車両サイバーセキュリティの未来（5）

車両開発におけるセキュアコーディング実施のポイント

PwC コンサルティング合同会社　シニアマネージャー　奥山 謙
PwC コンサルティング合同会社　シニアアソシエイト　吉田 万里子

前回までは、コンセプトフェーズにおける脅威分析とリスクアセスメント、開発フェーズにおけるセキュア設計、脆弱性分析と対策実施について紹介しました。第5回は、これらの活動の結果によって作成される設計書に従い、ソフトウェアを実装するフェーズで行うべき「セキュアコーディング」について考察します。

セキュアコーディング

セキュアコーディングとは、「サイバー攻撃に耐えられる堅牢なプログラムを書くこと」です。第4回で「設計段階で発生する脆弱性」と「実装段階で発生する脆弱性」について解説しましたが、セキュアコーディングは「実装段階で発生する脆弱性」に対する施策となります。

第1回で図解したISO/SAE 21434の6つの要素のひとつである「Product Development（製品開発）」に含まれる「ソフトウェアレベル」の中の最も詳細なレベルの活動です。

さらに、ソフトウェア全体の脆弱性としては「設計段階で発生する脆弱性」に比べて「実装段階で発生する脆弱性」が多く報告されており、脆弱性の「数」の観点でも重要と考えられます。

セキュアコーディングの具体的な作業

図表1に、セキュアコーディングのための具体的な作業を挙げます。コーディング作業に先立ち、プロジェクトの特性（製品が扱う情報や機能の重要度、コスト、納期など）に応じて全体計画を策定する必要があります。

図表 1 セキュアコーディングの具体的な作業

作業項目	概要
セキュアコーディング全体計画の策定	セキュアコーディングの全体計画を策定します。ソースコードレベルでのプログラムの堅牢性の確認をする方法（静的コード解析ツール利用、ピアレビュー※3など）についても検討します。
セキュアコーディング教育	セキュアコーディングの概要や、違反した場合に起こる問題について学びます。
コーディングルールの策定	自動車業界で一般的に利用されるCERT-C/C++※4や MISRA-C/C++※5などのコーディングスタンダードをベースとし、必要に応じて自社のルールを追加して策定します。
静的解析ツールの利用計画策定※静的解析ツール利用時	ソースコードレベルでのプログラムの堅牢性を確認するための静的解析ツールの選定や、静的解析ツール実行／対応計画の策定などを行います。
コーディング	コーディングを行います。開発ツールの警告メッセージを利用することで、プログラムの堅牢性をある程度、担保するこができます。
ソースコードレベルでのプログラムの堅牢性の確認	ツール利用による静的解析ツールや、ソースコードレビューをして、ソースコードレベルでのプログラムの堅牢性を確認します。ここで問題が検出された場合は、ソースコードの修正と確認作業を繰り返します。

セキュアコーディング時の課題と解決方法

多くの組織は何らかの静的解析ツールを利用してソースコードレベルでのプログラムの堅牢性の確認をしていますが、この時の大きな課題として、「検出される問題が多すぎて対応しきれない」、「ツールの誤検知かどうかの判断だけでも大変」ということが挙げられます。

この課題を解決するために、一般的なソフトウェアの開発では、静的解析ツールが示す重要度（Critical/HighやModerate/Lowなど）に応じた対応をする場合が多くあります。例えば、Critical/Highの問題については、修正、あるいは、リスクを分析したのち可能な場合はリスクを受容し、Moderate/Lowの問題については、誤検知かどうかの判定もしない方針とするケースなどもあり得ます。

一方、自動車業界では、静的解析ツールで検出された問題を修正しなかった場合、静的解析ツールが重要度=Lowと判定した問題でも、人命に関わる被害が発生する可能性があるため、静的解析ツールが示す重要度などにより自動的に対応を決定することは難しいのが実情です。

では、自動車のソフトウェア開発においては、どのような解決策があるのでしょうか。

PwCとしてクライアントにアドバイスしている解決策は、「ツールが検出した問題は全て対応する」、そのために「全ての問題に対応するための施策を考える」ということです。ここで、「対応する」とは、「誤検知と判定する」または「修正する」ことを指します。

ポイントは、「そもそも脆弱なコードを可能な限り入れない」ことと、「ツールが検出した問題に計画的に対応する」ことです。

そのための具体的な施策として、以下が挙げられます。

- セキュアコーディング教育を実施する
- 開発フェーズの早い段階から静的解析ツールを実行する
- CIツール※6に静的解析ツールを組み込むなど、静的解析ツールの実行を自動化する

以上、今回は「Product Development（製品開発）」に含まれる「ソフトウェアレベル」の中で最も詳細なレベルに該当するセキュアコーディングについて解説しました。

次回は、脆弱性診断によるHW（ハードウェア）/SW（ソフトウェア）レベルのセキュリティテストについて考察します。

※1 バッファオーバーフロー：プログラムによって確保されたメモリ領域を超えるデータが送り込まれることにより誤動作が生じること

※2 SQLインジェクション：セキュリティ上の不備を利用して攻撃者が任意のSQL文（データベースへの命令文）を実行させてデータベースを不正に操作すること

※3ピアレビュー：立場や職種が同じ（または近い）者によって、成果物を検証して改善を図る品質保証の手法

※4 CERT-C/C++：セキュアなソフトウェアをつくるためのコーディングガイドライン。約300項目のルールが規定されている。

※5MISTRA-C/C++：自動車の機能安全を目的として作成されたC/C++言語用のコーディングガイドライン。2016年にセキュリティに関するルールも発行された。

※6Continuous Integration(CI)ツール：ソースコードのビルド（実行可能ファイルや配布パッケージを作成する処理）やテストを継続的に実施するためのツール。静的解析ツールをJenkinsなどの自動ビルド機能を備えたツールと連携することで、自動ビルドのタイミングで静的解析を実行することができる。

お問い合わせ

PwCコンサルティング合同会社
〒100-6921 東京都千代田区丸の内2-6-1 丸の内パークビルディング
Tel : 03-6250-1200（代表） Mail : jp_cyber_inquiry@pwc.com