セキュリティ情報融合基盤 CURE

津田 侑[1]　井上 大介[1]　鈴木 宏栄[1]　高木 彌一郎[1]　田中 秀一[1]　金谷 延幸[1]　竹本 亜希[2]
古本 啓祐[1]

概要：
組織のセキュリティを向上させるため，多種多様なサイバーセキュリティ関連情報の利活用が求められる．サイバーセキュリティ関連情報には組織内ネットワーク上のアラート，無差別型攻撃や標的型攻撃対策に特化した個々の観測システムのデータ，外部組織が発信する脅威情報などがある．しかし，これらの収集した情報は性質や粒度・量・形式が異なり，情報間の繋がりを分析・検索するには多大な処理時間を要する．そこで本稿では異種かつ大規模なサイバーセキュリティ関連情報からデータ間の繋がりを高速かつ自動的に発見するために，セキュリティ情報融合基盤 CURE を提案する．CURE では Pub/Sub モデルに倣ったメッセージングで各観測システム上からデータを受信し，キー・バリュー型インメモリ DB に蓄積する．さらに観測データに関連する攻撃グループやマルウェア名，攻撃技術で意味付けし横断的に分析することで，大規模データからサイバー攻撃の隠れた構造を抽出する．本稿では CURE の設計・実装について述べ，横断分析で発見されたサイバーセキュリティ関連情報間の繋がりをケーススタディとして示す．

キーワード：観測データ，脅威情報，大規模データ，セキュリティ・オペレーション

CURE: Cybersecurity Universal Repository

Yu Tsuda[1]　Daisuke Inoue[1]　Koei Suzuki[1]　Yaichiro Takagi[1]　Hidekazu Tanaka[1]
Nobuyuki Kanaya[1]　Aki Takemoto[2]　Keisuke Furumoto[1]

Abstract: Utilizing cybersecurity-related data, which includes alerts from security appliances on an enterprise network, observation data on monitoring systems, cyber threat intelligence (CTI) shared from security specialists, etc., has been needed for improving security operation in an organization. These data differ in their characteristics, information granularities, quantities and formats each other. Consequently, it is hard to analyze the heterogeneous data then to find relations among the data. Therefore, we develop CURE (Cybersecurity Universal Repository), which makes it possible to aggregate huge amount of heterogeneous cybersecurity-related data and to rapidly search across the all data. CURE adopts publish-subscribe messaging model. When CURE receives observation data from each monitoring system, it stores the data into its key-value in-memory database. Then, CURE gives cybersecurity-related meaning (e.g. threat actor, malware, attack technique, etc.) to the data. In this paper, we describe the design and implementation of CURE, and present relations among heterogeneous data which we find through case study.

Keywords: Observation Data, Threat Intelligence, Big Data, Security Operation

1. はじめに

多岐にわたるサイバー攻撃への対策として，サイバー攻撃の特徴に合わせて構築された観測システムにより多種多様な観測データが収集され，組織のセキュリティ向上に活

[1] 国立研究開発法人情報通信研究機構
National Institute of Information and Communications Technology
[2] NTT アドバンステクノロジ株式会社
NTT Advanced Technology Corporation

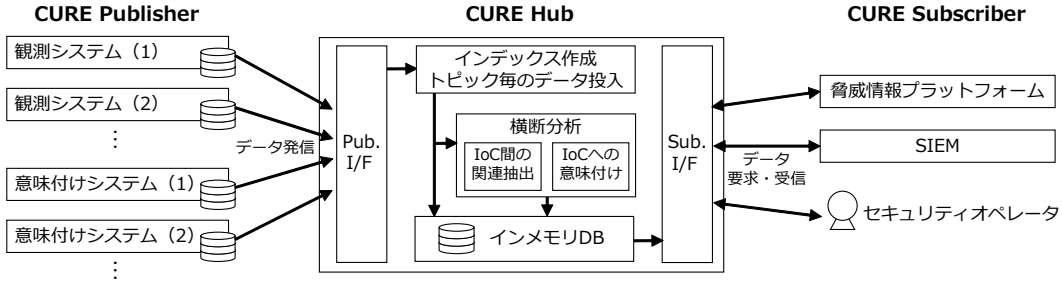

図 1 CURE のシステム概要

用される．たとえば，マルウェアの感染拡大活動の分析のためにはダークネット観測や各種ハニーポットによる観測が効果的である．組織内ネットワークの観測にはネットワーク上やエンドポイントの複数個所にセキュリティ機器やソフトウェアを導入し，それらが発するアラートを蓄積する．セキュリティ・オペレーションでは，このように大規模に収集された多種多様な観測データから相互の関連性を分析し，組織の対策基準に合わせて対処すべき事象を発見・優先付けすることで早期の対処を実現する．

他方で，現在では脅威情報がセキュリティ専門家である個人や団体によってインターネット上で広く公開されるようになり，セキュリティ・オペレーションを迅速に進める上での脅威への理解や脅威ハンティングでの利活用が進められている [1,2]．脅威情報の発信には，IP アドレスやファイルハッシュ値などの IoC（Indicator of Compromise）や攻撃キャンペーンやマルウェア解析結果を解説した個人および団体のブログ記事，SNS 上での投稿など，商用・オープンソースの様々な媒体が利用されている．さらに，脅威情報を共有する仕組みも多く提案されている [3,4]．

このような脅威情報や先述の多種多様な観測データを含むサイバーセキュリティ関連情報を有効活用し，その間の繋がりを発見できるとより効果的なセキュリティ・オペレーションを実現できる．しかし，このような情報は各種観測システムや脅威情報サービス上に散在し，その中の情報の性質や粒度，量が各々で差があるため横断的に分析することは容易ではない．さらに，多くの場合においてサイバーセキュリティ関連情報は各々が大規模なデータベースであり，その中からサイバーセキュリティ関連情報の関連性を分析・検索するためには多大な処理時間を要する．

そこで本稿では，多種多様かつ散在しているサイバーセキュリティ関連情報の中からその繋がりを自動的かつ高速に発見するために，データを大規模に集約・横断分析するセキュリティ情報融合基盤 CURE を提案する．CURE では，全てのデータをインメモリ DB に搭載し高速な検索・横断分析を可能にする．CURE が扱うデータの種類には，各々の観測システムが大規模に収集している**観測データ**と，観測データに対し関連する攻撃グループやマルウェア名，攻撃技術などのキーワードを付与するための**意味付けデータ**がある．CURE への各種観測データや意味付けデータの発信および，CURE 上のデータの検索・受信は CURE の API や Web インタフェースを介して行える．

本稿の構成は以下の通りである．CURE の設計を第 2 章，実装を第 3 章で述べる．また，性能評価を第 4 章に述べる．そして，CURE を用いたケーススタディを第 5 章に述べ，それを基に第 6 章で考察する．最後に第 7 章で関連研究を示し，第 8 章でまとめと今後の課題を述べる．

2. セキュリティ情報融合基盤 CURE

本章では，大規模データの横断分析や高速な検索を可能とする CURE の設計について述べる．

2.1 システム概要

図 1 に CURE のシステム概要を示す．CURE では Pub-/Sub 型メッセージングの構成に倣い，データ発信者の **CURE Publisher**, データ利用者の **CURE Subscriber**, データの横断分析機能を持ち CURE Publisher/Subscriber を仲介するインタフェースを提供する **CURE Hub** を主要素とする．CURE Publisher には観測データを扱うシステムと観測データに意味付けを与えるシステムの 2 種類がある．CURE Subscriber には SIEM や脅威情報プラットフォームなどの自動化された仕組みやセキュリティ・オペレータによる手動での利用を想定する．CURE はこれらの 3 要素が各々に独立動作する疎結合なシステム構成である．

また，CURE Hub で扱うデータは各 CURE Publisher が選定・要約した IoC や意味付けのキーワードのみに絞り，これらを後述のデータ構造で管理する．そして，データの完全性は各々の CURE Publisher で保証することを前提とし，CURE Hub 上のデータストアの容量が超過した際は参照されていない期間が長いものから順に消去する．こうした設計の割り切りにより，検索の高速性を獲得する．

次に，CURE におけるデータの流れについて説明する．まず，各 CURE Publisher は CURE Hub で提供されるインタフェースが要求するフォーマットに従いデータを成型する．このとき，各々の CURE Publisher は CURE Subscriber が持つ関心とは独立して動作し，CURE Hub が設定した**トピック**（IP アドレス，ドメイン名，ファイル

ハッシュ値，意味付けデータなどのサイバーセキュリティ関連情報の注目すべき事柄）に対するデータ発信が唯一の役割である．次に，CURE Hub は受け取ったデータ形式を解釈し，各トピック内についてインデックスの作成およびトピックへのデータの投入を行う．これと並行してデータの横断分析を実行する．横断分析には，1）IoC 間の関連の抽出と 2）IoC への意味付けの 2 種類がある．最後に各 CURE Subscriber は，利用したいトピックを CURE Hub に要求することで，そのトピックの該当データを受信する．

次節以降では，CURE Hub，CURE Publisher，CURE-Subscriber のそれぞれの設計の詳細を述べる．

2.2 CURE Hub: 仲介者

本節では CURE Publisher/Subscriber のインタフェースとデータ管理・分析を担う CURE Hub の設計を述べる．

2.2.1 インタフェース

CURE Hub では，CURE Publisher/Subscriber に対してデータ発信・受信用のインタフェースを提供する．

まず，CURE Publisher 用の API 形式の例をトピック毎に図 2 に示す．API 形式の共通部分には，CURE Publisher の識別子を表す *publisher_id* とデータ発信日時の *published_at*，観測・意味付けデータの詳細が格納される *data* がある．*data* 内の形式はトピック毎に異なり，観測データのひとつである IP アドレストピック（図 2(a)）では IP アドレスおよび国コードの文字列と，その IP アドレスで観測した事象を *detail* に記載できる．ドメイン名トピック（図 2(b)）とファイルハッシュ値トピック（図 2(c)）もこれと同様に，ドメイン名とファイルハッシュ値の文字列とそれについて観測した事象を *detail* に記載できる．意味付けデータトピック（図 2(d)）では，各種観測データについて意味付けするためのキーワードを *tags* に設定する．そして，*detail* にそのキーワードと関連する IP アドレス，ドメイン名，ファイルハッシュ値の文字列を列挙する．なお，全てのトピックに共通して観測した事象や意味付けの詳細を自由記述できる *content* が用意される．

次に，CURE Subscriber 用 API の例として IP アドレストピックのデータを受信するための形式を図 3 に示す．IP アドレスの一覧取得時（図 3(a)）には，受信時刻を表す *subscribed_at* と各 IP アドレスが列挙される *data* がある．一覧取得時には，IP アドレスと紐づくキーワード *tags* とデータ発信元の一覧 *publishers* がある．*publishers* の中では CURE Publisher の識別子と最終データ発信時刻の組が記され，複数の組が記載されている場合は複数の CURE Publisher から発信されたことを意味する．個別 IP アドレスのデータ取得時（図 3(b)）には，キーワード *tags* とともに観測した事象と CURE Publisher の識別子が *data* に記載される．この内容は前節の図 2(a) 中の *detail* である．他のトピックにおいても同様の API 形式が用意される．

```
{
  "publisher_id": 500,
  "published_at": 1593569493,
  "data": [ {
    "ip": "203.0.113.1",
    "cc": "JP",
    "detail": {
      "timestamp": 1593569471,
      "src_ip": "",
      "src_port": 56789,
      "src_cc": "JP",
      "dst_ip": "203.0.113.1",
      "dst_port": 80,
      "dst_cc": "JP",
      "content": "Malicious C2 Connection",
    }
  }, ...,  ]
}
```

(a) IP アドレストピック

```
{
  "publisher_id": 300,
  "published_at": 1593746002,
  "data": [ {
    "name": "example.net",
    "cc": "JP",
    "detail": {
      "timestamp": 1593745932,
      "ip": "203.0.113.1",
      "url": "http://example.net?c=get",
      "content": "Malicious C2 Connection",
    }
  }, ...,  ]
}
```

(b) ドメイン名トピック

```
{
  "publisher_id": 200,
  "published_at": 1593937051,
  "data": [ {
    "hash": {
      "md5": "73336a8e9ea4345084c903336be41264",
      "sha1":
       "2ddce7c75e117c7617a3bd73e75bf713b57ad34f"
    },
    "detail": {
      "timestamp": 1593937037,
      "filename": ["mal.exe"],
      "ip": [],
      "domain": ["example.net"],
      "url": ["http://example.net?c=post"],
      "content": "Suspicious Network Activity",
    }
  }, ...,  ]
}
```

(c) ファイルハッシュ値トピック

```
{
  "publisher_id": 1000,
  "published_at": 1593937051,
  "data": [ {
    "tags": ["abcRAT", "Group123"],
    "detail": {
      "timestamp": 1593937037,
      "ip": ["203.0.113.1"],
      "domain": ["example.net"],
      "file": [],
      "content": "OpABC by Group123",
    }
  }, ...,  ]
}
```

(d) 意味付けデータトピック

図 2 Publisher API

2.2.2 データの管理

CURE Hub が CURE Publisher から受け取ったデータは**トピック**の単位で管理する．ここでトピックとは具体的には，IP アドレスやドメイン名，ファイルハッシュ値，意味付けデータなどのインデックスおよびそれぞれの個別の

```
{
  "subscribed_at": 1593570624,
  "data": [ {
    "ip": "203.0.113.1",
    "tags": ["abcRAT", "Group123"],
    "publishers": [
      { "id": 500,
        "last_published_at": 1593569493 },
      { "id": 300,
        "last_published_at": 1593569712 },
    ]
  }, ...,  ]
}
```

(a) IP アドレス一覧トピック

```
{
  "subscribed_at": 1593570624,
  "ip": "203.0.113.1",
  "cc": "JP",
  "tags": ["abcRAT", "Group123"],
  "data": [ {
    "timestamp": 1593569471,
    "publisher_id": 500,
    "src_ip": "",
    "src_port": 56789,
    "src_cc": "JP",
    "dst_ip": "203.0.113.1",
    "dst_port": 80,
    "dst_cc": "JP",
    "content": "Malicious C2 Connection",
  }, ...,  ]
}
```

(b) 個別 IP アドレストピック

図 3 Subscriber API

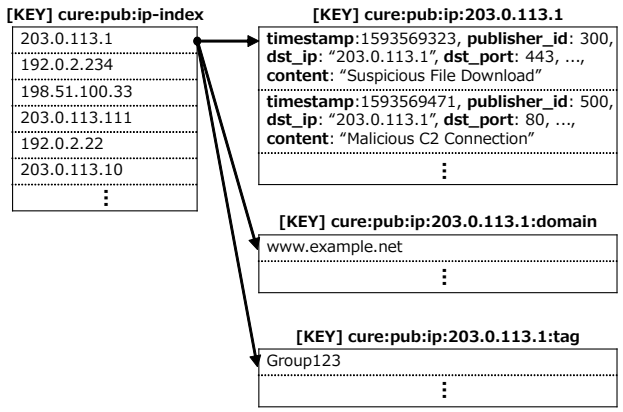

図 4 CURE Hub で扱うデータ構造の例

値のことを指す．そして，トピック名がデータ管理のために一意に定められるキーとなり，CURE Publisher から受け取ったデータを適したトピックを選択して値を蓄積する．

CURE Hub で扱うデータ構造を IP アドレスのトピックを例に図 4 に示す．トピック名には命名規則があり，IP アドレスのインデックスを表すトピックでは *cure:pub:ip-index*，個別 IP アドレスのトピックでは *cure:pub:ip:[ipaddr]* となる．そして，横断分析の結果は *cure:pub:ip:[ipaddr]:domain* や *cure:pub:ip:[ipaddr]:tag* の命名規則とし，個別 IP アドレスと別トピックのデータの繋がりを表現する．このようなデータ構造をキー・バリュー型のインメモリ DB に実装することにより，高速な検索を実現できる．

2.2.3 データの横断分析

横断分析では，まず各 CURE Publisher が発信した IP アドレスやドメイン名，ファイルハッシュ値などの観測データのトピックを対象に値の完全一致で全データを検索しデータ間の繋がりを発見する．たとえば，IP アドレス *203.0.113.1* と紐づくドメイン名 *example.net*，ドメイン名 *example.net* と紐づくファイルハッシュ値 *73336a8e9ea4345084c903336be41264* がある場合，IP アドレス *203.0.113.1* とファイルハッシュ値 *73336a8e9ea4345084c903336be41264* が繋げられる．

次に，観測データに対して意味付けを付与する．これには図 2(d) の形式で取得した意味付けデータを利用する．図中の *detail* に記載された IP アドレス，ドメイン名，ファイルハッシュ値に対して *tags* に記載された文字列を紐づける．たとえば，意味付けデータとして *Group123* という文字列が設定され *ip* に *203.0.113.1* が記載されている場合，これらが関連するものとして DB に格納する．

これらの横断分析処理は CURE Publisher からデータを受け取るたびに実行される．前節で述べた通りキー・バリュー型インメモリ DB による実装を想定しているため，リアルタイムな分析処理性能を見込める．

2.3 CURE Publisher: データ発信者

CURE Publisher には**観測層**と**意味付け層**の 2 種類の役割に分類され，それぞれに発信できるトピックが異なる．

観測層 Publisher は各種観測システムで収集したサイバーセキュリティ関連の事象の発信のみを担う．扱うトピックは IP アドレス，ドメイン名，ファイルハッシュ値といった観測データに関するものに特化する．

一方で意味付け層 Publisher は，観測データに対して任意のキーワードを付与する意味付けデータの作成し，意味付けデータトピックへの発信のみを担う．観測データへの意味付けや解釈は状況や環境次第で異なるものと仮定し，各観測層 Publisher と独立して動作させることで流動的な意味付けを可能にする．

観測層と意味付け層で発信できるデータの関心を分離することにより，CURE 全体として拡張性のあるシステム構成を実現できる．さらに，負荷の大きなデータの意味付けや解釈の処理を CURE Hub からも分離できる．これにより CURE Hub は各種インタフェースとデータ管理，横断分析の機能にのみ集中でき高速化を実現できる．

2.4 CURE Subscriber: データ利用者

CURE Subscriber はデータ受信を担うデータ利用者を表す．データ受信時は図 3 の API 形式に従う必要がある．

ただし，CURE Hub と CURE Subscriber 間の通信プロトコルは各々の実装によって選択できる．オンデマンドでデータを受信する場合には HTTP のような要求/応答型のパラダイムを持つ通信プロトコルを採用し，ストリームでデータを受信する場合は WebSocket や WebSub といった

リアルタイム性のあるメッセージングを採用すればよい．

3. 実装

CURE Hub の実装には全て Rust 言語 [*1]を用い，インメモリ DB には Redis [*2]を採用した．また，CURE Publisher/Subscriber 用の API および Web インタフェースには Rust 言語用 Web フレームワークの Rocket [*3]を用い，API は RESTful な実装とした．セキュリティ・オペレータは，Web インタフェースにて CURE 上のデータを閲覧・検索でき，自身で API を用い分析フローを自動化できる．

また，以下の通り，情報通信研究機構が定常運用するシステムを観測層 Publisher として，インターネット上のサービスを意味付け層 Publisher として実装した．

NICTER [5] （観測層 Publisher）
無差別型攻撃観測．ダークネット観測で得たパケットを送信元 IP アドレス・宛先ポート番号，プロトコルでグループ化したもの．扱うトピックは IP アドレス．

ハニーポット群 （観測層 Publisher）
無差別型攻撃観測．Dionaea，Cowrie，Glastopf の 3 種で観測したサイバー攻撃関連の通信．扱うトピックは IP アドレス，ファイルハッシュ値．

STARDUST [6] （観測層 Publisher）
標的型攻撃観測．標的型攻撃関連のマルウェアを介して発生した攻撃者の LAN 内での活動や C2 サーバとの通信．扱うトピックは IP アドレス，ドメイン名，ファイルハッシュ値．

NIRVANA 改 [7] （観測層 Publisher）
アラート情報統合分析．セキュリティ機器から発報されるアラートを毎分グループ化したもの．扱うトピックは IP アドレス，ドメイン名，ファイルハッシュ値．

エンドポイント [8] （観測層 Publisher）
エンドポイント情報収集．端末に導入されたエージェントソフトウェアによって収集された不審プロセスおよびそのプロセスによって発生した通信．扱うトピックはファイルハッシュ値．

EXIST [9] （観測層 Publisher）
脅威情報収集．15 種類のオープンソース脅威情報サービスから収集したレピュテーション情報．扱うトピックは IP アドレス，ドメイン名．

WarpDrive [10] （観測層 Publisher）
Web 媒介型攻撃対策．収集した Web アクセス履歴のうち，WarpDrive が備えるブラックリストと合致したもの．扱うトピックはドメイン名．

ATT&CK [11] （意味付け層 Publisher）
サイバー攻撃ナレッジフレームワーク．各 Techniques，Groups，Software を意味付けキーワードとして登録．各詳細ページに記載の参考文献から各種 IoC を抽出しキーワードと紐づけて保存．

[*1] https://www.rust-lang.org/
[*2] https://redis.io/
[*3] https://rocket.rs/

4. 性能評価

本章では，CURE Hub へのデータ発信および CURE Hub からのデータ受信の性能評価について述べる．性能評価に用いた機器構成を表 1 に示す．これらは全て 1000 Base-T のネットワーク機器で接続した．データ発信およびデータ受信の計測方法を以下に述べる．

(1) CURE Hub へのデータ発信
IP アドレストピックに対して，異なるデータ件数毎に発信の処理時間を計測した．データ発信は curl による HTTP の POST メソッドを用い，評価用データには観測層 Publisher の NICTER のものを用いた．

(2) CURE Hub からのデータ受信
IP アドレストピックに対して，上述の発信データについて全 IP アドレスの一覧取得，任意の IP アドレスの部分一致検索，任意の IP アドレスの完全一致検索の 3 種類の処理時間を計測した．データ受信には curl による HTTP の GET メソッドを用いた．

データ発信の結果を表 2 に，データ受信の結果を表 3 に示す．データ発信において件数が増加すると処理時間がそれぞれ線形に増加するが，100,000 件のデータ発信の場合でも数秒で完了できた．データ受信（一覧取得）においても同様に処理時間が線形に増加するが 100,000 件のデータ受信においても 1 秒未満で完了できた．データ受信（部分一致）では件数が増加するとともに処理時間も線形に増加した．これは全データの中から部分一致するものを検索するためである．一方でデータ受信（完全一致）では，データの件数が増加しても 0.005 秒程度で検索を完了した．これは第 2 章で述べた通り，キー・バリュー型インメモリ DB に特化したデータ管理の設計を採用したためである．

表 1 性能評価における機器・ソフトウェアの構成

システム	機器の仕様
CURE Hub	CPU: AMD EPYC 7401P RAM: 512 GB NIC: 1000 Base-T Ethernet OS: Ubuntu 18.04 LTS (Kernel 4.15.0) Other: Rust 1.46.0-nightly, Rocket 0.4.5
インメモリ DB	CPU: Intel Xeon Gold 5215M 2.50GHz ×4 RAM: 4 TB NIC: 1000 Base-T Ethernet OS: CentOS 8.2.2004 (Kernel 4.18.0) Other: Redis 5.0.3
CURE Publisher / Subscriber	CPU: AMD EPYC 7401P RAM: 256 GB NIC: 1000 Base-T Ethernet OS: CentOS 8.2.2004 (Kernel 4.18.0) Other: curl 7.61.1

表 2 データ発信の評価結果

件数		100	1,000	10,000	100,000
時間（秒）	最小値	0.012	0.039	0.327	3.649
	最大値	0.019	0.043	0.368	4.044
	平均値	0.013	0.040	0.341	3.770
	中央値	0.013	0.040	0.331	3.740

表 3 データ受信の評価結果

件数		100	1,000	10,000	100,000
時間（秒）	最小値	0.008	0.015	0.086	0.858
一覧取得	最大値	0.012	0.017	0.101	0.887
	平均値	0.009	0.016	0.093	0.872
	中央値	0.008	0.016	0.091	0.872
時間（秒）	最小値	0.006	0.014	0.081	0.821
部分一致	最大値	0.007	0.016	0.086	0.839
	平均値	0.007	0.015	0.083	0.828
	中央値	0.007	0.014	0.083	0.929
時間（秒）	最小値	0.005	0.005	0.005	0.005
完全一致	最大値	0.005	0.006	0.006	0.006
	平均値	0.005	0.005	0.005	0.005
	中央値	0.005	0.005	0.005	0.005

5. ケーススタディ

CURE によって抽出されたサイバーセキュリティ関連情報の繋がりについて述べる．まず，収集したデータの概要を表 4 に示す．なお，このときのインメモリ DB におけるデータ量は約 62.7 GB であった．各 CURE Publisher によってデータの期間や観測データ数は異なるが，多くの場合は同一の IoC を複数回発信していた．また，ほとんどの IoC に対して意味付けキーワードは付与されなかった．

次に，2020 年 8 月 15 日時点でのデータの繋がりを基に実施したケーススタディのまとめを表 5 に示す．ケース 1 では意味付けされたトピックに着目した．意味付けされたトピックのうち複数の CURE Publisher が共起しているものが個別 IP アドレストピックで 2 件，個別ドメイン名トピックで 1 件あった．このうち，1 件の個別 IP アドレストピックと個別ドメイン名トピックは正常な接続先についての意味付けであったため，残りの個別 IP アドレストピックの 1 件について 5.1 節で詳細を述べる．

ケース 2 では複数の CURE Publisher から発信されたトピックに着目した．CURE 上のトピックのうち，273 件の個別 IP アドレストピックにおいて 4 種類の CURE Publisher が共起してデータを発信していた．この中でドメイン名とファイルハッシュ値の他トピックを横断して繋がりがあったものが 1 件あり，これについて 5.2 節で詳細を述べる．

5.1 ケース 1

本ケースの発見の起点は，ハニーポット群と NIRVANA 改で共起した個別 IP アドレストピックに意味付けキーワードが付与されていたことである．まず，それぞれの CURE Publisher から発信されたデータに着目する．PHP-CGI のクエリ文字列処理の脆弱性を狙う攻撃試行についてハニーポット群と NIRVANA 改がともにデータ発信しており，それぞれの観測日が 2 カ月程度の差があるが同種の攻撃者・マルウェアに利用されている可能性がある．

次に意味付けキーワードを確認すると，ATT&CK の Software の項目として *Chaos* が付与されていた．また，Techniques の項目として *Symmetric Cryptography*，*Brute Force*，*Multi-Stage Channels*，*Unix Shell* が付与され，これらは Chaos に関連する手法であった．当該 IP アドレスが記載された記事は 2018 年 5 月のもので，当時はこの IP アドレスからの SSH ブルートフォース攻撃が観測されていた．加えて，当該 IP アドレスが Tor ネットワークの出口ノードであることも記載されており，我々の観測データとこの記事との関連性は現段階では調査できなかった．

5.2 ケース 2

本ケースの発見の起点は，NICTER の IP アドレストピックへのデータ発信により共起する CURE Publisher が全データ中最大の 4 種類になったことである．横断分析結果より，2020 年 5 月 20 日から 2020 年 5 月 22 日の間に NIRVANA 改によって発信されていた観測データはマルウェア添付のメールを検知したアラート情報であり，このアラート情報により当該 IP アドレスと 1 件のドメイン名と 5 件のファイルハッシュ値が紐づけられていた．また，アラート情報を NIRVANA 改と連携するセキュリティ機器上で精査すると，ドメイン名はメール署名に記載されたもの，ファイルハッシュ値はどれもメール本体と添付ファイルの動的解析でマルウェアと判定されたものであった．

2020 年 5 月当時は当該 IP アドレスはメールサーバとして機能していたが，2020 年 7 月，8 月におけるハニーポット群や NICTER での観測データからは異なる種別のサイバー攻撃関連の通信を発生させていた．このことより，当該 IP アドレスが攻撃インフラとして別の攻撃者やマルウェアに再利用されていた，もしくは同一の攻撃者・マルウェアが取り扱う攻撃の範囲が広がったことが示唆される．

6. 議論

6.1 考察

本節では性能評価やケーススタディを踏まえて CURE について考察する．まず性能評価については，IoC の完全一致検索でのデータ受信の結果を見るとキー・バリュー型のインメモリ DB に特化したデータ構造の設計にした効果が現れていると言える．さらに，一覧取得や部分一致の検索でのデータ受信では，どちらも件数が増加すると必要な時間は線形に増加した．しかしこれは，たとえ今回のケーススタディで長期間に渡って収集した全ての IoC を検索対象としたとしても，全データを数秒で検索できるとも言え

表 4 収集データ（数値は左から，観測総数，ユニーク数，意味付けユニーク数）

CURE Pub.	データの期間	IP アドレス			ドメイン名			ファイル			キーワード	
NICTER	2020/04/10-2020/08/15	80,434,394	210,961	3		-			-			-
ハニーポット群	2019/09/01-2020/08/15	1,625,469	247,858	4		-		4,630	110	1		-
STARDUST	2020/01/01-2020/08/15	1,389,305	6,448	2	17,216	159	20	1,476	396	0		-
NIRVANA 改	2020/04/10-2020/08/15	1,341,067	49,009	1	80,521	2,271	6	36,503	30,478	0		-
エンドポイント	2019/01/07-2020/08/14		-			-		122	99	2		-
EXIST	2020/04/09-2020/08/15	128,904	68,424	2	231,328	128,750	0		-			-
WarpDrive	2018/06/04-2020/07/30		-		5,795	312	1		-			-
ATT&CK	2020/04/15-2020/07/22		-			-			-		900	900
全体	-	84,919,139	500,499	11	334,860	131,490	23	42,731	31,068	2	900	900

表 5 ケーススタディのまとめ（各ケースの CURE Publisher は観測期間で昇順）

#	CURE Pub.	起点トピック	関連トピック	観測期間	意味	観測内容
1	ハニーポット群 NIRVANA 改	IP アドレス	なし	2020/05/06-2020/05/24 2020/07/21	あり	PHP-CGI の脆弱性を狙う攻撃と SQLi の試行 PHP-CGI の脆弱性を狙う攻撃試行
2	NIRVANA 改 ハニーポット群 EXIST NICTER	IP アドレス	ドメイン名 ファイル	2020/05/20-2020/05/22 2020/07/10-2020/07/20 2020/07/13 2020/08/10-2020/08/13	なし	マルウェアが添付されたメールの送信 23/tcp 宛の接続試行 CINS Army List [*4]に記載 22/tcp, 23/tcp, 5984/tcp, 7547/tcp 宛の接続試行

る．すなわち，これはセキュリティ・オペレーションを迅速に進めるには十分な検索性能であると言える．

次にケーススタディの結果から考察する．観測データへの意味付けに ATT&CK の Web ページおよびそこに記載される参照記事を利用したが，結果としては今回収集したデータにはあまり意味付けキーワードを付与できなかった．一方で，CURE Publisher は CURE Hub と独立して動作し，新たな CURE Publihher を CURE のフレームワークに参加させることは容易であるため，ATT&CK 以外の情報源による意味付けや，自然言語処理や機械学習などによる高度な手法を採用した意味付けによって，CURE 上のデータの質を向上させることも可能である．また， 5.2 節のように複数の CURE Publisher から得られた観測データを横断的に検索することで大量の観測データの中に隠された繋がりを発見できうることもわかった．異種の観測層 Publisher との連携により観測の範囲が広がるため，インテリジェンス創出のための基盤として利用可能と考える．

6.2 制限事項

CURE では高速な検索を実現するためにキー・バリュー型のインメモリ DB を採用した．高速化を実現できた一方で，1）長期間のデータを蓄積するためには大容量メモリが必要である，2）揮発性メモリを利用する場合にはデータを損失する可能性がある，という 2 点の制限事項がある．

第 5 章より，8 種の CURE Publisher からのデータ発信で約 62.7 GB のデータ量だった．セキュリティ・オペレーションで必要なデータ量と金銭的コストを考慮して搭載するメモリ量を決定する必要がある．また，CURE では各 CURE Publisher でのデータの完全性の保証を前提とする．そのため，CURE 上のデータを損失した際は，CURE Publisher からデータを再発信する運用で回復できる．

7. 関連研究

本節では，サイバーセキュリティ関連情報の集約・分析および意味付けの観点から関連研究について述べる．

7.1 情報の集約・分析

Modi らはグラフ DB でサイバーセキュリティ関連情報を扱うフレームワーク ATIS を提案している [12]．ATIS は 5 プレーン（収集・分析・コントローラ・データ・アプリケーション）で構成され，各プレーンを拡充することで ATIS に機能を統合できるモノリシックな構成を持つ．Azevedo らは PURE を提案している [13]．PURE は収集した IoC のクラスタを作成し，それを相関分析することで注目すべき IoC を抽出する．伊藤らはダイヤモンドモデルに基づき脅威情報をリレーショナル DB に蓄積する仕組みを提案し，その有効性を示している [14]．

一方，CURE は各要素間のインタフェースのみが規定された互いに独立動作する疎結合な構成である．そのため，CURE Publisher/Subscriber を新たに実装した場合にも容易に CURE のフレームワークに参画できる拡張性がある．また，全ての IoC やその間の繋がりをキー・バリュー型のインメモリ DB で取り扱うため，データ量が増加してもデータ受信の性能は大きく劣化しない．

7.2 情報の意味付け

Alves らは観測対象のインフラに適した脅威情報の生成のために，Twitter のセキュリティ関連のアカウントによる発言から情報の分類モデルを検討している [15]．Amir-

[*4] http://cinsscore.com/list/ci-badguys.txt

reza らも同様に Twitter を情報源とし，ハッシュタグなどの Twitter の特性を活かした IoC への意味付けを実現する [16]．Liao らは自然言語で記述された非構造的な文章を対象として固有表現抽出や関係抽出といった自然言語処理の手法を用いて IoC への意味付けを提案している [17]．

CURE による情報への意味付けは，横断分析によるデータ間の繋がりの発見や意味付け層 Publisher によるデータ発信でのみ行われる．CURE Hub 上の処理を軽量にしつつ，複雑な意味付けの処理は意味付け層 Publisher に委譲することで，流動的かつ多面的な意味付けを実現できる．たとえば自然言語処理やディープラーニングといった処理に時間を要するものを CURE Hub と独立する CURE Publisher のひとつとして実現することで，全体構成の中で役割分担をしつつ機能拡張を実現できる．

8. おわりに

本稿では，多種多様かつ大規模なサイバーセキュリティ関連情報を集約・横断分析し，高速な検索を可能にしたセキュリティ情報融合基盤 CURE を提案し，その設計および実装を示した．性能評価では，データ発信・受信ともに短時間で実行できることを示した．ケーススタディでは，意味付けデータが付与された事例と観測データが複数 CURE Publisher・複数トピックに横断する事例ついて述べ，CURE による分析の有効性について述べた．

今後はより多様なサイバーセキュリティ関連情報を扱うためにトピックを追加し，並行して観測層 Publisher を拡充する．また，意味付け層 Publisher としてインターネット上の記事への自然言語処理や時系列を考慮した意味付けを実現する．さらに，CURE を用いたデータ分析を実践し，大規模データからのインテリジェンスの創出を目指す．

参考文献

[1] Matt Bromiley. Threat Intelligence: What It Is, and How to Use It Effectively. Technical report, 2016.

[2] Xander Bouwman, Harm Griffioen, Jelle Egbers, Christian Doerr, Bram Klievink, and Michel van Esten. A Different Cup of TI? The Added Value of Commercial Threat Intelligence. In *Proceedings of the 29 th USENIX Security Symposiumb*, pp. 443–450, 2020.

[3] Cynthia Wagner, Alexandre Dulaunoy, Gérard Wagener, and Andras Iklody. MISP - The Design and Implementation of a Collaborative Threat Intelligence Sharing Platform. In *Proceedings of the 2016 ACM Workshop on Information Sharing and Collaborative Security (WISCS 2016)*, pp. 49–56, 2016.

[4] Eric W. Burger, Michael D. Goodman, Panos Kampanakis, and Kevin A. Zhu. Taxonomy Model for Cyber Threat Intelligence Information Exchange Technologies. In *Proceedings of the ACM Conference on Computer and Communications Security*, pp. 51–60, 2014.

[5] Daisuke Inoue, Katsunari Yoshioka, Masashi Eto, Masaya Yamagata, Eisuke Nishino, Jun'ichi Takeuchi, Kazuya Ohkouchi, and Koji Nakao. An Incident Analysis System NICTER and Its Analysis Engines Based on Data Mining Techniques. In *Advances in Neuro-Information Processing*, pp. 579–586, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[6] 津田侑, 遠峰隆史, 金谷延幸, 牧田大佑, 丑丸逸人, 神宮真人, 高野祐輝, 安田真悟, 三浦良介, 太田悟史, 宮地利幸, 神薗雅紀, 衛藤将史, 井上大介, 中尾康二. サイバー攻撃誘引基盤 STARDUST. コンピュータセキュリティシンポジウム 2017 論文集, 2017.

[7] 津田侑, 金谷延幸, 遠峰隆史, 神薗雅紀, 神宮真人, 高木彌一郎, 鈴木宏栄. NIRVANA 改によるライブネット分析. 情報通信研究機構研究報告, Vol. 62, No. 2, pp. 59–66, 2016.

[8] Yu Tsuda, Junji Nakazato, Yaichiro Takagi, Daisuke Inoue, Koji Nakao, and Kenjiro Terada. A Lightweight Host-Based Intrusion Detection Based on Process Generation Patterns. In *Proceedings of the 13th Asia Joint Conference on Information Security (AsiaJCIS 2018)*, pp. 102–108, 2018.

[9] nict-csl/exist. `https://github.com/nict-csl/exist/`.

[10] Takeshi Takahashi, Christopher Kruegel, Giovanni Vigna, Katsunari Yoshioka, and Daisuke Inoue. Tracing and Analyzing Web Access Paths Based on User-Side Data Collection: How Do Users Reach Malicious URLs? (in press). In *Proceedings of the 23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020)*, 2020.

[11] MITRE ATT&CK. `https://attack.mitre.org/`.

[12] Ajay Modi, Zhibo Sun, Anupam Panwar, Tejas Khairnar, Ziming Zhao, Adam Doupé, Gail Joon Ahn, and Paul Black. Towards Automated Threat Intelligence Fusion. In *Proceedings of the 2nd International Conference on Collaboration and Internet Computing (IEEE CIC 2016)*, pp. 408–416, 2017.

[13] Rui Azevedo, Iberia Medeiros, and Alysson Bessani. PURE: Generating Quality Threat Intelligence by Clustering and Correlating OSINT. *Proceedings of the 18th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/the 13th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE 2019)*, pp. 483–490, 2019.

[14] 伊藤大貴, 永井達也, 野村健太, 近藤秀紀, 神薗雅紀, 白石善明, 古本啓祐, 瀧田愼, 毛利公美, 高野泰洋, 森井昌克. スレットインテリジェンスのためのダイヤモンドモデルに基づく脅威情報分析システム. 電子情報通信学会論文誌, Vol. J101-D, No. 10, pp. 1427–1437, 2018.

[15] Fernando Alves, Pedro M Ferreira, and Alysson Bessani. Design of a Classification Model for a Twitter-based Streaming Threat Monitor. In *Proceedings of the 2019 IEEE/IFIP International Conference on Dependable Systems and Networks Workshops*, pp. 9–14, 2019.

[16] Amirreza Niakanlahiji, Lida Safarnejad, Reginald Harper, and Bei-Tseng Chu. IoCMiner: Automatic Extraction of Indicators of Compromise from Twitter. In *Proceedings of the 2019 IEEE International Conference on Big Data*, pp. 4747–4754, 2019.

[17] Xiaojing Liao, Kan Yuan, XiaoFeng Wang, Zhou Li, Luyi Xing, and Raheem Beyah. Acing the IOC Game: Toward Automatic Discovery and Analysis of Open-Source Cyber Threat Intelligence. In *Proceedings of the 23rd ACM Conference on Computer and Communications Security (CCS 2016)*, pp. 755–766, 2016.