東芝グループのソフトウェア開発工程における セキュリティー向上への取り組み

Efforts of Toshiba Group to Improve Security in Software Development Processes

古賀　国秀 KOGA Kunihide

現代のシステムは，様々な機器やクラウドシステムなどと連携して動作している。このようなシステムにおいて，製品のソフトウェア開発でセキュリティーの確保を行うには，セキュリティー脆弱（ぜいじゃく）性が入り込みにくい設計と，仮にセキュリティー脆弱性が入り込んだとしても早期に取り除くことができる仕組みが求められる。

東芝グループは，ソフトウェアの開発工程別に，⑴セキュアコーディング規約の策定，⑵セキュアコーディング教育講座の作成と実施，⑶セキュアコーディング遵守度合いを検証するツールの選定と導入，⑷OSS（オープンソースソフトウェア）のセキュリティー脆弱性を検証するツールの導入，⑸不正な入力データで攻撃を試行するツールの選定と導入，及び⑹各種ツール運用を効率化するツールの開発，の六つの具体的な施策を講じることで，ソフトウェア開発でのセキュリティー向上に取り組んでいる。

Many systems now work in synchronicity with other systems and with the cloud. To ensure the security of software for these systems, a design methodology that makes it difficult to introduce vulnerabilities into software and resources for removing them at an early stage are necessary.

The Toshiba Group is endeavoring to improve software security through the application of six measures: (1) establishment of secure coding standards, (2) development and provision of education courses on secure coding, (3) selection and use of tools for validating compliance with secure coding rules, (4) adoption of tools for evaluating the vulnerabilities of open-source software, (5) selection and use of tools for emulating attacks using false input data, and (6) development of tools for improving the efficiency of various tools.

1. まえがき

現代のシステムは，ネットワークを介して様々な機器やクラウドシステムなどと連携しながら動作している。こうしたシステムでは，ネットワーク経由でサイバー攻撃（注1）が波及し，更にソフトウェアで制御されているほかの機器にも影響を与えるおそれがある。このため，セキュリティーの確保が非常に重要となっている。

製品のソフトウェア開発におけるセキュリティーの確保には，開発工程の全体を通じ，セキュリティー脆弱性を入り込ませない設計をすることと，仮にセキュリティー脆弱性が入り込んだとしても，早期に取り除く仕組みが求められる。また，セキュリティー向上のためにソフトウェアを修正する場合，設計工程でのコストを基準とすると，実装工程で6.5倍となり，運用工程では60～100倍もの修正コストが必要になることが報告されており（**図1**）[1]，コストの視点でも早期に対策する必要がある。

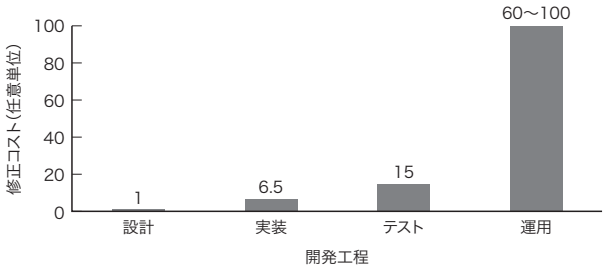

図1．開発工程別のセキュリティー脆弱性の修正コスト

セキュリティー対策として，設計工程でのコストを基準とした場合，実装工程では6.5倍，運用工程では60～100倍の修正コストが必要になる。

Costs of correcting security vulnerabilities in each development process

東芝のソフトウェア技術センターは，東芝グループ製品のソフトウェア開発の中で，セキュリティー確保に関する研究開発の一翼を担っている。

ここでは，ソフトウェア開発工程別のセキュリティー対策の概要と，東芝グループにおける，セキュリティー対策の具

（注1）コンピューターシステムやインターネットなどを利用して標的のコンピューターやネットワークに不正に侵入し，データの詐取や，破壊，改ざんなどを行ったり，標的のシステムを機能不全に陥らせたりすること。

体的な施策について述べる。

2. ソフトウェア開発工程別のセキュリティー対策

マイクロソフト社では，ソフトウェア開発でのセキュリティーに関する信頼性を向上させるため，ほぼ全てのソフトウェア開発で，同社が提唱しているセキュリティー開発ライフサイクル（SDL）[2]を適用している。SDLには，要件定義，設計，実装，テスト，リリース，及びサポートの六つの工程がある。SDLでは，ソフトウェア本体とソフトウェアが処理する情報を保護し，攻撃を防ぐように設計し，実装する必要がある。しかし，攻撃手法の高度化・巧妙化が日々進む中で，完全なセキュリティーを実現することは困難である。そこで，設計者は，残存するセキュリティーの欠陥を攻撃者が標的にした場合の被害を最小限に食い止めることを検討しなければならない。

SDLの要件定義からテスト工程までの各工程でのセキュリティー対策を，**図2**に示す。

要件定義工程では，対象製品に求められているセキュリティーの保証水準に対して，盛り込むべきセキュリティー対策を明らかにする（図2の①）。

設計工程では，システム構成や資産から脅威を洗い出し，資産がどのように保護されるべきかを明らかにする（同②）。また，守るべき資産を特定し，必要な保護機能を設計する（同③）。

実装工程では，全ての設計者がコーディング規約に従って，セキュアなプログラミングを行う（同④）。コーディング規約を適用することで，開発者がセキュリティー脆弱性につながる欠陥を入れ込むことを防ぐ。ルールを遵守し，一貫した入出力データの処理を行うことで，バッファーオーバーフローなどの攻撃を防ぐことができる。また，手作業によるコードレビューで，膨大なソースコードの修正箇所を全て検出することは困難であるため，ツールによって解析し，修正すべき箇所を明らかにする（同⑤）。更に，オープンソースソフトウェア（OSS）を利用している場合には，製品に搭載されているOSSで知られている，既知の脆弱性の混入状況について，情報を把握する（同⑥）。

テスト工程では，定義したセキュリティー要件を製品が遵守しているかの達成度合いを評価する（同⑦）。また，対象製品に対し，不正な入力データでの攻撃を試行し，セキュリティー脆弱性につながるエラーを検出する（同⑧）。

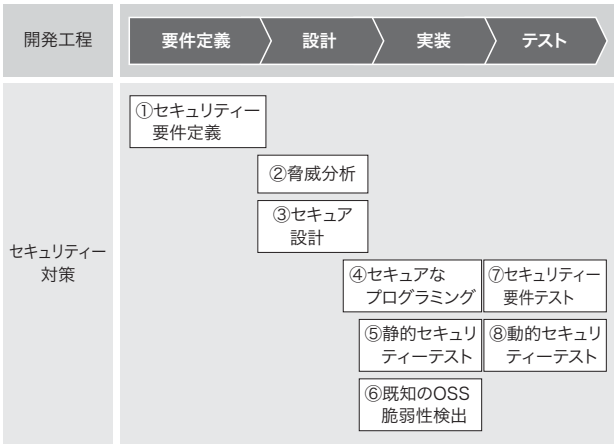

図2. 開発工程別のセキュリティー対策
マイクロソフト社が提唱しているSDLの中で，要件定義からテスト工程までには，大きく八つのセキュリティー対策がある。
Security measures for each development process proposed by Microsoft Corporation

3. セキュリティー対策の具体的な施策

IPA/SEC（独立行政法人 情報処理推進機構／ソフトウェア高信頼化センター）の調査[3]によると，「セキュリティ設計の品質をどのように「客観的」に確認していますか?」という質問に対して，評価手法やツールを用いて確認していると回答した割合は9％にすぎない。確認を行っていない理由は，客観的なレビューを行う人手やコストがない，適当な評価手法やツールが見当たらない，などである。

東芝グループは，この阻害要因を除去し，更にセキュリティーを向上させるために，①セキュアコーディング規約の策定，②セキュアコーディング教育講座の作成と実施，③セキュアコーディング遵守度合いを検証するツールの選定と導入，④OSSのセキュリティー脆弱性を検証するツールの導入，⑤不正な入力データで攻撃を試行するツールの選定と導入，及び⑥各種ツール運用を効率化するツールの開発，の6点の具体的な施策を講じた（**図3**）。それぞれ取り組みについて，以下に述べる。

3.1 施策① セキュアコーディング規約の策定

セキュアコーディングとは，開発者が攻撃者からの攻撃に耐えられる堅牢（けんろう）なプログラミングを行うことである。この目的のために，CERT/CC（Computer Emergency Response Team/Coordination Center）が中心となって，コーディング規約 SEI CERT コーディングスタンダード[4]をまとめている。この規約は，プログラミング言語を使ってセキュアコーディングを行うために必ず適用すべきルールと，製品の特性に応じて選択的に適応するルールを定めて

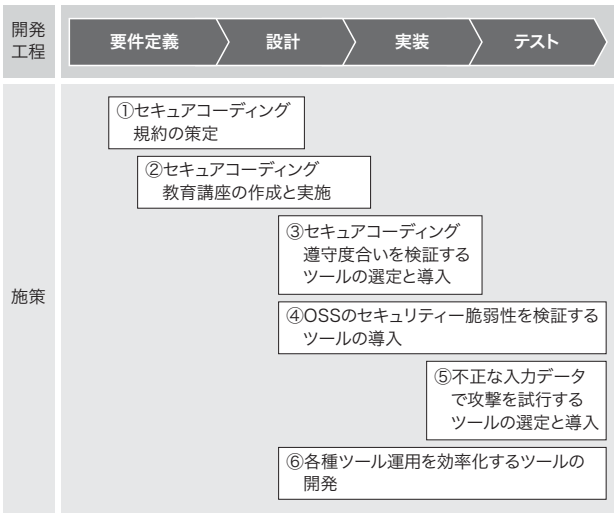

図3. セキュリティーを向上させるための六つの施策

東芝グループにおける具体的なセキュリティー対策として，六つの施策を講じた。

Six measures to improve software security

いる。東芝グループはSEI CERTコーディングスタンダードをベースとして，セキュアコーディング規約を作成する技術支援を行っている。これにより，各ソフトウェア開発部門において，開発者全員が均質なプログラミングを行えることを目指している。

3.2 施策② セキュアコーディング教育講座の作成と実施

施策①で策定したセキュアコーディング規約を開発者に提示するだけでは，修得することが難しい。そこで，開発者向けに，講義に演習を交えた教育を通してセキュアコーディングに関する知識とスキルを修得する支援を行っている。教育講座の作成にあたっては，製品の事業領域別に，最も使用されるプログラミング言語を中心に改訂した。そのプログラミング言語で陥りやすい誤ったプログラミング事例を題材とし，演習によって問題点と修正方法を修得する。製品の事業領域に関係する最新の攻撃動向を盛り込むことで，より身近な問題であることを開発者に意識させるようにしている。

3.3 施策③ セキュアコーディング遵守度合いを検証するツールの選定と導入

施策①と②を行っても，人がプログラミングを行っている限り，間違えることを前提に考えねばならない。そのために，セキュアコーディング遵守度合いを検証するツールが必要になる。このようなツールは数多く存在しており，しかも短い周期で機能向上するため，開発者自身で調査して最適なツールを選択することは，コストが掛かり過ぎる。そこで，商用，フリーを問わず，複数のツールを入手して詳細に評価し，東芝グループ全体に結果を報告している。各ツールの機能が向上する度に再評価を実施し，評価レポートを更新している。これにより，開発者は常に最適なツールを選択できるようになっている。

3.4 施策④ OSSのセキュリティー脆弱性を検証するツールの導入

OSSのセキュリティー脆弱性を検証するツールには，製品に搭載されているOSSについて，既知の脆弱性の混入状況を全て抜け漏れなく検出する正確性，全てのOSSの脆弱性情報を保有する網羅性，及び刻々と追加される既知の脆弱性情報をタイムリーに反映する迅速性が要求されるが，それらの全てを兼ね備えたツールは少ない。当社のソフトウェア技術センターでは，詳細な評価で高い能力を示したBlack Duck Hub™を導入し，東芝グループ全体で利用できる環境を構築した。更に，各ソフトウェア開発部門のセキュリティー要件に応じて，適切な図表を自動生成する機能を開発している。

3.5 施策⑤ 不正な入力データで攻撃を試行するツールの選定と導入

施策③と④は，プログラムを動かすことなく静的な解析で脆弱性のある箇所を検出する静的セキュリティーテストだが，システム全体を結合することで発生する脆弱性や，並行処理に起因する脆弱性の発見には，プログラムを動作させる動的セキュリティーテストが必要である。動的セキュリティーテストの手法として，ファジングツールがある。

ファジングツールは，意図的に作成した不正なパターンを含む信号を対象機器に入力し，機器が想定外の挙動をするかをチェックする。信号を受信したシステムがクラッシュ，あるいは想定しない動作をした場合は，そこに脆弱性があると判断される。テストに使用される不正メッセージは，ランダムではなく，プロトコルの構造上問題が発生しやすいと思われるものが選定されることで，効率的な脆弱性検出が可能となる。

東芝グループで導入実績があり，高い能力を持つファジングツール Defensics™を導入し，東芝グループ全体で利用可能とする環境を構築した。

3.6 施策⑥ 各種ツール運用を効率化するツールの開発

実装工程及びテスト工程で使用される検証ツール群の運用を効率化するツール TCANAを開発している（**図4**）。特長は，主に以下の四つである。

(1) 全てのツール及び解析結果を一元管理することで，

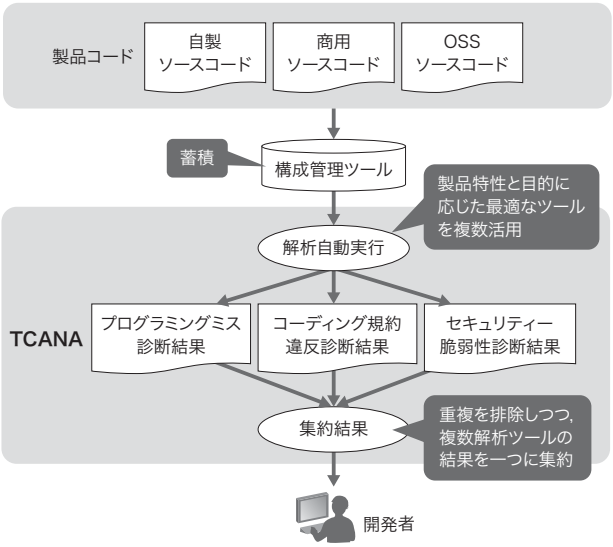

図4. 検証ツールの管理を効率化するツール TCANA

開発工程で使用される検証ツール群の運用を効率化するツールとして，TCANAを開発している。

TCANA tool for enhancing efficiency of verification tool management

導入コストや運用コストを削減している。また，ソフトウェア開発部門の検証ツールに掛かるコストを最小化している。

⑵ 構成管理ツールと連携することで，開発者は，自動化された解析環境をいつでも利用できる。

⑶ 開発者は，TCANAのユーザーインターフェースを通して結果を閲覧する。つまり，開発者が個々の検証ツールを意識することはない。

⑷ 複数の検証ツールの結果をマージし，重複した結果を排除した上で，集約して表示する。これにより，確認すべき指摘数が削減され，開発者は，原因特定までの時間を95％削減できる。また，セキュリティー脆弱性だけを表示するなど，目的に応じて選択的に結果を排除するフィルタリング機能も備えている。

4. あとがき

ソフトウェア開発工程におけるセキュリティー向上に対する取り組みとして，具体的な六つの施策を講じてきた。

今後，これらの施策の展開を加速させるため，セキュアコーディング規約とセキュアコーディング教育講座について，対応可能な事業領域やプログラミング言語を拡充するとともに，セキュアコーディング遵守度合いを検証するツールだけでなく，OSSのセキュリティー脆弱性を検証するツールや不正な入力データで攻撃を試行するツールにおいても，定期的な再評価による最適なツール選定を行うことで，更に高機能な検証ツールを低コストで運用できる環境を構築していく。また，TCANAについても，検証ツールの利用を完全に自動化させる技術の実現を目指し，研究開発を続けていく。

文 献

⑴ Hoo, K. S. et al. Tangible ROI through Secure Software Engineering. Security Business Quarterly. Fourth Quarter 2001, 1, 2.
⑵ Lipner, S.; Howard, M.“信頼できるコンピューティングのセキュリティ開発ライフサイクル”. Microsoft Developer Network. <https://msdn.microsoft.com/ja-jp/library/ms995349.aspx>,（参照 2018-05-25）.
⑶ IPA/SEC. セーフティ設計・セキュリティ設計に関する実態調査結果. 2015, 73P.
⑷ Carnegie Mellon University Software Engineering Institute (SEI). "SEI CERT Coding Standards". SEI External Wiki Home. <https://wiki.sei.cmu.edu/confluence/display/seccode/SEI+CERT+Coding+Standards>, (accessed 2018-05-25).

・Black Duck Hub，Defensicsは，米国Synopsys社の商標。

古賀 国秀 KOGA Kunihide
研究開発本部 ソフトウェア技術センター
ソフトウェアエンジニアリング技術部
Software Engineering Technology Dept.