修士論文

脅威データベースのマッピングを用いた脆弱性検証補助システム

須藤 嵩

主指導教員 Beuran Razvan

北陸先端科学技術大学院大学
先端科学技術研究科
(情報科学)

令和 6 年 3 月

Abstract

The purpose of this research is to solve the problem of human error in the existing threat modeling field and the difficulty of implementation due to high introduction costs, by automatically estimating threat information from network design information. It is to support the implementation of modeling. First, we estimated potential threat information and mitigation measures from the threat database MITRE ATT&CK and network configuration information. In contrast to Rak et al.'s method, which is a research that automates existing threat modeling, we estimate threat information using MITRE ATT&CK, which is a more general threat database, to compare and analyze discovered threats. We compared the accuracy. As a result, we confirmed that our method had a vulnerability detection performance similar to that of existing methods. Regarding the input of design information for network systems, we were able to use fewer features compared to existing methods. This made it possible to reduce work costs during the preparation stage for handling the automation system. By using MITRE ATT&CK in the vulnerability database, we succeeded in adding mitigation, subtechniques, and tactics to techniques. This can be expected to make it easier to counter vulnerabilities using mitigation compared to using a threat list consisting only of techniques. Also, by knowing the tactics in advance, it became possible to predict the attacker's objectives. One of the future challenges for this research is that the threat database is only MITRE ATT&CK, so it is difficult to compare the results with other methods that use different vulnerability databases. Improvements can be expected by using a mapping technology called BRON. The second challenge for the future is that by simplifying the design information of the network system, a lot of unnecessary vulnerability information will be detected. In the current method, design information is input manually by humans. By automating this process, it may be possible to refine the input information and narrow down the detected threat information.

目 次

図 目 次

表 目 次

第1章　序論

1.1　研究背景

昨今、企業活動においてITシステムの導入が盛んに行われている．こうした背景には、新型コロナウイルスの感染拡大によりオンライン化が必要とされ，社外からでも接続できるようなITシステムの導入が進んだことが挙げられる．加えて、企業競争により提供するサービスの品質が向上した事で，データやデジタル技術を製品やサービスに取り込んだDX化が進んだことが挙げられる．このような時代背景や利便性により，企業活動におけるITシステムの重要性は増加している．それに伴い，企業の機密情報を狙ったサイバー攻撃の増加が問題視されている．警察庁によれば(図1.1)2023年のサイバー空間における脆弱性探索行為とみられる無差別送信パケットのアクセス件数は1日・1IPアドレスあたり8219件と前年と比べて5.4%増加しており[1]、サイバー攻撃を原因とする被害は近年も問題視されている．こうした現状により，ITシステム設計におけるネットワークセキュリティの重要性は増加していると言える．一般的に，ITシステムの設計において致命的な脆弱性が発生する理由には、コーディングミスなどの偶発的なヒューマンエラーに加えて，特定の通信プロトコルの使用や不適切な権限の配置など設計上の潜在的な危険性によるものがある．

後者のようなシステムの構成情報から事前に予見できるような脆弱性について，

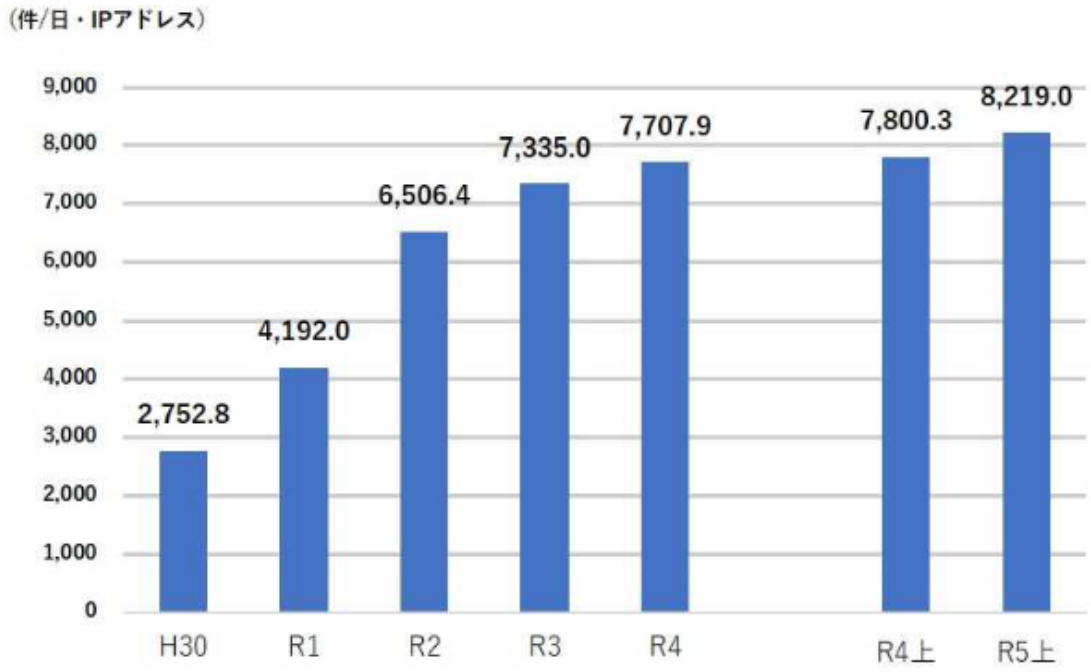

図 1.1: 警察庁 無差別パケットの受信件数 [1]

構築するシステムの定義や攻撃者の目的や方法の仮定を行い，想定される脅威を予め把握することで，設計前の段階から対策を講じる手法を脅威モデリングと呼ぶ [2]．一般的にシステム開発の工程が進むほど新たに発見された脆弱性へのフォローは難しくなるため，脅威モデリングを行い予め対策を講じることでより安全性の高いシステム設計を行うことが出来る．こうした脅威モデリングに関する取り組みはネットワークセキュリティに関して専門的な知識を要するため，一般的に外部に委託されることが多い．しかしながら，アジャイル開発と呼ばれるような開発からリリースまで短い期間で行うような現場も多く，そのような場合設計・開発を並行して進める手法を取られる事がある．以上のように専門家を雇い脆弱性や脅威の検出を事前に行う事は技術面・コスト面など複数の点で困難を伴うため，設計段階から詳細に脆弱性の分析を行う事は難しい現状がある．

1.2 研究目的

本研究の目的は，ネットワークシステムの提供者を対象とし，脅威モデリング導入の支援を行う事である．そのためにネットワークシステム設計情報から潜在的脅威および軽減策情報を提供し，脆弱性検証の補助を行うシステムを開発する．このシステムではまずネットワークシステムの構成要素を入力として，MITRE ATT&CK をベースとして想定される攻撃者の戦術および手法を出力する．攻撃者の手法と関連のあるネットワークシステム上の欠陥やセキュリティ攻撃パターン，軽減策から脅威診断書を作成し，システム設計者が脅威モデリングを行うことを支援する．このシステムを実現するために必要な工程は以下の 2 段階に分かれる．

1. MITRE ATT&CK を用いた脅威検索
2. 脅威診断・分析システムの開発

1.3 研究成果

本研究では脆弱性自動予測システムである AVPS を作成し，システム設計者が脅威モデリングを行う事を支援する．既存の脅威モデリングの自動化を行う研究手法と比較して次のような功績が得られた．

- 脅威の推量に用いるキー値の削減．これによりシステムを使用する準備の段階での作業コストの低下を実現した．
- 少ないキー値を用いて既存手法と同様の脆弱性の検知を行うことが出来た．

- 検出した脅威リストについて，MITRE ATT&CK を用いた高度な分析を行うことが出来た．tactics や mitigation 情報を付与する事により潜在脅威の軽減や二次被害への対策が期待できる．

1.4 本論文の構成

本論文は全 5 章で構成される．第 2 章では関連研究として脅威モデリングの自動化に関する研究，および脅威データベースのマッピングに関する研究を紹介し，既存の脅威モデリングの自動化領域における課題を提起する．第 3 章では研究手法について紹介し，作成するシステムの概要及び使用データセットに関する説明を行う．第 4 章では MITRE ATT&CK をデータセットとして用いた脆弱性の検知実験を行う．第 5 章で全体のまとめ，および今後の課題を示す．

第2章　関連技術

2.1　関連研究

2.1.1　脅威モデリングの自動化

脅威モデリングとは、ネットワークシステムの設計に際して、構築するシステムの定義や攻撃者の目的や方法の仮定を行うことで、事前に予見できるような脅威を予め把握することで、設計前の段階から対策を講じる手法である．脅威モデリングの実装は専門的な知識を要するため、導入コストが大きく実装が難しい．Xiong らの調査によれば、脅威モデリングは手動で行われる事例が多いため、多くの時間を要し人為的なミスも発生しやすい現状が報告されている [3]．また脅威モデリングの研究領域については多くは手動による手法の提案であり、自動化手法を用いた提案は比較的研究が進んでいない実情が報告されている [3]．既存の脅威モデリングに関する先行研究としては、脅威モデリングの支援をする手法や、人為的ミスなどの課題に対して予めネットワークシステムの構成要素を定義し、脅威データベースを作成することで脅威モデリングの自動実装を試みるものがある．Mohammad らは特定脅威の被害想定度やシステム使用者の IT セキュリティ理解度により危険度を数値化し、脅威モデリングの支援をする手法を提案した [4]．Rak らはオープンな脅威データベースである MITRE ATT&CK に基づいて作成された MITRE Knowledge Base を脅威データベースとして、予め作成したネットワークシステム

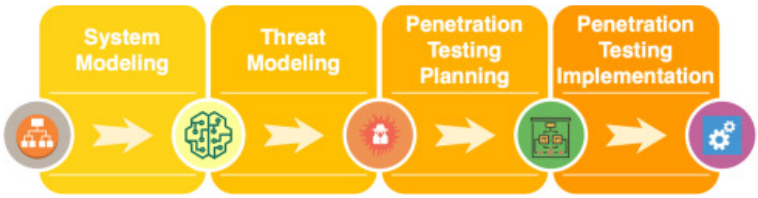

図 2.1: ESSecA 概要 [5]

モデルから想定される脅威の特定と攻撃計画のリストの作成を行う事で脅威モデリングの自動化を行う ESSecA システムを提案した (図 2.1) [5]．この提案手法はまずシステムへの入力となるネットワークシステムの分析を人間が行い、分析情報から MACM モデルで定義されるグラフシステムモデルを作成する．次に MACM モデルを入力として、ノードの asset 情報及びノード間の relationship 情報に基づき、脅威モデリングを行う．その後、分析結果を脅威リストとして出力し、出力脅威を実環境で検証する事を一連の工程としている．こうした研究もある一方で，脅威モデリングに関する研究は自動化の難易度の高さにより、依然として手動で行う手法による提案が多い．脅威モデリング自動化における課題とは完全な自動化は難しいため支援の段階に留まる手法が多いことや、単一の小規模脅威データベースを用いた脅威モデリングであるため、分析の精度が低いことが挙げられる．

2.1.2 脅威データベースのマッピング

オープンソースである複数の脆弱性データベースをリンクさせることで、脆弱性分析への応用を試みる研究が存在する．Erik らは MITRE ATT&CK, CVE, CWE, CAPEC など図 2.2 に掲載されるデータソースについて、図 2.3 に示されるように複数の脆弱性データベースの紐づけて BRON グラフを作成することで 各データベースの相互参照分析を行うコストを下げる手法を提案した [6].

脆弱性データソース	説明
ATT&CK Tactics	攻撃者の用いる戦略
ATT&CK Techniques	戦略を細分化した攻撃手法
CAPEC ATT&CK Patterns	攻撃手法を分類するためのパターン情報
NVD CWE CommonWeakness Enumeration	アーキテクチャや設計に関する脆弱性
NVD CVE Common Vulnerabilities and Exposures	ソフトウェアやアプリケーションに関する脆弱性
NVD CVE entry field: Known Affected Hardware or Software Configuration	脆弱性の影響をうける構成機器
NVD CVE entry field: Severity score	脆弱性の影響度を10段階で表したスコア

図 2.2: BRON 脅威データソース

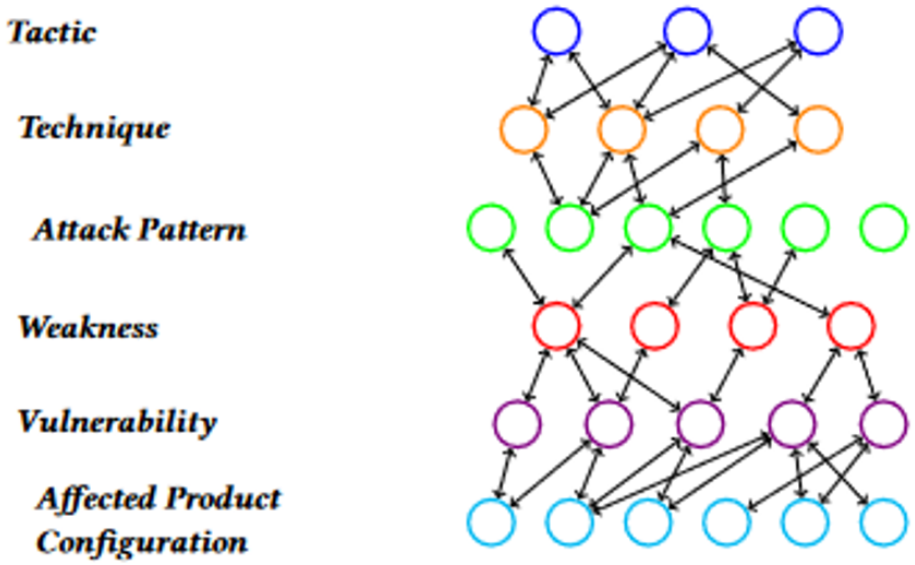

図 2.3: 脅威データマッピングの概要 [6]

2.1.3 本研究の位置づけ

既存の脅威モデリング領域における課題として，以下のような課題が想定される．

1 : 脅威モデリングは多くの工程を挟むため人為的ミスを誘発しやすい事 [3],

2 : 1 を解消するための自動化によるアプローチはあまり研究が進んでいない事 [3]

3 : 脅威モデリングの自動化事例は小規模の脅威データベースによる分析であるため精度が十分であるとは言えない事 [5]

脅威モデリングの自動化を行った Rak らの研究と比較して，脅威データベースとして広く用いられている MITRE ATT&CK 脅威データベースとして用いる事で，脅威モデリングの分析の観点において高度な情報支援を行う事を目指す．

2.2 脅威データベース

脅威データベースとは，既知の不正プログラムや悪意のある URL，既知の脆弱性情報など，脅威の名前や兆候を包括的に提供するデータベースを指す．本節ではそれらの概要について紹介する．

2.2.1 MITRE ATT&CK

MITRE ATT&CK とは MITRE 社が公開しているサイバー攻撃の手法を分類・体系化したオープンソースのフレームワークである [7]．過去に実際にあったサイバー攻撃の手法を分類し，図 2.4 の関係に基づき，以下の 5 種類の要素に基づいて構成されている．

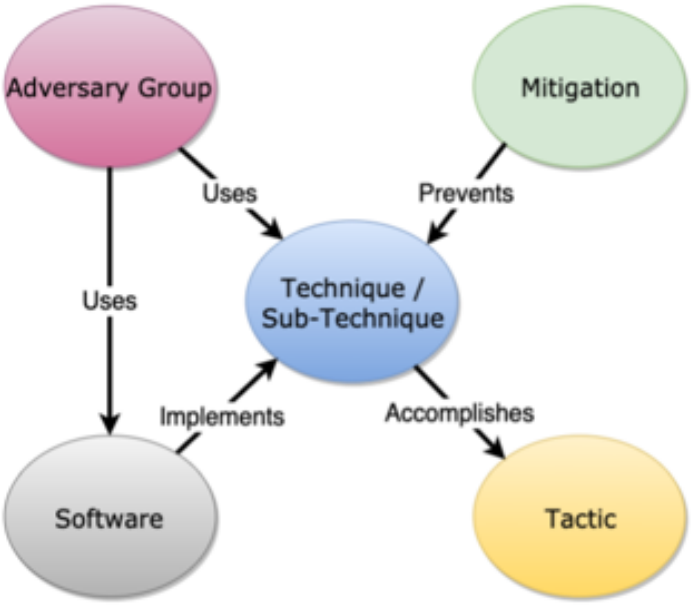

図 2.4: MITRE ATT&CK 概要 [7]

1. **Tactics**：攻撃によって達成される目的の一覧
2. **Techniques**：攻撃に用いられる技術や手法の一覧
3. **Groups**：攻撃者・攻撃グループの一覧
4. **Software**：攻撃に使用する攻撃ツールの一覧
5. **Mitigations**：攻撃をされた際の緩和策の一覧

Tactics

MITRE ATT&CK Tactics とは，ATT&CK データベースのうち，攻撃者の戦術・達成目標を一覧にしたデータベースである．ATT&CK データベースにおいて，Tactics は表 2.1 のような戦術が含まれる．それぞれの tactics を達成するための攻撃手段が techniques に当たる．ATT&CK データベースは tactics を頂点として，戦

術 (tactics)，攻撃手法 (techniques)，攻撃手法に関連する関連手法 (sub-techniques) の 3 つの階層的なデータ構造を形成している．

戦術 (tactics)	概要
初期アクセス (Initial Access)	ネットワークへの不正侵入
実行 (Execution)	悪意のあるコードの実行
永続化 (Persistence)	不正アクセスする環境の確保
権限昇格 (Privilege Escalation)	より高いレベルの権限の取得
防衛回避 (Defense Evasion)	防衛手段に対する検知回避
認証情報アクセス (Credential Access)	アカウント名とパスワードの不正取得
探索 (Discovery)	不正アクセス先の環境理解
水平展開 (Lateral Movement)	不正アクセス先の環境移動
収集 (Collection)	目標に関心のあるデータの収集
C&C（Command and Control）	侵害したシステムとの通信・制御
持ち出し (Exfiltration)	データの不正取得
影響 (Impact)	システムとデータの操作・中断・破壊

表 2.1: 戦術とその概要 [10]

Techniques

MITRE ATT&CK Techniques とは，ATT&CK データベースのうち，攻撃に用いられる技術や手法を一覧にしたデータベースである．Techniques 内の個別の攻撃技術について，攻撃名称や管理を行うための脅威番号，攻撃手法に関連する軽減策などの情報が与えられている．本実験で利用したキー値について解説を行う．

1. **name**：攻撃手法を表す名称.
2. **id**：各脅威情報を一意に定める脅威番号．例えば盗聴（Network Sniffing）を示す id は “attack-pattern–3257eb21-f9a7-4430-8de1-d8b6e288f529” である
3. **external_id**：可読性の高い脅威番号．例えば盗聴 (Network Sniffing) を示す external_id は “T1040” である．

4. **x_mitre_is_subtechnique**：攻撃手法が Sub-technique である場合 True を，そうでない場合は False を格納する．

Sub-Techniques

攻撃手法のうち，関連手法として techniques に従属する形で分類される攻撃手法を指す．例として phishing 攻撃に関連した sub-techniques を紹介する．

(T1566) **Phishing**：攻撃対象に対して攻撃の手掛かりになるメッセージを送り情報を得る攻撃手法

(T1566.001) **Spearphishing Attachment**：悪意のある添付ファイルを持つ電子メールを送り攻撃を行う手法

(T1566.002) **Spearphishing Link**：悪意のあるリンクを含む電子メールを送り攻撃を行う手法

(T1566.003) **Spearphishing via Service**：サードパーティのサービスを介してメッセージを送り攻撃を行う手法

(T1566.004) **Spearphishing Voice**：音声通信を使用してユーザーを操作し攻撃を行う手法

2.2.2 CVE

CVE (Commmon Vulnerabilities and Exposures) とは MITRE 社が公開しているオープンソースのデータベースであり，ソフトウェア製品・アプリケーションの脆弱性に対し，一意に定めるために与えられた世界共通の識別番号群である [8].

CVE 識別番号 (CVE-ID) は脆弱性の登録年号及び，一意に特定するための番号によって構成される．例として，CVE-2016-1018 は 2016 年に登録された Adobe Flash の脆弱性を示している．CVE には以下の 3 つの情報が与えられている [9].

1. **Description**：CVE-ID に割り当てられた脆弱性の概要．プログラム上のセキュリティ問題について記載されている．

2. **References**：CVE-ID に関連する URL 群．例えばプログラムに関連する製品の URL や，CVE の情報源の URL などが記載される．

3. **Status**：登録された脆弱性の審査状態を示す．ある CVE-ID について脆弱性であることが審査中の場合は Candidate, 報告された CVE-ID が脆弱性だと認められた場合は Entry のステータスが与えられる．

2.2.3 CWE

CWE (Common Weakness Enumeration) とは MITRE 社が公開しているオープンソースのデータベースであり，多種多様な脆弱性をいくつかの脆弱性タイプで分類し，CWE 識別子を用いて体系化した世界共通の識別番号群である [10]. 脆弱性タイプは (1) View, (2) Category, (3) Weakness, (4) Compound Element の 4 種類に分類される．脆弱性タイプごとに更にタイプが定義され，View として 22 個, Category として 105 個，Weakness として 638 個, Compound Element として 12 個，計 777 個の脆弱性タイプが分類されている．

1. **View**：特定の観点から脆弱性タイプをいくつか選択し，集めたもの．例として，NIST では実際に公表されている脆弱性を考慮して，CWE の中から

19 個の脆弱性タイプを選択して NVD に掲載している．

2. **Category**： 共通の特性を持つ脆弱性タイプをカテゴライズし，まとめたもの．

3. **Weakness**：個々の脆弱性を表したもの．以下の 3 属性が与えられている．

   (a) Class：最も抽象的な脆弱性の属性．

   (b) Base：特定のリソースや技術に依存しない脆弱性の属性．

   (c) Variant：個々のリソースや技術が特定できる脆弱性の属性．

4. **Compound Element**：複数の脆弱性による複合的な原因により引き起こされる脆弱性を表したもの．以下の 2 属性が与えられている．

   (a) Composite：複数の脆弱性が混合して発生する脆弱性の属性．

   (b) Chain：ある問題から連鎖的に発生した脆弱性の属性．

第3章　研究手法

本研究で作成したシステムの概要及び技術的な背景知識について説明を行う．

3.1　作成するシステム

本研究で作成したシステムの概要を図3.1に記載する．本システムは以下のような過程で脅威モデリングの支援を行う．

- 脅威情報の登録

  (1-1) 脅威情報入力：脅威情報を脅威モデリングシステムに登録する．システムに入力されるネットワークシステム設計情報はListing3.1のように記述される．Listing3.1の各キー値は先行研究であるRakらの実験を参考に設定した[5]．ネットワーク構成情報各キー値の詳細は次の通りである．

  1. name：構成要素の名称
  2. category：構成要素の基本的な属性
  3. host：node間での関係値．対象ノードに対してデータベースやアプリケーションサービスを提供する

4. connect：node 間での関係値．対象ノードに対してネットワークを提供する．

5. uses：node 間での関係値．対象ノードのサービスを利用する．

Listing 3.1: ネットワークシステム設計情報の抜粋

```
1   [
2   {
3       "name":"EmonBase",
4       "category":["server", "hardware", "device","host
            "],
5       "host":["database","mosquitto"],
6   },
7   {
8       "name":"radionetwork",
9       "category":["radio","network","Internet"],
10      "connect":["emonth","emontx","EmonBase"]
11  },
12  {
13      "name":"emoncms",
14      "category":["web","service", "software", "
            RaspberryPi"],
15      "uses":["mosquitto","database","emonth","emontx
            "]
16  }
17  ]
```

(1-2) データ登録：入力として受け取ったネットワークシステム構成情報を診断書作成システムに登録する．

- 脅威モデリングの実行

(2-1) 脅威・軽減策検索：脅威モデリングシステムに渡されたシステム設計情報のうち，構成要素の名称 (name), および属性 (category) を用いて，脆弱性データベースと脅威の照合を行い，脅威情報を獲得する．

(2-2) データ登録：登録する脅威情報として今回はネットワークシステム設

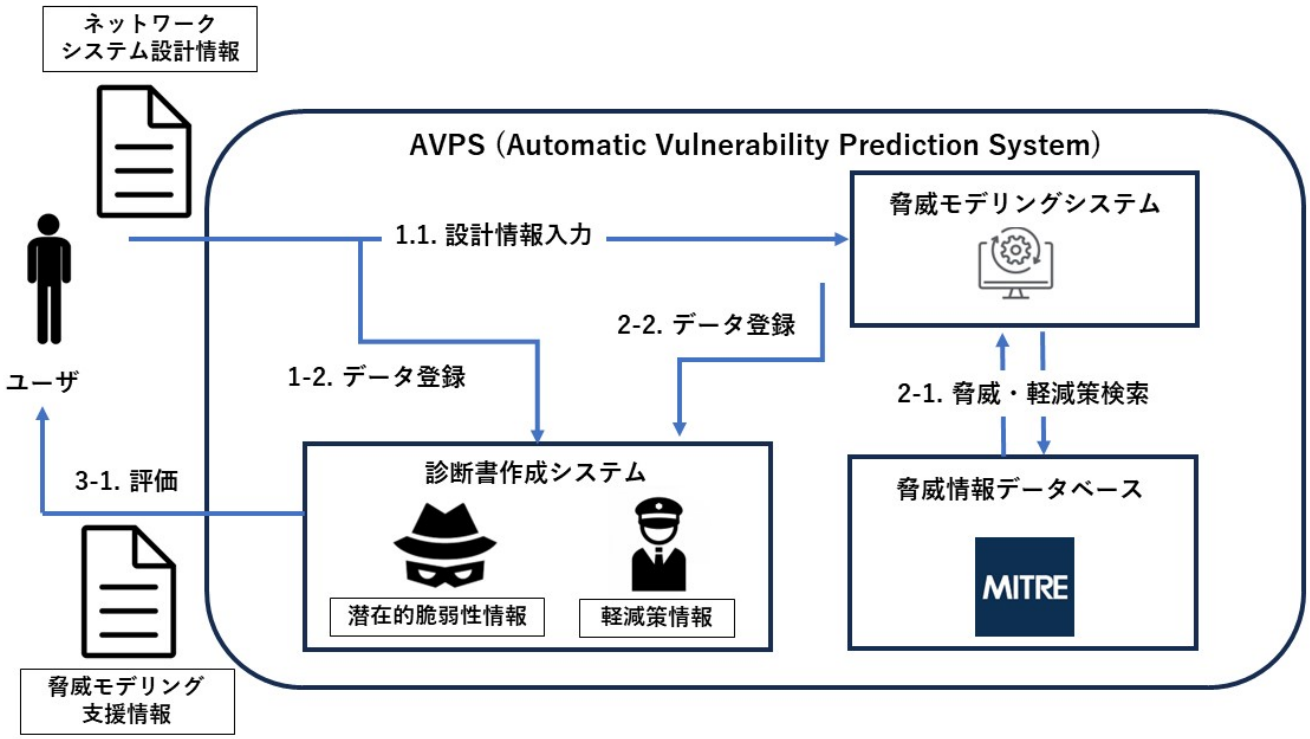

図 3.1: 提案システム：AVPS

計情報から想定される潜在的脆弱性 (techniques)，およびそれを軽減するための軽減策 (mitigation) , techniques から想定される攻撃者の戦略・目的 (tactics) を用いる．

- 脅威モデリング支援情報の出力と評価

(3-1) 評価：1-2, 2-2 で登録されたシステム設計情報，潜在的脆弱性，および軽減策情報から人間が脅威モデリングを支援する事を目的とした評価情報を出力する．

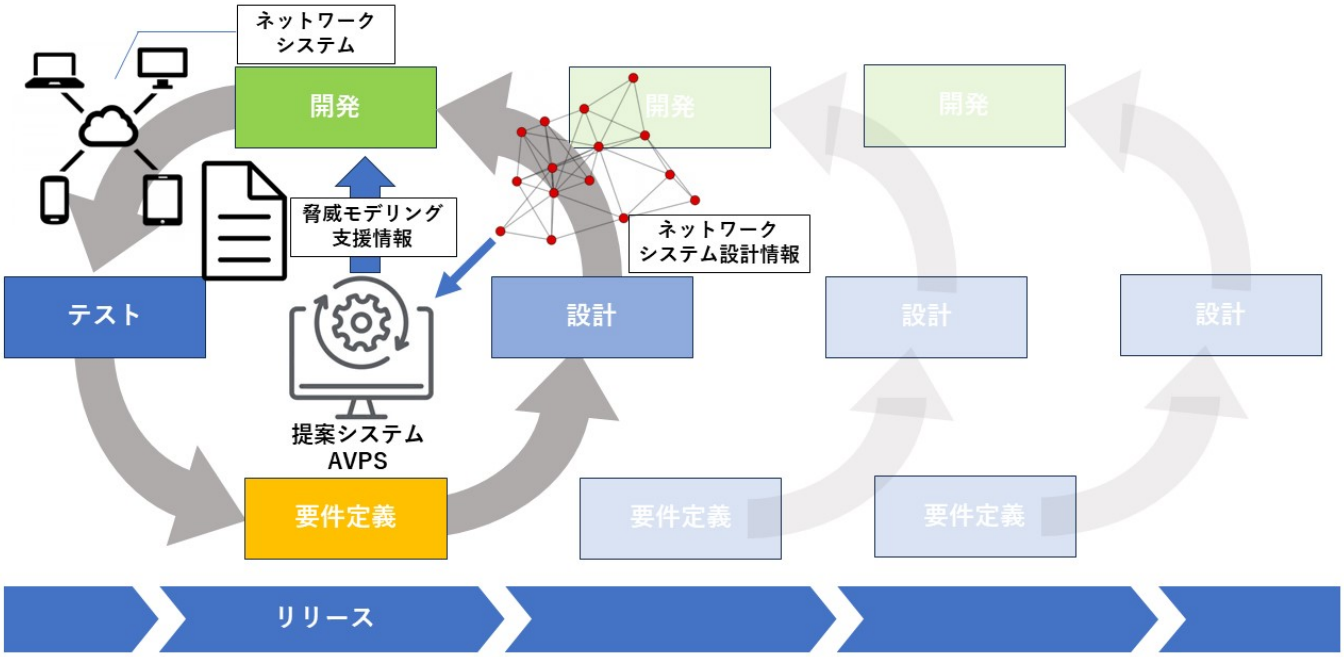

図 3.2: 研究で想定する状況

3.2 研究で想定する状況

システム設計者がネットワークシステムの開発に取り掛かる際に，低コストで脅威モデリングを導入したい状況を想定する．より具体的には，図 3.2 のようにアジャイル開発のような設計の工程に多くの時間を割けない場合にも，設計情報から脅威情報の類推を支援する事を目指す．本システムを使用するシステム設計者は設計の工程を予め終えており，システムのネットワークモデル情報を予め受け取っている状況を想定する．

3.3 実験で使用するデータベース

本実験では実験で使用したオープンソースの脅威データベースである MITRE ATT&CK の中でも企業向けのデータベースである，ATT&CK Enterprise を用いた．MITRE ATT&CK を使用するにあたり使用したモジュールについて紹介する．

3.3.1 mitreattack-python

MITRE ATT&CK を python 環境で動作させるためのモジュール．ソースコード内で使用した関数については次の通り．

1. **MitreAttackData**：引数の json ファイルから，MitreAttack オブジェクトを返す
2. **get_techniques**：techniques リストを返す
3. **get_tactics**：tactics リストを返す
4. **get_techniques_by_tactics**:引数で指定された tactics に従属する techniques リストを返す
5. **get_mitigations_mitigating_technique**：引数で指定された techniques に従属する mitigation リストを返す

3.4 脅威検索アルゴリズム

実験に使用したプログラムのうち，脅威モデリングの仕様を担うプログラムおよび MITRE ATT&CK に関連するモジュールについて紹介する．

3.4.1 潜在脅威取得関数

まず入力となるネットワークモデル情報を”component.json”として受け取る．このjsonファイルはRakらのシナリオ[5]で用いられたネットワークモデルの情報を引用したものである．今回はネットワークモデルの構成情報のうち，ノード本体の名称(name)及び機能名称(category)を参照した．次に脅威データベースであるMITRE ATT&CK から 脅威の情報として techniques，techniques の達成目標となる tactics を読み込む．techniques には脅威の詳細が記述されている description が存在する．本システムでは description に含まれる言葉のうち，構成情報に関連のある言葉を複数種類含む場合に潜在脅威であるとして脅威の出力を行う．

Listing 3.2: 潜在脅威取得関数

```
1  def threat_search():
2
3      with open('component.json', 'r') as f:
4          component_file = json.load(f)
5
6      components = []
7      for i in component_file:
8          components.append(i["name"])
9          components.extend(i["category"])
10
11     mitre_attack_data = MitreAttackData("enterprise-attack.json")
12
13     techniques = mitre_attack_data.get_techniques(
           remove_revoked_deprecated=True)
14     tactics = mitre_attack_data.get_tactics(
           remove_revoked_deprecated=True)
15
16     tactech_list = []
17     for i in tactics:
18         tactech = mitre_attack_data.get_techniques_by_tactic(i.
               x_mitre_shortname,'enterprise-attack',
               remove_revoked_deprecated=True)
19         tactech_list.append(tactech)
20
21     ID_tactech_list = []
```

```
22
23      for i in tactech_list:
24          sublist = []
25          for j in i:
26              sublist.append(j.external_references[0].external_id)
27          ID_tactech_list.append(sublist)
28
29
30      comtech_list = []
31
32      for i in components:
33          for j in techniques:
34              if i in j.description:
35                  comtech_list.append(j.external_references[0].
                        external_id)
36
37
38      common_list = []
39
40      for i in ID_tactech_list:
41          sublist = list(set(i) & set(comtech_list))
42          common_list.append(sublist)
43
44      M = 0
45
46      for i in common_list:
47          print(' There are %d techs related to %s' %(len(i),
                tactics[M].name))
48          print(i)
49          print('--------\n')
50          M = M + 1
51
52      M = 0
53      target = '.'
54      with open('data/threat_list.csv', 'w') as f:
55          writer = csv.writer(f)
56          for i in common_list:
57              for j in i:
58                  for k in techniques:
59                      tech_id = k.external_references[0].external_id
60                      if tech_id==j:
61                          if '.' in tech_id:
62                              idx = tech_id.find(target)
63                              subtech_parent=mitre_attack_data.
                                    get_object_by_attack_id(tech_id[:
                                    idx], 'attack-pattern')
```

```
64                          writer.writerow([tactics[M].name,
                                subtech_parent.name, k.name])
65                      else:
66                          writer.writerow([tactics[M].name, k.
                                name, 'none'])
67              M = M + 1
```

第4章　脆弱性検知実験

4.1　概要

本実験ではMITRE ATT&CKを用いた脅威モデリングの補助実験を行い，Rakら [5] の開発したESSecAシステムを用いた自動脅威モデリングの手法と比較した場合の分析結果を比較する．本実験では以下の工程で脅威の検出を行う．

1. 図4.1のネットワークモデルからシステムの構成情報を受け取る．
2. 脅威データベースと1で受け取ったシステムの構成情報を照合する．
3. 関連のある脅威・軽減策について出力し，ESSecAの結果と比較を行う．

4.2　実験方法

今回作成したAVPSシステムの動作手法や実験環境について紹介を行う．

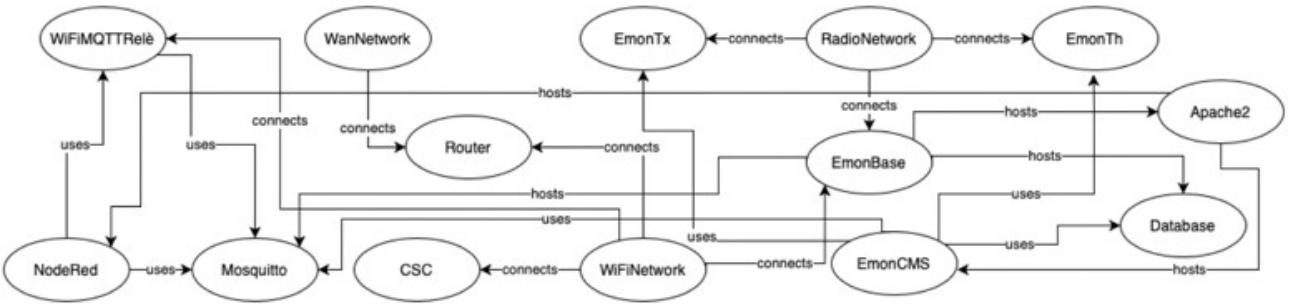

図 4.1: Rakらの実験で用いられたネットワークモデル [5]

4.2.1 分析対象

Rak らの研究で用いられた実験シナリオを脅威モデリングの分析対象とする [5]. なお入力とするネットワークモデル (図 4.1) については，各ノード間の接続関係など本来のネットワークモデルの情報を欠損しない形でデータセットに加工した．参考研究で使用されたネットワークモデルの情報を表 4.1 に記載する．それらのデータセットを元に本実験用に作成したデータセットは，Listing3.1 のように記述される．また作成したデータセットのうち，本実験で使用した特徴量を抜粋したものを表 4.2 に記載する．

Name	**Asset type**	**Labels**	**Description of asset type**
EmonBase	HW.Server	HW	A physical hosting hardware
EmonCMS	Service.Web	Network, LAN	Software (typically COTS) offered as a service
EmonTh, EmonTx	HW.IoTDevice	Service, SaaS	A physical hosting hardware
(Radio, WiFi) Network	Network.WiFi	Network, LAN	Network, the assets differs depending on the involved technologies
Mosquitto	Service.MQTT	Service, SaaS	Software (typically COTS) offered as a service
MQTTRelay	HW.IoTDevice	HW	A physical hosting hardware
NodeRED, Apache	Service.Web	Service, SaaS	Software (typically COTS) offered as a service

表 4.1: ESSecA で用いられたネットワークモデルの情報

4.2.2 実験環境

脅威データベースや実験システムを予め用意した vscode, linux ターミナル上で動作を行った．使用したシステムについては github で公開している．また作成し

Name	Category
EmonBase	server, hardware, host, device
EmonCMS	web,service, software, RaspberryPi
EmonTh	service,device,hardware,host
EmonTx	service,device,hardware,host
RadioNetwork	radio,network,Internet
WiFiNetwork	network,lan,wifi
Mosquitto	protocol, software, message
MQTTRelay	hardware, device, host
NodeRED	web,service, software, programming, edditer
Apache	Web, service, HTML, server

表 4.2: ネットワークモデル情報を圧縮したデータセット

たシステムについては python3 上で動作を行う．

4.2.3 実験手順

作成した実行ファイル (AVPS.py) を実行する事で，脅威モデリングを行うことが出来る．操作手順を以下にまとめる．

1. ネットワークモデルを参考にネットワークシステム設計情報を作成する．この時，設計情報の多くは先行研究のデータベースで使用されていたパラメータを引用した．

2. AVPS ファイルを実行し，システムをを動作させる．

3. 図 4.2 のように攻撃手法 (techniques) と軽減策 (mitigation) の組みで検出結果が出力される．検出された脅威情報および軽減策はターミナルに出力される．

4. 検出された脅威情報に対応する戦略 (tactics) および関連する攻撃手法 (sub-techniques) を示す，検出結果が実行ファイルに対して doctest-output / all-graph.gv.pdf に出力される．

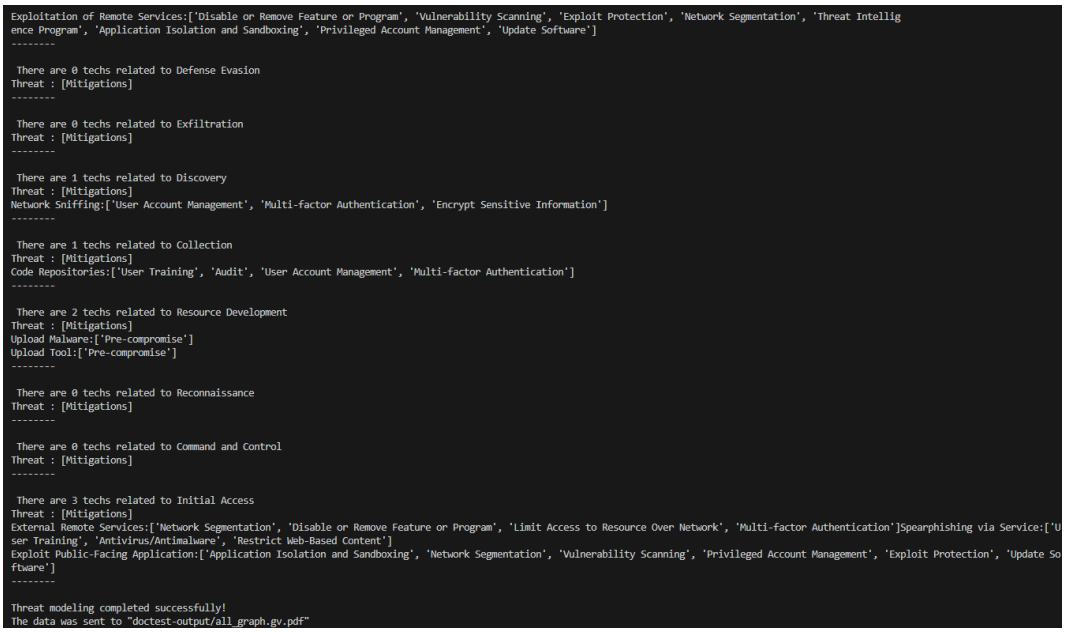

図 4.2: AVPS 実行中のターミナルへの出力

5. 脅威リストおよび関連情報について確認する.

4.3 結果

AVPS が，表 4.2 の各構成部品に対して検知した攻撃手法を表 4.3 に記載する．また検出された脅威情報に対して，subtechniques, tactics 情報、およびそれらに関する分析を付与した報告書を出力する．

Component	Threat
EmonBase	Compromise Hardware Supply Chain, Exploit Public-Facing Application, Traffic Signaling, Hardware, System Checks, Hybrid Identity, System Checks, Traffic Signaling, Hybrid Identity, Traffic Signaling, Endpoint Denial of Service, Network Denial of Service, Dynamic Data Exchange
EmonCMS	External Remote Services, Spearphishing via Service, Exploit Public-Facing Application, Upload Malware., Upload Tool, Code Repositories, Network Sniffing, Exploitation of Remote Services, External Remote Services, Compromise Client Software Binary, Service Exhaustion Flood
EmonTh, EmonTx	Exploit Public-Facing Application, Traffic Signaling,Device Driver Discovery, System Time Discovery, Network Service Discovery, System Checks, Network Sniffing, System Information Discovery, Traffic Signaling, System Checks, Traffic Signaling, Network Denial of Service, Endpoint Denial of Service, Native API, Steal or Forge Authentication Certificates, Multi-Factor Authentication Interception
Radio Network, WiFi Network	Hardware Additions, Socket Filters, CDNs, Client Configurations, Malvertising, Upload Malware, Acquire Access, ARP Cache Poisoning, Network Sniffing, System Checks, Network Provider DLL, Network Device Authentication, Patch System Image, Socket Filters, Obfuscated Files or Information, System Checks, Kernel Modules and Extensions, Network Provider DLL, Network Device Authentication, Socket Filters, Kernel Modules and Extensions, Cloud API, Network Sniffing, Network Device Authentication, ARP Cache Poisoning, Password Guessing, Forced Authentication
Mosquitto	Exploit Public-Facing Application, Spearphishing, Mail Protocols, Internal Proxy, Non-Standard Encoding, Ingress Tool Transfer, Develop Capabilities Malware, Obtain Capabilities, Malware, DHCP Spoofing, LLMNR/NBT-NS Poisoning and SMB Relay, Network Device Configuration Dump, Network Share Discovery, Network Sniffing, Exfiltration Over Alternative Protocol, Spoof Security Alerting, Patch System Image, Install Root Certificate, Remote Service VNC, Lateral Tool Transfer, Service Exhaustion Flood, XPC Services, DHCP Spoofing, LLMNR/NBT-NS Poisoning and SMB Relay, Steal Application Access Token, Network Sniffing
MQTT Relay	Mail Protocols, Non-Standard Encoding, LLMNR/NBT-NS Poisoning and SMB Relay, ARP Cache Poisoning, DHCP Spoofing, Remote Services VNC, Inter-Process Communication XPC Services, ARP Cache Poisoning, DHCP Spoofing
NodeRED, Apache	Exploit Public-Facing Application, Spearphishing via Service, External Remote Services, Upload Tool, Broken Upload Malware, Code Repositories, Network Sniffing, Exploitation for Defense Evasion, Exploitation of Remote Services, Exploitation for Privilege Escalation, Compromise Client Software Binary, External Remote Services, Service Exhaustion Flood, Native API, Network Sniffing, Network Denial of Service, Service Execution, Serverless Execution, LSASS Memory, LLMNR/NBTNS Poisoning and SMB Relay, Steal Application Access Token, Forge Web Credentials

表 4.3: 提案システム AVPS における 図 4.1 のシナリオを用いた脅威リスト [5]

4.3.1 脅威情報報告書

実際に出力された報告書のうち，Emonbase に関する報告書を図 4.3 に掲載する．報告書には以下の情報を含む．

1. Tactics 毎に想定される脅威情報

2. 危険性の高い Tactics. 出力される Techniques の数の多い Tactics を脅威の高い戦略と分析して出力する

3. 軽減効果の高い Mitigation. Mitigation のうち, より多くの Techniques を軽減することに期待できる Mitigation を より軽減効果を見込めると分析し，出力する．

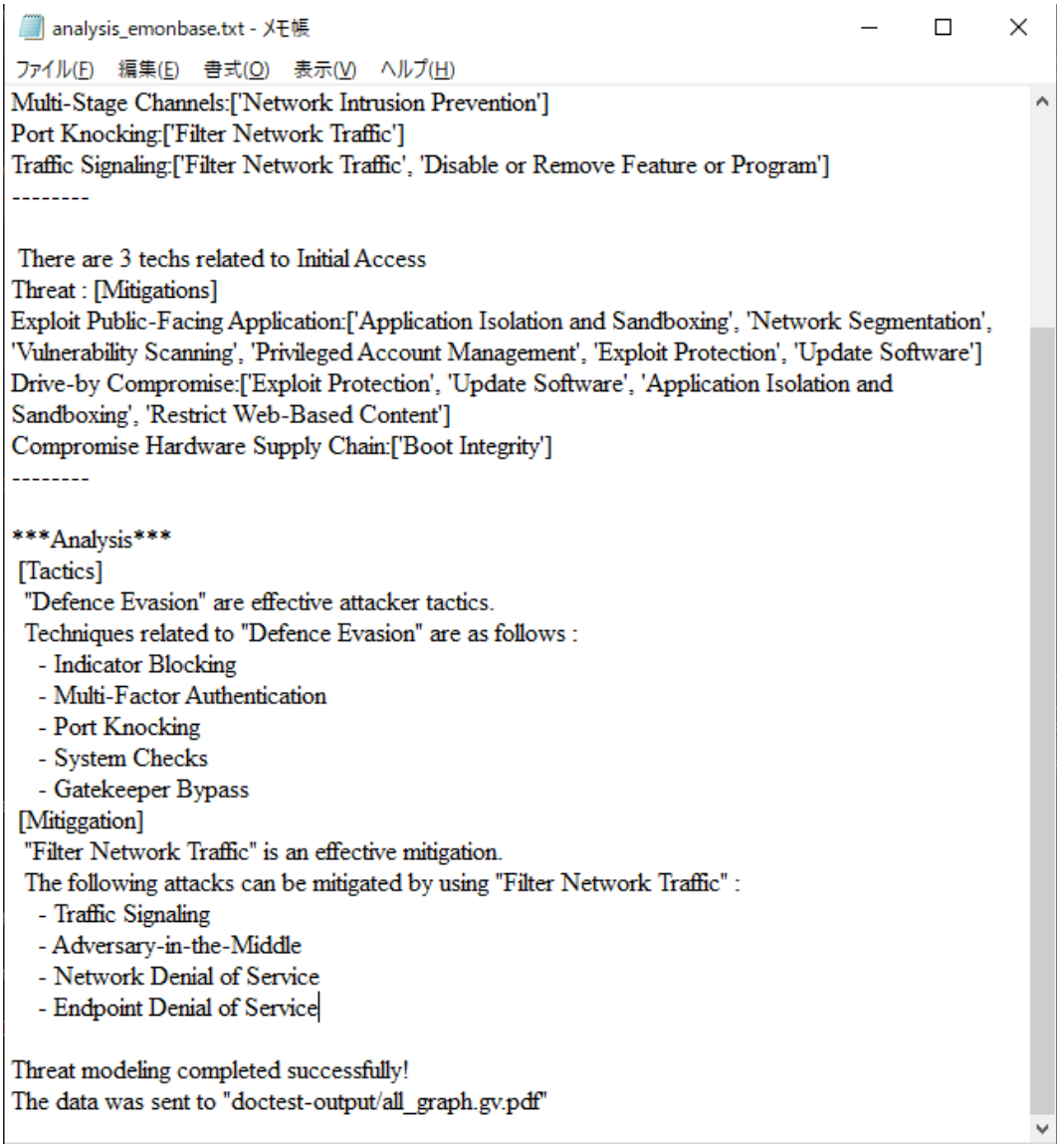

図 4.3: 出力される脅威の報告書

4.3.2 脅威情報グラフ

構成要素ごとの脅威出力のうち，EmonBase に関するグラフを図 4.4 に，Mosquitto に関するグラフを図 4.5 に記載する．

グラフのノードはグラフ中の左から以下の情報を示す．

1. MITRE ATT&CK データベース の Tactics 情報を示す．

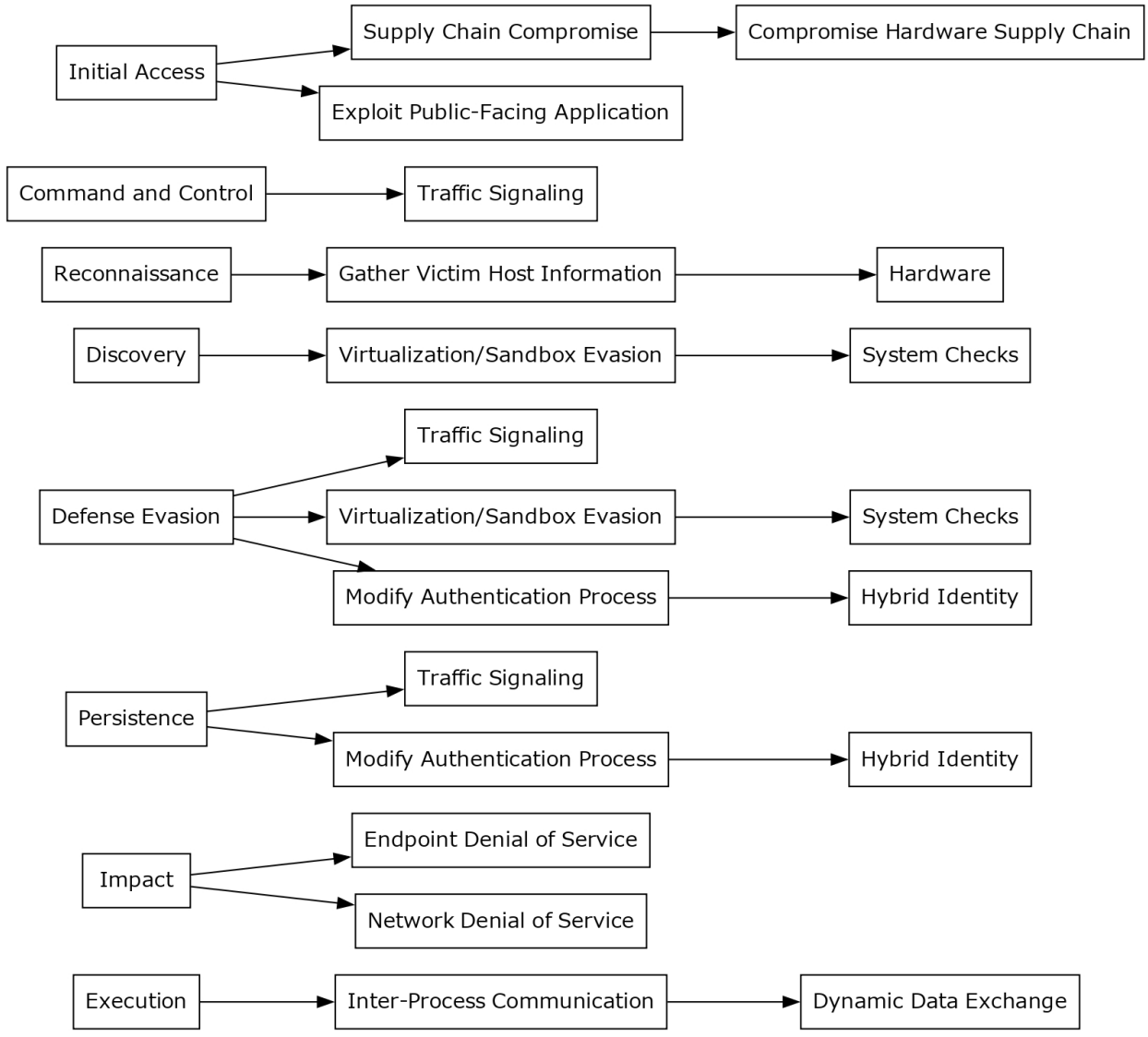

図 4.4: 提案システム AVPS を用いた EmonBase の脅威出力結果

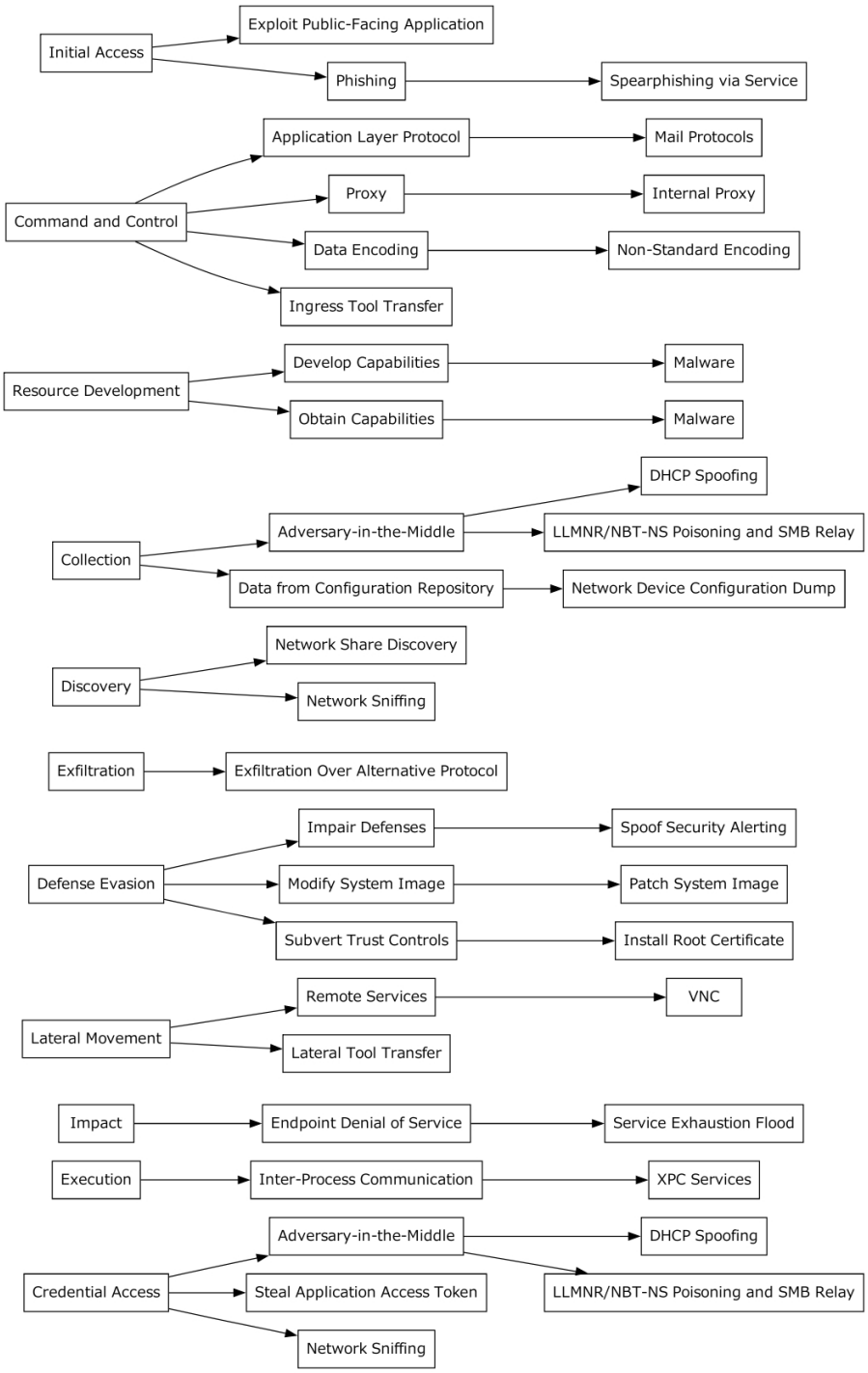

図 4.5: 提案システム AVPS を用いた Mosquitto の脅威出力結果

2. MITRE ATT&CK データベース の Techniques 情報を示す.

3. MITRE ATT&CK データベース の Techniques 情報を示す.

脅威情報・軽減策情報についてはソースコード実行時に報告書に出力されるものと同様の情報がターミナルに出力される.

4.4 評価

今回開発した AVPS について，以下の観点で評価を行う.

1. 先行研究で検知できた脅威についてもれなく検知できているか.

2. 先行研究での検知結果と比べて出力脅威の妥当性はどの程度あるか.

3. 先行研究と比べて本手法の新規性はどこにあるか.

4.4.1 包括性の検証

以下の観点で包括性の検証を行う．先行研究である ESSecA のシナリオのうち，EmonBase と Mosquitto の二つの構成要素に関する脅威情報について比較を行った．ESSecA は MITRE とは異なるデータセットを用いているため，完全一致の比較はできない．従って ESSecA から検出された脅威のうち，MITRE ATT&CK と類似しているかどうかを主観で判断し，評価を行った．EmonBase と Mosquitto のそれぞれについて，ESSecA が出力した脅威リストをそれぞれ表 4.4，4.5 に示す．先行研究で検出された脅威のうち，Emonbase から検出される脅威について AVPS との類似性を比較した結果を表 4.4 に示す．同様に Mosquitto から検出される脅威について AVPS との類似性を比較した結果を表 4.5 に示す．それぞれの構成要素か

ら推定される脅威に対して，7 割以上の精度で同様の脅威が出力できていることが確認される．ESSecA で出力されたすべての脅威についてマッピング精度を検証した結果を表 4.8 に示す．マッピング精度の比較により，先行研究で検出した脅威のうち本システムでも検出できた脅威の割合は 0.70 にあたる．

脅威名称	説明
Communication Lock	通信プロトコルの封鎖
Eavesdropping	通信プロトコルを利用した盗聴
Action Spoofing	非公開のトピックへのアクセス
Message Tampering	送信パケットの盗聴，書き換え
Data leakage	データの漏洩

表 4.4: ESSecA が EmonBase に対して脅威モデリングを行った場合の脅威リスト

脅威名称	説明
Communication Lock	通信プロトコルの封鎖
Eavesdropping	通信プロトコルを利用した盗聴
Action Spoofing	非公開のトピックへのアクセス
Message Tampering	送信パケットの盗聴，書き換え
Data leakage	データの漏洩

表 4.5: ESSecA が Mosquitto に対して脅威モデリングを行った場合の脅威リスト

脅威名称	AVPS で検出した類似脅威
Data leakage	Gather victim host information
Denial of Service	Endpoint Denial of Service
Impersonation	Network Denial of Service
Device isolation	類似脅威の検出はなし

表 4.6: EmonBase について AVPS が ESSecA の検出結果を包含している事の検証

脅威名称	AVPS で検出した類似脅威
Communication Lock	Command and Control, Application Layer Protocol, Mail Protocol
Eavesdropping	Network Sniffing
Action Spoofing	DHCP Spoofing
Message Tampering	類似脅威の検出はなし
Data leakage	Steal Application Access Token

表 4.7: Mosquitto について AVPS が ESSecA の検出結果を包含している事の検証

Component	**AVPS での検出脅威数**	**ESSecA で検出した脅威の検出割合**
EmonBase	12	0.75 (3/4)
EmonCMS	11	0.60(3/5)
EmonTh, EmonTx	16	0.66(6/9)
Radio Network, WiFi Network	27	0.62(8/13)
Mosquitto	24	0.80(4/5)
MQTTRelay	10	0.80(4/5)
NodeRED, Apache	22	0.78(7/9)
Total	97	0.71(33/46)

表 4.8: ESSecA との出力脅威の割合の比較

4.4.2 分析精度の比較

本システムで検知できた脅威のうち，先行研究 ESSecA では検知できなかった脅威の妥当性について評価する．ESSecA で検知された脅威を表 4.9 に記載する．まず AVPS の出力脅威のうち ESSecA とマッピングが行われた脅威を取り除いた脅威リストを表 4.10 に記載する．表 4.10 内の太字で記載した脅威は，検出脅威のうち危険性の高い脅威や ESSecA では検出できなかった種類の脅威であると私が判断したものである．表より多くの脅威が太字で表されるように新たな危険性を本システムを用いて検出できることが分かる．AVPS を用いて脅威モデリングを

行うことで，ESSecA では検知できず，新たな脅威の可能性を示唆するような脆弱性を予め知ることができる．

Component	Threat
EmonBase	Data Leakage, Denial of Service, Impersonation, Device isolation
EmonCMS	Denial of Service, Injection, Sensitive Data Exposure, Broken Auth., Broken Access Control, Insecure Deserialization, Functionality Misuse, Web Communication Channel Manipulation, System Manipulation
EmonTh, EmonTx	Denial of Service, Impersonation, Data Leakage, Exhaustion of Power, Device isolation
Radio Network, WiFi Network	Eavesdropping, Message Elimination, Message Modification, Message Injection, Network Partitioning, Selective Forwarding, Denial of Service, Spoofing, Communication Lock, Message Reply, Topology Disclosure, Network Abusive Access, Jamming
Mosquitto	Communication Lock, Eavesdropping, Action Spoofing, Message Tampering, Data Leakage
MQTT Relay	Impersonation, Denial of Service, Data Leakage, Eavesdropping, Device message tampering
NodeRED	Denial of Service, Injection, Sensitive Data Exposure, Broken Auth, Broken Access Control, Insecure Deserialization, Functionality Misuse, Web Communication Channel Manipulation, System Manipulation
Apache	Denial of Service, Injection, Sensitive Data Exposure, Broken Auth., Broken Access Control, Insecure Deserialization, Functionality Misuse, Web Communication Channel Manipulation, System Manipulation

表 4.9: 先行研究 EESecA における 図 4.1 のシナリオを用いた脅威リスト [5]

Component	Threat
EmonBase	**Compromise Hardware Supply Chain**, Exploit Public-Facing Application, System Check, System Checks, **Traffic Signaling**, Hybrid Identity, **Dynamic Data Exchange**
EmonCMS	External Remote Services, **Spearphishing via Service**, Upload Tool, Code Repositories, Exploitation of Remote Services, External Remote Services, Compromise Client Software Binary, Service Exhaustion Flood
EmonTh, EmonTx	**Traffic Signaling**,Device Driver Discovery, System Time Discovery, **System Information Discovery**, **System Checks**, Native API, Multi-Factor Authentication Interception
Radio Network, WiFi Network	Hardware Additions, Socket Filters, CDNs, Client Configurations, Malvertising, Upload Malware, Acquire Access, ARP Cache Poisoning, System Checks, Network Provider DLL, Network Device Authentication, Patch System Image, Socket Filters, Obfuscated Files or Information, System Checks, Kernel Modules and Extensions, Network Provider DLL, **Network Device Authentication**, Socket Filters, Kernel Modules and Extensions, Cloud API, Network Sniffing, Network Device Authentication, ARP Cache Poisoning, Password Guessing, Forced Authentication
Mosquitto	**Exploit Public-Facing Application**, Spearphishing, Mail Protocols, Internal Proxy, Non-Standard Encoding, Ingress Tool Transfer, Develop Capabilities Malware, Obtain Capabilities, Malware, DHCP Spoofing, LLMNR/NBT-NS Poisoning and SMB Relay, Network Device Configuration Dump, Network Share Discovery, Network Sniffing, Exfiltration Over Alternative Protocol, **Spoof Security Alerting**, Patch System Image, Install Root Certificate, Remote Service VNC, Lateral Tool Transfer, Service Exhaustion Flood, XPC Services, DHCP Spoofing, LLMNR/NBT-NS Poisoning and SMB Relay, Steal Application Access Token, Network Sniffing
MQTT Relay	Mail Protocols, **Non-Standard Encoding**, LLMNR/NBT-NS Poisoning and SMB Relay, ARP Cache Poisoning, DHCP Spoofing, Remote Services VNC, Inter-Process Communication XPC Services, ARP Cache Poisoning, DHCP Spoofing
NodeRED, Apache	Exploit Public-Facing Application, Spearphishing via Service, External Remote Services, Upload Tool, **Broken Upload Malware**, Code Repositories, Network Sniffing, Exploitation for Defense Evasion, Exploitation of Remote Services, Exploitation for Privilege Escalation, Compromise Client Software Binary, External Remote Services, Service Exhaustion Flood, Native API, Network Sniffing, Network Denial of Service, Service Execution, Serverless Execution, LSASS Memory, LLMNR/NBTNS Poisoning and SMB Relay, Steal Application Access Token, Forge Web Credentials

表 4.10: AVPS でのみ検出された脅威リスト [5]

4.4.3 新規性の比較

先行研究と比較した場合の本システムの新規性として，AVPS システムは検出した攻撃手法に関連する軽減策および戦術，サブテクニックの検出が可能である．これにより，ネットワーク設計者に対して脅威の推量に上乗せして，脅威モデリング分析を助ける情報支援が可能である．加えて図 4.3 で出力されるような脅威の分析および，図 4.4 に代表されるような脅威グラフの作成により視覚的・感覚的な理解を促すことができる．これにより専門性を一段階下げたユーザーに対しても理解のしやすい情報の提供が可能になる．

第5章　まとめと課題

5.1　本研究の功績

本研究の目的は，既存の脅威モデリングの領域において人為的ミスを生じやすい事や，導入コストの高さによる実装の困難性に対し，ネットワーク設計情報から自動で脅威情報の推定を行う事で脅威モデリングの実装を支援する事である．まず脅威データベースである MITRE ATT&CK とネットワーク構成情報から潜在的脅威情報および軽減策の推定を行った．既存の脅威モデリングの自動化を行う研究である Rak らの手法に対して，より一般的な脅威データベースである MITRE ATT&CK を用いて脅威情報の推定を行う事で，発見脅威の比較や分析精度の比較を行った．その結果次のような功績が得られた．

1. ESSecA で検知できた脅威について，本システムでも類似した脆弱性の検知を行うことが出来た．

2. ネットワーク設計情報の読み込みにあたり，既存の手法と比較して脅威の推量に用いる特徴量を少なく抑える事が出来た．これにより，自動化システムを扱う準備の段階での作業コストを下げる事が出来た．

3. AVPS で検知できた techniques について，MITRE ATT&CK を用いて mitigation を加えた分析を行うことが出来た．これにより techniques のみで構

成された脅威リストと比較して，システム設計者に対して軽減手法を交えたより高い脅威モデリング支援の効果が期待できる．

4. 得られた脅威リストについて，sub-techniques および，tactic 情報を付与することが出来た．これにより特定の攻撃手法に対して，関連する sub-techniques 情報の支援により精度の高い分析を行うことが出来る．加えて，techniques に従属する tactics 情報の支援により，攻撃者の目的についての推量が可能になった．

5.2 今後の課題

以下に残された課題についてまとめる．

1. ESSecA との実験結果の比較について，使用した脅威データベースの違いから脅威リストの厳密な比較を行う事は難しい．これを解決する手段として，異なる脅威データベース間でのマッピングを行う BRON と呼ばれるシステムが存在する．今回作成した AVPS システムに対して BRON システムを適用し，複数の脅威データベースを交えた分析を行う事でより精度の高い分析を行うことが期待できる．

2. ネットワークシステム設計情報の入力において，入力情報のフォーマットに用いる特徴量を減らすことが出来た反面，検出される脅威情報が増大してしまい，検出される脅威情報の絞り込みが甘くなってしまった．入力情報の自動化を試みる事で，入力される設計情報の特徴量を増やしながら作業量の易化に期待することができる．

参考文献

[1] 警察庁, 令和５年上半期におけるサイバー空間をめぐる脅威の情勢等について. https://www.npa.go.jp/publications/statistics/cybersecurity/

[2] SOMPO CYBER SECURITY, 脅威モデリングとは, https://www.sompocybersecurity.com/column/glossary/threat-modeling

[3] Xiong Wenjun and Robert Lagerstrom. Threat modeling – A systematic literature review, Computers & Security, Volume 84, July 2019, pp 53-69.

[4] Mohammad Ali Ramazanzadeh, Behnam Barzeggar, and Homayun Motameni. ASATM: Automated security assistant of threat models in intelligent transportation systems, IET Computers & Digital Techniques, Volume 16, Issue 5-6, 2022.

[5] Massimiliano Rak, Giovanni Salzillo, and Daniele Granata. ESSecA: An automated expert system for threat modelling and penetration testing for IoT ecosystems, Computers and Electrical Engineering, volume 99, April 2022.

[6] Erik Hemberg, Jonathan Kelly, et al. BRON – Linking Attack Tactics, Techniques, and Patterns with Defensive Weaknesses, Vulnerabilities and Affected Platform Configurations, arXiv:2010.00533v1, 2020.

[7] Blake E. Strom, Andy Applebaum, Doug P. Miller, et al. MITRE ATT&CK: Design and Philosophy, 2020. https://www.mitre.org/sites/default/files/2021-11/prs-19-01075-28-mitre-attack-design-and-philosophy.pdf

[8] CVEdetails.com, https://www.cvedetails.com/cvss-score-distribution.php

[9] IPA 共通脆弱性識別子CVE解説, https://www.ipa.go.jp/security/vuln/scap/cve.html

[10] NTT データ先端技術株式会社, “IPA MITRE ATT&CK その 1 〜概要〜”, https://www.intellilink.co.jp/column/security/2020/060200.aspx

[11] IPA 共通脆弱性識別子CWE解説, https://www.ipa.go.jp/security/vuln/scap/cwe.html

[12] IBM Security, データ侵害のコストに関する調査, 2023. https://www.ibm.com/account/reg/jp-ja/signup?formid=urx-52258