インシデントの再現を目的とした サイバーレンジ構築支援システムの提案

砂川 真範[1]　知念 賢一[1]　篠田 陽一[2]

概要：コンピュータ・ネットワークの発展や IoT 製品の普及により，様々な機器がネットワークに接続され，それらの機器を踏み台とした攻撃や，それらの機器を対象としたサイバー攻撃が横行している．結果，様々なセキュリティインシデントが発生している．過去に発生したインシデントを再現することはサイバー攻撃対策として重要である．そこで，本論文では発生したインシデントの記録である，インシデントデータに着目しインデントの再現を目的としたサイバーレンジ構築支援システム“Hack 祭”の提案を行う．まず，サイバー攻撃の現状とそれらの記録を述べ，サイバー演習およびサイバーレンジの定義を行う．そして，類似する既存研究について述べたのち，Hack 祭で必要となるサブシステムを述べる．

キーワード：インシデントレポート,Cyber Range, サイバー攻撃

Proposal of Cyber Range Construction Support System for Reproducing Incidents

Masanori SUNAGAWA[1]　Ken-ichi CHINEN[1]　Yoichi SHINODA[2]

1. はじめに

コンピュータ・ネットワークの発展や IoT 製品の普及により，様々な機器がネットワークに接続されている．しかし，その一方それらの機器やそれらの機器を踏み台としたサイバー攻撃が横行している．そららのサイバー攻撃により様々なセキュリティインシデントが発生している．過去に発生したインシデントを題材にサイバー演習を行うことは，サイバー攻撃への対策として重要であり実際のインシデントが発生した場合の手順の確認との訓練やインシデントへの対応方法が最適であったかの振り返りを含めた評価を行うことを可能とする．また，インシデントの記録は，発生した事象やその対応を文章化し一般に公開するインシデントレポートと内部向けに詳細な情報を含め専用のフォーマットで記録したインシデントデータの 2 種類が存在する．一般的にインシデントレポートやインシデントデータを元にインシデントが発生した時の状況をサイバーレンジで再現するのは，トポロジー，OS やアプリケーションのバージョンなどの前提条件やインシデント発生のトリガーやその結果などが必要となり難しい．図 1 はセキュリティインシデントの記録例である．WEB アプリケーションを

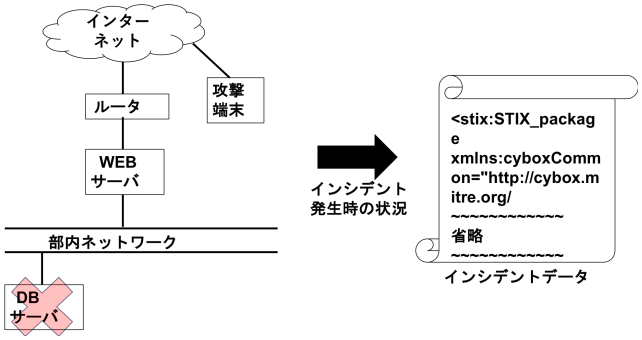

図 1　インシデントの記録例

トリガーとして，DB サーバへ攻撃を行なっている．このように，単純なインシデント例であってもルータ，WEB サーバ，DB サーバのコンフィグレーション等による動作しているアプリケーションの状態や部内ネットワークの接続形態などの情報が必要である．

2. 目的

本研究は，1 章で述べたように，過去に発生したインシデントをサイバー演習の題材として容易に使用できる，インシデントの再現を目的とするサイバーレンジ構築支援システム《Hack 祭》の設計と実装を目的とする．使用するインシデントの記録として，1 章で述べたしたインシデントデータを対象とする．これには，右記の理由によるためである．

(1) インシデントレポートを対象とした場合，文章として

[1] 北陸先端科学技術大学院大学 先端科学技術研究科
Graduate School of Advanced Science and Technology, Japan Advanced Institute of Science and Technology
[2] 北陸先端科学技術大学院大学 情報社会基盤研究センター
Research Center for Advanced Computing Infrastructure, Japan Advanced Institute of Science and Technology

記載されておりフォーマットが統一されておらず自由記述のため解析が難しい

(2) 多くの場合 PDF 形式で公開されているため，プログラムでの処理が難しい

(1) および (2) の問題を解決するためフォーマットが統一されプログラムでの処理が容易な XML 形式で記録されたインシデントデータを採用する．また，既存のインシデントデータのフレームワークとして，STIX[1]，CybOX[2]，TAXII[3] などがある．しかし，それらの既存のフレームワークを使用した場合でも 1 章で述べた前提条件が記録されていない．そのため，本研究の一環としてインシデントの再現で必要となる前提条件を明確にするとともに，フォーマットとして提案する．

3. サイバー演習およびサイバーレンジの定義

2 章で述べたサイバー演習には目的に合わせて様々なスタイルがある．同様に，目的に合わせた様々なサイバーレンジの種類がある．本章では，それらの一部について述べる．

3.1 サイバー演習

3.1.1 攻防型

サイバー演習参加者が相手のチームに対して，攻撃を行うとともに自らのチームを守る演習である．これは，攻撃と防御の両方の知識を身につけることを目的としている．

3.1.2 防御型

攻防型に比較的近いが，サイバー演習参加者は，自らのチームに対して行われる攻撃を守る演習である．これは，サイバー演習実施者が攻撃を行うことにより，攻撃手法などをサイバー演習実施者が学ぶ機会を減らしているため学んだ知識を悪用し実際にサイバー攻撃を行うリスクを低くしている．また，防御や対策などを中心に行うため実際にサイバー攻撃が発生した状況下に近く，実践的なサイバー演習といえる．

3.1.3 クイズ型

単純に，ジャンルごとの問題を解くサイバー演習である．これは，知識として身についているかなどを調べるのに適している．また，攻撃型や防御型と比較し，必要となるサーバ等の機材などが少ないためポータビリティがある．

3.2 サイバーレンジの種類

3.2.1 対称型

対称型のサイバーレンジは，3.1.1 で述べた攻防型のサイバー演習で使用することを目的とする．これには，各サイバーレンジ間のネットワークの連携や攻撃などのトラヒックがサイバーレンジ外へ流出しない仕組みが必要となる．

3.2.2 非対称型

非対称型のサイバーレンジは，3.1.2 で述べた防御型のサイバー演習で使用することを目的とする．これには，サイバーレンジ内のマシンに対して，自動で攻撃が行われる仕組みが必要となる．

3.2.3 コース

コース型のサイバーレンジは，3.1.3 で述べたクイズ型のサイバー演習を発展させたサイバー演習で使用することを目的とする．サイバーレンジのマシン内のユーザ，パスワード，カーネルバージョンなどをクイズの問題として出題し演習者がサイバーレンジのマシン内から答えを導く．そのため，サイバーレンジのマシンとクイズの問題を掲示するシステムで連携し情報の参照ができる仕組みが必要となる．

3.2.4 SandBox 型

SandBox 型のサイバーレンジでは，マルウェアの解析やハニーポットなどに使用することを目的とする．そのため，ネットワークやマシンへの影響が発生しないように，強固なセキュリティ機能が必要である．また，マルウェアや BOT の多くは，プロセスやマシンの様々な状態から実際に使用されているマシンか判定する機能を有しており，SandBox と判定した場合，解析されるのを避けるため，マルウェアの場合プログラム本体の自己削除やハニーポットの場合コネクションの切断などが行われる．SandBox と判定されないよう，背景トラヒックや動作しているアプリケーションなどを制御できる仕組みが必要となる．

4. 既存研究

4.1 Alfons

Alfons[4] は，ビルディングブロック型模倣環境構築システムである．OS のインストール，アプリケーションインストール，共通設定を行いテンプレートとして保存し，そららのテンプレートを複製配布し，ネットワーク設定を行ったのち，固有設定やコンテンツの配置を行っている．

4.2 CABIN

CABIN[5] は，サイバー演習統合管理システムである．複数の物理サーバーをクラスター化し 1 つの仮想環境として使用しサイバー演習環境用にそれぞれのリソースを分割し使用している．各サイバー演習の主催者は，提供を受けたリソース上でサイバー演習環境を構築する．また，CABIN では雛形トポロジーが提供されておりそららを用いてサイバー演習環境のネットワークを構築できる．さらに，CABIN はリアリティ支援機能や監視/観測機能を提供し主催者の選択によって利用が可能である．

4.3 CyRIS

CyRIS[6] は，サイバーレンジ構築支援システムである．サイバーレンジで使用するテンプレートイメージ，ネットワークトポロジ，作成するユーザ名，インストールするア

プリケーションが記述された YAML 形式の設定ファイルを使用する．それらの設定ファイルを使用することによりサイバーレンジの作成と管理を自動化をはかり，サイバーセキュリティトレーニングを容易にするためのツールである．有する機能として

(1) ユーザの作成/変更
(2) ファイヤーウォールの設定変更
(3) パッケージによるアプリケーションのインストール
(4) ソースコードによるアプリケーションのインストール
(5) カスタムインストール
(6) 攻撃のエミュレート
(7) キャプチャしたトラヒックの再生
(8) マルウェアのエミュレート
(9) コンテンツのコピー
(10)スクリプトの実行
(11)ネットワークの設定
(12)仮想マシンの複製

などが挙げられる．(1)，(2) は，システム設定の支援機能である．(3)，(4)，(5) は，アプリケーションなどのツールのインストール支援機能である．(6)，(7)，(8) は，インシデントエミュレーションの支援機能である．(9)，(10) は，サイバー演習で使用するコンテンツの管理支援機能である．(11)，(12) は仮想マシン展開の支援機能である．

4.4 CyTrONE

CyTrONE[7] は，サイバーセキュリティのトレーニングフレームワークである．問題掲示システムへの問題を登録し対応したサイバーレンジを 4.3 で述べた CyRIS を使用し構築する．これにより，実施者はサイバーレンジの構築と問題掲示システムへの登録を手動で行うことなくセキュリティトレーニングの実施できる．

4.5 STIX

STIX[1] は，脅威情報構造化記述形式である．キャンペーン，攻撃者，攻撃手法，検知指標，観測事象，インシデント，対処措置，攻撃対象などの情報群から構成される．また，XML ベースで記述されているため，脅威情報を自動で処理することが可能である．

4.6 CybOX

CybOX[2] は，サイバー攻撃観測記述形式である．コンピュータやネットワークの状態などの観測可能な属性の記述を定義する．観測対象には，ファイル，セッション，証明書やシステム構成要素などが含まれ，コンピュータシステムと観測対象の動作を記述するために拡張された言語である．

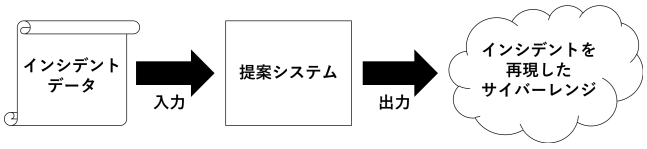

図 2　Hack 祭の動作イメージ

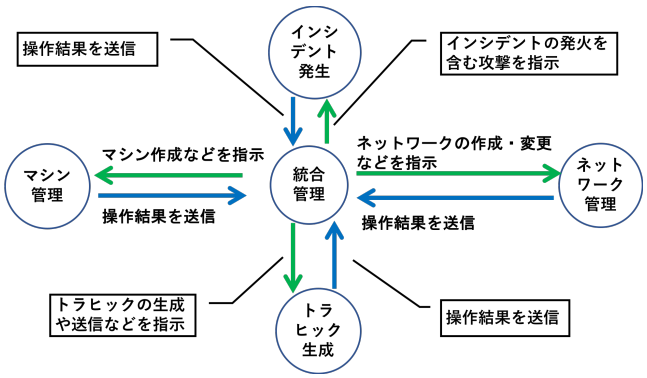

図 3　Hack 祭のアーキテクチャ

4.7 TAXII

TAXII[3] とは，検知指標情報自動交換手順である．脅威情報の転送と交換などに使用される．また，4.5 の STIX や 4.6 の CybOX と組み合わせて使用される．

4.8 サイバーセキュリティ経営ガイドライン

サイバーセキュリティ経営ガイドライン [8] とは，経済産業省が提唱する，経営者を対象するサイバーセキュリティのガイドラインである．サイバー攻撃から守るため，経営者が認識する必要がある 3 原則と経営者がサイバーセキュリティ対策の責任者に指示する重要１０項目が記載されている．「付録 C インシデント発生時に組織内で整理しておくべき事項」は，項目ごとに表形式で記録を行う．

5. Hack 祭

Hack 祭では，図 2 のように，入力されたインシデントデータを元にインシデント発生時のサーバやクライアントの上で実行されていたサービス，インシデントのトリガーとなった攻撃などを再現したサイバーレンジの構築支援を行う．

5.1 サブシステム

Hack 祭では，インシデント発生時の状態を再現することを目的とする．そのため，様々なインシデントでも対応できるよう汎用性を持たせる必要があり，機能ごとにサブシステム化する．Hack 祭とサブシステムの関係を図 3 に示す．

5.1.1 マシン管理システム

インシデント発生状況下におけるマシン数や物理マシン・仮想マシンといった種別はインシデントが発生した状況の再現を行うレベルにより異なる．そのため，物理マシ

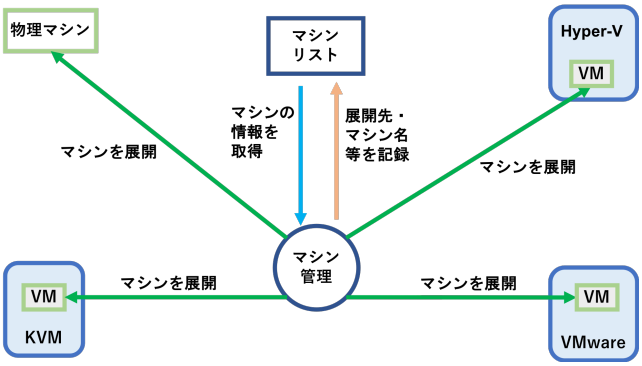

図 4　マシン管理システムのイメージ

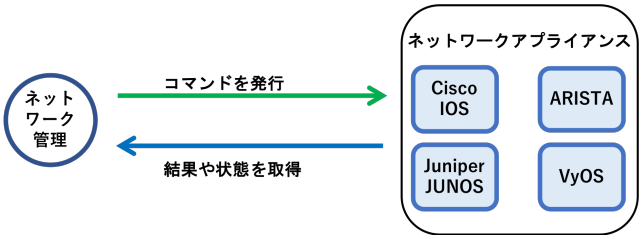

図 5　ネットワーク管理システムのイメージ

ン・仮想マシンだけではなく仮想マシンを動作させるハイパーバイザにより仕組みが異なるため，サブシステム内で利用者が拡張や定義を可能とする．また，専用のサブシステムで管理する．図 4 にイメージを示す．

5.1.2　ネットワーク管理システム

インシデント発生状況下におけるネットワークはインシデントが発生した状況の再現を行うレベルにより，物理的なアプライアンス，ベンダーが販売している専用のバーチャルアプライアンス，それら以外の仮想的なネットワークに分類することができる．そのため，複数のアプライアンスに対応する必要があり，アプライアンスごとに設定コマンドや設定方法が異なるため，左記に述べたマシン管理システム同様にサブシステム内で利用者が拡張や定義を可能とする．また，インシデントのケースにより発生前と発生下においてネットワークトポロジが変更されている場合も想定されるため，それらに対応できるようにサブシステムで管理を行う．図 5 にイメージを示す．

5.1.3　トラヒック作成システム

インシデントの再現を行う上で，トラヒックの状態は重要である．また，トラヒックには攻撃者による攻撃時に発生したトラヒック，攻撃時以外で発生したトラヒック，その他の IoT 機器などで意識せずに発生したトラヒックに分類することができる．本サブシステムで取り扱うトラヒックとして攻撃時以外で発生したトラヒックと意識せずに発生したトラヒックである．しかし，実際のインシデント発生状況下ではトラヒックの記録が行われていない場合が多い．そのため，擬似トラヒックの生成が必要となり，擬似トラヒックを使用することにより実際のインシデント発生下の状況に近くなる．また，トラヒックの種類や内容をカスタマイズできるように利用者が拡張や定義を可能とする．これにより，模擬トラヒックとして使用するためキャプチャしたトラヒックの再生やソフトウェアによりトラヒックの生成，トラヒックジェネレーターなどの専用アプライアンスによるトラヒックの生成を可能とする．

一般的に 1 日のトラヒックは以下のように分類でき内訳は以下の通りである．使用したデータは WIDE MAWI WorkingGroup[9] で公開されている 2018 年 5 月 12 日である．

(1) IPv4 (97.32%)
(2) IPv6 (2.68%)

(1)，(2) より，IPv4 による通信が大半であると言える．また，(1) の内，tcp による通信は，54.93% であり，http(19.19%) および https(19.81%) による通信がトップである．tcp 以外では icmp(25.27%) による通信の割合が高い．それらの結果より，トラヒック生成システムでは，IPv4 を使用した http，https，icmp の各トラヒックを生成すると現実的なバックグラウンドトラヒックとなる．

5.1.4　インシデント発生システム

Hack 祭では，インシデントの再現を目的としている．そのため，インシデント発生システムは重要である．インシデントの発生方法として，タイミングによるプロセスの実行などである．インシデントを発生させるタイミングは，4 パターン存在する．

(1) 時間経過
(2) 時間指定
(3) イベントトリガー
(4) ランダム

(1) では，開始された時間を記憶しておき，そこからの指定した時間が経過した場合，インシデントを発生させる手法である．

(2) では，インシデントを発生させる時間を指定し，その時間になるとインシデントが発生する手法である．特定の日時などを対象としたインシデントの発生を再現することが可能となる．しかし，日時を定義する必要があり，事前のテストを実施する際などに変更が必要となる．

(3) では，イベントを引き金として，受講者の操作を監視し，それらに合わせインシデントを発生させる手法であり，対策が行われた場合の攻撃パターン変更などを可能とする．しかし，各パターンの記述が必要となるだけではなく，操作を監視するロガーを受講者が不審に感じる場合や攻撃の一部と勘違いする場合がある．そのため，偽装を行う必要がある．偽装が必要となるのは以下の 2 つの項目である．

(1) プロセスの偽装
(2) 通信の偽装

(1) では，プロセスを一般的な OS で動作しているプロ

セス名に変更することにより，ロガーであることを認識しにくくできる．

（2）では，ロギングを行い，プロセスを実行させる際などに行う通信を 5.1.3 で述べたバックグラウンドトラヒックと同じプロトコルにすることによるバックグラウンドトラヒックとの区別がつきづらくなり，認識しずらくなる．

（4）では，乱数によりインシデントが発生するタイミングが決定する．ユニークとなるため，同じインシデントを題材として複数回実施したい場合に適している．

6. おわりに

インシデントの再現を目的としたサイバーレンジ構築支援システムに必要な機能は，マシン管理，ネットワーク管理，トラヒック作成，インシデント発生である．トラヒック作成，インシデント発生は，インシデントの再現を行う上で重要度が高い機能である．しかし，実際のインシデント発生状況下ではトラヒックの記録が行われている場合が少ないため，一般的な 1 日のトラヒックから分類した結果に基づき，http，https，icmp の各トラヒックをバックグラウンドトラヒックとして使用する．インシデント発生では，インシデントを発生させるタミングを定義しイベントトリガーで操作の監視で必要となるロガーの偽装について定義を行った．Hack 祭はこれらの機能を実現し，インシデントの再現を容易とする，インシデントの再現を目的としたサイバーレンジ構築支援システムである．本論文では，サイバー演習のスタイル，サイバーレンジの種類，類似する既存のシステム，既存のインシデントデータのフォーマット，Hack 祭の機能の定義やサブシステムの定義を述べた．今後は，Hack 祭の各サブシステムの実現と各課題についての研究開発を行う．

参考文献

[1] Stix - structured threat information expression (archive) — stix project documentation. `https://stixproject.github.io/`.

[2] Cybox - cyber observable expression — cybox project documentation. `http://cyboxproject.github.io/`.

[3] Trusted automated exchange of indicator information (taxii) — taxii project documentation. `https://taxiiproject.github.io/`.

[4] Shingo Yasuda, Ryosuke Miura, Satoshi Ohta, Yuki Takano, and Toshiyuki Miyachi. Alfons: A Mimetic Network Environment Construction System. In *Testbeds and Research Infrastructures for the Development of Networks and Communities (TridentCom 2016)*, pp. 59–69, November 2016.

[5] 太田悟史, 安田真悟, 湯村翼, 高野祐輝. 次世代サイバー演習環境に向けて. マルチメディア，分散協調とモバイルシンポジウム 2016 論文集, pp. 1776–1782, jul 2016.

[6] Razvan Beuran, Pham Cuong, Tang Thanh Dat, Ken ichi Chinen, Yasuo Tan, and Yoichi Shinoda. Cybersecurity Education and Training Support System: CyRIS. *IEICE Transactions on Information and Systems*, Vol. E101-D, No. 3, March 2018.

[7] Razvan Beuran, Pham Cuong, Tang Thanh Dat, Ken ichi Chinen, Yasuo Tan, and Yoichi Shinoda. CyTrONE: An Integrated Cybersecurity Training Framework. In *International Conference on Information Systems Security and Privacy (ICISSP 2017)*, pp. 157–166, February 2017.

[8] サイバーセキュリティ経営ガイドライン. `http://www.meti.go.jp/policy/netsecurity/mng_guide.html`.

[9] Wide mawi workinggroup. `http://mawi.wide.ad.jp/`.