(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2021-157401
(P2021-157401A)
(43)公開日　令和3年10月7日(2021.10.7)

(51)Int.Cl.		F I		テーマコード（参考）
G06F 8/20	(2018.01)	G06F 8/20		5B376
G06F 21/57	(2013.01)	G06F 21/57	320	

審査請求 未請求　請求項の数 10　ＯＬ　（全 11 頁）

(21)出願番号　特願2020-56103(P2020-56103)
(22)出願日　令和2年3月26日(2020.3.26)

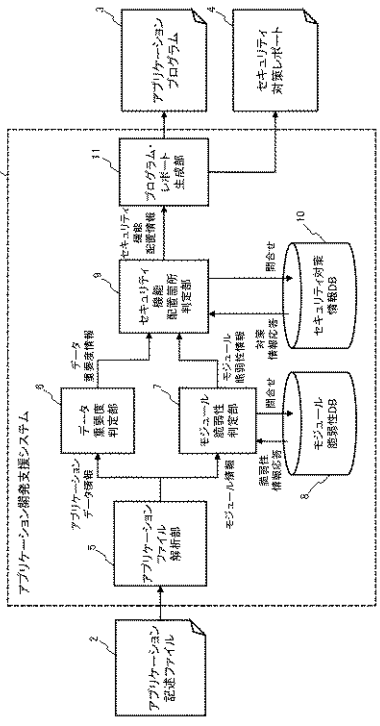
(71)出願人　000005108
株式会社日立製作所
東京都千代田区丸の内一丁目６番６号
(74)代理人　110000350
ポレール特許業務法人
(72)発明者　藤田　淳也
東京都千代田区丸の内一丁目６番６号　株式会社日立製作所内
(72)発明者　小笠原　英道
東京都千代田区丸の内一丁目６番６号　株式会社日立製作所内

Ｆターム(参考)　5B376 BC16　BC32　FA13

(54)【発明の名称】アプリケーション開発支援システム、アプリケーション開発支援方法

(57)【要約】　（修正有）
【課題】フローダイアグラムを用いたアプリケーション開発環境において、必要とする箇所に最適なセキュリティ対策を設定する技術を提供する。
【解決手段】アプリケーション開発支援システムにおいて、入力されたアプリケーション記述ファイルを解析し、アプリケーションデータ情報及びモジュール情報を出力する解析部と、アプリケーションデータ情報に基づいてモジュール間でやり取りされるデータの重要度を決定するデータ重要度判定部と、モジュール情報及びモジュール脆弱性データベースから読み出された脆弱性情報に基づいて各モジュールの脆弱性スコアを決定する脆弱性判定部と、データ重要度判定部で決定したデータ重要度情報、脆弱性判定部で決定した各モジュールの脆弱性スコア及びセキュリティ対策情報データベースから読み出されたセキュリティ対策情報に基づいて、セキュリティ機能の配置箇所を決定するセキュリティ機能配置箇所判定部と、を備える。
【選択図】図１

図1



【特許請求の範囲】
【請求項１】
フローダイアグラムを用いたアプリケーション開発を支援するアプリケーション開発支援システムであって、
入力されたアプリケーション記述ファイルを解析し、アプリケーションデータ情報およびモジュール情報を出力するアプリケーションファイル解析部と、
前記アプリケーションデータ情報に基づいてモジュール間でやり取りされるデータの重要度を決定するデータ重要度判定部と、
前記モジュール情報およびモジュール脆弱性データベースから読み出された脆弱性情報に基づいて各モジュールの脆弱性スコアを決定するモジュール脆弱性判定部と、
前記データ重要度判定部で決定されたデータ重要度情報、前記モジュール脆弱性判定部で決定された各モジュールの脆弱性スコア、セキュリティ対策情報データベースから読み出されたセキュリティ対策情報に基づいてセキュリティ機能の配置箇所を決定するセキュリティ機能配置箇所判定部と、
を備えることを特徴とするアプリケーション開発支援システム。
【請求項２】
請求項１に記載のアプリケーション開発支援システムであって、
前記各モジュールは、アプリケーションフローを構成し、
前記脆弱性スコアと前記モジュール間でやり取りされるデータの重要度が高いフロー、かつ、当該フロー間のモジュールの脆弱性スコアが高い箇所にセキュリティ機能を割り当てることを特徴とするアプリケーション開発支援システム。
【請求項３】
請求項２に記載のアプリケーション開発支援システムであって、
前記各モジュールは、階層化された機能ブロック型モジュール構造を有し、
前記モジュール脆弱性判定部は、各モジュール内のデータフローおよび各モジュールを構成するサブモジュールの脆弱性スコアを基に各モジュールの脆弱性スコアを決定することを特徴とするアプリケーション開発支援システム。
【請求項４】
請求項２に記載のアプリケーション開発支援システムであって、
前記各モジュールは、基本モジュール群から構成され、
前記基本モジュール群の各々は、各モジュールの特徴に応じた固有の脆弱性スコアを有することを特徴とするアプリケーション開発支援システム。
【請求項５】
請求項２に記載のアプリケーション開発支援システムであって、
前記セキュリティ対策情報データベースに格納されたセキュリティ対策情報は、対策の信頼性を基に生成した信頼性スコア情報を有し、
前記信頼性スコア情報に基づいて、信頼性の高いセキュリティ機能を優先的に割り当てることを特徴とするアプリケーション開発支援システム。
【請求項６】
フローダイアグラムを用いたアプリケーション開発を支援するアプリケーション開発支援方法であって、
アプリケーション記述ファイルを読み込んで解析し、アプリケーションデータ情報およびモジュール情報を取得し、
前記アプリケーションデータ情報に基づいてモジュール間でやり取りされるデータの重要度を決定し、
前記モジュール情報およびモジュール脆弱性データベースから読み出された脆弱性情報に基づいて各モジュールの脆弱性スコアを決定し、
前記モジュール間でやり取りされるデータの重要度および前記各モジュールの脆弱性スコア、セキュリティ対策情報データベースから読み出されたセキュリティ対策情報に基づいてセキュリティ機能の配置箇所を決定することを特徴とするアプリケーション開発支援

方法。
【請求項７】
請求項６に記載のアプリケーション開発支援方法であって、
前記各モジュールは、アプリケーションフローを構成し、
前記脆弱性スコアと前記モジュール間でやり取りされるデータの重要度が高いフロー、かつ、当該フロー間のモジュールの脆弱性スコアが高い箇所にセキュリティ機能を割り当てることを特徴とするアプリケーション開発支援方法。
【請求項８】
請求項７に記載のアプリケーション開発支援方法であって、
前記各モジュールは、階層化された機能ブロック型モジュール構造を有し、
各モジュール内のデータフローおよび各モジュールを構成するサブモジュールの脆弱性スコアを基に各モジュールの脆弱性スコアを決定することを特徴とするアプリケーション開発支援方法。
【請求項９】
請求項７に記載のアプリケーション開発支援方法であって、
前記各モジュールは、基本モジュール群から構成され、
前記基本モジュール群の各々は、各モジュールの特徴に応じた固有の脆弱性スコアを有することを特徴とするアプリケーション開発支援方法。
【請求項１０】
請求項７に記載のアプリケーション開発支援方法であって、
前記セキュリティ対策情報は、対策の信頼性を基に生成した信頼性スコア情報を有し、
前記信頼性スコア情報に基づいて、信頼性の高いセキュリティ機能を優先的に割り当てることを特徴とするアプリケーション開発支援方法。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、フローダイアグラムを用いたソフトウェア開発環境におけるアプリケーション開発を支援する技術に係り、特に、アプリケーションのセキュリティ対策に有効な技術に関する。
【背景技術】
【０００２】
ソフトウェア開発において、フローダイアグラムで記述するプログラミング言語を採用することで、プログラミング未経験者でも比較的簡単にアプリケーションを開発できる環境を実現することができる。一方、セキュリティ対策が十分に考慮されておらず、開発したアプリケーションのセキュリティ確保が重要な課題となっている。
【０００３】
本技術分野の背景技術として、例えば、特許文献１のような技術がある。特許文献１には「情報処理装置上で動作し、１つ以上のコンポーネントから構成される対象システムの各コンポーネントについて、セキュリティに関する保証レベルを導出するツールであって、前記対象システムが想定する脅威とコンポーネントが備えるセキュリティ機能との対応関係が格納された、脅威－コンポーネント対応関係テーブルと、前記各脅威のリスクの大きさを示すリスク値の情報が格納された、リスク値テーブルと、前記各脅威に対抗するコンポーネントが備えるセキュリティ機能の効果の大きさに関する情報が格納された、コンポーネント寄与率テーブルと、リスク値とこれに応じて要求される保証レベルの対応関係が格納された、リスク－保証レベル対応関係テーブルと、前記脅威－コンポーネント対応関係テーブルと、前記リスク値テーブルと、前記コンポーネント寄与率テーブルと、前記リスク－保証レベル対応関係テーブルとの各間において、互いの重複項目に基づき、前記各テーブルの連係関係を特定する、テーブル連結利用部と、前記各テーブルの連係関係に基づき、前記脅威のリスク値をコンポーネント寄与率により重み付けすることで、コンポーネントが備えるセキュリティ機能が対抗する脅威のリスク値に見合った各コンポーネン

トの保証レベルを導出する、コンポーネント保証レベル導出部と、を備えるシステムセキュリティ設計・評価支援ツール」が開示されている。

【０００４】

また、特許文献２には「（Ａ）フローダイアグラムを用いたモデル開発環境における編集中のフローに関して第１ノードが選定された場合に、第２ノードの１以上の候補である１以上の第２ノード候補を提示し、前記第１ノードは、入力ノードと出力ノードのうちの一方の種類のノードに該当するいずれかのノードであり、前記第２ノードは、入力ノードと出力ノードのうちの他方の種類のノードと入力ノードと出力ノードのいずれにもなれる種類のノードである入力／出力ノードとのうちのいずれかの種類に該当するノードのうちの、前記第１ノードに対応するノードであり、（Ｂ）前記１以上の第２ノード候補からいずれかの第２ノード候補が第２ノードとして選定された場合に、前記第１ノードと前記第２ノード間の部分フローの１以上の候補である１以上の部分フロー候補を提示する、ことをコンピューター装置に実行させるコンピュータープログラム」が開示されている。

【０００５】

また、特許文献３には「メモリまたはプロセッサのうちの少なくとも１つで実装され、クライアントデバイスおよび一組のリソースと電子的に通信するように構成される、通信モジュールと、（１）一組の脅威からの各脅威に対する一組のリスク緩和スコアに関連付けられている脅威信頼度ベクトル、および（２）前記一組の脅威からの各脅威に対する一組のリソース脆弱性スコアに基づき、前記一組のリソースからの各リソースに対するリソース信頼度基準を定義するように構成されているポリシー定義モジュールと、（１）前記通信モジュールを介して、前記一組のリソースからのリソースに関連付けられている認証要求を示す信号を受信し、（２）前記認証要求に関連付けられている脅威信頼度ベクトルおよび前記一組のリソース脆弱性スコアに基づき前記一組のリソースからの前記リソースに対するリソース信頼度値を定義するように構成されている、ポリシー適用モジュールと、を備え、前記ポリシー適用モジュールは（１）前記一組のリソースからの前記リソースに対する前記リソース信頼度値と前記一組のリソースからの前記リソースに対する前記リソース信頼度基準とを比較し、（２）前記一組のリソースからの前記リソースに対する前記リソース信頼度基準が満たされたときに、前記通信モジュールを介して肯定的認証を示す信号を送信し、前記クライアントデバイスが前記リソースへのアクセスを認められるように構成される装置」が開示されている。

【先行技術文献】

【特許文献】

【０００６】

【特許文献１】特開２００６－３３１３８３号公報

【特許文献２】特開２０１９－１４８８５９号公報

【特許文献３】特表２０１７－５２２６６７号公報

【発明の概要】

【発明が解決しようとする課題】

【０００７】

ところで、アプリケーション開発において、情報セキュリティ（Information Security）は情報の「機密性」（Confidentiality）、「完全性」（Integrity）、「可用性」（Availability）の３要素からなると定義されている。この３要素の頭文字をとって「CIA」と呼ばれる。

【０００８】

また、セキュリティ対策の優先度は「影響（Impact）×可能性（Likelihood）」の度合い、つまり「（セキュリティ）リスク」の大きさによって決定される。

【０００９】

「影響（Impact）」は「流れているデータの価値」、すなわち、「機密性」（Confidentiality）、「完全性」（Integrity）、「可用性」（Availability）の３要素で決まる。

また、「可能性（Likelihood）」は「脆弱性（Exploitability）」の数と度合い（脆弱性

スコア）で決まる。
【００１０】
従って、アプリケーション開発におけるセキュリティ対策には、これらの要素を考慮した開発支援が必要になる。
【００１１】
しかしながら、上記特許文献１，２には、いずれにも「脆弱性スコア」に関する記載はなく、優先度を考慮したセキュリティ対策が行われていないため、開発システムの負荷の増大や必要とする箇所に最適なセキュリティ対策が行われない可能性がある。
【００１２】
また、上記特許文献３は、フローダイアグラムを用いたアプリケーション開発を対象とするものではなく、フローダイアグラムを用いた場合のセキュリティ対策については言及されていない。
【００１３】
そこで、本発明の目的は、フローダイアグラムを用いたアプリケーション開発環境において、必要とする箇所に最適なセキュリティ対策を設定可能なアプリケーション開発支援システム及びアプリケーション開発支援方法を提供することにある。
【課題を解決するための手段】
【００１４】
上記課題を解決するために、本発明は、フローダイアグラムを用いたアプリケーション開発を支援するアプリケーション開発支援システムであって、入力されたアプリケーション記述ファイルを解析し、アプリケーションデータ情報およびモジュール情報を出力するアプリケーションファイル解析部と、前記アプリケーションデータ情報に基づいてモジュール間でやり取りされるデータの重要度を決定するデータ重要度判定部と、前記モジュール情報およびモジュール脆弱性データベースから読み出された脆弱性情報に基づいて各モジュールの脆弱性スコアを決定するモジュール脆弱性判定部と、前記データ重要度判定部で決定されたデータ重要度情報、前記モジュール脆弱性判定部で決定された各モジュールの脆弱性スコア、セキュリティ対策情報データベースから読み出されたセキュリティ対策情報に基づいてセキュリティ機能の配置箇所を決定するセキュリティ機能配置箇所判定部と、を備えることを特徴とする。
【００１５】
また、本発明は、フローダイアグラムを用いたアプリケーション開発を支援するアプリケーション開発支援方法であって、アプリケーション記述ファイルを読み込んで解析し、アプリケーションデータ情報およびモジュール情報を取得し、前記アプリケーションデータ情報に基づいてモジュール間でやり取りされるデータの重要度を決定し、前記モジュール情報およびモジュール脆弱性データベースから読み出された脆弱性情報に基づいて各モジュールの脆弱性スコアを決定し、前記モジュール間でやり取りされるデータの重要度および前記各モジュールの脆弱性スコア、セキュリティ対策情報データベースから読み出されたセキュリティ対策情報に基づいてセキュリティ機能の配置箇所を決定することを特徴とする。
【発明の効果】
【００１６】
本発明によれば、フローダイアグラムを用いたアプリケーション開発環境において、必要とする箇所に最適なセキュリティ対策を設定可能なアプリケーション開発支援システム及びアプリケーション開発支援方法を提供することができる。
【００１７】
これにより、最適なセキュリティ対策が施されたアプリケーションを比較的簡単に開発することができる。
【００１８】
上記した以外の課題、構成及び効果は、以下の実施形態の説明により明らかにされる。
【図面の簡単な説明】

【００１９】
【図１】本発明の実施例１に係るアプリケーション開発支援システムの全体構成を示す図である。
【図２】本発明の実施例１に係るアプリケーション開発支援方法（主要処理）を示すフローチャートである。
【図３】本発明の実施例１に係るセキュリティ機能配置箇所判定部によるセキュリティモジュールの割り当てを概念的に示す図である。
【図４】本発明の実施例２に係るアプリケーション開発プラットフォームを概念的に示す図である。
【図５】本発明の実施例３に係るアプリケーション開発システムを概念的に示す図である。
【発明を実施するための形態】
【００２０】
以下、図面を用いて本発明の実施例を説明する。なお、各図面において同一の構成については同一の符号を付し、重複する部分についてはその詳細な説明は省略する。
【実施例１】
【００２１】
図１から図３を参照して、本発明の実施例１に係るアプリケーション開発支援システム及びアプリケーション開発支援方法について説明する。図１は、本実施例のアプリケーション開発支援システムの全体構成を示す図である。図２は、本実施例のアプリケーション開発支援方法（主要処理）を示すフローチャートである。図３は、図１のアプリケーション開発支援システムによるセキュリティモジュール（セキュリティ機能）の割り当てを概念的に示す図である。
【００２２】
本実施例のアプリケーション開発支援システム１は、図１に示すように、主要な構成として、アプリケーションファイル解析部５と、データ重要度判定部６と、モジュール脆弱性判定部７と、モジュール脆弱性データベース８と、セキュリティ機能配置箇所判定部９と、セキュリティ対策情報データベース１０と、プログラム・レポート生成部１１を備えている。
【００２３】
アプリケーションファイル解析部５は、入力されたアプリケーション記述ファイル２を解析し、アプリケーションデータ情報とモジュール情報を取得する。アプリケーションデータ情報はデータ重要度判定部６へ出力され、モジュール情報はモジュール脆弱性判定部７へ出力される。なお、アプリケーション記述ファイル２には、アプリケーションフロー（プロジェクト）情報が含まれている。アプリケーションフローは、複数のモジュールから構成される。（図５参照）
データ重要度判定部６は、入力されたアプリケーションデータ情報に基づいてモジュール間でやり取りされるデータの重要度を決定する。
【００２４】
一方、モジュール脆弱性判定部７は、入力されたモジュール情報と、モジュール脆弱性データベース８から読み出された脆弱性情報に基づいて各モジュールの脆弱性スコア（モジュール脆弱性情報：「脆弱性（Exploitability）」の数と度合い）を決定する。
【００２５】
セキュリティ機能配置箇所判定部９は、データ重要度判定部６で決定されたデータ重要度情報と、モジュール脆弱性判定部７で決定された各モジュールの脆弱性スコア（モジュール脆弱性情報）と、セキュリティ対策情報データベース８から読み出されたセキュリティ対策情報に基づいて、アプリケーションフローを構成する複数のモジュールにおけるセキュリティモジュール（セキュリティ機能）の配置箇所を決定する。
【００２６】
セキュリティモジュール（セキュリティ機能）は、脆弱性スコア（モジュール脆弱性情

報）と、各モジュール間でやり取りされるデータの重要度が高いフロー、かつ、当該フロー間のモジュールの脆弱性スコア（モジュール脆弱性情報）が高い箇所に優先的（選択的）に割り当てられる。

【００２７】

セキュリティ機能配置箇所判定部９で決定されたセキュリティ機能配置情報は、プログラム・レポート生成部１１に入力される。プログラム・レポート生成部１１は、入力されたセキュリティ機能配置情報に基づいて、アプリケーションプログラム３を生成し、セキュリティ対策レポート４を発行する。生成されたアプリケーションプログラム３は、必要とする箇所に最適なセキュリティ対策が施されたセキュアなコンピュータプログラム（アプリケーション）となる。

【００２８】

なお、図１では、モジュール脆弱性データベース８及びセキュリティ対策情報データベース１０がアプリケーション開発支援システム１に内蔵される構成例を示したが、モジュール脆弱性データベース８及びセキュリティ対策情報データベース１０を外部に設置し、モジュール脆弱性判定部７とモジュール脆弱性データベース８、セキュリティ機能配置箇所判定部９とセキュリティ対策情報データベース１０を、それぞれ通信ネットワーク等を介して接続する構成としても良い。

【００２９】

上述したアプリケーション開発支援システム１による主要な処理フローを図２に示す。

【００３０】

先ず、アプリケーション記述ファイルを読み込んで解析し、アプリケーションデータ情報およびモジュール情報を取得する。（ステップＳ１）

次に、取得したアプリケーションデータ情報に基づいて、モジュール間でやり取りされるデータの重要度を決定する。（ステップＳ２）

続いて、取得したモジュール情報と、モジュール脆弱性データベースから読み出された脆弱性情報に基づいて、各モジュールの脆弱性スコアを決定する。（ステップＳ３）

次に、決定したデータの重要度と、決定したモジュール毎の脆弱性スコア（情報）と、セキュリティ対策情報データベースから読み出されたセキュリティ対策情報に基づいて、セキュリティ機能の割り当て候補箇所（配置箇所）を決定する。（ステップＳ４）

続いて、割り当て可能なセキュリティモジュール（セキュリティ機能）が存在するか否かを判定する。（ステップＳ５）

割り当て可能なセキュリティモジュール（セキュリティ機能）が存在すると判定された場合（Ｙｅｓ）、セキュリティモジュール（セキュリティ機能）の割り当てを実行する。（ステップＳ６）

一方、割り当て可能なセキュリティモジュール（セキュリティ機能）が存在しないと判定された場合（Ｎｏ）、割り当て可能なセキュリティモジュール（セキュリティ機能）が存在しない旨のセキュリティ対策レポートを発行し（ステップＳ９）、処理を終了する。

【００３１】

ステップＳ６において、セキュリティモジュール（セキュリティ機能）の割り当てを実行した後、モジュールの脆弱性スコアを再計算する。（ステップＳ７）

次に、残存リスクが許容範囲であるか、もしくは、割り当て可能な対策が無いかを判定する。（ステップＳ８）

残存リスクが許容範囲内である、もしくは、割り当て可能な対策が無い（割り当て可能なセキュリティ機能が存在しなくなった）と判定された場合（Ｙｅｓ）、セキュリティ機能を含むアプリケーションプログラムを生成し、対策割り当てレポートを発行する。（ステップＳ９）

一方、残存リスクが許容範囲外であり、かつ、割り当て可能な対策が有る（割り当て可能なセキュリティ機能が存在する）と判定された場合（Ｎｏ）、ステップＳ４へ戻り、残存リスクが許容範囲内である、もしくは、割り当て可能な対策が無い（割り当て可能なセキュリティ機能が存在しなくなった）と判定されるまで、ステップＳ４からステップＳ８

までの処理フローを繰り返し実行する。

【００３２】

上述したセキュリティ機能配置箇所判定部９によるセキュリティモジュール（セキュリティ機能）の割り当てを図３に示す。

【００３３】

図３において、アプリケーションモジュール１（１３）およびアプリケーションモジュール２（１４）は、アプリケーションフローを構成している。

【００３４】

セキュリティ機能配置箇所判定部９は、モジュール脆弱性判定部７を介して重み付け（脆弱性スコア化）されたアプリケーションモジュール１（１３）のＣＶＳＳ情報（モジュールの脆弱性）とアプリケーションモジュール２（１４）のＣＶＳＳ情報（モジュールの脆弱性）、セキュリティ対策情報データベース１０から読み出されたセキュリティ対策情報、データ重要度判定部６を介して重み付け（重要度を付与）されたモジュール間でやり取りされるデータ（図示せず）に基づいて、対策機能１（１２）の配置箇所を決定する。

【００３５】

セキュリティ対策情報データベース１０には、適用可能な条件と効果のある脆弱性の組合せが複数の対策機能として格納されている。

【００３６】

以上説明したように、本実施例のアプリケーション開発支援システム及びアプリケーション開発支援方法では、リスク値が高い箇所を順にセキュリティ機能モジュールを割り当て、１つのセキュリティ機能モジュールを割り当てるたびに脆弱性が減ることから、全体のリスクが変わるため、特定のリスクスコア（受け入れ可能なリスク）になるまで再計算し、収束条件として全体のリスクが受け入れ可能な状態になるか、もしくは、割り当て可能なセキュリティ機能モジュールが存在しなくなったタイミングで計算を終了する。

【００３７】

これにより、フローダイアグラムを用いたアプリケーション開発環境において、必要とする箇所に最適なセキュリティ対策を設定することができるため、最適なセキュリティ対策が施されたアプリケーションを比較的簡単に開発することができる。

【００３８】

［変形例１］

なお、各モジュールが、階層化された機能ブロック型モジュール構造を有する場合、モジュール脆弱性判定部７は、各モジュール内のデータフローおよび各モジュールを構成するサブモジュールの脆弱性スコアを基に各モジュールの脆弱性スコアを決定するように構成しても良い。

【００３９】

これにより、抽象化されたモジュールの脆弱性を決定することができる。

【００４０】

［変形例２］

また、各モジュールが、基本モジュール群から構成される場合、基本モジュール群の各々は、各モジュールの特徴に応じた固有の脆弱性スコアを有するように構成しても良い。

【００４１】

これにより、未知モジュールの脆弱性を既知モジュールから決定することができる。

【００４２】

［変形例３］

また、セキュリティ対策情報データベース１０に格納されたセキュリティ対策情報は、対策の信頼性を基に生成した信頼性スコア情報を有し、その信頼性スコア情報に基づいて、信頼性の高いセキュリティ機能を優先的に割り当てるように構成しても良い。

【００４３】

これにより、セキュリティ機能の誤動作リスクを軽減することができる。

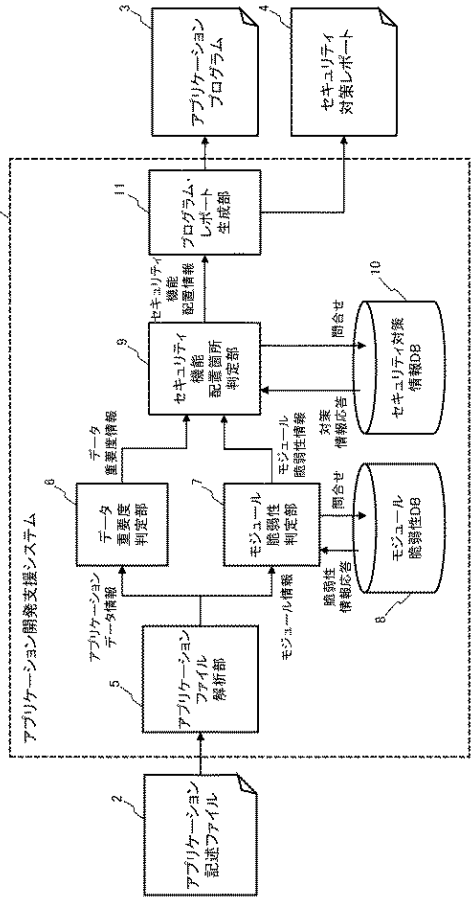
【実施例２】

【００４４】
図４を参照して、本発明の実施例２に係るアプリケーション開発プラットフォームについて説明する。図４は、本実施例のアプリケーション開発プラットフォームの全体構成を概念的に示す図であり、実施例１で説明した本発明のユースケースの一例を示している。
【００４５】
図４に示すように、それぞれ脆弱性情報と紐付けされたアプリケーション部品Ａ，Ｂ，Ｃからなる部品群（開発プラットフォーム）を用いてアプリケーション開発を行う場合、部品群（開発プラットフォーム）に登録されたアプリケーション部品を基に開発者が必要とする機能を実行するアプリケーションを開発する。この際、実施例１で説明したアプリケーション開発支援システムやアプリケーション開発支援方法を用いる。
【００４６】
汎用性が高いアプリケーション部品については、新たな部品として部品群（開発プラットフォーム）に登録され、他のアプリケーションフロー（プロジェクト）にも流用される。
【００４７】
これにより、新たにアプリケーション開発を行う際に、汎用性が高いアプリケーション部品のセキュリティ対策を改めて考慮する必要がなくなり、アプリケーション開発期間を短縮することができる。
【実施例３】
【００４８】
図５を参照して、本発明の実施例３のアプリケーション開発システムについて説明する。図５は、本実施例のアプリケーション開発システムの全体構成を概念的に示す図であり、実施例１で説明した本発明のユースケースの一例を示している。
【００４９】
図５において、アプリケーション生成エンジンが実施例１（図１）のアプリケーション開発支援システム１に相当する。
【００５０】
セキュリティの非専門家が作成したアプリケーションフロー（プロジェクト）情報（図１のアプリケーション記述ファイル２に相当）とモジュール脆弱性情報がアプリケーション生成エンジンに入力され、セキュリティ処理エンジンが有する３つの機能（セキュリティリスク分析，セキュリティ機能決定，セキュリティ機能コード生成）により、必要とする箇所に最適なセキュリティ対策が設定されたセキュアなコンピュータープログラム（アプリケーション）が出力される。セキュリティ処理エンジンの３つの機能の内、セキュリティ機能決定に本発明が適用される。
【００５１】
なお、本発明は上記した実施例に限定されるものではなく、様々な変形例が含まれる。例えば、上記の実施例は本発明に対する理解を助けるために詳細に説明したものであり、必ずしも説明した全ての構成を備えるものに限定されるものではない。また、ある実施例の構成の一部を他の実施例の構成に置き換えることが可能であり、また、ある実施例の構成に他の実施例の構成を加えることも可能である。また、各実施例の構成の一部について、他の構成の追加・削除・置換をすることが可能である。
【符号の説明】
【００５２】
１…アプリケーション開発支援システム
２…アプリケーション記述ファイル
３…アプリケーションプログラム
４…セキュリティ対策レポート
５…アプリケーションファイル解析部
６…データ重要度判定部
７…モジュール脆弱性判定部

８…モジュール脆弱性データベース
９…セキュリティ機能配置箇所判定部
１０…セキュリティ対策情報データベース
１１…プログラム・レポート生成部
１２…対策機能１
１３…アプリケーションモジュール１
１４…アプリケーションモジュール２

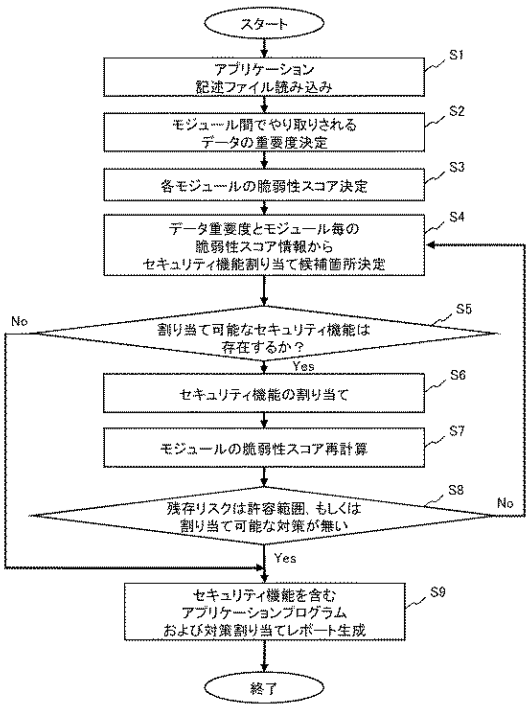
【図１】

図 1



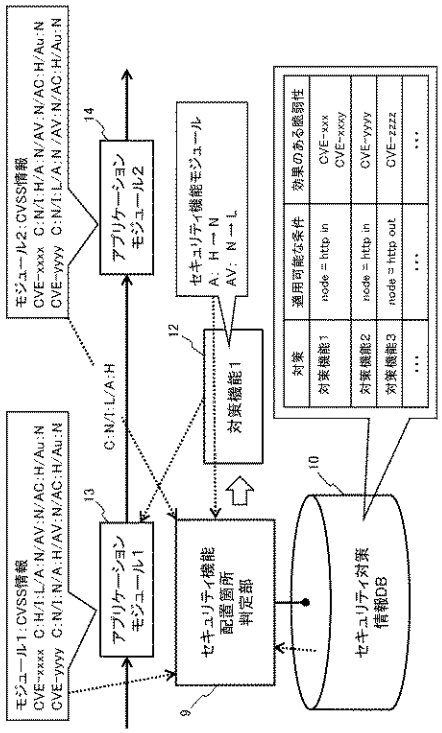
【図２】

図 2



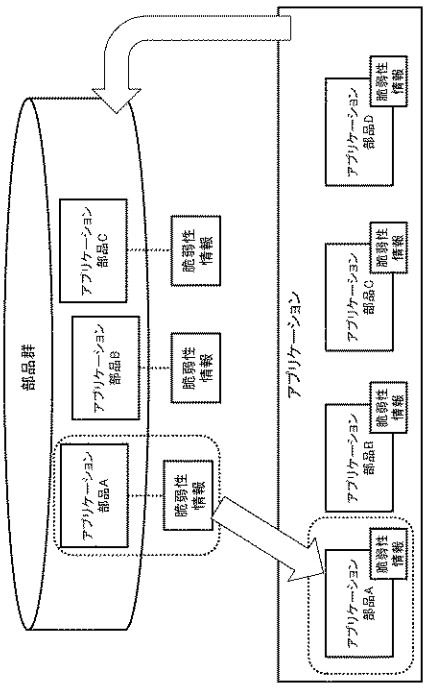
【図３】

図 3



【図４】

図 4



【図５】

図 5