Vol.2021-DPS-186 No.19
Vol.2021-CSEC-92 No.19
2021/3/15

複数組織の接続傾向を用いた自律進化型防御システムの提案

西嶋克哉[1]　川口信隆[1]　植木優輝[1]　重本倫宏[1]　近藤賢郎[2]　中村修[3]

概要：サイバー攻撃の激化にともない，悪性サイトへの接続を防止する技術が求められている．既存手法としては，ブラックリストやホワイトリストを用いたものがある．しかし，インターネット上に無数にあるサイトを全て，ブラックリストやホワイトリストに分類することは不可能である．従って，これらリストに存在しないサイトが良性か悪性かを判断できないという課題がある．本稿ではこの課題を解決する手法を提案する．提案手法では組織が保有する，サイトへの接続ログを元に接続傾向を分析し，その傾向との乖離度を利用してサイトの良性，悪性を判定する．また，自組織の接続傾向だけでなく，他組織の接続傾向を利用することで判定の精度向上を狙う．更に，判定で誤って悪性と判断した良性サイトへの接続を即時遮断するのではなく，機械には突破困難な追加認証を課し，突破できなかった場合のみ接続を遮断する．提案システムにより，業務遂行に必要なサイトを誤って悪性と判定した際の業務阻害を緩和しつつ，悪性サイトへの接続を遮断できることが期待される．

キーワード：悪性 Web サイト，追加認証，異常検知，情報共有

Proposal of Automated Defense System Using Access Trend in Multiple Organizations

KATSUYA NISHIJIMA[†1]　NOBUTAKA KAWAGUCHI[†1]　YUKI UEKI[†1]
TOMOHIRO SHIGEMOTO[†1]　TAKAO KONDO[†2]　OSAMU NAKAMURA[†3]

1. はじめに

近年，標的型攻撃に見られるように，攻撃が高度化しており，企業や国家にとって重大な脅威となっている．ここで，マルウェアのダウンロード[1]に加えて，マルウェアとの通信，フィッシングサイトの表示，スパムの発信[2]等，悪意を持ったサイト（以降，悪性サイト）がサイバー攻撃において重要な役割を有している．このことから，サイバー攻撃の被害を抑制するためには，Firewall や Web Proxy 等のゲートウェイにより悪性サイトへの接続を遮断する，いわゆる出口対策[3]が重要であるといえる．悪性サイトへの接続を遮断する方法としては，悪性サイトと判定したサイト群であるブラックリスト（以下 BL）への接続を遮断する方法や，反対に良性サイトと判定したサイト群であるホワイトリスト（以下 WL）以外への接続を遮断する方法が存在する．しかし，インターネット上の無数にあるサイトを全て，BL や WL に分類することは不可能である．従って，これらリストに存在しないサイト群（未知サイト）への接続が良性か悪性かを判定できないという課題がある．そこで本稿では，未知サイトへの接続の不審度を接続傾向との乖離度から算出し，当該不審度の高低により未知サイトへの接続が良性か悪性かを判定する方法を提案する．これは，悪性サイトへの接続は，普段の接続傾向とは異なる可能性が高い，という考えに基づいている．尚，接続傾向は，組織が保有するサイトへの接続ログを分析し算出する．

上述の方法は，普段の接続傾向と異なる接続を全て悪性サイトへの接続と見做す．このため特に，良性サイトを悪性サイトと判定してしまう誤検知が多く発生する．そこで誤検知の削減方法として，普段の接続傾向がより多くの良性サイトをカバーできるように，多種多様な正常の接続情報をインプットとして利用することが考えられる．ここで日立製作所では，慶應義塾大学と協力して，複数組織の SOC（Security Operation Center）を跨ったセキュリティ・オペレーション連携により，サイバー攻撃への集団防御を実現する，分散 SOC アーキテクチャを提案している[4]．本アーキテクチャを用い，自組織が保有する接続ログだけでなく，他組織が保有する接続ログを利用する．これにより，例えば自他組織が共に不審と判断した時のみ悪性と判定する，といった方法で誤検知の削減が期待できる．一方で，組織が保有する接続ログは一般的に機微情報であり，且つログ量も大きくなる傾向があるため，他組織への共有は困難である．ここで我々は，生データの開示を伴わない分析を支援する，秘匿データ分析システムの研究を進めている[5]．本稿では，接続ログといった生データを直接共有するのではなく，他組織上で接続傾向と，接続傾向を用いた不審度の算出を行う分析ロジックを実行し，不審度のみを共有するシステムを提案する．

また，他組織の接続ログを用いることで精度が向上できたとしても，依然として誤検知の可能性は残っており，業務で利用する良性サイトが悪性サイトと判定され，接続が

1 株式会社日立製作所 研究開発グループ
Hitachi, Ltd. R&D Group
2 慶應義塾情報セキュリティインシデント対応チーム
Computer Security Incident Response Team, Keio University
3 慶應義塾大学環境情報学部
Faculty of Environment and Information Study, Keio University

Vol.2021-DPS-186 No.19
Vol.2021-CSEC-92 No.19
2021/3/15

遮断されることにより，業務が阻害される可能性がある．そこで，接続先が悪性サイトと判定された際に，即座に接続を遮断するのではなく，いったん追加認証を課すことにより，人間による業務上必要な接続は許可しつつ，マルウェア等による機械的な接続は遮断する．

これらにより提案システムは，未知の悪性サイトにも有効性があり，業務遂行に必要なサイトを誤って悪性と判定した際の業務阻害を緩和することが可能である．

本稿の構成は次のとおりである．まず，2 章で既存の悪性サイトへの接続防止手法とその課題について述べる．3 章で同課題を解決する手法を提案し，その後 4 章で関連研究について述べ，最後に 5 章でまとめを述べる．

2. 背景と課題

本章では，著者らが研究を進めてきた悪性サイトへの接続防止手法である，自律進化型防御システム（AED：Autonomous Evolution of Defense） [6], [7], [8]の概要とその課題を述べる．

2.1 AED の概要

AED では，ユーザが未知サイトへ接続しようとした場合に，プロキシで追加認証を要求する．これにより，マルウェアによる機械的な接続を遮断しつつ，人間による業務上必要な接続は許可することができる．また，ホワイトリスト型の AED[8]では，一定数のユーザが追加認証を突破し接続したサイトを WL に追加することにより，追加認証の回数を減らすことで，ユーザの利便性低下を抑えることができる．

2.2 AED の課題

(1) 未知サイト接続時の利便性低下

AED では，WL を動的に拡張していく仕組みがあるが，未知サイトへ接続する際には必ず追加認証が発生してしまい，ユーザの利便性が低下する．

(2) 小規模組織での WL 拡張が困難

AED で WL を拡張する仕組みは，組織内の他ユーザが接続した記録を用いるため，ユーザ数が少ない小規模の組織では十分に WL の拡張が行えない．実際に先行研究[8]では，ユーザが許容できる利便性を維持するために，1000 人以上のユーザが必要という結果が出ている．これは大きな組織でないと実現が困難である．

3. 分散型 AED

2 章で述べた課題を解決するシステムとして，複数組織の接続傾向を利用して接続を制御する，分散型 AED を提案する．本章では，提案システムの設計について述べる．図 1 に提案システムの全体像を示す．本システムは，(1) インテリジェンス共有プラットフォーム， (2) ドメイン分析ロジック，(3)AED に分類される．

インテリジェンス共有プラットフォームは，データ分析を行う機能である分析ロジック（本ケースでは後述するドメイン分析ロジック）を他組織で実行することで，他組織が保有する生データである各種インテリジェンス（本ケースでは接続ログ）を直接取得せずに分析結果（本ケースでは接続の不審度）のみを取得可能とする．これにより，効率的且つセキュアに他組織のデータを活用することが可能となる．このように，他組織の接続ログを使い，接続ログのユーザ数を仮想的に増やすことで，2.2 節課題(2)を解決する．

また，ドメイン分析ロジックは，異常検知アルゴリズムを利用し，組織が保有するサイトへの接続ログから接続傾向を分析し，あるサイトへの接続の不審度を，当該接続傾向との乖離度から算出する．これにより，WL に依らない判定が可能となり，2.2 節課題(1)を解決する．

そして AED は，WL，BL，インテリジェンス共有プラットフォームから得られた不審度を元に，接続の可否や追加認証を制御する．

以下の節にて，それぞれの設計について詳述する．

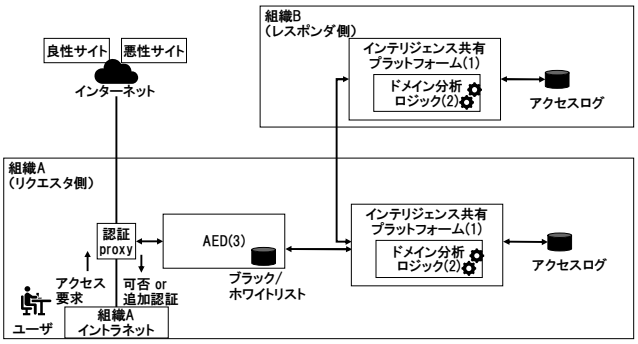

図 1　提案システム（分散型 AED）の全体像

3.1 インテリジェンス共有プラットフォーム

3.1.1 構成

図 2 にインテリジェンス共有プラットフォームの概要図を示す．組織 A には組織 B へ分析リクエストを送るリクエスタが在り，リクエスタには GUI のツールや AED 等からリクエスト情報が送られる．組織 B には分析リクエストを受信し処理を行うレスポンダ，および接続ログといった共有・分析対象のインテリジェンスが存在する．本プラットフォームでは，分析の実行主体である分析ロジックは，リクエスタ側が作成しレスポンダ側で実行される．レスポンダ内には，分析ロジックの実行基盤であるプラットフォーム，およびインテリジェンスへの接続制御を行うリソース接続 API が存在する．分析ロジックとその実行基盤を分離して別組織が管理することで，リクエスタ側の要望に応じて様々な分析を実施することができる．また，各コンポ

Vol.2021-DPS-186 No.19
Vol.2021-CSEC-92 No.19
2021/3/15

ーネント間のやり取りは，WebAPI を介して行う．これについては，次節にて詳述する．また，本プラットフォームでは，リクエスタ側，レスポンダ側双方の情報流出を極力抑えるように設計している．セキュリティモデルの設計については，3.1.3 節にて詳述する．

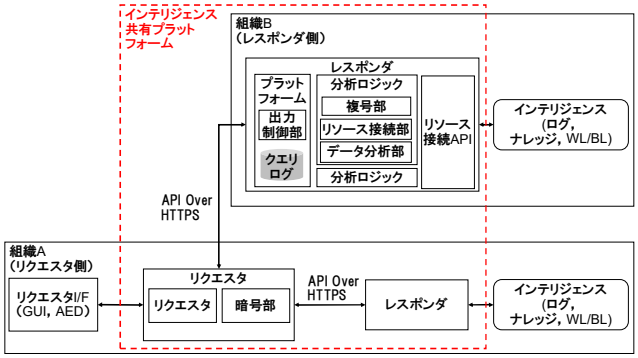

図 2　インテリジェンス共有プラットフォームの概要図

3.1.2 WebAPI 設計

図 3 に WebAPI の概要図を示す．本節では，各 WebAPI の設計について述べる．

まず，プラットフォームの WebAPI について説明する．プラットフォームの WebAPI が提供する機能は，(1)分析ロジックの制御，(2)分析ロジックによる分析の実行，に分類される．

分析ロジックの制御では，分析ロジックの状態表示，インストール・起動，停止・アンインストール機能を提供する．また，リクエスタは分析ロジックによる分析の実行を，プラットフォームの WebAPI を介して実施する．これは，全ての接続をプラットフォームに集約させることにより，3.1.3 節にて詳述するセキュリティモデルにおいて効率性，網羅性を高めるためである．プラットフォームは，受け取ったリクエストからどの分析ロジックへのリクエストであるかを判断し，適切な分析ロジックへリクエストを転送する．これらの機能を持つプラットフォームは，各組織が共通のものを利用する．これにより，本情報連携への組織の新規参入を効率的にする．

次に分析ロジックが提供する WebAPI について説明する．分析ロジックが提供する WebAPI は分析ロジック毎に異なる．本稿で提案するドメイン分析ロジックでは，リクエスタ側は，レスポンダ側のインテリジェンス（接続ログ）を用いた分析モデルの作成・削除，分析モデルを利用したドメインの不審度問合せを行うことができる．ここでいう分析モデルとは，不審度スコアを算出するのに使用するデータ構造であり，接続ログを前処理することで作成される．データ構造を事前に作成することで，分析リクエストに対して迅速に結果を返すことができる．作成されたデータ構造は レスポンダ内に保存される．この分析モデル作成のフェイズを以降「モデル作成フェイズ」と，分析モデルを用いて未知ドメインへの接続の不審度を算出するフェイズを以降「分析フェイズ」と呼ぶ．

最後に，リソース接続 API について説明する．リソース接続 API は，各分析ロジックがレスポンダ側の保有するインテリジェンスを取得する際に利用する WebAPI である．

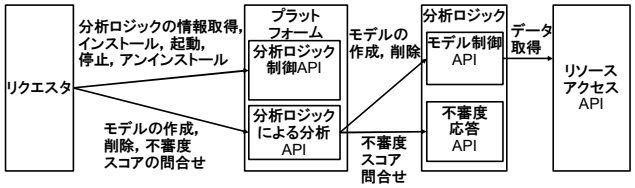

図 3　Web API の概要図

3.1.3 セキュリティモデル

本節では，インテリジェンス共有プラットフォームのセキュリティモデルについて説明する．まず前提として”Trust-but-verify”, “Honest-but-curious” のモデルを採用する．Trust-but-verify というのは，相手のことを信頼するが，念のため検証を行うという考え方である．Honest-but-curious というのは，積極的に悪意を働くことはないが，容易な機会があれば不正を働く可能性があるという考え方である．インテリジェンス共有プラットフォームはインターネット上の不特定多数が参加するものでなく，比較的身元が保証された組織間での利用を想定するため，前述のモデルを前提としている．このため，リクエスタ側，レスポンダ側はお互いが積極的に悪意ある行動を行う可能性は低いと判断しつつ，後に相手の行動を検証する手段を確保することを必要とする．

表 1 にリクエスタへの脅威を示す．まず，リクエスタへの脅威として，リクエストの内容が平文でレスポンダ側が取得可能なことがあげられる．そこで，リクエスタは分析ロジックとの間で共通鍵の交換を行い，暗号部によりリクエストを暗号化し，分析ロジックの複号部で復号してから分析を行う．暗号に使う鍵は，DH[9]等，Forward Secrecy を担保できるものを用いる．すなわち，暗号化されたトラフィックを受信していた攻撃者が鍵を入手した場合でも，過去のクエリを復元することをできないようにする．このリクエストの内容を保護する方法の注意点としては，レスポンダ側が管理している分析ロジック内に復号の鍵が存在するため，メモリの解析や分析ロジックのリバースエンジニアリングにより鍵がレスポンダ側に漏洩する可能性があることがあげられる．従って，より強固にリクエストを保護したい場合は，組織間のデータを開示することなく，共通する要素だけを得るプロトコルである Private Set Intersection のような，リクエスタ側で暗号化したリクエストを複号せずに分析を行う方法を選択すべきである．ただし，暗号化した状態での分析には，分析手法に制限があることや，分析処理時間が増加するといった欠点があるため，

実施したい分析とリクエストの秘匿化必要性に応じて，手法を選択すべきである．

次に，レスポンダ側により分析ロジックが改ざんされるリスクがある．本リスクに対しては，分析ロジックを難読化することで，改ざんを防ぐことができる．また，分析ロジックに電子署名を付与し，それを検証することにより，分析ロジックの改ざんやレスポンスの改ざんを防ぐことができる．

最後に，分析ロジックは他の分析ロジックやレスポンダに不正アクセスされるリスクがある.本リスクに対しては，分析ロジックへの接続をプラットフォームからのみに制限する．また，分析ロジックに対して不正アクセスを検知する仕組みや，脆弱性管理，不正アクセスされた際にイメージから分析ロジックを復元する機能を導入することが対策として考えられる．

次に，表 2 にレスポンダへの脅威を示す．まず，レスポンダへの脅威として，分析ロジックが分析に必要とする以上の情報を取得するリスクがある．本リスクに対しては，リソース接続 API で分析ロジックが接続可能なインテリジェンスを制御することで対策する．また，分析ロジックに付与された電子署名を検証することにより，分析ロジックを認証し，認証の結果に応じて分析ロジックの認可を制御する．また，許可されていないリソース接続 API への接続をリアルタイムに監視することで，異常を検知することができる．その他，リソース接続 API への接続ログを保管することで，監査することができる．

次に，分析ロジックがレスポンダから取得したデータを漏洩させるリスクについて説明する．本リスクに対して，分析ロジックと外部との通信は必ずプラットフォームを介するようにすることで，一元的な監査・制御を行う．具体的には，出力制限部によって，分析ロジックから外部へ送られるデータ量を制限することや，リクエスタのリクエスト，分析ロジックのレスポンスをクエリーログとして保管しておき，必要に応じてリクエスタ側にリクエストの復号・開示を要求できるようにする．尚，リクエスタ側は，開示要求に応じるために，暗号化に利用した鍵を保管しておく必要がある．

次に，インテリジェンス共有プラットフォームが，本プラットフォームに参加をしていない不特定多数に利用されるリスクについて説明する．本リスクに対しては，プラットフォームの WebAPI に認証機能を具備することで対策する．

次に，分析ロジックがレスポンダのシステムに不正アクセスするリスクについて説明する．本リスクに対しては，レスポンダのシステムで不正侵入を検知する仕組みや，脆弱性管理，不正アクセスされた際にイメージから復元することで対策する．

最後に分析ロジックがリソースを大量に消費し，レスポンダのサービスを妨害するリスクについて説明する．本リスクに対しては，分析ロジックが利用可能な CPU・メモリ量を制限することにより，分析ロジックのリソース消費を制御する．また，分析ロジックがリソース接続 API を利用できる回数，同時接続数を制限することで，インテリジェンスを格納しているデータベースのリソース消費量を制御する．

表 1　リクエスタへの脅威

#	分類	脅威
1	機密性	リクエスト内容の漏洩
2	完全性	レスポンダによる分析ロジックの改竄
3	完全性	他の分析ロジックが不正にリクエスタの分析ロジックに接続する

表 2　レスポンダへの脅威

#	分類	脅威
1	機密性	分析ロジックによる必要以上のインテリジェンス取得
2	機密性	分析ロジックによるレスポンダのデータ漏洩
3	機密性	第 3 者によるシステムの利用
4	完全性	分析ロジックによるレスポンダのシステムへの不正接続
5	可用性	分析ロジックによるリソース（CPU，メモリ，DB 接続）大量消費

3.2 ドメイン分析ロジック

本節で述べるドメイン分析ロジックで用いる異常検知アルゴリズムは，著者らが研究してきた AED[8]をベースに，[12]にある N-gram による分析を追加で援用している．

3.2.1 分析ロジックの要件

自組織だけではなく他組織上で実行されるという特徴から，分析ロジックには(1) 使用するメモリを一定量以内に抑える，(2) CPU 負荷を一定内に抑える，(3) インターネットへの通信を制限する，(4) 必要に応じてレスポンダが分析内容を監査できるようにする，(5) 必要な情報の一部がレスポンダ上に無い場合でも，分析を実施できるようにする，という固有要件がある．後述の設計では，上記要件をどのように満たしているのか言及する．尚，要件(4)については，3.1.3 節にて言及済みである．

3.2.2 入力

本実装では接続ログの各エントリは表 3 のカラムを持つことを想定する．

表 3　接続ログのカラム

#	カラム	例
1	接続日時	2021-02-01T12:00:00.000
2	接続元の識別子	192.168.0.2
3	接続先サイトのドメイン名	www.example.com
4	ドメイン名に対応する IP アドレス	10.0.0.10

Vol.2021-DPS-186 No.19
Vol.2021-CSEC-92 No.19
2021/3/15

接続日時，接続元の識別子，接続先サイトのドメイン名，ドメイン名に対応する IP アドレスは何れも一般的な Web Proxy ログ，DNS ログから取得できる情報である．分析フェイズにおいて リクエスタから送信される分析リクエストに含まれるクエリは，リクエスタ側で接続が発生した日時，問合せ対象ドメイン名，ドメイン名に対応する IP アドレスの 3 つであり，接続元識別子は必要としない．

3.2.3 ドメインの良性・悪性判定

図 4 にドメインの異常検知アルゴリズムの概要を示す．アルゴリズムは，接続に関する情報（接続日時，ドメイン，IP アドレス）を入力として与えられ，不審度を出力する．ドメイン悪性判定技術の中には WHOIS 情報やレジストリ情報を問い合わせるためにインターネットへの接続が必要なものも多いが[10][11]，本アルゴリズムは追加情報を取得するための外部接続を必要としない．これにより 3.2.1 節の要件(3)が満たされる．本アルゴリズムは「頻度ベースフィルタ」,「特徴ベースフィルタ」の 2 段階から構成される．

頻度ベースフィルタは，リクエスタ側組織に於いて，問合せ対象ドメインへの接続が過去に発生した頻度を基に，非悪性である確度が高いドメインをフィルタリングする処理である．たとえば，問合せ対象ドメインに対して，当該組織内の多数のユーザから過去何度も接続があるならば，当該ドメインは良性である確度が高いと判断できる．同様に，悪性ドメインの生存期間は一般的に短いことから，古くから接続していた履歴があるドメインは良性である確度が高いと判断することができる．問い合わせ内容がこのフィルタに合致した場合，アルゴリズムは不審度スコア=0 として良性判定を返して処理を終了する．一方，問合せ内容がフィルタに合致しない場合は，次の特徴ベースフィルタに処理が移る．

特徴ベースフィルタは，問合せ対象接続の特徴（ドメイン名，ドメイン階層，IP アドレス，国番号・ASN，接続日時等）が，問合せ先組織での接続履歴と比べてどの程度乖離しているのかを基に，不審度スコアを算出する．不審度スコア算出は，(1)近傍性，(2)カテゴリ類似性，(3)ドメイン名類似度という 3 つの観点を基に行う．(1)近傍性では，(a)問合せ対象ドメインと 3rd level ドメインが一致する別ドメインに接続したことがある自組織内ユーザ数，(b)問合せ対象 IP アドレスと/24 で一致するアドレスに接続したことがある自組織内ユーザ数，を基に近傍性を算出する．(2)カテゴリ類似性の観点では，過去接続の各特徴（接続した時刻や接続ドメインのトップレベルドメイン等）をカテゴリとして分け，問合せ対象接続がどのカテゴリに属するかにより，接続の類似性を算出する．(3)ドメイン名類似度の観点では，ドメイン名の n-gram を基に，組織が過去に接続したことがあるドメイン名との類似性を算出する．

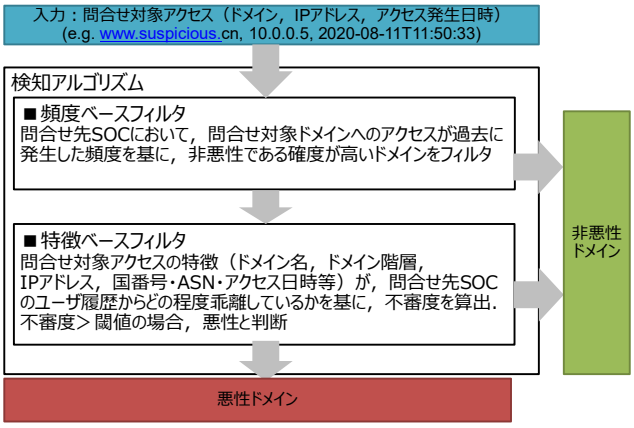

図 4　アルゴリズムの概要

3.2.4 疑似コード

次に，リスト 1 に示す疑似コードを用いてアルゴリズムの詳細について説明する．アルゴリズムの入力には，先述のドメイン(q_domain), IP アドレス(q_IP), 接続日時(q_time)以外に，Country Code (q_CC), Autonomous System Number (q_AN)がある．q_CC, q_ASN は q_IP を基に解決することができる．具体的なライブラリ名は後述する．それ以外にも，アルゴリズムで用いる閾値として，TH_hosts, TH_day, TH_close, w_closeness, w_fitness, w_normality がある．w_closeness, w_fitness, w_normality は特徴ベースフィルタの各観点の重みづけを示すものであり合計値は 1.0 となる．Dataset はモデル作成に使用する，レスポンダ側組織での接続ログを指す．また N は後述の n-gram で用いる値である．

1 行目～3 行目は頻度ベースフィルタの処理を示す．2 行目にあるように，問合せ対象の q_domain に接続したことがある Dataset 内ユーザ数が TH_hosts を超えるか，q_domain への接続履歴が TH_day より前にある場合，q_domain は良性である確度が高いと判断し，不審度スコア=0.0 を返す（3 行目）．これは，多くのユーザが接続するドメインまたは古くから接続があるドメインは良性である確度が高いという知見からくるものであり，既存研究[10]や我々の過去の研究[8]でも活用されている．分析フェイズで q_domain と Dataset 内ユーザの突合を高速に行うために，分析フェイズで予めドメインごとの接続元ユーザ数や接続日時を記録する必要がある．この時，問合せ先組織のユーザ数や接続ドメイン数によっては多くの記憶領域を要する必要がある．そこで我々は，記憶領域のメモリサイズを一定に保つために，ドメインごとの接続ユーザ数や TH_days より過去に接続があるドメインの一覧を，Count-min Sketch 及び Bloom Filter を用いて管理することにした．Count-min Sketch は key に対応するカウント数を確率的に保持する機構であり，key 数が増えてもメモリサイズは変わらない．同様に Bloom Filter はアイテム一覧を一定のメモリサイズで保持する機構である．これにより，分析ロジックの使用メモリサイズを一定以内に抑えるという 3.2.1 節の要件(1)を解決する．

Vol.2021-DPS-186 No.19
Vol.2021-CSEC-92 No.19
2021/3/15

```
1   #q_domain にアクセスしたユーザ数が TH_hosts を超える，
    もしくは TH_day より過去にアクセスがある場合，無条件に score = 0 とする
2   if | {h | h in Dataset and h.hasAccesed(q_domain)}| > TH_hosts
    or | {h | h in Dataset and h.hasAccesedBefore(q_domain,TH_day)}| > 0:
3           return score = 0.0
4   #q_domain の 3rdlevel domain または g_IP/24 にアクセスしたユーザ数に応じて，
    通常アクセス先との近さ closeness を計算.
5   domain3rd = get3rdLevelDomain(q_domain)
6   Closeness = | {h | h in Dataset
    and h.hasAccesed(domain3rd)}| + |{h | h in Dataset and h.hasAccessed(g_IP/24)}|
7   Closeness = min(closeness/TH_close, 1.0)
8   #CC, AN, TLD, 日中／夜間，weekday/weekend，ドメイン階層数・サブドメイン長から，
    カテゴリ観点での fitness（正常度合）を計算
9   domain1st = getTopLevelDomain(q_domain)
10  type_of_hour = getTypeOfHour(q_time)
11  type_of_day = getTypeOfDay(q_time)
12  domain_length_range = log16(maximum length of subdomaind of g_domain)
13  domain_hierarchy_range = log2(#of dot in g_domain)
14  fitness = WDOD(q_CC, q_AN, q_agent, domain1st, type_of_hour,
    type_of_day, domain_length_range, domain_hierarchy_range)
15  #SLD 以降の各レベルのドメインを対象とした n-gram の rank の観点，および階層数から，
    ドメイン名に関するデータセットとの類似度を計算する
16  for token in getN-gram(q_domain):
17          name_rank += log2(rank(token)) / log2(#unique tokens in Dataset)
18  normality = 1.0 - name_rank / (#tokens in g_domain)
19  #ネットワーク上の近さ，およびカテゴリ観点の異常度を基に最終的な outlier score を計算する
20  return score = 1.0-(w_closeness*closeness+w_fitness*fitness+w_normality*normality)
```

リスト 1　異常検知アルゴリズムの疑似コード

また，Count-min Sketch，Bloom Filter 共に処理に必要な計算量は key 数やアイテム数に因らず一定であり，要件(2)も満たす．尚，両者の課題として，保持する key 数或いはアイテム数が増えるほど，実際には保持していない key やアイテムを保持していると誤判断する false positive の発生頻度が高くなることが挙げられる．

4 行目～7 行目は, 特徴ベースフィルタの中の近傍性に関する処理である．ここでは，q_domain の 3rd level ドメインおよび q_IP の/24 アドレスへ接続したユーザ数をカウントし，その数に応じて closeness を計算する．ユーザ数≧TH_close のとき closeness=1.0 となる．この処理は，一定数以上のユーザから接続履歴があるドメインと同じ階層下にあるドメイン, あるいは接続履歴がある IP アドレスと同サブネット下の IP アドレスは良性である可能性が高いというヒューリスティックに基づいている．モデル作成フェイズでは，Count-min Sketch を使用して 3rd level でのドメイン接続ユーザ数及び/24 での IP アドレス接続ユーザ数をカウントしておく．

8 行目～14 行目は，特徴ベースフィルタの中のカテゴリ類似性に関する処理である．ここでは，IP アドレスの国名・AS 番号，ドメインの TLD・階層数・サブドメインの最大ラベル長，接続日時の日中／夜間・平日／週末，という 7 種類の特徴量を基に, Dataset 内の履歴と比較した, q_IP, q_domain, q_time の類似度 fitness を求める．ここで，各特徴量は numerical ではなく categorical なデータとして処理する．例えば，日中／夜間の違いや AS 番号の数値に大小関係は無く, カテゴリとして処理するのが好ましい. 階層数, ラベル長は数値として扱うことも可能だが，今回は階層数

が浅い（2 以下）／深い（3 以上），ラベル長が一定値（16）以上／未満，という区分にわけ，カテゴリとして扱うことにする．また，GeoIP[13]を使うことでインターネット接続無しに，IP アドレスから国名・AS 番号を求めることができる．

モデル作成には one-class SVM のような機械学習を用いる方法も考えられるが，演算コストを抑えるために本研究では Weighted density-based outlier detection（WDOD）を用いる[14]．WDOD の詳細は割愛するが，特徴量ごとに各カテゴリの相対発生頻度を求めることで入力データの類似度を算出する，特徴量間の依存関係を考慮しない，という特性がある．特徴量間の依存関係を考慮しないというのは精度上の欠点になりうるが，本研究のように，他組織上で動作する分析ロジックでは 2 つの理由でそのシンプルさが功を奏する．1 つは WDOD の計算は軽量であることがあり，もう 1 つは Dataset が複数のログに分かれていた場合でも演算を行えることである．例えば，問合せ先組織で使用できるデータセットが<ユーザ ID, 接続先ドメイン，接続日時>，<ユーザ ID, 接続先 IP アドレス，接続日時>の 2 つログに分かれている場合を想定する．この場合ドメイン名と IP アドレスを一対一に紐づけることはできず, モデル作成フェイズで名前解決をするのは 3.2.1 節の要件(2)(3)に反することになる．しかし WDOD では特徴量ごとに演算を行うため, ドメインと IP アドレスが紐づいていなくとも類似度の算出が可能である．以上，分析に必要な特徴量間の紐づけが不要となることで，3.2.1 節の要件(2)及び(5)を満たす処理が可能になる．理論上，ドメイン名，IP アドレス，接続日時は別々のログとして与えられたとしても処理を実行することが可能である．モデル作成フェイズでは WDOD のモデル作成を行い，分析フェイズではクエリと WDOD モデルとの類似度を求める．

15 行目～18 行目はドメイン名類似度 normality の算出処理であり，Dataset 内にあるドメインと q_domain との類似度を求める．ドメイン名類似度は，q_domain の各ラベルの n-gram が，Dataset 内のドメインの n-gram に含まれる頻度を基に算出する．ここで n-gram とは，ある文字列を n 文字ごとに区切ったサブ文字列の集合である. 例えば,”hitachi”の 3-gram は”hit”，“ita”，“tac”，“ach”，“chi”の 5 つである．ドメイン名の類似度に着目した既存研究としては[12][15]などがある. 特に[12]は本研究と同じく n-gram を用いている．違いとしては，本研究では類似度の値を 0~1 の間に入るよう正規化するために出現頻度のランクを求めている点がある．モデル作成フェイズでは Dataset 内のドメインの n-gram の抽出及びランク付けを行い, 分析フェイズでは q_domain の n-gram のランクを求める．

19 行目～20 行目は，前段で求めた closeness, fitness, normality を基に不審度スコアを算出する．算出に当たっては各値を w_closeness, w_fitness, w_normality で重みづけし加重平均を求める．前述の通り不審度スコアは 0~1 の値を取り，値が大きいほど不審度が高いことを指す.

3.3 AED

AED は未知サイトへの接続が発生した際に, 自他組織内のドメイン分析ロジックに不審度の問合せを行う．得られた不審度の内，最小のものが，予め定めた閾値を超えた際に, 悪性サイトへの接続であると判定し, 追加認証を行う．これにより，自組織で不審と判定された場合でも，他組織の観点で正常であれば正常と見做され，誤検知を削減できる．

4. 関連研究

悪性サイトへの機械的な接続を遮断する方法として，インターネットへの出口に設置したプロキシのユーザ認証機能を利用する方法がある[16]．しかし，遠隔操作型マルウェアの中にはユーザ認証機能を突破するものも存在している[17]．一方提案手法は，悪性サイトへの接続時に機械には突破困難な追加認証を課すことで対応している．

追加認証による対策は，日々の業務の妨げになるという課題が存在する．悪性サイトへの接続を防ぎつつも，安全性の高いサイトへの接続時には CAPTCHA を省略することで業務への影響を軽減する手法が角田らにより提案されている[18]． また，中鉢ら[19]は複数組織がブロックチェーン技術を用いて WL を共有することにより, AED が抱えていた，小規模組織で WL の拡張が小さく，追加認証が多く発生する課題を解決しようとしている．これらいずれの対策も, 未知サイトへ接続する際に必ず追加認証が発生する．提案手法では，アクセス傾向を元に良性悪性の判定を行うことで, 未知サイト接続時の追加認証の発生数を削減する．

未知サイトの良性悪性判定についても様々な研究が行われている．文献[20]は，Convolutional Neural Network を拡張することで, プロキシログに含まれる宛先 URL の系列から, ドライブバイダウンロード攻撃に関する悪性 URL 系列を検知する手法を提案している．文献[21]は，Bayesian sets と呼ばれる類似要素探索アルゴリズムを用いて，既知の悪性 URL 群と類似した URL を検索する方法を提案している．文献は[22]，DNS ログからドメインと IP アドレスの関連をグラフ化し, 確率伝播アルゴリズムを改良することで, 悪性ドメインを高精度で検知する手法を提案している．これらの手法では，悪性サイトの判定に予め用意した悪性サイトのデータを必要とする．一方提案手法は，インターネットへのアクセスログと，接続先 IP アドレスの ASN，国番号のみを使用し，悪性サイトのデータを必要としない点で異なる．

5. おわりに

本稿では悪性サイトへの通信遮断を目的として，(1)組織が保有する，サイトへの接続ログを元に接続傾向を分析し，その傾向との乖離度を利用して接続の良性，悪性を判定する，(2)自組織の接続傾向だけでなく，他組織の接続傾向を利用することで判定の精度向上を狙う，(3)判定で誤って悪性と判断した良性サイトへの接続を即時遮断するのではなく，機械には突破困難な追加認証を課し，突破できなかった場合のみ接続を遮断する，という特徴を持つ分散型 AED というシステムを設計，提案した．

このシステムにより，未知の悪性サイトにも有効性があり，業務遂行に必要なサイトを誤って悪性と判定した際の業務阻害を緩和することが期待できる．

今後は，提案システムの実装，評価を行っていく．

参考文献

[1] “Norton: What Are Malicious Websites?”, available from https://nz.norton.com/internetsecurity-malware-what-are-malicious-websites.html (accessed 2021-02).
[2] Zhao, B.Z.H., Ikram, M., Asghar, H., Kaafar, M.A., Chaabane, A. and Thilakarathna, K.. A decade of malactivity reporting: A retrospective analysis of internet malicious activity blacklists, Proc. 14th ACM Asia Computer Communication and Security (ASIA CCS’19), pp.1-13 (2019).
[3] “IPA :「新しいタイプの攻撃」の対策に向けた設計・運用ガイド 改訂第 2 版”．入手先 (https://www.ipa.go.jp/files/000017308.pdf) (参照 2021-02）．
[4] 近藤 賢郎，細川 達己，重本 倫宏，藤井 康広，中村 修．分散型 SOC アーキテクチャに基づいた複数組織間におけるセキュリティ・オペレーションの連携，マルチメディア，分散協調とモバイルシンポジウム 2018 論文集，pp.872-878 (2018).
[5] 西嶋 克哉，川口 信隆，重本 倫宏，近藤 賢郎，中村 修．セキュリティ・オペレーションにおける秘匿データ分析システムの提案，マルチメディア，分散協調とモバイルシンポジウム 2020 論文集，pp.284-289 (2020).
[6] 仲小路博史，藤井康広，磯部義明，重本倫宏，鬼頭哲郎，川口信隆，林 直樹，下間直樹，菊池浩明．人間行動を用いた自律進化型防御システムの提案，2016 年暗号と情報セキュリティシンポジウム（SCIS2016），pp.1‐8 (2016).
[7] Nakakoji, H., Fujii, Y., Isobe, Y., Shigemoto, T., Kito,T., Hayashi, N. Kawaguchi, N., Shimotsuma, N. and Kikuchi, H.. Proposal and Evaluation of Cyber Defense System Using Blacklist Refined Based on Authentication Results, The 19th International Conference on Network-Based Information Systems (NBiS2016), pp.135-139 (2016).
[8] 重本倫宏，藤井翔太，来間一郎，鬼頭哲郎，仲小路博史，藤井康広，菊池浩明．WL を用いた自律進化型防御システムの開発，情報処理学会論文誌，Vol.59，No.3，pp.1050‐1060 (2018).
[9] W. Diffie, M. E. Hellman. New Directions in Cryptography, IEEE Transactions on Information Theory, vol.IT-22, No.6, pp.644-654 (1976).
[10] Alina, O., Zhou, L., Robin, N., Kevin, B.. MADE: Security Analytics for Enterprise Threat Detection, Proc. of ACM ACSAC’18, pp.124-136 (2018).
[11] Michael, W., Jun, W.. Unsupervised Clustering for Identification of Malicious Domain Campaigns, Proc. of ACM RESEC’18, pp.33-39 (2018).
[12] Zhao, H., Chang, Z., Bao, G., Zeng, X.. Malicious Domain Names Detection Algorithm Based on N-Gram, Hindawi, pp.1-9 (2019).
[13] “GeoIP”, available from https://dev.maxmind.com/geoip/ (accessed 2021-02).
[14] Ayman, T., Ali S.H.. Anomaly Detection Methods for Categorical Data: A Review, ACM Computing Surveys, Vol.52, No.38 (2019).
[15] Jayaram, R., David, J.M.. Unsupervised, low latency anomaly detection of algorithmically generated domain names by generative probabilistic modeling, Journal of Advanced Research, pp.423-433 (2014).
[16] “IPA :「高度標的型攻撃」対策に向けたシステム設計ガイド”．入手先 (https://www.ipa.go.jp/files/000046236.pdf) (参照 2020-02).
[17] “IIJ：新型 PlugX の出現”．入手先 (https://sect.iij.ad.jp/d/2013/11/197093.html) (参照 2020-02).
[18] 角田朋，大鳥朋哉，藤井康広，谷口信彦，木城武康．グレーリストを用いたホワイトリスト／ブラックリストの自動生成によるマルウェア感染検知方法の検討，情報処理学会研究報告．SPT，Vol.2014, No.16, pp.1-7 (2014).
[19] 中鉢かける，中村嘉隆，稲村浩．標的型攻撃対策のためのブロックチェーン技術を用いたホワイトリスト方式防御システムの実現性に関する検討，研究報告モバイルコンピューティングとパーベイシブシステム（MBL），2018-MBL-89, No.6, pp.1-8 (2018).
[20] 山西 宏平，芝原 俊樹，高田 雄太，千葉 大紀，秋山 満昭，八木 毅，大下 裕一，村田 正：畳み込みニューラルネットワークを用いた URL 系列に基づくドライブバイダウンロード攻撃検知，コンピュータセキュリティシンポジウム 2016 論文集，pp.811-818 (2016).
[21] 孫 博，秋山 満昭，八木 毅，森 達哉．既知の悪性 URL 群と類似した特徴を持つ URL の検索，コンピュータセキュリティシンポジウム 2014 論文集，pp.1-8 (2014).
[22] Hau, T., An, N., Phuong, V., Tu, V.. DNS graph mining for malicious domain detection, 2017 IEEE International Conference on Big Data (Big Data), pp.4680 - 4685 (2017).