Computer Security Symposium 2020
26 - 29 October 2020

セキュアなソフトウェア開発の阻害要因分析

金井 文宏[1,a)]　長谷川 彩子[1]　塩治 榮太朗[1]　秋山 満昭[1]

概要：複数人のプロジェクトで行われるソフトウェア開発の現場では，非機能要件であるセキュリティよりも，明示的な要件が優先される傾向にある．ソフトウェア開発の現場におけるセキュリティを妨げる要因を明らかにすることは，開発時に活用できるセキュリティ対策技術の設計やシフトレフトを推進する上で重要である．本研究では，プロフェッショナルのソフトウェア開発者（N=375）を対象としたオンラインアンケートにより，セキュアなソフトウェアの開発を妨げる要因を定量的に分析した．我々は，開発対象のソフトェアのユーザやプロジェクトの契約関係など，開発物/開発現場の持つ特性がソフトウェア開発者のセキュリティ意識や行動に与える影響を明らかにした．さらに我々は，デベロッパーとマネージャーの 2 群に分けてグループ間分析をすることで，2 者間のセキュリティ意識や行動のギャップがセキュリティに与える影響を明らかにした．この調査によって，セキュリティに関する意思決定を補助する技術の重要性など，ソフトウェア開発者にとって利用しやすいセキュリティ対策を検討する上での知見を得た．

キーワード：ユーザブルセキュリティ, ソフトウェア開発者, セキュア開発

Analysis of constraints that prevent the development of secure software

Fumihiro Kanei[1,a)]　Ayako Hasegawa[1]　Eitaro Shioji[1]　Mitsuaki Akiyama[1]

Abstract: Software security, which is often regarded as a non-functional requirement, tends to be less prioritized than other explicit requirements in a development project with multiple members. For designing security measures that can be used in software development, it is important to clarify the constraints that prevent the adoption of secure software development practices. In this study, we quantitatively analyzed such constraints through an online survey of software development professionals(N=375). We revealed how certain characteristics of a development project, such as the project's contractual relationships or the software's target users, influence developers' security awareness and behaviors. In addition, by comparing between the survey results of the two groups, developers and managers, we revealed how the gap in their security awareness and behavior influence software security. The results provide insights toward designing security measures usable for software developers, such as how technology might play an important role in assisting security-related decision-makings.

Keywords: Security, Software developer, Secure development

1. はじめに

昨今のサイバー攻撃の原因の一つとしてソフトウェア脆弱性が挙げられる．攻撃者は，攻撃対象のソフトウェアの脆弱性を悪用する事で，マルウェア感染や DDoS 攻撃など，様々な攻撃を行う．

このような脆弱性への対策として，ソフトウェア工学を始めたとした様々な分野で，脆弱性の検知や修正を目的とした技術が検討されている [1–4]．また，近年では，前述のソフトウェアを対象にした分析に加えて，ソフトウェアを開発する人間，つまりソフトウェア開発者の心理や行動に着目して，ソフトウェアに脆弱性が生じる原因の分析やそれらを防ぐ方法の研究も行われている．文献 [5,6] では，暗号化 API を始めとしたライブラリが提供する API に着

[1] NTT セキュアプラットフォーム研究所
NTT Secure Platform Laboratories
[a)] fumihiro.kanei.sw@hco.ntt.co.jp

目し，それらの API のユーザビリティの検証や改善に向けた検討が行われている．文献 [7,8] では，ソフトウェア開発者へのインタビューにより開発現場で実際に実施されているセキュア開発のための慣習を調査した上で，それとパブリックなセキュアコーディングガイドとのギャップを明らかにしている．また，文献 [9] では，ソフトウェア開発者を対象としたオンラインサーベイに基づいて，昨今の開発現場におけるセキュリティ戦略の傾向や，ソフトウェア開発者がセキュリティに注意を払う上でのモチベーションと阻害要因を明らかにしている．

上記のようなソフトウェア開発者に着目した研究が行われている一方で，既存研究ではソフトウェア開発現場や開発物が持つ特性がセキュリティに与える影響はこれまで十分に調査されていない．ソフトウェア開発現場において，しばしばセキュリティは明確な要件として記述されない非機能要件として扱われ，より明示的な要件との競合により軽視されやすい事が知られている [10,11]．このようなセキュリティとその他の要件の競合は様々な原因で発生する．一部の文献 [9,12] では，このような競合の原因の実態調査が行われており，主にセキュリティ対策にかかるリソース（時間，人材，予算 など）の不足について考察が行われている．一方で，複数人のプロジェクトで行われるような開発現場にフォーカスを当てた上で，プロジェクトの契約関係，開発物の種別，利用形態などの開発現場や開発物の持つ特性が，ソフトウェア開発者や開発されるソフトウェアのセキュリティにどのような影響を与えるかはこれまでに定量的に調査されていない．

開発現場や開発物の特性に加えて，プロジェクトでの開発に関わる人の役割に着目した調査についても十分に行われていない．プロジェクトで行われる開発には，プログラマやテスターなどの開発者（デベロッパー）の他に，それら開発者を管理する管理者（マネージャー）も関わっているが，多くの既存研究ではデベロッパーを対象とした調査を行っておりマネージャーに着目した分析は十分に行われていない．チーム/プロジェクトで行われる開発現場においては，コーディングやテストなどを実施するデベロッパーと，プロジェクトのスケジュール管理や各種リソースのマネジメントなどを実施するマネージャーで役割が分担されている．特に，開発物のセキュリティを担保する為の各種対策（セキュアコーディングガイドの利用, 脆弱性検査ツールの利用など）を導入する上で，プロジェクトのマネージャーは，それら対策のコストや効果を正しく認識した上で，導入判断を行う必要がある．セキュアなソフトウェア開発を行う上でマネージャーのセキュリティ意識や行動は重要な要素である一方で，具体的なマネージャーのセキュリティ意識，またデベロッパーとマネージャーの間の認識のギャップがセキュリティにどのような影響を及ぼすかの量的な分析は十分に行われていない．

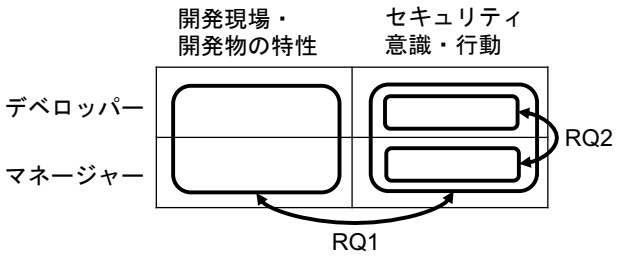

図 1 アンケートにおける質問内容と RQ の関係

上記を踏まえ，本研究では以下の 2 つの研究的問い（RQ）に答えるための調査を実施する．

- **RQ1**：ソフトウェア開発における開発現場/開発物の特性は，ソフトウェア開発者のセキュリティ意識や行動にどのような影響を与えるか？
- **RQ2**：デベロッパーとマネージャーの間でセキュリティ意識や行動にギャップが存在するか？また，それらはソフトウェアのセキュリティにどのような影響を与えているか？

これらの結果により，ソフトウェア開発者向けのセキュリティ対策技術において考慮すべき開発現場の課題が明らかとなり，実際の開発者にとって “利用しやすい” セキュリティ対策技術の設計につながる事が期待できる．

RQ を解明する為に，我々は 375 名のソフトウェア開発者を対象とするオンラインアンケート調査を実施した．図 1 に本調査における質問内容と RQ の関係を示す．アンケートは，(1) 開発現場/開発物の特性に関する質問，および（2）ソフトウェア開発者のセキュリティ意識/行動に関する質問から構成されており，我々は各質問項目の相関を元に分析を実施した．アンケートはデベロッパーとマネージャーの 2 つのグループに対して実施し，各グループにおける回答の傾向や差異，類似性などを分析した．

本論文における貢献は以下の通りである．

- プロフェッショナルのソフトウェア開発者を対象としたオンラインアンケートを実施し，開発現場/開発物の特性および参加者のセキュリティ意識や行動について定量的に調査した．
- 開発現場/開発物の特性に関する回答とセキュリティ意識/行動に関する回答の比較分析の結果から，ソフトウェアを利用するユーザ（不特定多数，限られた特定のユーザ）や，プロジェクトの契約関係（自社開発，受託開発）などの特性は，ソフトウェア開発者のセキュリティ意識や行動に強く影響を与えている事を確認した．
- 契約関係の下位に位置する開発プロジェクトは裁量でセキュリティ対策を実施しにくいことが明らかになったため，契約元や上位の開発プロジェクトにアプローチすることが重要である．
- デベロッパーとマネージャーのセキュリティ意識と行

動に関する回答の分析から，セキュリティ対策を実施する上での意思決定権の有無や意思決定の困難さが，セキュアなソフトウェア開発を妨げる影響が強い要因である事を明らかにした．

- デベロッパーはプロジェクト全体でのセキュリティ対策の実施状況を把握できていない傾向がある事を明らかにした．
- ソフトウェア開発者向けのセキュリティ対策技術を検討する上で，意思決定権を持つマネージャーにアプローチする事が重要であり，今後は意思決定プロセスを補助するような技術が求められる．

2. 関連研究

ソフトウェア開発者を対象としたユーザブルセキュリティの研究として，開発者や開発者を取り巻く環境とセキュリティとの関係の調査を目的とした研究 [9,13] が挙げられる．Assal らは，ソフトウェア開発者へのオンラインサーベイにより，開発者のセキュリティに対する方針，セキュリティ対策を実施する動機，およびセキュリティを妨げる要因について調査を行った [9]．また，ソフトウェアの開発手法，開発者の所属する会社の規模，テスト駆動開発の実施状況が開発者のセキュリティ意識や行動に与える影響についても合わせて調査を行っている．Palombo らは，実際のソフトウェア会社における開発業務に参加しながらセキュリティに関連する開発者の行動や認識を調査することで，脆弱性が発生する原因や対処方法の傾向について質的な分析を行った [14]．

本研究は，ソフトウェアの利用ユーザや開発プロジェクトの契約関係など，これまでにセキュリティとの関連が定量的に評価されていない特性を分析対象としている点，またデベロッパーとマネージャーの違いに着目し，両者の間の意識/行動のギャップがセキュリティに与える影響を定量的に分析している点が既存研究と異なっている．

3. 調査方法

3.1 アンケートの設計

我々のアンケートは，(1) 参加者属性に関する質問，(2) 開発現場/開発物の特性に関する質問，(3) セキュリティ意識/実践内容に関する質問の 3 つの大項目で構成される．本アンケートは，既存研究 [9] のアンケートをベースとして，我々が追加した開発現場の特性やセキュリティ意識/行動の観点を含めて調査ができるよう修正を加えたものである．参加者属性に関する質問には，参加者の基本属性（年齢, 性別, 国籍 など）に加え，ソフトウェア開発の経験年数や所属する企業/開発プロジェクトの構成に関する質問が含まれる．開発現場/開発物の特性に関する質問には，参加者が所属する開発プロジェクトや開発しているソフトウェアの特性に関する質問が含まれる．セキュリティ意識/実践内容に関する質問には，ソフトウェア開発に関する参加者のセキュリティ意識や，実践しているセキュリティ対策の内容，セキュリティを担保する上での阻害要因に関する質問が含まれる．具体的な質問の項目については， 3.2 節にて説明を行う．アンケートの作成にあたっては，回答者の負担を減らすため，我々は質問紙レビューとパイロット調査を繰り返して質問紙の質の向上に努めた．パイロット調査は，計 6 名の開発エキスパートの協力を得て実施した．

3.2 質問項目

3.2.1 開発物と開発現場の特性に関する質問

我々は，ソフトウェアのセキュリティに影響を与える可能性がある特性として，ソフトウェアが利用される業界，ソフトウェアが対象とするユーザ，プロジェクトの契約関係，開発手法の 4 つの要素に着目し質問項目を作成した．ユーザに関しては，参加者が開発しているソフトウェアが対象とするユーザが不特定多数であるか，もしくは限られた特定のユーザであるかを尋ねた．契約関係については，参加者のプロジェクトが自社開発，受託開発のどちらに該当するか，また受託開発の場合元請けであるか，下請けであるかを尋ねた．なお，自社開発とは自社が提供するサービスの為のソフトウェア開発を指し，受託開発とは，他社からの依頼に基づいたソフトウェア開発を指す．また，元請けとはサービスを提供する顧客から直接開発の依頼を受けている場合であり，下請けとは元請けやその他の下請けから開発の依頼を受けている場合である．開発手法については，参加者のプロジェクトで採用されている開発手法が，ウォータフォール，アジャイル，ウォータフォールとアジャイルのハイブリットのどれに該当するか尋ねた．

上記のうち，ソフトウェアが利用される業界，ユーザ，プロジェクトの契約関係については既存研究では調査されておらず，我々が新たに追加した観点である．

3.2.2 セキュリティ意識および実践内容に関する質問

我々は，参加者セキュリティ意識および実践内容を調査するために，表 1 に示す質問を行った．

質問 R1 はセキュリティにかけるリソースに関する質問である．我々は，プロジェクトの持つ開発リソース（時間, 予算，人員 など）全体のうち，セキュリティ対策にかけるリソースの割合を質問した．また，質問 E1–E15, 質問 A1–A5, 質問 C1–C11 は，それぞれ実践しているセキュリティ対策，セキュリティ意識，セキュリティを阻害する要因に関する質問である．これらの質問では，各記述に対してどの程度同意できるかを 5 段階のリッカート尺度で質問した．ただし，セキュリティ対策に関する各質問においては，参加者が記述されたセキュリティ対策の実施状況を把握していないケースを踏まえて，リッカート尺度の選択肢と合わせて「分からない」の選択肢を設置した．また，参加者が表 1 に含まれないその他のセキュリティ対策を実践

表 1　ソフトウェア開発におけるセキュリティに関するアンケート項目

セキュリティにかけるリソースに関する質問 （0〜100 の数値）
R1. あなたのプロジェクトでソフトウェア開発工程全体にかけるリソースのうち，セキュリティに対する取り組みにかけるリソースは何%を占めますか？
セキュリティ対策に関する質問 （5 段階リッカート尺度，強く同意できる - 全く同意できない）
E1. 私のプロジェクトでは，ソフトウェアが満たすべきセキュリティ要件を定義している．
E2. 私のプロジェクトでは，ソフトウェアに対して発生しうる攻撃についてリスク分析を行っている．（例: 攻撃経路の洗い出し，攻撃手法の調査など）
E3. 私のプロジェクトでは，ソフトウェア開発時に発生しうる脆弱性についてリスク分析を行っている．（例: 重要データの取り扱い など）
E4. 私のプロジェクトでは，仮に攻撃者によって脆弱性が攻撃された場合でも，被害を軽減できる（最小化できる）ようにソフトウェアを設計している．
E5. 私のプロジェクトには，ソフトウェアを安全に開発するためのドキュメント/チェックリストが存在する．（例: セキュアコーディングガイド など）
E6. 私のプロジェクトでは，セキュリティを担保するために既存のライブラリやフレームワークに用意されたセキュリティ機能（API）を利用している．
E7. 私のプロジェクトでは，セキュアコーディングを行っているかチェックを行うツールを利用している．
E8. 私のプロジェクトでは，開発中のソフトウェアに対してツールによる自動的な脆弱性検査を行っている．
E9. 私のプロジェクトでは，開発中のソフトウェアに対して手動で脆弱性検査を行っている．
E10. 私のプロジェクトでは，コードレビュー時にセキュリティ要件を満たしているかチェックしている．
E11. 私のプロジェクトでは，内部のセキュリティ診断チームを有している．
E12. 私のプロジェクトでは，外部のセキュリティ診断サービスを利用している．
E13. 私のプロジェクトでは，公開されている最新の脆弱性情報を継続的にチェックしている．（例：CVE/JVN など）
E14. 私のプロジェクトでは，セキュリティに関する問題が発見された場合スケジュールを延長することができる．
E15. 私のプロジェクトでは，セキュリティを担保するためのタスクをそれ以外のタスクより優先させている．
セキュリティ意識に関する質問 （5 段階リッカート尺度，強く同意できる - 全く同意できない）
A1. 私のプロジェクトにとって，ソフトウェアセキュリティは重要なトピックであると思う．
A2. 私のプロジェクトが開発しているソフトウェアは，攻撃者にとって攻撃が容易であると思う．
A3. 私のプロジェクトが開発しているソフトウェアは，攻撃者にとって攻撃による利益（インセンティブ）が大きいと思う．
A4. 私のプロジェクトが取り組むセキュリティ対策は十分であると思う．
A5. 私のプロジェクトの開発物において脆弱性は生じないと思う．
セキュリティ阻害要因に関する質問 （5 段階リッカート尺度，強く同意できる - 全く同意できない）
C1. 私のプロジェクトでは，ソフトウェアのセキュリティを担保するための十分な時間が無い．
C2. 私のプロジェクトでは，ソフトウェアのセキュリティを担保するための十分な知識が無い．
C3. 私のプロジェクトでは，ソフトウェアのセキュリティを担保するための十分な人員がいない．
C4. 私のプロジェクトでは，ソフトウェアのセキュリティを担保するための十分な予算が無い．
C5. 私のプロジェクトでは，その他の要件の優先順位が高いためセキュリティを後回しにしている．
C6. 私のプロジェクトの顧客は，セキュリティを必要な要件としていない．
C7. 私のプロジェクトは，開発しているソフトウェアの要件に合うセキュリティ対策技術（ツール，フレームワーク，開発手法 など）を把握していない．
C8. 私のプロジェクトでは，セキュリティ対策技術（ツール，フレームワーク，開発手法 など）の導入判断に時間や手間がかかる．
C9. 私のプロジェクトにとって，既存のセキュリティ対策技術（ツール，フレームワーク，開発手法など）はセキュリティを向上させる上では有用ではあるが，実装/運用のコストの観点からは費用対効果が低い．
C10. 私には，セキュリティ対策技術（ツール，フレームワーク，開発手法 など）を導入する意思決定権が無い．
C11. 私のプロジェクトには，長年取り組んできた開発プロセスがあるため，セキュリティのためにそれを変更するのは困難である．

している場合や，その他のセキュリティ阻害要因に直面している場合を想定し，それらを回答する為の自由記述設問も合わせて設置した．

3.3　参加者募集

我々は，サーベイ企業 [15] のサービスを用いて参加者を募集し，2020 年 8 月に調査を実施した．調査では，サーベイ企業の保有する日本の 18 歳以上のモニターに対してスクリーニング調査を実施し，結果に応じて参加者の抽出，及びグループ分けを行った．具体的には，複数人で構成されるソフトウェア開発プロジェクトにて業務を行っており，プロジェクトにおける役職が，デベロッパー（実装/テストなどの開発業務を担当）もしくはマネージャー（スケジューリングなどの管理業務を担当）に該当する参加者を抽出した．ただし，マネージャーを選択した回答者のうちのソフトウェア開発におけるセキュリティ対策を実施する意思決定権を一切持っていない回答者は，スクリーニングにて調査対象から除外された．一般的に，ソフトウェア開発現場では様々な役職のマネージャー（プロジェクトマネージャ，プロダクトマネージャ など）が存在しており，それぞれの立場や権限が異なっている．我々は，開発現場でのセキュリティ対策を実施する上での意思決定権を持つマネージャーのセキュリティ意識/行動が，開発物のセキュリティに与える影響が大きいと考え，そのような権限を持たない立場のマネージャーは研究のフォーカス外とした．スクリーニング質問を通過し本調査に回答した全ての参加者には，1,000 円相当の報酬が支払われた．この報酬額はパイロット調査におけるアンケートの回答時間に基づいて算出した最低賃金を優に超える額である．

我々は回答データの質を担保する為に，以下のような場合に回答を分析対象から除外した：回答者が注意力チェックの設問に適切に回答していない場合，矛盾する回答がある場合，回答時間が 5 分より短い場合，自由記述に不適切な回答が書かれている場合．

4. 分析結果

本章では，（RQ1）ソフトウェア開発における開発現場および開発物の特性がソフトウェア開発者のセキュリティ意識や行動に与える影響，および（RQ2）デベロッパーとマネージャーの間にあるセキュリティ意識や行動のギャップ，についての分析結果を説明する．なお，本調査では全ての質問を任意回答としているため，空欄の回答は全て無視して分析を行った．

4.1 参加者属性

今回の調査では， 3.3 節で説明した対象者のスクリーニング，及び無効回答の除外の結果，最終的にデベロッパー 185 名，マネージャー 191 名の合計 375 名からの回答を分析対象とした．有効な回答を行ったアンケート参加者の平均回答時間は 28.5 分（Md=10.1）であった．参加者には，20 代から 60 代までの世代のソフトウェア開発者が含まれていた．また，参加者のソフトウェア開発経験は，平均 19.5 年（Md=20）であり，デベロッパーにおいては平均 17.2 年（Md=18），マネージャーにおいては平均 21.6 年（Md=20）であった．参加者が所属する会社には従業員が 100 名以下の中小企業から，1,000 人を超える大企業まで様々な規模の会社が含まれていた．

4.2 RQ1. 開発現場および開発物の特性とセキュリティ

開発現場および開発物の特性に着目し，それらがソフトウェアのセキュリティに与える影響について調査した．具体的には，開発現場および開発物の特性に関する質問の結果ごとに回答をグループ分けし，グループごとにセキュリティに関する質問の回答を比較した．本分析では，回答に有意差があるか検定する際に Mann-Whitney の U 検定を利用した．また検定における有意水準は 0.05 とした．

我々は，3.2.1 節で説明した開発現場および開発物に関する 4 つの特性の中で，ソフトウェアのユーザ，およびプロジェクトの契約関係が，ソフトウェア開発者のセキュリティ意識，行動に強く影響を与えている事を確認した．以降の節では上記の 2 つの特性についての分析結果を述べる．また，セキュリティに与える影響が小さかった 2 つの特性（ソフトウェアの利用業界，開発手法）の分析結果は紙面の都合上割愛する．

4.2.1 ソフトウェアが対象とするユーザ

我々は，ソフトウェアが対象とするユーザに関する質問の回答について，不特定多数のユーザに利用される場合（N=118，32%）と，限られた特定のユーザに利用される場合（N=256，68%）に回答をグループ分けして分析した．グループ間でのセキュリティに関する質問の比較結果を図 2 に示す．なお，図 2 における各質問の平均スコアは，リッカート尺度の選択肢に+2（強く同意できる）から-2（全く同意できない）までのスコアを割り振った上で質問ごとのスコアの平均値を算出した値であり，スコアが大きい程，質問項目に対して同意できる傾向が強い事を表している．質問ごとにグループ間の回答の差異を検定した結果，セキュリティ対策に関する質問 E5–E13，セキュリティ意識に関する質問 A1，セキュリティ意識に関する質問 C7，C9 のそれぞれにおいて有意差を確認した．

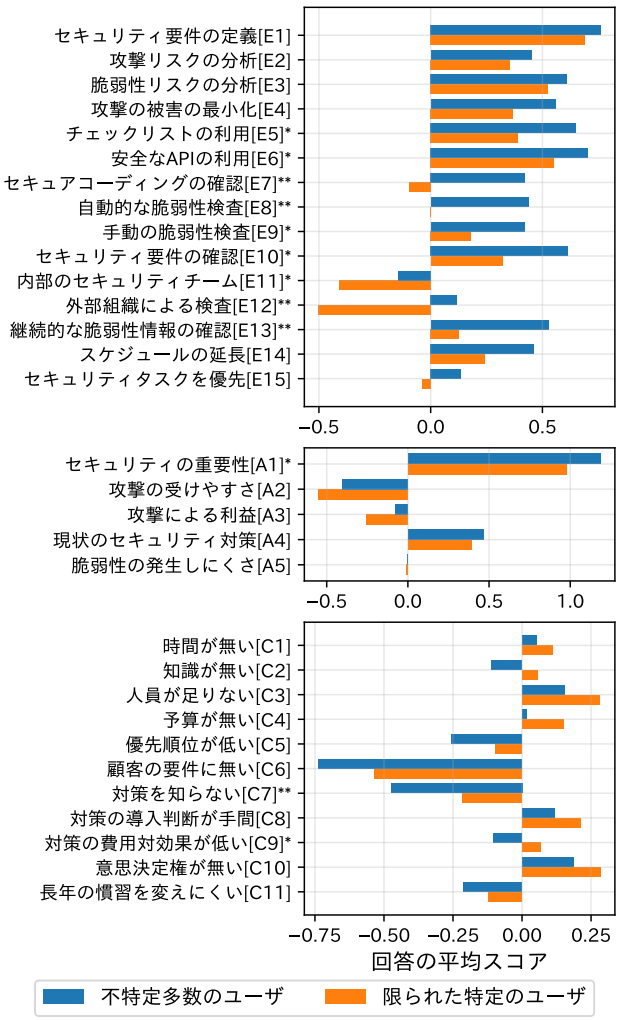

図 2 ソフトウェアのユーザごとの回答結果の比較（セキュリティ対策，セキュリティ意識，セキュリティ阻害要因）*有意差あり（p<.05），**有意差あり（p<.01）

上記の結果から，ソフトウェアがどのようなユーザに利用されるかは，開発物のセキュリティに対して影響を与えていると言える．特に，限られたユーザにしか利用されないソフトウェアを開発しているプロジェクトでは，セキュリティが軽視されがちであり，セキュリティ対策が不十分な傾向があると言える．

4.2.2 プロジェクトの契約関係

我々は，契約関係に関する質問の回答に基づいて，自社開発（N=110，30%）と受託開発（N=259，70%）に回答をグループ分けして分析した．また，受託開発の回答については，さらに元請け（N=148，58%）と下請け（N=109，

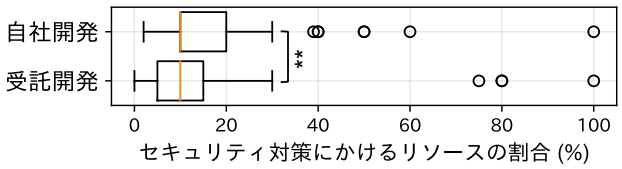

図 3 契約関係ごとの回答結果の比較（セキュリティにかけるリソース） **有意差あり（p<.01）

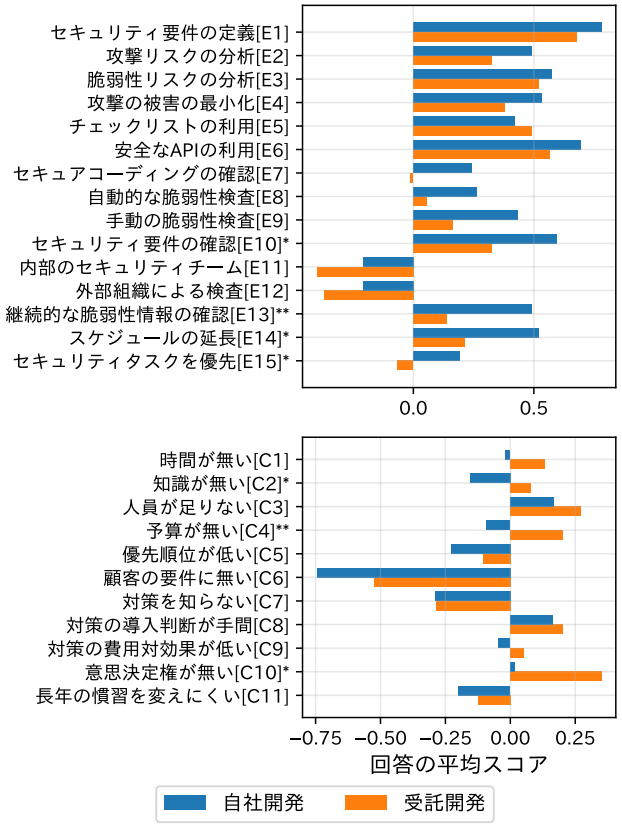

図 4 契約関係ごとの回答結果の比較（セキュリティ対策，セキュリティ阻害要因）*有意差あり（p<.05），**有意差あり（p<.01）

42%）の 2 つのグループに分けて分析を行った．

自社開発，および受託開発の間でのセキュリティに関する質問の比較結果を図 3，図 4 に示す．図 3 はセキュリティにかけるリソースの割合に関する質問 R1 の結果を表している．また，図 4 は 4.2.1 節の図と同様の方法で作成された．自社開発と受託開発を比較すると，セキュリティにかけるリソースに関する質問 R1，セキュリティ対策に関する質問 E10，E13–E15，セキュリティを阻害する要因に関する質問 C2，C4，C10 で有意差が確認された．有意差が確認された質問の内容から，自社開発の方がセキュリティに多くのリソースを掛けており，多くのセキュリティ対策を実施している事が分かる．また，受託開発は自社開発と比較して，予算不足や知識の不足，およびセキュリティに関する意思決定権が無い事に対する制約を感じている事が分かる．元請けと下請けを比較すると，セキュリティ対策に関する質問 A4，及びセキュリティを阻害する要因に

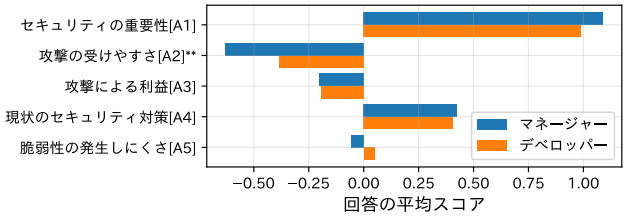

図 5 デベロッパー，マネージャーのセキュリティ意識（A1–A5）の比較 **有意差あり（p<.01）

関する質問 C7，C10 において有意な差が確認された．質問 A4 においては，元請けの方が肯定的な回答をしており，質問 C7，C10 においては下請けの方が肯定的な回答をしている傾向があった．つまり，下請けの方がセキュリティ対策が不十分であると感じており，セキュリティ対策を把握しておらず，かつ意思決定権に関する制約を強く感じていると言える．

上記の結果から，ソフトウェア開発に関わる契約関係は，開発物のセキュリティに対して影響を与えていると言える．特に，質問 C10 は自社開発と受託開発の間，および元請けと下請けの間の両方で大きな差異が見られた事から，契約関係の下位に位置する開発プロジェクトの方が，契約元の要求や優先事項により自分たちのの裁量でセキュリティ対策を実施できない状況にある事が示唆される．上記の裏付けとして，受託開発を選択した参加者のセキュリティを阻害する要因に関する自由記述には，顧客（契約元）からのセキュリティ以外の優先事項が原因となる制約が含まれていた．具体的には，“顧客側が経営陣の求めるサービス開始時期にのみ着目し，その他の要素を軽視している”，“セキュリティをこちらが心配していても，顧客側がその重要性に気付いていないときがある” などである．

4.3 RQ2. デベロッパーおよびマネージャーの間のセキュリティ意識や行動の差異

デベロッパーおよびマネージャーのセキュリティに関する質問の結果を比較分析することで，両者の間にセキュリティ意識と行動に関するギャップが存在するかを調査した．なお，本分析においてデベロッパーとマネージャーの回答に有意差が存在するかは，4.2 節の分析と同様の方法で検定を行った．

4.3.1 デベロッパーとマネージャーのセキュリティ意識

デベロッパーおよびマネージャーのセキュリティ意識に関する質問について 4.2 節と同様の方法で回答の平均スコアを算出した．図 5 に示す質問 A1–A5 の回答結果より，参加者はセキュリティの重要性を認識しているにもかかわらず，自身の開発しているソフトウェアが攻撃されやすいと考えていない，つまりセキュリティに対して楽観的な傾向があるといえる．これらの傾向は，既存研究 [9] における分析の結果と一致する．また，デベロッパーとマネー

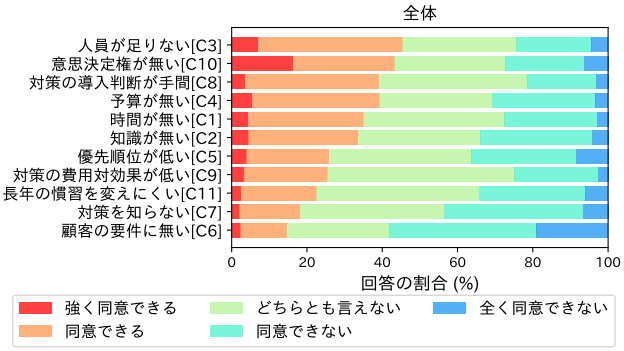

図 6　セキュリティ阻害要因に関する質問（C1–C11）への回答

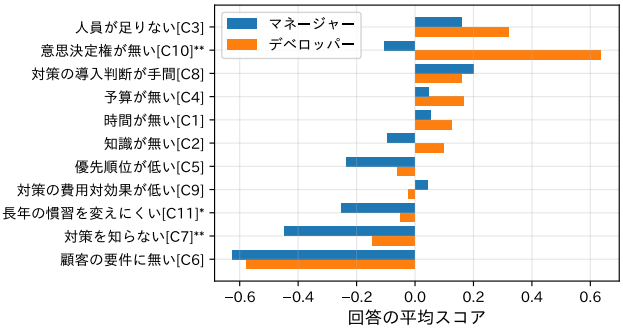

図 7　デベロッパーおよびマネージャーのセキュリティ阻害要因（C1-C11）の比較 *有意差あり（p<.05），**有意差あり（p<.01）

ジャーを比較すると，マネージャーの方がセキュリティの重要性を強く認識しているにも関わらず，質問 A2 においてデベロッパーとマネージャーの間に有意差が確認された事から，上記の傾向はデベロッパーよりマネージャーにおいてより強いといえる．

4.3.2　セキュリティの阻害要因

セキュリティを妨げる要因に関する質問 C1–11 の回答結果を図 6 に示す．なお図 3 におけるグラフは参加者全体での回答の結果を示しており，質問項目は “強く同意できる”，“同意できる” の回答数の合計が多い順に記載されている．既存研究 [9,12] でも指摘されている通り，時間や予算などのセキュリティにかけるリソース不足に関連する制約（質問 C1–C4）は，本調査においても同意できる回答の割合が多い要因である．加えて，我々が新たに質問した，セキュリティ対策を導入する際の意思決定に関する制約（質問 C8，C11）も，同様に同意できる回答の割合が多い事が確認された．これらの項目は今回調査したセキュリティを阻害する要因の中で同意できる割合が上位 3 位に含まれており，リソース不足に起因する制約と同様に，ソフトウェア開発におけるセキュリティを妨げる影響が大きい要因であると言える．

図 7 に，デベロッパーおよびマネージャーにおける質問 C1–C11 の回答の平均スコアを示す．なお，図 7 における質問項目は，図 6 と同じ順序で記載されている．質問 C8，C9 を除いた全ての質問において，デベロッパーの方がマネージャーより強くセキュリティを阻害する要因を認識している事が分かる．また，有意差が確認された質問 C7，C10，C11 の阻害要因は特にその傾向が強いと言える．これらの結果から特筆すべき点として，マネージャーはセキュリティ対策を実施する意思決定権を持っており，かつ既存の開発プロセスをセキュリティの為に変更することに積極的にも関わらず，実際に作業を行うデベロッパーは，マネージャーと比較してセキュリティの為に開発プロセスに変更を加えることに消極的である点が挙げられる．

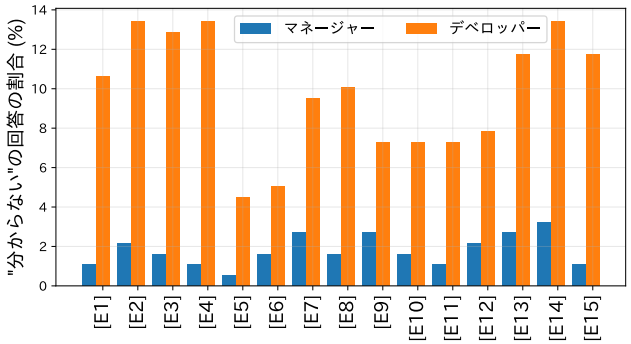

図 8　質問 E1–E15 における “分からない” の回答の割合

4.3.3　プロジェクトの取り組みに関する理解

図 8 にセキュリティ対策に関する質問 E1–E15 の中で “分からない” の回答が選択された数を示す．なお，ここでいう “分からない” の回答とは，回答者が明示的に “分からない” の選択肢を選んだ場合であり，回答者が空欄とした回答は分析に含まれていない点には注意されたい．質問 E1–E15 全体を通して，デベロッパーの方が，マネージャーと比較して “分からない” と回答している数が多い事が分かる．特に，一部の質問においては，14 %近いデベロッパーが “分からない” を選択していた．これらの結果は，デベロッパーはマネージャーと比較して開発プロジェクト全体での取り組みを把握できていない事を示している．

5.　議論

5.1　セキュリティを考慮したソフトウェア開発に向けて

デベロッパーはプロジェクト全体のセキュリティ対策を把握出来ておらず，また対策を実施する意思決定権を有していない事が 4.3.2 および 4.3.3 節の結果から明らかになった．よって，ソフトウェア開発者向けにセキュリティ対策技術を検討する上では，意思決定権を持つマネージャーにアプローチする事が重要であると言える．また，セキュリティ対策にかかるリソース不足の課題と合わせて，対策を実施する上での意思決定の困難さがセキュアなソフトウェア開発を妨げている傾向があることから，今後はマネージャーの意思決定プロセスを補助するような技術が求められる．セキュリティ対策に関する意思決定に必要となる情報には様々なものが考えられるが，例えばセキュリティ対

策がカバーする脅威や対策のコストを正確かつ効果的に提示する事で，意思決定に纏わる困難さを軽減できる可能性がある．また，契約関係の下位に位置する開発プロジェクトは裁量でセキュリティ対策を実施しにくいことが 4.2.2 節で明らになったため，契約元や上位の開発プロジェクトにアプローチすることも考えられる．

5.2 制約事項

今回の調査は参加者の自己申告に基づいている．特にセキュリティに関する質問については，参加者の所属や立場によって社会的に好まれるであろう回答を行う「社会的望ましさのバイアス（Social Desirability Bias，SDB）」の影響を受ける可能性がある．我々は SDB の影響を減らすために，アンケート冒頭で，全ての設問を任意回答であること，調査が匿名で行われる事を説明した．

5.3 研究倫理

本研究は，ICT・サイバーセキュリティの研究倫理原則を記した Menlo Report に従って実施しており，また著者所属組織のプライバシー評価委員会によってアンケート内容や実施手順について承認を得ている．具体的には，実験参加者は事前にアンケート内容について知られされており，自由意志に基づいた同意のもと参加している．また，個人情報の取り扱いは，アンケート実施国の個人情報保護法に準拠した方法に基づいている．

6. まとめと今後の課題

本研究では，プロフェッショナルのソフトウェア開発者を対象としてセキュアなソフトウェア開発を妨げる原因について調査を行った．アンケートの結果の分析により，開発物を利用しているユーザや，開発プロジェクトの契約関係は，ソフトウェア開発者のセキュリティ意識や行動に強く影響を与えている事が明らかになった．特に，契約関係の下位に位置する開発プロジェクトは裁量でセキュリティ対策を実施しにくいことが明らかになったため，契約元や上位の開発プロジェクトにアプローチすることが考えられる．また，デベロッパーとマネージャーのアンケート結果の比較により，デベロッパーはプロジェクトにおけるセキュリティ対策の実施状況を十分に把握しておらず，セキュリティに関する意思決定権を有していない傾向が明らかになった．加えて従来研究で指摘されてきた対策に掛かるコストやその他のタスクとの優先事項との兼ね合いと同様に，セキュリティ対策を実施する意思決定の困難性が，セキュアなソフトウェア開発を妨げる強い要因の一つであることが確認された．これらの結果から，セキュリティ対策を実施する意思決定者であるマネージャーを補助する技術が求められていることが示唆される．例えば，セキュリティ対策を実施する上での意思決定に必要となる情報（対策の効果，コストなど）を正確かつ効果的に示す技術によりセキュアなソフトウェア開発の普及に貢献できる可能性がある．

参考文献

[1] Livshits, V. B. and Lam, M. S.: Finding Security Vulnerabilities in Java Applications with Static Analysis, *Proc. of SSYM* (2005).

[2] Zheng, Y. and Zhang, X.: Path Sensitive Static Analysis of Web Applications for Remote Code Execution Vulnerability Detection, *Proc. of ICSE* (2013).

[3] Sounthiraraj, D., Sahs, J., Greenwood, G., Lin, Z. and Khan, L.: SMV-Hunter: Large Scale, Automated Detection of SSL/TLS Man-in-the-Middle Vulnerabilities in Android Apps, *Proc. of NDSS* (2014).

[4] Zhang, M. and Yin, H.: AppSealer: Automatic Generation of Vulnerability-Specific Patches for Preventing Component Hijacking Attacks in Android Applications., *Proc. of NDSS* (2014).

[5] Oliveira, D. S., Lin, T., Rahman, M. S., Akefirad, R., Ellis, D., Perez, E., Bobhate, R., DeLong, L. A., Cappos, J., Brun, Y. and Ebner, N. C.: API Blindspots: Why Experienced Developers Write Vulnerable Code, *Proc. of SOUPS* (2018).

[6] Gorski, P. L., Iacono, L. L., Wermke, D., Stransky, C., Moeller, S., Acar, Y. and Fahl, S.: Developers Deserve Security Warnings, Too: On the Effect of Integrated Security Advice on Cryptographic API Misuse, *Proc. of SOUPS* (2018).

[7] Assal, H. and Chiasson, S.: Security in the Software Development Lifecycle, *Proc. of SOUPS* (2018).

[8] Acar, Y., Stransky, C., Wermke, D., Weir, C., Mazurek, M. L. and Fahl, S.: Developers Need Support, Too: A Survey of Security Advice for Software Developers, *Proc. of SecDev* (2017).

[9] Assal, H. and Chiasson, S.: "Think Secure from the Beginning": A Survey with Software Developers, *Proc. of CHI* (2019).

[10] Tahaei, M. and Vaniea, K.: A Survey on Developer-Centred Security, *Proc. of EuroUSEC* (2019).

[11] Poller, A., Kocksch, L., Türpe, S., Epp, F. A. and Kinder-Kurlanda, K.: Can Security Become a Routine? A Study of Organizational Change in an Agile Software Development Group, *Proc. of CSCW* (2017).

[12] Xie, J., Lipford, H. R. and Chu, B.: Why do programmers make security errors?, *Proc. of VL/HCC* (2011).

[13] Alomar, N., Wijesekera, P., Qiu, E. and Egelman, S.: "You've Got Your Nice List of Bugs, Now What?" Vulnerability Discovery and Management Processes in the Wild, *Proc. of SOUPS* (2020).

[14] Palombo, H., Tabari, A. Z. and Ou, D. L. J. L. X.: An Ethnographic Understanding of Software (In) Security and a Co-Creation Model to Improve Secure Software Development, *Proc. of SOUPS* (2020).

[15] Macromill Group: `https://group.macromill.com/`.