Vol.2020-DPS-182 No.20
Vol.2020-CSEC-88 No.20
2020/3/12

機械学習を用いた CVE から CAPEC への関連付け手法の提案

鹿子木健太[1] 野寄祐樹[1] 鷲崎弘宜[1] 深澤良彰[1] 小形真平[2]
大久保隆夫[3] 加藤岳久[4] 鹿糠秀行[4] 櫨山淳雄[5] 吉岡信和[6]

概要：システム管理者が自組織に関係する脆弱性の調査や対処をする際，既知の脆弱性情報を参照するが，その数は非常に膨大であり脆弱性の対処に多くの時間を必要とする．脆弱性の対処を効率化するために，既知の情報を蓄積する Common Vulnerabilities and Exposures (CVE) や Common Attack Pattern Enumeration and Classification (CAPEC) は有用である．CVE は脆弱性情報を識別する共通識別子 (CVE-ID) のリストである．CAPEC は脆弱性に関係する攻撃をパターン化したアタックパターンの辞書であり，パターンごとに共通識別子 (CAPEC-ID) が付与されている．しかし CVE と CAPEC とは独立しているので，CVE の脆弱性の情報から関係した CAPEC の攻撃の情報を特定するのには経験を有する．そこで本論では CVE-ID に関連する CAPEC-ID を半自動的に特定する手法を提案する．最初に CVE-ID を CAPEC に記載の Impact ごとに機械学習の Random Forest を用いて分類する．次に分類された CVE-ID と，この Impact に該当する CAPEC-ID とを Doc2Vec を用いてそれぞれの文章の類似度を算出し，類似度が高い CAPEC-ID と CVE-ID とを関連付けする．実験を通じて，扱った CAPEC に記載されていた脆弱性合計 43 個の内の 17 個が，CVE-ID から直接特定できた．

Tracing CAPEC Attack Patterns from CVE Vulnerability Information by using Machine Learning

KENTA KANAKOGI[†1] YUKI NOYORI[†1] HIRONORI WASHIZAKI[†1] YOSHIAKI FUKAZAWA[†1]
SHINPEI OGATA[†2] TAKAO OKUBO[†3] TAKEHISA KATO[†4] HIDEYUKI KANUKA[†4]
ATSUO HAZEYAMA[†5] NOBUKAZU YOSHIOKA[†6]

1. はじめに

脆弱性の数は非常に膨大であり，現在 Common Vulnerabilities and Exposures (CVE)に 13 万個以上の脆弱性が報告されている．システム管理者は脆弱性への対応に多くの時間を費やさなければならない．効率的に脆弱性へ対応するためには，脆弱性情報を迅速にかつ効率的に把握，収集することが重要であり，既知の情報を蓄積する CVE は有用である．しかし CVE には攻撃手法に関する情報が含まれている事が少なく，脆弱性がわかっても具体的な攻撃手法がわからず対策に苦慮している．そこで Common Attack Pattern Enumeration and Classification (CAPEC) が役立つ．CAPEC は脆弱性に対する攻撃をパターン化した辞書で，パターン毎に共通識別子 (CAPEC-ID)が付与されている．しかし CVE と CAPEC とは独立しているので，CVE の脆弱性の情報から関係した CAPEC の攻撃の情報を特定するのには経験を有する．

そこで本論では CVE-ID に関連する CAPEC-ID を半自動的に特定する手法を提案する．CAPEC に記載されている 514 個の攻撃パターンから関連のある攻撃パターンを探すのは難しいため, 攻撃パターンの数を絞り込む必要がある．そこで, 機械学習の 1 つである Random Forest を使用して，CVE の Description を Impact パラメーターで分類する．Impact とは，攻撃に成功した場合に発生する負の技術的影響を表している．次に分類された CVE-ID とラベル付けされた Impact に該当する CAPEC-ID を, 機械学習の 1 つである Doc2Vec を用いてそれぞれの文章の類似度を算出し，類似度が高い CAPEC-ID と CVE-ID とを関連付ける．

実験を通じて，扱った脆弱性合計 43 個の内の 17 個が，CVE-ID から直接特定できた．

この研究の最終目標は，ペンテスターやレッドチームが発見した脆弱性に対して攻撃シナリオの提案することである．

我々の研究手法を評価するために以下，2 つの RQ を調査する．

- RQ1:今現在 CVE(脆弱性)から CAPEC(攻撃パターン)は対応づいているのか? その対応づけの精度は?
- RQ2:我々の研究手法は，CVE(脆弱性)から CAPEC(攻撃パターン)へ自動的かつ直接的に対応づけることは可能か? その精度は?

RQ1 は提案手法の精度を比較するために必要である．この質問に答えるために既存の脆弱性情報データベースの関係を追跡する．RQ2 は我々の手法の有用性を証明するために必要である．この質問に答えるためにテストケースを設定し，実験する．

1 早稲田大学
Waseda University
2 信州大学
Shinshu University
3 情報セキュリティ大学院大学
Institute of Information Security
4 日立製作所
Hitachi,Ltd.
5 東京学芸大学
Tokyo Gakugei University
6 国立情報学研究所
National Institute of Informatics

Vol.2020-DPS-182 No.20
Vol.2020-CSEC-88 No.20
2020/3/12

本論の貢献は以下の 2 つである．1 つ目は現在の脆弱性情報データベース間の対応づけの精度の実態を明らかにしたこと．2 つ目は脆弱性が分かっていても具体的な攻撃手法が分からず対策を打てない人に具体的な対策情報を提供できることである．

2. 背景

2.1 脆弱性データベースについて

脆弱性に関する情報を広く一般に公開するデータベースが多く存在する．その中から，本論で扱う脆弱性データベースについて説明する．

2.1.1 Common Vulnerabilities and Exposures (CVE)

CVE は既知の脆弱性情報を識別する共通識別子のリストである．

CVE[1]に掲載されている個々の脆弱性についての主な情報は以下の通りである．

- CVE-ID: CVE 内の識別子としての番号．CVE-{データが登録された年 4 桁}-{年内での通し番号}で表される．ここに同期している NVD のページへのリンクも記載されている．
- Description: セキュリティの脆弱性または露出の簡単な説明．
- Reference: 関連する脆弱性関連情報の一覧で, CVE 情報源サイトや製品開発ベンダサイトの URL などがリストアップされている．

2.1.2 National Vulnerability Database (NVD)

NVD は，Security Content Automation Protocol (SCAP)を使用して表される脆弱性管理データに基づく標準の米国政府リポジトリである[2]．CVE リストに基づいて構築され，CVE リストと完全に同期されている．NVD のページから CVSS，CWE，CPE およびその他の関連データベースが得られる．

2.1.3 Common Weakness Enumeration (CWE)

CWE は, 一般的なソフトウェアセキュリティの弱点のリストである[3]．脆弱性のカテゴリを識別するために作成された．CWE-ID がそれぞれに割り当てられ，階層構造で提供される．各 CWE-ID は，説明，原因，関連攻撃パターン(CAPEC-ID)を含む一連の情報とともに文書化されている．

2.1.4 Common Attack Pattern Enumeration and Classification (CAPEC)

CAPEC は, 既知の弱点を悪用するために敵が使用する攻撃パターンの包括的な辞書である．攻撃パターンは，既知の弱点を悪用するために攻撃者が使用する一般的な属性とアプローチの説明である．CAPEC は，攻撃の成功と攻撃の特定の要素を防ぐ方法を理解するのに役立つ. CAPEC の主な情報は以下の通りである[4].

- CAPEC-ID: CAPEC 内の識別子としての番号.
- Description: 攻撃パターンの概要.
- Consequences: 攻撃パターンに関連するさまざまな個々の結果を示す. 範囲と影響に関する情報が書き込まれる．Scope は，違反しているセキュリティプロパティを表す．Impact は，攻撃者が攻撃に成功した場合に発生する負の技術的影響を示す.
- Prerequisites: 攻撃が成功するための前提条件が記載されています．防御側は，攻撃されないために，記載されている内容を満たさないことが要求される.

2.2 動機付けの例

2.1 で説明した脆弱性情報データベースは, 様々なユースケースで使用される．本論文では，VAPT(Vulnerability Assessment and Penetration Test)での使用方法について説明する．VAPT のプロセスはいくつかの研究で説明されている[5][6]．図 1 において，PHASE2 はさらに 5 つのステップから構成される．

S1: 情報収集．テスターは攻撃対象のセキュリティステータスの効率的かつ正確なプロファイルを作成する．

S2: 脆弱性スキャン．主にツールを使うことで自動化される．OpenVAS(a)や Vuls(b)を使用することで，システムに存在する CVE-ID を網羅的に検知する．

S3: 脆弱性のマッピングと分析．脆弱性の重大度と影響を特定することにより，脆弱性に優先順位を付ける．

S4: 攻撃シナリオの生成．上記のステップから優先順位の高い脆弱性を使用した攻撃シナリオを考える．

S5: 脆弱性の活用．S4 で作成した攻撃シナリオを実際に実行し，テストを行う．

S2 で発見される脆弱性は膨大な数になるため，S3 での優先順位付けが重要となる．優先順位付けを行うためには多くの情報が必要である．そこで役立つのが CVE, CWE, CAPEC である．CWE は脆弱性に関する詳しく情報を提供し, CAPEC はその弱点を利用した攻撃方法の情報を提供できる．さらに CAPEC は S4 での攻撃シナリオの作成にも利用できる．よって，CVE-ID が CWE や CAPEC へ対応付いていると VAPT を効率的に行える．

a) https://www.openvas.org/

b) https://vuls.biz/

Vol.2020-DPS-182 No.20
Vol.2020-CSEC-88 No.20
2020/3/12

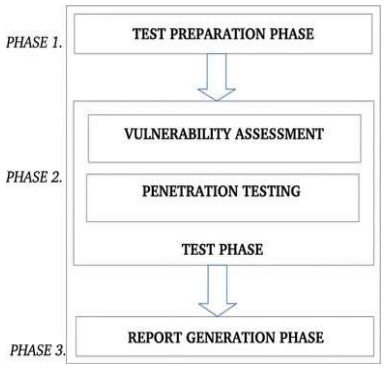

図 1　VAPT プロセス [6]

3. 提案手法

この手法の目的は，CVE の Description を入力し，その CVE-ID に関連付けられた CAPEC-ID を取得することである．

本手法の入出力と概要を図 2 に示す．本手法は図 2 に示す 3 つのステップから構成されている．

3.1 STEP1: 攻撃手法の抽出

入力された CVE の Description に特定の攻撃手法名が含まれているか確認する．特定の攻撃手法名とは，「SQL Injection」「Cross Site Request Forgery」「Buffer Overflow」「Cross Site Scripting (XSS)」の 4 つである．この 4 つの攻撃手法は，CVE で 2000 件以上報告されている．また，これら攻撃手法は，CAPEC にて 1 つの識別子だけではない．例えば，XSS(CAPEC-63)は，DOM-Based XSS(CAPEC-588), Reflected XSS(CAPEC-591), Stored XSS(CAPEC-592)に具体化できる．XSS に関わりのある CAPEC-ID は 37 個ある．よって，これら 4 つの攻撃手法は CVE に報告されるときは Description に攻撃手法名を明記されることが多い．また複数の CAPEC-ID と関係を持つため，その攻撃手法名に特化した CAPEC-ID 群から探し出すべきである．CVE の Description にこれらの攻撃手法名が含まれていた場合は STEP2 を飛ばす．含まれていない場合は STEP2 に進む．

3.2 STEP2: Impact の分類

入力された CVE の Description を 8 種類の Impact に分類する．Impact とは，攻撃者が攻撃に成功した場合に発生する負の技術的影響を示す．Impact を用いたラベル付けを行うために，分類器を作成する．図 3 は，Impact を用いたラベル付け方法の概要を示す. Impact ラベル付けした CVE の Description を教師データとして作成し，教師あり学習を行うことで分類器を作成する．教師データ数は表 1 に示す．機械学習では，数値データを入力として提供する必要があるため Bag-Of-Words を使用した．分類器は Random Forest を使用した．これにより，作成した Random Forest に CVE の Description を入力すると，予測結果に則り，入力された CVE の Description に対して Impact ラベルが自動的に付与される．Impact をラベルとして入力された CVE の Description へ付与することで，Impact に応じた攻撃パターンの検索対象の絞込みを可能とする．機械学習には Python の[Scikit-learn]を使用した．Bag-Of-Words 作成には Python の[genism]を使用した．

3.3 STEP3: 類似度測定

CVE の Description と類似度が高い文章を検索する． 図 4 は，類似度測定の処理を示す． 図 4 は，STEP2 で「Gain Privileges」とラベル付けされた後の流れを示している．類似度測定として Doc2Vec を使用する．Doc2Vec は，任意の文章をベクトル化する技術であり，テキストなどの分散表現を取得できる．ベクトル間の類似性を測定することで類似した文章を検索することができる. CVE の Description と CWE の Description をトレーニングデータとして学習することにより Doc2Vec モデルを作成した． そのモデルを使用して CVE の Description と CAPEC の Description または Prerequisites の各文との類似性を計算する. 類似度の高い文章を含んだ CAPEC-ID と CVE-ID とを関連付けする．

表 1　教師データの数

Impact	データ数
Gain Privileges	613
Execute Unauthorized Commands	492
Bypass Protection Mechanism	572
Hide Activities	29
Modify Data	594
Read Data	527
Resource Consumption	343
Alter Execution Logic	2
Unreliable Execution	1

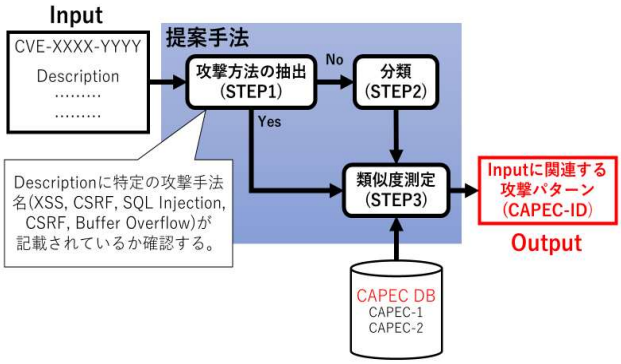

図 2　提案手法の概要

Vol.2020-DPS-182 No.20
Vol.2020-CSEC-88 No.20
2020/3/12

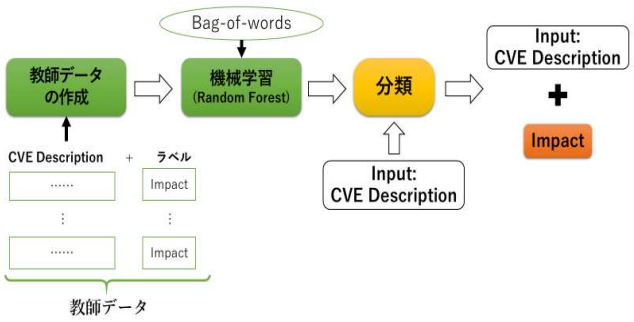

図 3　ラベリング手法の概要

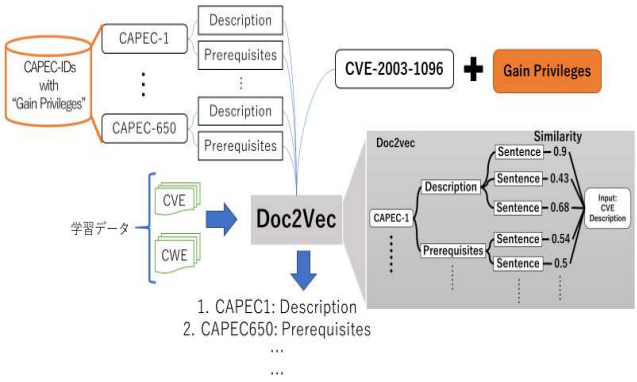

図 4　類似度測定

4. 実験結果

提案手法を用いて CVE-ID を入力し関連のある CAPEC-ID を探せるか実験を行う.

4.1 実験時のテストデータ

514 個のうち 35 個の CAPEC-ID が CVE で報告された 1 つもしくは複数の識別子を参考に作成された．これは CAPEC の Example Instance に記載されている. よって, 我々の提案手法を用いて Example Instance に記載されている 43 個の CVE-ID を入力して，それぞれに該当する CAPEC-ID を特定できるか実験する.

4.2 STEP2 の結果

43 個のうち Description に特定の攻撃手法名が記載されていない 31 個の CVE-ID が STEP2 を通る．実際に，31 個の CVE-ID を Random Forest によって Impact ベースのラベル付けを行った．Random Forest の精度について表 2 に示す．再現率は 0.774 であり，適合率が 0.558，　F 値が 0.649 となった.

4.3 STEP3 の結果

特定の攻撃手法名を含んでいた 12 個の CVE-ID と STEP2 で正しくラベリングされた 18 個を類似度判定し，類似度の高い CAPEC-ID の中に正解があるかを確かめる．提案手法の閾値の変動に応じた精度の結果を図 5 に示す．また，図 5 には従来手法の精度の結果も示した. 従来の手法とは,

表 2　Random Forest の精度

再現率	適合率	F 値
0.774	0.558	0.649

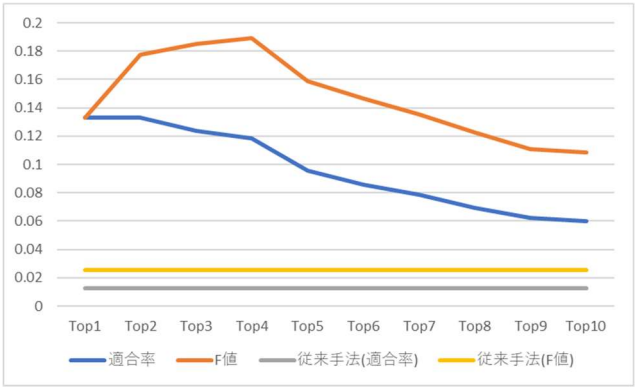

図 5　従来手法と提案手法の実験結果の比較

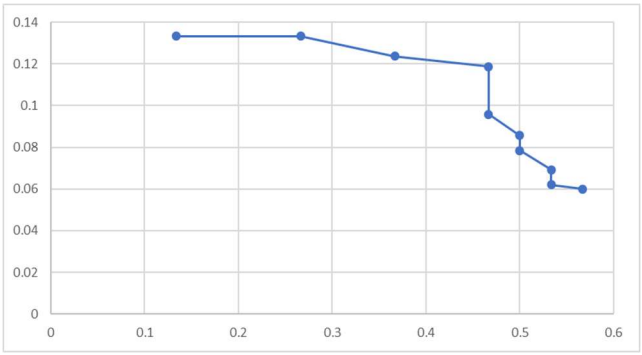

図 6　PR 曲線

CVE の Weakness Enumeration で CWE-ID を特定し，そして Related Attack Pattern で CAPEC-ID を特定する方法である. 図 6 に PR 曲線の結果を示す．それぞれの結果を比較した結果，すべての閾値において我々の提案手法の方が精度がよいことがわかる．我々の提案手法では，閾値が 4 位のときに F 値が最大となり, 再現率の最大値は 0.567 となった.

5. 考察

5.1 RQ1:今現在 CVE から CAPEC は対応づいているのか? その対応づけの精度は?

2.1 で述べたように CVE と NVD は同期している．NVD に Weakness Enumeration というフィールドがある．そこに CWE-ID が記載されている場合がある．また CWE には “Related Attack Pattern”というフィールドに CAPEC-ID が記載されている場合がある. よって, 間接的ではあるが, CVE から CWE を経由することで CAPEC へ対応づけることが可能である．しかし，この手法で CVE-ID から関連する CAPEC-ID を特定するには問題がある．それは Weakness Enumeration である．問題点を二つ挙げる.

1 つ目は抽象度の高い CWE-ID に対応づいていることが

Vol.2020-DPS-182 No.20
Vol.2020-CSEC-88 No.20
2020/3/12

多いことである．Weakness Enumeration には，CWE-ID か NVD-CWE-noinfo，NVD-CWE-Other のうちの 1 つもしくは複数を所持，または空白である場合がある．実際にこれらの割合はどれくらいなのか．NVD と CWE の関係の比率を図 7 に示す．CVE-ID の 28%が CWE-ID に対応づいていない．しかし，図 8 の結果から年代別で見ると昨年の CVE-ID は CWE-ID に対応付けられる割合が多くなってきた．CVE と対応づいている CWE-ID による脆弱性分布を図 9 に示す．上位 20 個の CWE-ID を掲載する．図 9 で 3,4 番目に数が多い CWE-20 と CWE-200 に注目する．これらの弱点は抽象度が非常に高いため，Related Attack Pattern に多くの CAPEC-ID が記載されている．これではどれが自身の脆弱性に関連する攻撃パターンか特定することは難しい．さらに CWE-20 について詳細に分析する．CWE-20 は，図 10 から大分類であることがわかる．パストラバーサル，バッファエラー，XSS，およびインジェクション系の親ノードである．CWE-20 の適応範囲が広い理由は，CWE-20 のホームページに記載されている．

“Some people use "input validation" as a general term that covers many different neutralization techniques for ensuring that input is appropriate, such as filtering, canonicalization, and escaping. Others use the term in a narrow context to simply mean "checking if an input conforms to expectations without changing it."” (c)

さらに，CWE-20 の「Terminology」には次の説明が含まれている．

“The "input validation" term is extremely common, but it is used in many different ways. In some cases its usage can obscure the real underlying weakness or otherwise hide chaining and composite relationships.” (c)

上記の情報から，CWE-20 が NVD から多くマッピングされる理由を理解できるが，このような対応づけでは役立つ脆弱性情報を提供できるとは考えにくい．

2 つ目は，Weakness Enumeration は攻撃者が CWE-ID を利用する可能性や方法などを特徴づけるものではないことである．1 つ具体的な例を示す．CVE-2014-0160 は Buffer Overread に関する脆弱性である．CVE-2014-0160 の Description は以下の通りである．

“The (1) TLS and (2) DTLS implementations in OpenSSL 1.0.1 before 1.0.1g do not properly handle Heartbeat Extension packets, which allows remote attackers to obtain sensitive information from process memory via crafted packets that trigger a buffer over-read, as demonstrated by reading private keys, related to d1_both.c and t1_lib.c, aka the Heartbleed bug. ” (d)

CWE-126 には Buffer Overread に関する弱点の識別子が存在するのに，CVE-2014-0160 は CWE-119(Buffer Errors)と結びついている．これはなぜなのか．CVE-2014-0160 は，CWE-125, 126, 130 といった複数の弱点が絡んでいると考えられる．それを 1 つの識別子で示すことができ，複数の弱点が存在することを表すことが可能な抽象度の高い CWE-119 の Buffer Errors を割り当てたのではないかと考える．よって，Weakness Enumeration は攻撃者が脆弱性を利用して攻撃することを特徴づけるものではないことが言える．

これらの 2 つの問題点から CVE から CAPEC への間接的な対応づけは可能だが有用な攻撃に関する情報は得られない．

RQ1 の回答:
CVE から CAPEC へ対応づけることは可能だったが，有用な攻撃に関する情報を得られる精度ではない．

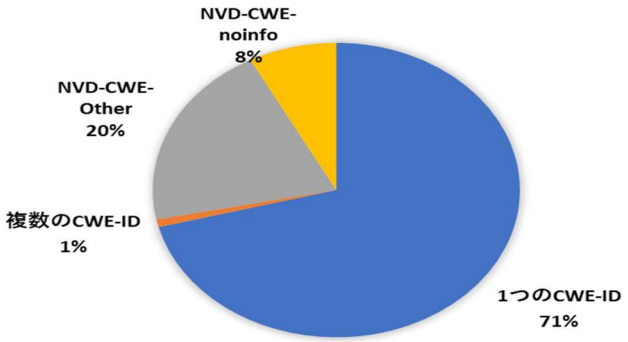

図 7 NVD と CWE の関係の割合

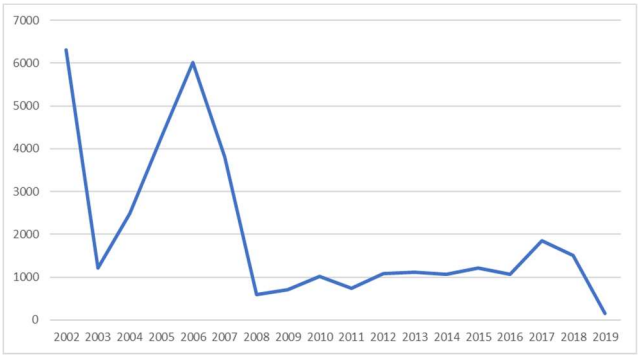

図 8 CWE-ID と対応づいていない CVE-ID の数の推移

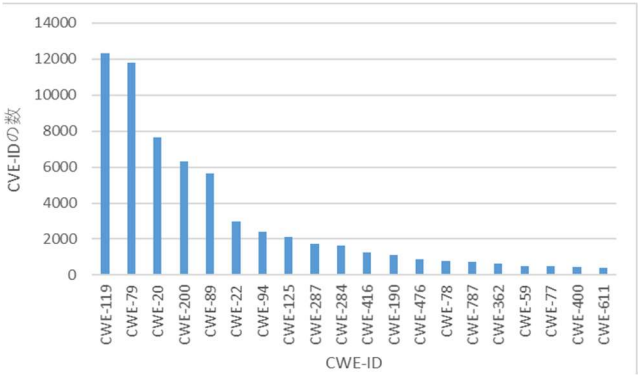

図 9 CWE-ID による脆弱性分布

c) CWE Version 3.2, https://cwe.mitre.org/data/published/cwe_v3.2.pdf

d) https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-0160

Vol.2020-DPS-182 No.20
Vol.2020-CSEC-88 No.20
2020/3/12

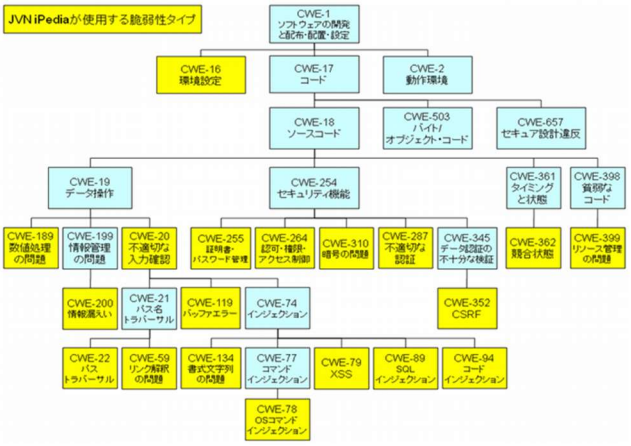

図 10　CWE の脆弱性タイプの階層構造図 (e)

5.2 RQ2: 我々の研究手法は，CVE から CAPEC へ自動的かつ直接的に対応づけることは可能か？ その精度は?

4 章において提案手法の各ステップの精度について述べた．手法全体の結果として 43 個のうち 17 個は CVE-ID が特定できた．図 5 の結果から従来の手法より F 値が向上することが明らかになった．従来の手法とは，CVE の Weakness Enumeration で CWE-ID を特定し，そして Related Attack Pattern で CAPEC-ID を特定する方法である．

しかし，我々の提案手法の正解率の低さは問題である．その原因として, STEP2 でのデータ数の偏りが考えられる．STEP3 に関しては学習データの数の少なさが原因だと考えられる．ネット上で公開されている日本語の学習済み Doc2Vec モデルは 8.86GB だが，今回作成した Doc2Vec モデルは 0.5GB だった．これはかなり小さい．

RQ2 の回答:
提案手法で CVE から CAPEC へ半自動的に直接特定することができた．従来手法より精度がよいことが明らかになった

5.3 妥当性への脅威

内的妥当性への脅威として，教師データを自分で作成することが含まれる．複数の専門家で教師データを作成することによる優れたアプローチが必要である．

外的妥当性への脅威は，43 個の CVE-ID のみで実験したことである．ただし，我々の提案手法は今回のテストケースに限定したものではない．CVE 全体に対して有効であることを証明する必要がある．CVE では日々新しい脆弱性が報告されている．本手法の有効性を確認するために更新されていく脆弱性を入力として実験する必要がある．

6. 関連研究

脆弱性オントロジーモデルに関する研究がある[7][8]．これらの研究では，CVE，CWE，CPE，CAPEC などのネットワークセキュリティの分野でよく知られている公開データベースに基づく脆弱性オントロジーモデルを研究している．

脆弱性情報データベースを使用したセキュリティ対策に関するいくつかの研究がある．ある論文では, CVE, CWE, CVSS，CAPEC などの一般的に使用される標準を統合して攻撃をランク付けする脆弱性管理オントロジーから脆弱性情報を取得している[9]．また，ある研究では脆弱性情報データベースで収集された関連性に基づいて，脆弱性に優先順位を付けるためのフレームワークを定義している[10]．

CVE と CAPEC 間のマッピングに関する研究は乏しい．唯一，CVE を CAPEC および ATT&CK に自動的にマッピングして適切な緩和策を見つける研究がある[11]．この研究では，CVE を CWE として自動的に分類するニューラルネットワークモデルを作成する．彼らは，CWE を CAPEC に分類するディープラーニングモデルを作成しようとしている．この論文は, CWE を介して CAPEC につなげている．CAPEC に直接対応づけされるものではない. 我々の研究は CAPEC に直接対応付けを行った．しかし，Impact パラメーターは CWE にも存在する．したがって，我々の提案手法を使用して，CVE から CWE も対応付けることが可能である．この研究には，新しい貢献を生み出す可能性がある．

7. おわりに

本論文では，CVE-ID に関連する CAPEC-ID を半自動的に特定する手法を提案した．CAPEC の Example Instance に記載されている 43 個の CVE-ID を入力データとして, 我々の提案手法を実行した．その結果，17 個の CVE が正確に対応付けされた．この研究では，攻撃手法名または Impact に基づくラベル付けする工程を考案した．これは，CAPEC から類似度の高い文を検索するときに，検索対象の数を減らすためである．

RQ 1 では, 現在の脆弱性情報データベースの関係を使用して，CVE-ID から CAPEC-ID の対応付ける方法を説明した. 今回実験で入力した 43 個の CVE-ID を従来手法で実験する．その結果，対応づいている CAPEC-ID を特定できるのは 2 つの CVE-ID のみだった．これらの結果を比較すると，従来方法より我々の提案手法の方が正確であることと既存の対応付け情報を参照しても有益な情報を獲得することは難しいと考えられる．

我々の提案手法の課題は各 STEP の精度を向上させることである．STEP2 では, CVE の Description に 1 つの Impact ラベルが与えられる．ただし，複数の Impact がある場合が

e) https://www.ipa.go.jp/security/vuln/CWE.html

ある．例えば以下の Description の場合，「Gain Privileges」か「Bypass Protection Mechanism」のどちらかに限定することは難しい．

“Telnet allows a remote client to specify environment variables including LD_LIBRARY_PATH, allowing an attacker to bypass the normal system libraries and gain root access.” (f)

したがって，複数のラベリングを行うことにより精度が向上すると考えられる．また，教師データ数の偏りはあったが，データ数は少なくはない．したがって，分類器を Random Forest から Deep Learning である Convolutional Neural Network (CNN)などに変更すると，精度が向上すると考えられる．Deep Learning の利点は，教師データが増えると精度が上がることである．STEP3 では，学習データの増加が必要である．そのため，英語の Wiki を使用して学習データの数が増やす．これにより，精度が大幅に向上する可能性がある．また妥当性の脅威がある．それも解決するべきである．

この研究の最終的な目標は，ペンテスターとレッドチームに有効な攻撃シナリオを提案することある．この論文では，CVE-ID に関連する CAPEC-ID を半自動的に特定する手法を提案した．CVE と CAPEC の他にも数多くの脆弱性情報データベースが存在する．CWE および ATT&CK に対応付ける方法を提案したい．

謝辞 本稿作成にあたりましては，SSR 産学戦略的研究フォーラムでの御助言を賜りました．ここに感謝の意を表します．また，本研究の一部は科学研究費補助金基盤研究(C)17K00475 の助成の下で行われました．記して謝意を表します．

参考文献

[1] The MITRE Corporation. Common Vulnerability and Exposures (CVE). https://cve.mitre.org/, 2020.
[2] National Institute of Standards and Technology. National Vulnerability Database (NVD). https://nvd.nist.gov/, 2020.
[3] The MITRE Corporation. Common Weakness Enumeration (CWE). https://cwe.mitre.org/, 2020.
[4] The MITRE Corporation. Common Attack Pattern Enumeration and Classification (CAPEC). https://capec.mitre.org/, 2020.
[5] Sugandh Shah, B. M. Mehtre.: An overview of vulnerability assessment and penetration testing techniques, Journal of Computer Virology and Hacking Techniques volume 11, pp. 27–49 (2015).
[6] Yugansh Khera, Deepansh Kumar, Sujay, Nidhi Garg.: Analysis and Impact of Vulnerability Assessment and Penetration Testing, Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Prespectives and Prospects, pp. 525-530 (2019).
[7] Lina Zhu, Zuochang Zhang, Guoen Xia, Caoqing Jiang.: Research on Vulnerability Ontology Model, Proceedings of the IEEE 8th Joint International Conference on Information Technology and Artificial Intelligence, pp. 657-661 (2019).
[8] Jian-bo Gao, Bao-wen Zhang, Xiao-hua Chen, Zheng Luo.: Ontology-based model of network and computer attacks for security assessment, Journal of Shanghai Jiaotong University (Science), 18(5), pp. 554-562 (2013).
[9] Wang, J.A., Wang, H., Guo, M., Zhou, L., Camargo, J.: Ranking attacks based on vulnerability analysis, Proceedings of the 43rd Hawaii International Conference on System Science, pp. 1–10 (2018).
[10] Ratsameetip Wita, Nattanatch Jiamnapanon, Yunyong Teng-amnuay.: An Ontology for Vulnerability Lifecycle, Proceedings of the Third International Symposium on Intelligent Information Technology and Security Informatics, pp. 553-557 (2010).
[11] Ehsan Aghaei, Ehab Al-shaer.: ThreatZoom: Neural Network for Automated Vulnerability Mitigation, Proceedings of the 6th Annual Symposium on Hot Topics in the Science of Security, No.24, pp. 1–3 (2019)