必要です。特に、脆弱性の作り込みを防止することと併せて、作り込んでしまった脆弱性を検出することは、ソフトウェアを出荷する前に脆弱性を低減するための施策として、根本的に重要です。これらの手法には具体的に、「セキュアプログラミング(*3)」、「ソースコードセキュリティ検査」、「脆弱性診断(*4)」などがあります。

これらの中で、ソースコード中に存在する脆弱性を網羅的に検出することができる「ソースコードセキュリティ検査」は特に有効です。しかし、IPAが独自に行ったアンケート結果では、開発現場での実施率は、「脆弱性診断」の約54%に対し、「ソースコードセキュリティ検査」は約16%と、まだまだ実施されていない状況です。この理由として、「ソースコードセキュリティ検査」の有効性や重要性が認識されていないことが考えられます。

本レポートでは、システムライフサイクルに沿ったセキュリティ対策のうち、「ソースコードセキュリティ検査」技術について説明し、加えて、具体的に有効な場面や、「ソースコードセキュリティ検査」の実施・成功事例を紹介することで、「ソースコードセキュリティ検査」の有効性と重要性を説明します。

(*1) 情報システムの企画から廃棄までの一連の工程。IPA では［1.要件定義］［2.設計］［3.実装］［4.テスト］［5.運用／利用］（［6.廃棄］）と分類しています。

(*2) ソフトウェアの設計図であるソースコードを機械的にチェックし、ソースコードに含まれる特定のパターンを抽出することで脆弱性を自動的に検出する手法。システムライフサイクルの［3.実装］工程において実施します。

(*3) ソフトウェアに脆弱性を作りこまないようなプログラムの書き方およびそれに関わる知識。

(*4) 検査対象のシステムやソフトウェアに対して、一般的に知られている攻撃（特定のパターン等）を実施し、特徴的な応答を観察することで、脆弱性を検出する手法。

1章では、システムライフサイクルに沿ったセキュリティ対策と、「ソースコードセキュリティ検査」の位置づけ、現在のセキュリティ対策の状況について説明します。

2章では、「ソースコードセキュリティ検査」の概要や他のセキュリティ検査技術との違いを解説します。2.4節では具体的に有効な場面を紹介します。

3章では、2章までに解説する、「ソースコードセキュリティ検査」の実施・成功事例を紹介することで、その有効性を示します。

4章では、「ソースコードセキュリティ検査ツール」の最新動向について解説します。

5章では、「ソースコードセキュリティ検査ツール」の導入を推進するための、「今後のIPAの施策」を紹介します。

IPAとしては、本レポートが「ソースコードセキュリティ検査」の普及の一助となり、安心・安全なIT社会の実現に寄与することを期待します。

■ 本件に関するお問い合わせ先
IPA 技術本部 セキュリティセンター　金野／大森／甲斐根
Tel: 03-5978-7527　Fax: 03-5978-7518　E-mail: vuln-inq@ipa.go.jp
■ 報道関係からのお問い合わせ先
IPA 戦略企画部広報グループ　横山／大海
Tel: 03-5978-7503　Fax: 03-5978-7510　E-mail: pr-inq@ipa.go.jp