離散シミュレータを用いた脆弱性対応優先度付けに係るCVSSとSSVCの比較評価

植木 優輝[1,a)]　藤井 翔太[1]　辻 大輔[1]　川口 信隆[1]

概要：脆弱性の増加や法規制の強化に伴い，脆弱性管理がより重要視されている．脆弱性管理においては，全ての脆弱性に対応することが現実的でないことから，対応優先度付けが行われる．この優先度付けには，脆弱性の深刻度を示すCVSSがデファクト的に活用されているが，実際にはCVSSの優先度付けへの利用は設計意図とは異なる運用であり，種々の問題点も指摘されている．こうした状況から，脆弱性対応に係る優先度付けのフレームワークとしてSSVCが提案されているが，依然として少なくない組織がCVSSを利用している．これは，実環境における再現性や有効性が十分に実証されておらず，SSVCへの移行に踏み切る判断が難しいことが一因と推察される．そこで，本稿では，CVSSとSSVCを用いて脆弱性対応影響の離散事象型シミュレーション分析を行い，その有効性・実用性を定量的に評価する．シミュレーション評価では，SSVCを用いた場合，CVSSベーススコアのみを考慮した評価と比較して，半分の要員数で法令期限内の脆弱性対応が可能なことが明らかになった．また，シミュレーション結果をもとに，SSVCの普及に係る検討結果や提言事項をまとめた．本研究における評価結果がサイバーセキュリティの現場での脆弱性の優先度付けタスクにおけるSSVC採用可否の検討に係る一助になれば幸いである．

キーワード：脆弱性，優先度付け，CVSS，SSVC

A Comparative Evaluation of CVSS and SSVC for Vulnerability Response Prioritization Using Discrete Simulator

Yuki Ueki[1,a)]　Shota Fujii[1]　Daisuke Tsuji[1]　Nobutaka Kawaguchi[1]

Abstract: With the growing number of vulnerabilities and the tightening of regulatory requirements, vulnerability management has become increasingly critical. Since addressing all vulnerabilities is impractical, prioritization is necessary. The Common Vulnerability Scoring System (CVSS) is widely used as a de facto standard for prioritization; however, its use in this context deviates from its original design intent, and several limitations have been noted. To address these issues, the Stakeholder-Specific Vulnerability Categorization (SSVC) framework has been proposed, yet many organizations continue to rely on CVSS. One reason is the limited empirical evidence demonstrating SSVC's reproducibility and effectiveness in real-world environments, which makes transitioning difficult. In this paper, we conduct a discrete-event simulation analysis comparing CVSS and SSVC to quantitatively evaluate their effectiveness and practicality in vulnerability response. The results show that, compared with prioritization based solely on CVSS base scores, SSVC enables vulnerability remediation within regulatory deadlines with only half the personnel. Based on these findings, we present considerations and recommendations for promoting the adoption of SSVC. We expect that the results of this study will support decision-making on the feasibility of employing SSVC for vulnerability prioritization tasks in cybersecurity practice.

Keywords: vulnerability, prioritization, CVSS, SSVC

1. はじめに

脆弱性は，システムやソフトウェアに存在する欠陥や弱点のことであり，これを悪用されると情報漏洩やシステム破壊といった重大な被害を招く可能性がある．近年の脆弱性の急増や，公開から 2 週間以内の脆弱性対応を求める BOD22-01 [1] 等の法規制の強化により，組織における迅速かつ効果的な脆弱性対応が求められている．しかし，全ての脆弱性に対応することは現実的に困難であり，限られたリソースを効率的に活用するためには，脆弱性の優先度付けが不可欠となっている [2].

脆弱性の優先度付けを行うための代表的な手法として，共通脆弱性評価システム（CVSS: Common Vulnerability Scoring System）が広く利用されている．CVSS は，脆弱性の深刻度をスコアとして定量化し，組織が対応すべき脆弱性を選定する指針を提供する．しかし，CVSS は脆弱性の技術的な深刻度を評価するための仕組みであり，そのスコアをそのまま優先度付けに使用することは，設計意図とは異なる運用である [2–4]．このような運用は対応の緊急性やリスクを正確に反映できない可能性があり，セキュリティの現場では課題として挙げられている．例えば，CVSS スコアが高い脆弱性でも，実際には攻撃の可能性が低い場合や組織内でのリスクは小さい場合があり，その場合 CVSS をもとにした対応は非効率となる可能性がある．

これに対し，脆弱性の深刻度ではなく脆弱性の対応優先度付けを行うことを目的としたフレームワークとして，ステークホルダー固有の脆弱性分類（SSVC: Stakeholder-Specific Vulnerability Categorization [5]）が近年注目されている．SSVC は，脅威と環境要因を考慮した脆弱性対応における意思決定プロセスをステークホルダーごとに提供し，脆弱性の技術的要因だけでなく，組織特有の要因やリスクを統合的に考慮することを特徴としている．このフレームワークは，脆弱性の対応優先度を Immediate（即時対応），Out-of-Cycle（迅速対応），Scheduled（定期対応），および Defer（不対応）といった具体的なカテゴリに分類する．これにより，説明性のある対応方針を提供するため，脆弱性の深刻度から優先度付けを決定する意思決定プロセスを効率化でき，属人性の抑制も期待できる．

他方で，SSVC の実用性や有効性については，十分な実証が行われているとは言い難い．特に，SSVC の導入が実際の業務プロセスにどのような効果をもたらすのか，またその運用が従来の手法と比較してどれほど効率的なのかという点について，明確なデータや指針が不足しているという課題が存在する．この結果，依然として少なくない組織が脆弱性の対応優先度付けに CVSS を用いている [2,6].

そこで本研究では，CVSS と SSVC の実用性を比較するために，離散事象型シミュレータを用いた脆弱性対応プロセスのシミュレーション分析を行う．具体的には，脆弱性の対応優先度付けに CVSS ベーススコアと SSVC を用いた場合を比較し，両者の効率性や法令遵守の達成度を定量的に評価する．これにより，脆弱性の優先度付けに際して，組織の規模や捻出可能なコスト等から，CVSS と SSVC の何れを用いるのが良いかの参考とすることを目指す．

本研究の主な貢献は以下の通りである．

- 脆弱性の対応に係る離散事象型シミュレータを実装し，SSVC と CVSS のそれぞれで優先度付けを実施した際の影響を定量的に示した．
- 我々が想定する環境において，SSVC による脆弱性評価は，脆弱性の深刻度のみを評価する CVSS ベーススコアを用いた評価と比較して，半分の CSIRT 要員数で法令期限内の脆弱性対応が可能なことを示した．
- シミュレーション結果をもとに，今後の脆弱性優先度付けや研究の方向性について提言事項をまとめた．

2. 背景

2.1 脆弱性への対応

脆弱性が発見された際の典型的な対応として，自組織の機器が影響を受けるかを調査し，影響を受ける場合はパッチを適用することによって修正を図ることが挙げられる [2,6–8]．これにより，自組織に内在する脆弱性を緩和・解消し，セキュリティの担保を図る．

他方で，脆弱性の数は年々増加しており，複雑化する IT 環境も相まって，全ての脆弱性に対して即時的にパッチを適用することは現実的ではない．例えば，アメリカ連邦政府に係る CISA（Cybersecurity and Infrastructure Security Agency）の KEV（Known Exploited Vulnerabilities）[9] で設定されている 3 週間以内に完遂できたものは 62.2%に留まるという調査も存在する [8]．こうした状況から，対応する脆弱性に優先度付けを行い，より優先度の高い脆弱性に対してパッチを適用することが一般的に行われる [2,6–8].

2.2 脆弱性の優先度付けに用いられる指標

前述の通り脆弱性の優先度付けは重要なタスクの一つであり，様々な指標の提案や研究の実施が進められている [10]．本節では，代表的な脆弱性の優先度付け手法，具体的には，デファクト的に活用されている [2,6] CVSS と後発の SSVC について述べる．

2.2.1 CVSS

CVSS は，ソフトウェアの脆弱性の特性と重大度を伝えるためのオープンフレームワークであり，脆弱性の主要な特徴を捉え，その深刻度を反映する数値スコアとそのスコアのテキスト表現を生成する方法を提供する．2025 年 8 月現在の最新版である CVSS v4.0 では，基本評価基準，脅威評価基準，環境評価基準の 3 つの基準で脆弱性を評価する．

[1] 株式会社 日立製作所
[a)] yuki.ueki.pr@hitachi.com

CVSS は脆弱性の深刻度を適切にスコアリングできる一方で, 同スコアは脆弱性の優先度付けには適していない [2]. まず，CVSS は脆弱性の深刻度を測るものであり，リスク評価に直接使用するのは誤用であると CVSS の開発者は主張している [3,4]．また，Allodi [11] らは，脆弱性の優先順位付けにおいて，CVSS スコアよりもランダムな優先付けの方が優れた結果になる場合があることを実証している．

2.2.2 SSVC

SSVC は，脆弱性対応に関する意思決定プロセスやステークホルダーの違いを反映した脆弱性分類を実現するフレームワークであり，CISA によってガイドライン化されている．SSVC では，以下 4 つの Decision Points を持つ決定木を用いて脆弱性の対応方針を決定する．なお，各 Decision Points で判定される結果の詳細については，紙面の都合上割愛するため，文献 [5] を参照されたい．

- **Exploitation（悪用実績）**
  攻撃コードの公開有無や脆弱性の悪用実績の存在により，Active, PoC, None の何れかで評価する
- **Exposure（露出状況）**
  インターネットに接続しているなど，資産の外部への露出度により Open, Controlled, Small の何れかで評価する
- **Automatable（自動化可否）**
  攻撃の自動化可能性に応じて，Yes，No のどちらかで評価する
- **Human Impact（安全性およびミッション影響）**
  安全への影響（Safety Impact）と，業務に不可欠な機能への影響（Mission Impact）の 2 項目の評価によって決定される．Safety Impact では，経済的，環境的，金銭的, 心理的の 4 つの観点で評価を行い，None, Minor, Major, Hazardous, Catastrophic の何れかで評価する．Mission Impact では，業務について，None, Degraded, MEF support crippled, MEF failure, Mission failure の何れかで評価する．

決定木により出力される対応方針は Immediate，Out-of-Cycle，Scheduled，Defer の 4 つである．SSVC では意思決定プロセスの設計目標として，意思決定に関与する者を明確に定義すること，証拠の分類を適切に使用すること，信頼性の高い証拠に基づくこと，透明性があること，説明可能であることを挙げている．また，数値での出力は解釈を要することや，統計的重なりの解消のために定性的な出力となっている．このような特徴から SSVC は脆弱性の深刻度ではなく，脆弱性に関する意思決定の手法として有用であるといえる．

2.3 研究課題

前述の通り，CVSS は脆弱性の対応優先度付けの文脈においては，その利用が必ずしも最適であるとは言い難い．他方で，いまだに脆弱性の対応優先度付けに CVSS が用いられている組織も存在する [2,6]．これは，SSVC をはじめとする様々な手法が提案されているものの，ホスト構成やネットワークアーキテクチャといった要素が考慮されておらず，実環境における再現性や有効性が十分に実証されていないこと [10] が一因であると考えられる．また，既に CVSS を利用している状況も相まって，移行コストに対して上述の効果が不明瞭であるという点も，CVSS の継続利用をさらに助長していると考えられる．

こうした背景を受けて，本研究では，脆弱性の対応優先度付けにおける CVSS と SSVC の定量的な評価を試行する．具体的には，以下の研究課題（Research Questions，以降 RQs）を制定し，その解決を図る．

RQ1. 脆弱性対応の優先度付けについて，CVSS と SSVC でどのような差異が生じるか?

RQ2. 脆弱性の評価手法として，CVSS と SSVC をどのように選択すべきか?

まず，SSVC と CVSS のそれぞれについて，同条件下で脆弱性の優先度付けを行った場合，どのような結果になるか定量的に評価する．この際，想定した運用環境における脆弱性対応フロー内において，脆弱性が滞在する時間や，対応中の脆弱性より優先度の高い脆弱性が発生したがために生じる，フロー内での待機時間を比較評価する（**RQ1**）．

さらに，上記評価の結果をもとに，優先度付けの有用性に加えて，算出コストや組織編成に必要な人員等，実践的に利用するに際して考慮する必要のある観点から，どのような場合・組織において CVSS と SSVC のどちらを利用するのが望ましいかの検討を行う（**RQ2**）．

本研究の全体像をまとめると，**図 1** の通りである．

3. 脆弱性対応影響評価のための離散事象型シミュレータを用いた評価

3.1 概要

脆弱性対応のプロセスは，脆弱性の発覚や公開，組織内での探索，対応優先度付け，対応方針の決定，対応といった流れで行われるのが一般的である [2,5–7]．これら一連の事象は連続的に変化するのではなく，特定の時点で状態が変化するイベントを持つ離散的な状態遷移の連鎖として表現可能である．したがって，本評価では離散事象型シミュレーションを用いて，脆弱性が検出してから対応されるまでの脆弱性の状態を観測する．

CVSS を用いた脆弱性の優先度付けでは，投入する脆弱性データから CVSS ベーススコアを抽出し，この値を優先度として用いる．例えば，ベーススコアが 2.0 の脆弱性が脆弱性対応フロー内に存在しているとき，ベーススコアが 8.0 の脆弱性が公開されたとする．この時，対応中のベーススコアが 2.0 の脆弱性は保留され，新規に公開された，よりベーススコアの高い 8.0 の脆弱性が優先して対応される．

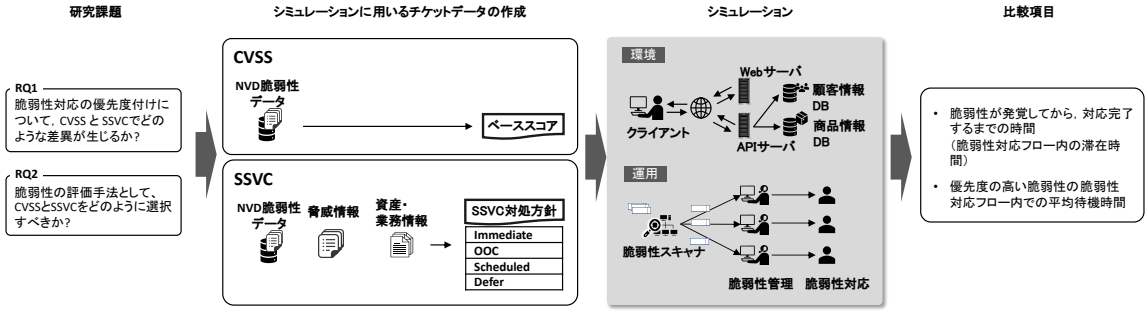

図 1　本研究の全体像

表 1　評価での想定環境における構成情報

	Web サーバ	API サーバ	顧客情報 DB	商品情報 DB
ソフトウェア	Apache HTTP server	Python	MySQL	MongoDB
SSVC Exposure	Open	Controlled	Small	Small
業務プロセス	サービス提供	サービス提供	顧客情報管理	商品情報管理

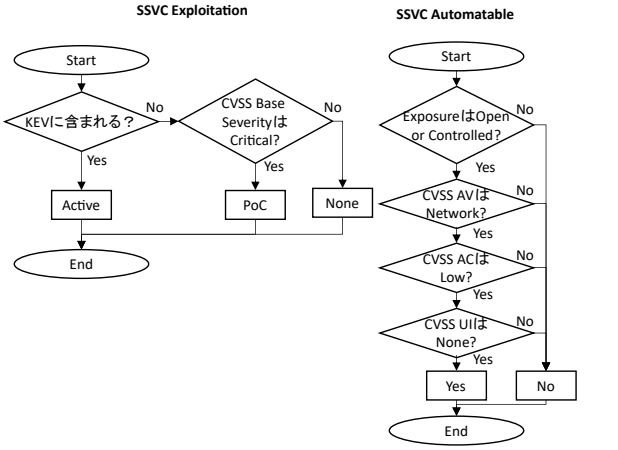

図 2　Vulnrichment に情報が含まれていない脆弱性の Exploitation, 及び Automatable の算出手法

SSVC を用いた脆弱性の優先度付けでは，事前に設定した資産・環境情報と，脆弱性に関する脅威情報を用いる．図 1 において，より上にある対象方針（最上段は Immediate）の方が対応優先度が高いとみなす．

3.2　想定環境

シミュレーションを実施するにあたり，我々は利用者数十万人程度の E コマースサービスを提供する組織を想定し，サービスを提供するシステムを定義した．

表 1 に想定環境における構成情報を示す．本環境は，図 1 の「シミュレーション」における「環境」に示したものである．想定した組織は，クライアントユーザに E コマースサービスを提供する Web サーバ，クライアントサイドからの通信を取得してデータベースへアクセスする API サーバ，顧客情報を管理するデータベース，および商品情報を管理するデータベースを有することとした．なお，表 1 の業務プロセスの行では，各資産が持つ業務プロセスを示している．これは，3.3.3 項で後述する Human Impact の評価が業務ごとに実施されるため設定したものである．

3.3　想定環境における SSVC の評価

2.2.2 項で述べた通り，SSVC では，Exploitation, Exposure, Automatable, および Human Impact の 4 つの Decision Points の評価が必要である．このうち，Exploitation と Automatable は脆弱性ごとに決まる Decision Points であり，残る Exposure と Human Impact は環境に依存するため事前に評価しておく必要がある．本節では，脆弱性ごとの Exploitation と Automatable の評価方法と，想定環境における Exposure と Human Impact の事前評価結果を述べる．

3.3.1　Exploitation と Automatable の評価

先述の通り，Exploitation と Automatable は脆弱性ごとに決定するものである．一部の脆弱性における両者の評価結果は，CISA が公開している Vulnrichment プロジェクト [12] から取得可能であり，これを用いた．取得できない脆弱性については，SSVC [5] の文献を参考に，**図 2** で示す手法で評価を行った．Exploitation については，CVSS Base Severity が Critical であれば，概念実証が行われていると仮定し，PoC と評価している．Automatable についても，SSVC の文献 [5] の記載を参考として実施した．具体的には，図 2 で挙げた CVSS の各指標が攻撃の自動化可能性に関与する可能性があることや外部への露出が攻撃の自動化に関連することが述べられており，これらに基づいて判定手法を定めた．

3.3.2　Exposure の評価

表 1 の SSVC Exposure の行に Exposure の評価結果を示す．Web サーバはインターネットからアクセス可能であるため Open，API サーバはアクセス制御が施されているため Controlled，残る各データベースはローカルサービスであるため Small とした．

3.3.3　Human Impact の評価

表 2 に想定環境における各業務ごとの Human Impact の評価結果を示す．Human Impact は 2.2.2 項で説明した通り，Mission Impact と Safety Impact の二つの評価を用いて算出される．

Mission Impact の評価． Mission Impact は OWASP Risk Rating Methodology の Business Impact Factors を

用いて，4 つの観点で評価を行った [13]

サービス提供業務プロセスについて，サービス停止や障害が年間利益へ重大な影響があるため，経済的損害は 7 とした．また，サービス停止により信用失墜や，SNS での炎上など信用の棄損につながるとして，風評被害は 7 とした．コンプライアンス違反については，システム障害が直接違反となる可能性は低いため，2 とし，プライバシー侵害は関連しないとして 1 とした．

顧客情報管理業務プロセスでは，情報漏洩に伴う賠償・対策費用は非常に大きく倒産につながる可能性もあるとして，9 とした．風評被害については，信用を大きく棄損し，利用者低減につながる大きな問題であるため 9 とした．また，個人情報保護法違反など重大な違反につながるとしてコンプライアンス違反は 7 とし，プライバシー損害については，利用規模から 7 とした．

商品情報管理業務プロセスでは，商品データ改ざんにより売上・在庫管理に重大影響があるため，経済的損害は 7 とした．また，風評被害についても，信用の棄損につながるため，7 とし，コンプライアンス違反では景品表示法などの違反の可能性から，明らかな違反になるとし，5 とした．プライバシー侵害については，利用規模から 7 とした．

OWASP の基準では, 評価平均が 0〜3 であれば LOW, 3〜6 は MEDIUM, 6〜9 は HIGH である．したがって, LOW であれば, Mission Impact の評価結果は Degraded/Crippled, MEDIUM であれば, MEF failure, HIGH であれば Mission failure とした．

Safety Impact の評価． Safety Impact は SSVC で提示されている 4 つの評価観点をもとに評価を行った．本シミュレーションで想定する E コマースサービスでは，どの業務においても，身体的，環境的影響は考えられない．経済的影響においても，SSVC で定義されているような規模影響は与えず，心理的影響についても，カウンセリング治療が必要な大きな影響は与えないため，全ての評価は None/Minor となった．

3.4 シミュレーションの設定

シミュレーションではまず最初に，脆弱性の優先度が付与されたチケットが発行される．脆弱性の発見と，優先度付け，チケットの発行までは自動で行われるとし，シミュレーション上の時間に影響を与えないとする．

本シミュレーションにおいて投入する脆弱性には, Apache サーバに関連する深刻度の高い脆弱性（Log4j）が含まれる 2021 年に公開された CVE 情報を用いた．SSVC による脆弱性評価は，脅威・環境情報を考慮した評価である．そのため，脆弱性の深刻度が高い場合でも，脅威の悪用状況や，外部への露出度，業務への影響によって SSVC によって判定される対応優先度は低い可能性がある．一方，CVSS による脆弱性評価では，脆弱性の深刻度のみを評価しているため，このような状況においても対応優先度が減少することはない．したがって，脅威・環境情報の考慮により，優先度付けが実態と乖離し，真に優先して対応するべき脆弱性を優先できない可能性がある．このような仮説を検証するために，深刻度の高い脆弱性が含まれる脆弱性データを活用する．

シミュレーションに用いる脆弱性は，NVD（National Vulnerability Database）のデータベースから取得した．この際，表 1 に含まれるソフトウェアを CPE（Common Platform Enumeration）にもつ CVE 情報のみを取得した．

脆弱性が公開された日に脆弱性が発覚してチケットも発行されるとし，シミュレーション期間は 2021 年の 1 月 1 日から 12 月 31 日までの 1 年間とした．

脆弱性の対応優先度は, CVSS と SSVC で異なる．CVSS では，NVD の脆弱性データに含まれるベーススコアを優先度とする．SSVC については，3.3 節で述べた手法で評価を実施する．

脆弱性対応はパッチ適用を主とし，パッチの修正や脆弱性を取り除くためのソフトウェア自体の修正は行わない事を想定する．また，Nepel らの調査 [8] では，パッチ適用に要する時間は，平均 41.3 日，中央値 13.2 日と報告されている．これは脆弱性個体差による大きなばらつきが存在するためであると考え，本シミュレーションではこれらの外れ値を排除し，平均値と中央値が一致する分布を想定した．したがって，脆弱性管理と対応にかかる時間については，平均 μ=13.2 日，標準偏差 σ=4 日の正規分布に従うと仮定した．

以上の設定を用いて，シミュレーション用プログラムを Python で実装し，評価に用いた．

3.5 制限事項

本シミュレーションならびに研究では，いくつかの制限事項がある．まず，今回の評価は 3.2 節で述べた環境を想定している．このため，多くの先行研究と同様にケーススタディとなっており [8]，他環境においては必ずしも同様の結果が出るとは限らない．これに関連して，本研究の一般化可能性について 5.2 節で議論する．

また，脆弱性対応に係るシミュレーションに限界が存在する．具体的には，脆弱性対応はパッチ適用のみを対象としており，他の対応，例えば FW 等によるワークアラウンド対応やソフトウェアのコード修正等 [7] はシミュレーションに含まれていない．

さらに，脆弱性の優先度付けに資する他の指標，例えば Exploit Prediction Scoring System（EPSS [14]）等や各種研究 [10] は今回は対象外としている．EPSS は，脆弱性が今後 30 日以内に悪用される可能性を推定する統計モデルであり，脆弱性の対応優先度付けに応用することが可能である [10]．しかし，EPSS は公開元の FIRST がアクティブ

表 2　想定環境における Human Impact の評価結果

		サービス提供	顧客情報管理	商品情報管理
Mission Impact	経済的損害	7	9	7
	風評被害	7	9	5
	コンプライアンス違反	2	7	5
	プライバシー侵害	1	7	7
	評価結果	MEF failure	Mission failure	Mission failure
Safety Impact	Physical（身体的影響）	None/Minor	None/Minor	None/Minor
	Environment（環境的影響）	None/Minor	None/Minor	None/Minor
	Financial（経済的影響）	None/Minor	None/Minor	None/Minor
	Psychology（心理的影響）	None/Minor	None/Minor	None/Minor
	評価結果	None/Minor	None/Minor	None/Minor
Human Impact	評価結果	Medium	Very High	Very High

な悪用を示す他の証拠がない場合に使用するのが最適だと指摘していることや，悪用が確認された脆弱性の EPSS スコアは必ずしも高くないことから [15]，単体での利用が難しく，他の指標と組み合わせる必要があるため，今回は利用を見送った．また，各種研究についても，普及度や導入性等の観点から今回は実施を見送った．

Vulnrichment プロジェクトから取得できない脆弱性で行った Exploitation による評価において，仮定が存在する．PoC と判定するには，エクスプロイトコードが販売されている，よく知られた悪用方法がある等の詳細な情報が必要である．しかし，実際の脆弱性管理においてもそれぞれの脆弱性ごとに詳細な調査と判定をするのが困難であると推察し，本シミュレーションでは CVSS Severity が Critical であれば PoC レベルに該当すると判定した．

CVSS は，基本評価基準，脅威評価基準，環境評価基準の 3 つを組み合わせることを強く推奨している．このため，ベーススコアのみで評価を行うことは，CVSS の本来の活用方法を十分に反映しているとは言えない．しかし，本研究では環境評価基準を用いたスコア補正の作業が煩雑であり，実際にその利用率も低いことから [16]，ベーススコアのみを用いた．環境評価を行うには，年間数万件も報告される脆弱性 [6] それぞれに対して補正作業を実施する必要があり，現実的に困難である．一方で，SSVC も環境情報の考慮を必要とするが，その評価は資産や業務単位であり，一度実施すれば複数の脆弱性に対して結果を再利用することが可能である [5]．このため，定期的なメンテナンスは必要となるものの，CVSS の環境基準評価を採用した場合と比べ，評価にかかる負荷は大幅に軽減されると推察される．

4.　評価結果（RQ1）

4.1　脆弱性の対応優先度付け結果

図 3 にシミュレーションに使用した脆弱性データの CVSS スコアと SSVC 判定結果を示す．SSVC 判定結果については，Scheduled と Out-of-Cycle の二つの結果が得られた．Out-of-Cycle と判定された脆弱性は必ずしも高い CVSS ベーススコアでないことがわかる．

図 3 に示した結果について，脆弱性毎に一部抜粋した

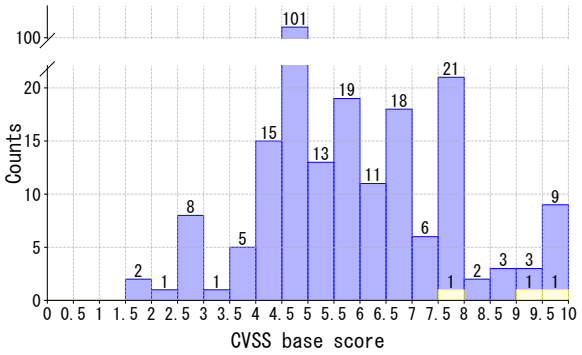

図 3　シミュレーションにおける CVSS と SSVC による脆弱性評価結果．CVSS による評価結果をヒストグラムで，SSVC による評価結果を色で図示．Out-of-Cycle が黄色（3 件），Scheduled が青（235 件），Immediate と Defer はそれぞれ 0 件．

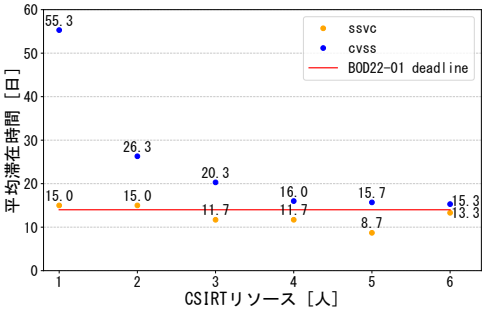

図 4　Out-of-Cycle と SSVC により判定された脆弱性が脆弱性対応処理に滞在する平均時間

ものを**表 3** に示す．具体的には，CVSS ベーススコアが 9.5〜10 のもの（全 9 件，1-1〜1-9）と 7.5〜8.0 のもの（全 21 件中 2 件抜粋，2-1〜2-2）を示す．CVSS ベーススコアが高い（9.8）9 件の脆弱性では，Exploitation の指標が Active である CVE-2021-42013（1-8）のみが SSVC において Out-of-Cycle と出力された．逆に，CVE-2021-41773（2-1）は，CVSS ベーススコアが中程度（7.5）であるものの，SSVC ではおいては Expoitation g が Active かつ Exposure が Open であること等から Out-of-Cycle と評価された．このように，SSVC は，CVSS ベーススコアのみを用いる場合よりも，脆弱性や資産の状況に応じてより適切に対応優先度を付与できているという示唆を得た．

4.2　脆弱性対応シミュレーションの結果

図 4 にシミュレーションによって得られた，Out-of-Cycle

表 3　脆弱性毎の CVSS と SSVC の評価結果（一部抜粋）

#	CVE ID	資産名	ソフトウェア	CVSS ベーススコア	SSVC 評価結果	Exploitation	Exposure	Automatable	Human Impact
1-1	CVE-2021-3177	API サーバ	Python	9.8	Scheduled	POC	controlled	No	Medium
1-2	CVE-2021-21344	顧客情報 DB	MySQL	9.8	Scheduled	POC	Small	No	Very High
1-3	CVE-2021-29921	API サーバ	Python	9.8	Scheduled	POC	controlled	No	Medium
1-4	CVE-2021-26691	Web サーバ	Apache	9.8	Scheduled	POC	Open	No	Medium
1-5	CVE-2021-22931	顧客情報 DB	MySQL	9.8	Scheduled	None	Small	Yes	Very High
1-6	CVE-2021-3711	顧客情報 DB	MySQL	9.8	Scheduled	POC	Small	No	Very High
1-7	CVE-2021-39275	Web サーバ	Apache	9.8	Scheduled	POC	Open	No	Medium
1-8	CVE-2021-42013	Web サーバ	Apache	9.8	Out-of-Cycle	Active	Open	Yes	Medium
1-9	CVE-2021-44790	Web サーバ	Apache	9.8	Scheduled	POC	Open	No	Medium
2-1	CVE-2021-31618	Web サーバ	Apache	7.5	Scheduled	None	Open	Yes	Medium
2-2	CVE-2021-41773	Web サーバ	Apache	7.5	Out-of-Cycle	Active	Open	Yes	Medium

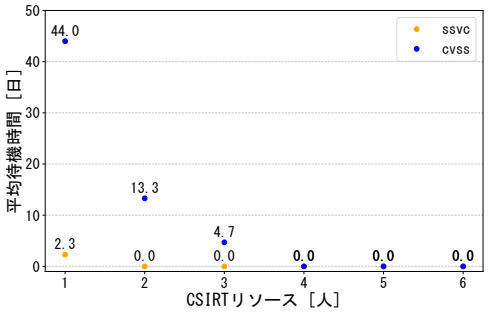

図 5　Out-of-Cycle と SSVC により判定された脆弱性の平均待機時間

と SSVC により判定された脆弱性が脆弱性対応フローに滞在する平均時間を示す．赤線は CISA が KEV に含まれる脆弱性に対して，BOD22-01 で定めている脆弱性の対応期限である 14 日を示している．また，**図 5** にシミュレーションによって得られた，Out-of-Cycle と SSVC により判定された脆弱性の平均待機時間を示す．脆弱性対応では優先度の高い脆弱性から対応が実施され，優先度の低い脆弱性は自身より優先度が高い脆弱性が対処されるまで待機するため，その平均時間を図 5 に示している．これらの結果から，SSVC では CSIRT の人数が 2 人以上，CVSS では CSIRT の人数が 4 人以上になると Out-of-Cycle と判定された脆弱性の待機時間が 0 になることが分かる．

RQ1 に対する回答：

何れの場合でも CVSS よりも SSVC のほうが対応に必要な人数が少なく，脆弱性の待機時間も短く済むことが明らかになった．また，SSVC は管理対象の資産の状況や脆弱性との関係をもとに，より適切に対応優先度を付与できることが確認された．さらに，SSVC を用いたほうが CVSS よりも各種法令期間内での対応可能性が高いことを示した．

5. 議論

5.1 脆弱性の対応優先度付けにおける SSVC と CVSS の比較（RQ2）

本評価では，KEV よりも対応期限が短く，法的義務を伴う指令である BOD22-01 を満たせるか確認した．CSIRT の人数が 3 人以下の場合，CVSS の優先度付けをもとに脆弱性対応を行うと BOD22-01 の期限である 14 日より長い期間が必要であることが分かった．また，この時，SSVC により Out-of-Cycle と判定された脆弱性に待機時間が生じることが分かった．これは，SSVC で Out-of-Cycle と判定された脆弱性より CVSS ベーススコアが高い脆弱性が優先して対応されるためである．したがって，CVSS による優先度付けを用いた場合，BOD22-01 によって定められた期限を守ることができないと推察する．

図 5 の結果から，SSVC による脆弱性評価を用いた場合，CSIRT は 2 人いれば SSVC で Out-of-Cycle と評価される脆弱性を待機時間なく優先して対応できることが分かった．これは，JNSA と ISOG-J が 4 段階で定めるセキュリティ対応組織の要員モデルにおいて，Level 3 において求められる要員数である．CVSS を用いた場合，4 人以上が求められるため，2 倍の要員数が必要である．したがって，これらの評価から SSVC を用いた場合，脆弱性の深刻度のみを評価する CVSS を用いた評価と比較して，半分の CSIRT 要員数で 法令期限内の脆弱性対応が可能なことが分かった．

また，図 4 において，SSVC と CVSS ともに CSIRT の人数が 4 人以上であっても BOD22-01 の期限を超えてしまっている場合がある．これは，脆弱性の対応平均時間が 13.2 日であり，正規分布に基づいた揺らぎによって期限である 14 日を超えてしまう場合が存在するためである．実際の対応では，作業を分担することや 1 日の稼働時間を増加させることで BOD22-01 によって定められた期限以内の脆弱性対応が可能であると考えられる．

RQ2 に対する回答：

本シミュレーションで想定する環境では，CVSS よりも SSVC のほうが対応に必要な人数や脆弱性の待機時間等，何れの面でも優位性があることを示した．SSVC に基づく評価には，専門性や環境評価のコストも必要であるため，人員のスキルや資産管理体制が十分に整備されていない組織においては，依然として CVSS を活用することが選択肢となり得る．一方で体制の整った組織では，初期に一定のリソースを投じて SSVC の評価環境を構築することにより，SSVC の恩恵を受けることができる．

5.2 一般化可能性

本研究の結果は，特定の評価環境（E コマース環境）における特定の年（2021 年）のシミュレーション結果をベースとしている．即ち，構成要素の異なる他の環境や脆弱性の数・性質の異なる他の年では全く同じ結果になるわけではない．また，本研究のシミュレーションでは専従の脆弱性対応担当者を想定しているが，中小規模の組織では IT 担当者がセキュリティ対応も兼任する場合やその IT 担当者が存在しない場合すらあることが知られている [17]．このため，中小規模の組織では必ずしもこの結果が当てはまるとは限らない．以上より，多くの先行研究と同様に，本研究はケーススタディであり，一般化可能性については注意が必要である．

他方で，本研究では，一般的に想定され得る E コマース環境における実際の 1 年間のシミュレーションを実施した．このため，評価環境や対象年（脆弱性の件数や性質）が異なる場合には，結果に一定の差異が生じ得るものの，CVSS と SSVC との関係性自体には大きな変化は無く，本研究におけるシミュレーション結果は一定程度の一般化可能性を有すると推察される．

5.3 提言事項

本研究におけるシミュレーションとその分析結果をもとに，以下の提言事項を示す．

SSVC 普及のための教育や啓蒙．評価結果より，CVSS よりも SSVC のほうが脆弱性優先度付けの観点では優れていることを定量的に示した．この一方で，現実には未だに CVSS が利用されている．この現状を改善するためには，SSVC 活用に係る周知や教育活動が重要であると考える．CISA では SSVC の利用促進のため Vulnrichment プロジェクトを公開しており，日本においても IPA から SSVC に関する資料が公開されている．これらの取り組みを活用することで，SSVC の採用が促進されると期待される．

SSVC 利用に係る省力化・効率化．SSVC を活用するにあたっては，CVSS と比較して資産情報に基づく環境評価が求められるため，スキルや時間的コストが必要となる．さらに，シャドー IT やインストール済みソフトウェアの把握，ライフサイクル管理を含む適切な資産管理が不可欠である．したがって，スキル育成に関する教育的研究，環境評価の省力化・効率化技術，資産管理を容易化・促進する研究開発は，SSVC の実用化と，ひいては脆弱性優先度付けの高度化・効率化に向けて重要であると推察される．

6. おわりに

本稿では，我々の実装した離散事象型シミュレータを用いて，CVSS と SSVC を用いて脆弱性対応の対応優先度付けを行った際の影響を定量的に比較評価した．この結果，我々が想定する E コマース環境において，SSVC による脆弱性評価は，脆弱性の深刻度のみを評価する CVSS を用いた評価と比較して，半分の CSIRT 要員数で法令期限内の脆弱性対応が可能なことが分かった．本研究における評価結果が SOC や CSIRT での脆弱性の優先度付けタスクにおける SSVC 採用可否の検討に係る一助になれば幸いである．

参考文献

[1] Cybersecurity & Infrastructure Security Agency: BOD 22-01: Reducing the Significant Risk of Known Exploited Vulnerabilities, `https://www.cisa.gov/news-events/directives/bod-22-01-reducing-significant-risk-known-exploited-vulnerabilities` (2021).

[2] Alomar, N. et al.: "You've got your nice list of bugs, now what?" vulnerability discovery and management processes in the wild, SOUPS '20, pp. 319–339 (2020).

[3] Spring, J. et al.: Towards improving CVSS, *Carnegie Mellon University, Tech. Rep* (2018).

[4] Spring, J. et al.: Time to Change the CVSS?, *IEEE Security Privacy*, Vol. 19, No. 2, pp. 74–78 (2021).

[5] Spring, J. M. et al.: Prioritizing Vulnerability Response: A Stakeholder-Specific Vulnerability Categorization (SSVC), Technical Report (draft) v2.1.0-edb6c97, Carnegie Mellon University, CERT Coordination Center (2024).

[6] de Smale, S. et al.: No One Drinks From the Firehose: How Organizations Filter and Prioritize Vulnerability Information, S&P '23, pp. 1980–1996 (2023).

[7] Jenkins, A. D. G. et al.: Not as easy as just update: Survey of System Administrators and Patching Behaviours, CHI '24, pp. 1–17 (2024).

[8] Napel, G. t. et al.: Speedrunning the Maze: Meeting Regulatory Patching Deadlines in a Large Enterprise Environment, S&P '25, pp. 504–521 (2025).

[9] Cybersecurity & Infrastructure Security Agency: CISA Known Exploited Vulnerabilities Catalog. `https://www.cisa.gov/known-exploited-vulnerabilities-catalog`.

[10] Jiang, Y. et al.: A Survey on Vulnerability Prioritization: Taxonomy, Metrics, and Research Challenges, *arXiv* (2025).

[11] Allodi, L. et al.: Comparing Vulnerability Severity and Exploits Using Case-Control Studies, *ACM Transactions on Information and System Security*, Vol. 17, No. 1, pp. 1–20 (2014).

[12] Github: cisagov/vulnrichment, `https://github.com/cisagov/vulnrichment` (2025).

[13] FutureVuls Blog: OWASP Risk Rating Methodology について -リスクの判断-. `https://vuls.biz/blog/articles/20231019a/`.

[14] Jacobs, J. et al.: Exploit Prediction Scoring System (EPSS), *Digital Threats*, Vol. 2, No. 3 (2021).

[15] PwC Japan Group: KEV を用いた EPSS の効果検証, `https://www.pwc.com/jp/ja/knowledge/column/awareness-cyber-security/epss-kev.html` (2024).

[16] Dugal, D. et al.: Announcing CVSS v4.0, Technical report, FIRST – Forum of Incident Response and Security Teams (2023).

[17] Parkin, S. et al.: Pragmatic Security: Modelling IT Security Management Responsibilities for SME Archetypes, MIST '16, pp. 69—80 (2016).