Vol.2016-CSEC-73 No.6
Vol.2016-IOT-33 No.6
2016/5/26

IoT向けSIEMシステム設計のための脅威シナリオ検討

三品 拓也[1]　小野 康一[1]　宗藤 誠治[1]　佐藤 直人[1]　初鳥 陽一[1]　佐藤 史子[1]

概要：近年，あらゆるモノがインターネットに接続される IoT (Internet of Things) の時代を迎えつつある．IoT デバイスは物理的アクセスとインターネット経由のアクセスがいずれも容易に行えるため，様々なセキュリティリスクが想定される．SIEM (Security information and event management) は，管理対象とする複数のデバイスで起こっているセキュリティ事象を一括して保管・分析して，攻撃の検知や将来の対策に役立つ情報を運用担当者に提示する仕組みであり，運用監視にとって重要なツールである．SIEM は，デバイスに対して想定される脅威のうち，ログ等のイベント情報を見て検知可能な脅威をユーザに対していち早く効率よく通知することで，被害の拡大を防ぐために使うことができる．しかし，IoT デバイスは一般的 IT 機器とは大きく異なる特性を持っており，SIEM が理解可能なログを出力してくれない・既知の攻撃検出ルールが対象としていないサービスを提供している・脅威モデルが大きく異なるという特徴を持つため，既存の SIEM システムをそのまま IoT デバイスに対して適用しても，把握しておきたいセキュリティインシデントを検出できないおそれが大きい．そこで本論文では，IoT デバイスの持つ特性を整理し，具体的な IoT デバイスに対する具体的な攻撃例に基づいて脅威シナリオを列挙し，その検出に必要な SIEM システム側の要件を検討する．

Takuya Mishina[1]　Koichi Ono[1]　Seiji Munetoh[1]　Naoto Satoh[1]　Yoichi Hatsutori[1]
Fumiko Satoh[1]

1. はじめに

近年，インターネットに接続される機器の種類が大幅に増えている．その種類は計算機とその周辺機器（プリンタなど）といった一般的 IT 機器から，携帯電話・スマートフォンへと幅を拡げ，更にあらゆるモノがインターネットに接続される Internet of Things (IoT) の時代を迎えつつある．それらのデバイスに対しては，IoT 時代以前からある脅威に加えて，ネットワークという新しい Attack Surface (攻撃対象領域) を経由する脅威が生じている．IoT デバイスを活用したサービスを継続的に提供するためには，セキュリティインシデントが発生しないように事前準備を行った上で，万が一のセキュリティインシデント発生時には，それを早期に発見し，迅速かつ的確なアクションを取る必要がある．

ネットワーク接続機器にセキュリティインシデントが発生していないかを 24 時間・365 日体制で監視するセキュリティ・オペレーション・センター（SoC）における重要なツールのひとつが SIEM (Security Information and Event Management) システムである．SIEM システムは，多種類のデバイスからやってくる大量のセキュリティイベントを分析して，対処が必要なセキュリティインシデントの発生を通知する [1]．SIEM システムの初回配備にあたっては，セキュリティインシデントの発生を検出するために，事前に以下のような処理を行って，攻撃検出ルールとログ解析ルールを準備しておく．

- SIEM の対象となる，セキュリティインシデントの発生が懸念される機器（攻撃対象機器）のリストを作る
- 攻撃対象機器ごとに，以下の作業を繰り返す
  - 攻撃対象機器に対する脅威モデルを作り，脅威シナリオのリストを作る
  - 脅威シナリオごとに，以下の作業を繰り返す
    - 脅威シナリオが実際に起こった場合に発生するイベントのシーケンスを定義する (攻撃検出ルール)
    - 多種・複数のデバイスから到着するログから，攻撃検出ルールを構成するのに必要なイベントを抽出するログ解析（集約・正規化）ルールを定義する（ログ解析ルール）

製品としての SIEM システムは，上記の作業を多数の機

[1] 日本アイ・ビー・エム（株）東京基礎研究所
IBM Research Tokyo, Chuo, Tokyo 103–8510, Japan

器・多数の脅威シナリオを対象として行って作成した攻撃検出ルール・ログ解析ルールと，セキュリティインシデント発生をオペレータに通知するためのダッシュボードを備え，これに加えて利用者が攻撃検出ルール・ログ解析ルールをカスタマイズする機能や，分析に必要な情報を外部から追加で取得する機能，分析結果を別のシステムに通知する機能を備える．

ある特定の IoT デバイスを SIEM システムの管理対象としたい場合，最も簡便な方法は，単にその IoT デバイスのログ出力先を SIEM システムにリダイレクトして，既存機器向けのルールをそのまま使う方法である．例えばある IoT デバイスが ssh プロトコルでリモートログイン機能を提供していて，ログの形式が一般的なもの（例えば `openssh` が Linux 上の `syslog` に記録するのと同じフォーマット）でログを記録していた場合，ssh ポートに対して総当たり攻撃を仕掛けられたことは，一般的な Linux 向けのルールを使って検出することが可能である．しかし，この方法では，以下のような場合に必要なセキュリティインシデントの検出を行うことができない．

- ログが独自形式であって，既知のログ解析ルールでは「ssh のログイン試行が発生した」というイベントや，そのイベントで同一のパスワードが繰り返し使われたというプロパティを SIEM システムがログから理解できない場合
- 既知の攻撃検出ルールが対象としていないサービスを検出対象としたい場合
- 脅威モデルが大きく異なり，あるセキュリティインシデントが発生するイベント・プロパティのパターンが既知の機器とは異なる場合 [*1]．

次節で示すように，IoT デバイスは一般的 IT 機器とは大きく異なる特性を持っているため，上記 3 点の全てに当てはまる可能性が高く，よって既存の SIEM システムをそのまま IoT デバイスに対して適用しても，把握しておきたいセキュリティインシデントを検出できないおそれが大きい．そこで本論文では，IoT デバイスの持つ特性を整理し，具体的な IoT デバイスを例に挙げて現実的な脅威シナリオを列挙し，その検出に必要な SIEM システム側の要件を検討する．以下ではまず IoT デバイスそのものの特性と，セキュリティの視点から見た特性を整理する（2 節）．次に IoT デバイス向け SIEM システムの設計を行うための例題となるシステムを決定して，IoT デバイスに対する具体的な攻撃例を列挙し，それらの具体例を元に脅威シナリオおよびその脅威を検出するためのルールを検討する（3 節）．関連研究を 4 節に示し，まとめと今後の課題を 5 節にて述べる．

[*1] 例えば DoS 攻撃を想定した脅威モデルを対象としたパターンを記述する場合，IoT デバイスの性能に応じて，どの程度の頻度に至った場合に警告を発するべきなのかを調整する必要がある．

表 1 文献 [4] による IoT デバイスの分類例
Table 1 Classification of IoT Devices [4]

クラス	機器特性
Class 1	超小型，8/16 bit CPU, real time OS
Class 2	32bit CPU, real time OS
Class 3	32bit CPU, Linux
Class 4	32/64bit CPU, Linux, IoT ゲートウェイ

2. IoT デバイスの特性

本節では，IoT デバイスの一般的特性及びセキュリティ上の特性について整理する．

2.1 IoT デバイスの一般的特性

ここではまず IoT デバイスの特性を，文献 [2], [3] を元に整理した形で列挙する．本論文では IoT デバイスの特性を，「機器数」「低コスト」「リソース制約」「管理困難な環境」「機微情報」の 5 つに分類する．

機器数

ひとつのサービスを構成する IoT デバイスは非常に多数となるため，SIEM を含む，機器を管理する側にスケーラビリティが要求される．また，膨大な数の機器を常に最新かつ想定通りの状況に維持するためには，デバイス管理を正確に行い，必要となった場合（例えば内部のソフトウェアに脆弱性が発見された場合），可能な限りすみやかに機器の更新作業を実施する必要がある．

低コスト

IoT デバイスは，一度に多数の機器がまとめて用いられることから単価を安くせざるを得ない．これは設計・実装にかけられるコストも低くなることを意味し，その帰結としてセキュリティに関する設計・実装・テストのクオリティにも影響を与える．

リソース制約

組み込み CPU を使っている，メモリが少ない，バッテリ駆動で大電力を使えない，といった理由で，機器本来の機能（温度測定・画像記録など）を行うことが精一杯で，セキュリティ関連の追加処理を行う余力がない IoT デバイスが存在する．なお，このリソース制約はほかの 4 つの特性とは異なり，IoT デバイスの種類によっては問題とならないことがある．デバイスの計算能力に基づいてなされた分類例 [4] では IoT デバイスを表 tabreftbl:iot-device-classes のように分類しており，Class 3 や Class 4 の機器においては，一般的 IT 機器では問題なく実行可能なセキュリティ処理（通信路暗号化や，それに必要な対称鍵・公開鍵暗号の処理）を，同様に問題なく実行可能なレベルの計算リソースを備えている．

管理困難な環境

IoT デバイスは，変化しやすく，監視の難しい環境に設

置されることが多い．設置位置や温度は変化し，また利用者も変化する．また，第三者が物理的にアクセスしやすい位置に設置されている場合が多く（例えば公共施設内に設置された監視カメラ，誰でも入れる駐車場に置かれた自動車），物理的アクセスがしやすいことも大きな特徴である．

機微情報

ウェアラブルデバイスであれば体温・位置など，監視カメラであれば顔情報といったように，IoT デバイスの多くがプライバシーに関わる情報を記録しており，これを適切に保護する必要がある．

2.2 IoT デバイスのセキュリティ特性

ここでは，2.1 節で整理した IoT デバイスの特性から得られる，IoT デバイスの持つセキュリティ特性を列挙する．まず，機器数と管理困難環境下という特性を考慮すると，膨大な数の機器が遠隔地に配備されている状態でセキュリティアップデートを適時に行うには，デバイス管理を適切に実施し，各デバイスの状態を把握して迅速に対象を絞ってセキュリティアップデートを行う必要がある．また，機器数・リソース制約を克服すべく開発された，IoT 向けに特化したプロトコルと，物理的アクセスの容易さが，セキュリティ上特性に影響を与える．IoT 向けのプロトコルとしては，低消費電力が特徴の ZigBee や，同じく低消費電力な MATT，簡易版 HTTP である CoAP などの標準的プロトコルに加えて，文献 [5], [6] にあるような独自プロトコルが用いられる．これらのプロトコルは一般的 IT 機器ではあまり使われないので，未知の脆弱性が存在する可能性が高い．物理的アクセスに関しては，IoT デバイスは誰でも入れる場所に設置されることも多いので，容易に物理的アクセスを行うことができ，デバイスの盗難や通信路の改ざん（タッピング）といった攻撃を受けやすいと言える．更に IoT デバイスはそれ単体では動作せず，取ってきた情報を別のサービスに転送する形で連携することが多いので，attack surface（攻撃対象領域）が広い．攻撃を検知するためには，それらのデバイス群・サービス群はまとめて管理する必要がある．

また，IoT デバイスにおいて検討すべきセキュリティ上の項目が文献 [7] に整理されている．

- Protocol and network security: scalable な暗号など，IoT の特性に応じたプロトコルを採用する
- Data and privacy: “privacy by design”, 誰がどのデータを管理しているのかを明確化する
- Identity management: 複数の identity，親子関係，匿名，筆名を持っている可能性を考慮する
- Trust and governance: 信頼できる通信相手を把握する
- Fault tolerance: secure by default（なるべく後でパッチを当てなくてもよいようにする）, 周りの状況を伝達して各デバイスが状況に適応する（例: 状況がおかしくなったらサービスレベルを落としてでもサービスを継続する）
- Information propagation: 1 対 1 では trust していても，trust chain によって，その情報の元の owner が承知していない相手に情報が伝播する可能性を考慮する

3. 脅威シナリオと検出ルール

本節では，具体的な攻撃事例を元に検討対象とする IoT デバイスを選定し，セキュリティ上の課題を列挙した上で,，各種のハッキングイベント・学術会議において IoT デバイスを実際に攻撃した事例に基づくシナリオを列挙し，続いて 2.2 節にて検討した，IoT デバイス独自の特性を狙った脅威を列挙する．

3.1 対象システム

脅威シナリオ作成に先立ち，その対象となるシステムとしてどのような IoT デバイスを仮定するのかを検討する．SIEM はセキュリティインシデントの予兆や実際の攻撃を検出するものであり，行うべき対策を打った上で更に不審なイベントが発生していないかを検出するツールである．次節（3.2 節）で取り上げる攻撃例の多くは，IoT デバイスの特徴のひとつである低コストに起因する．このようなデバイスの場合，確かに様々な脅威シナリオを考えることができるが，そのときの対応作としては，SIEM システムを使った事後検知ではなく，そもそもの設計を修正して，そのような攻撃が実現しないようにする対応が適切であると考える．そのため，そこで本論文ではそのようなデバイスについては検討の対象から外し，最低限必要な防御策については実施済みのデバイス，具体的にはネットワーク接続型監視カメラを対象とする．監視カメラの中には組み込み Linux を搭載しているものがあり [*2]，ネットワーク機能としては標準的な機能が提供されていると考えられる．また，監視カメラは現状でも既に興味目的でアタックが試みられており [*3]，今後テロ対策・東京オリンピック対策として増設が進むと予測される（例えば複数の鉄道会社 [*4] が監視カメラの増設を表明している）．インターネットに接続された監視カメラが増えることで，それらのデバイスが悪意ある利用者から攻撃を受ける可能性も増すと考えられることから本論文では，**図 1** に示すような，監視カメラとその付随システムを対象として考える（図中の S1 から S6 までの矢印は，後続の節で検討する脅威シナリオの番号とそ

[*2] 例えば http://www.kansi.jpn.org/cam_ip/index.htm では Linux を組み込んだ監視カメラを，http://armadillo.atmark-techno.com/armadillo-810 では組み込み Linux 搭載の監視カメラ用モジュールが販売されている．

[*3] http://www.insecam.org/ では安易なパスワードが設定されている監視カメラの映像を参照することができる

[*4] http://jr-central.co.jp/news/release/_pdf/000027351.pdf, http://www.tokyu.co.jp/company/news/list/?id=2401

の攻撃先を示している）．このシステムは撮影した映像をデータセンターに転送し，ユーザがスマートフォン上のアプリからその映像を確認できるシステムである．ユーザはまた，スマートフォンアプリを経由して，カメラ自体の設定を変更できるものと仮定する．

3.2 脅威の実例

脅威シナリオを検討するにあたり，2014 年以降に行われた IoT デバイスに対する具体的な攻撃事例について確認する．著名なハッカーイベントのひとつである Black Hat USA の 2014 年および 2015 年の発表において，IoT デバイスを標的として行われた攻撃の事例を**表 2** に示す．ICS/SCADA，自動車，RFID/Smart Card は両年いずれでも攻撃対象とされているほか，毎年多様なデバイスが攻撃対象となっている．なお，比較対象として，Android に対する攻撃事例はそれぞれ 5 件（2014 年）および 8 件（2015 年）となっている．攻撃の方法としては，プロトコルもしくは API の脆弱性を突いているものが多い．2.2 節で述べた通り，IoT デバイスで使われるプロトコルは，計算量・通信量を節約するため，一般的 IT 機器ではあまり使われないプロトコルや，独自プロトコルが利用される場合が多く，これまでに知られていない脆弱性を突くことが比較的容易であると考えられる．また，API の脆弱性を突く攻撃も多い．

2.1 節で述べたとおり，IoT デバイスの設計・製造にはコストをかけられないため，設計上の問題を突かれる事例も多く見られる．例えば車のコントロールを奪取できた事例では，認証キーとして使われる文字列の長さが 6 文字しかなかったため，総当たり攻撃による侵入を許している．また，別の文献 [8] では Smart TV への攻撃事例が挙げられている．対象の Smart TV では，ユーザが任意のアプリケーションを追加することができるが，オペレーティングシステムとアプリケーション間の隔離が不十分で，あるアプリケーションが平文で送っているパスワードを別のアプリケーションから盗聴したり，オペレーティングシステムが持つ証明書がアプリケーションから見えてしまう（それどころかファイルシステムが全て見てしまう）状態になっていた．また，監視カメラの事例 [9] では，遠隔管理用に telnet プロトコルが使われていたため，telnet プロトコルの telnet の脆弱性を突いて乗っ取りが行われた．別の監視カメラの事例 [10] では，監視カメラからカメラの映像を送信するためのプロトコルが TLS ではなかったため，中間者攻撃を実行してカメラの物理的位置を詐称することが可能であると報告されている．

3.3 実例に基づく脅威シナリオ

S1: 偽のファームウェアアップデート

監視カメラがファームウェアアップデート機能を持ち，それが管理用 API を経由して外部からアクセス可能であったと仮定する．攻撃者はまず管理用 API の脆弱性を探す．次に発見した脆弱性を使って設定データ中のファームウェアアップデートのデータ取得先を自らが管理する偽のファームウェアの URL に書き換え，ファームウェア更新を実行する．これにより，デバイスの特権を奪取して攻撃者の意図通りに遠隔操作できるようにする．この手順のうち，ファームウェアの改ざんに関して実例が存在する [*5]．検出ルールとしては，アップデートのデータ取得先の変更を含む重要な設定変更情報をログとして記録しておき，不審な取得先がセットされたときに警告を発することが考えられる．

S2: スマホアプリから証明書を盗んで中間者攻撃

モバイルアプリはリバースエンジニアリングが容易であることを利用して，IoT デバイスへ接続可能な状態となっているモバイルアプリをリバースエンジニアリングして各種 API へのログイン用証明書を盗み，中間者攻撃に悪用する攻撃があり得る [*6]．IoT デバイスは，それ単体では動作せず，取ってきた情報を別のサービスに転送したり，ユーザインタフェースのリッチなデバイスでデータを参照可能にするなど，多種類のサービス・組み合わさって動作するため，スマホも含めて攻撃対象となる間口（attack surface）が広い．その中からもっとも脆弱なデバイスを探して攻撃することで，IoT デバイスを含むシステムを攻撃することができる．

証明書を盗んで別のサービスに不正ログインを試みる攻撃の場合，IoT デバイス側で証明書を盗まれたことを検出するのは困難であり，不正ログインされた側のサービスのログで，ログインが不自然なものかどうかを判定するルールを作っておくことが考えられる．例えば，あるユーザの証明書が短時間に全く異なる 2 つの国からアクセスされた場合に，不正ログインの恐れありと警報を発するルールが考えられる．

S3: タッピング攻撃

IoT デバイスで収集されたデータがどの時点で暗号化されるのかは千差万別である．インターネット接続前であれば平文転送してもよい，という考え方で通信路を設計した場合，物理的アクセスが容易であるという IoT デバイスの特徴から，機器近くに露出しているネットワークケーブルをタッピングし，通信路上のデータを改ざんする攻撃があり得る [*7]．このような途中経路上の改ざんは，SIEM での検出が困難である．デバイス側とデータ受け取り側とでデータのインテグリティをチェックすれば検出可能である

[*5] Remote Exploitation of an Unaltered Passenger Vehicle (Black Hat USA 2015)
[*6] Drive It Like You Hacked It: New Attacks and Tools to Wirelessly Steal Cars (Black Hat USA 2015)
[*7] Looping Surveillance Cameras through Live Editing of Network Streams (DEFCON 23)

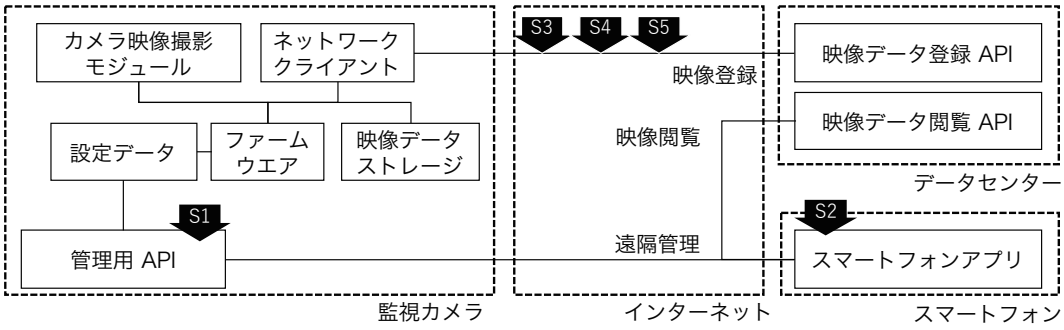

図 1 監視カメラサービスのコンポーネント図

表 2 Blach Hat USA における IoT デバイスの攻撃事例
Table 2 IoT Device Attack Examples in Blach Hat USA

発表年	攻撃対象
2014 年	ICS/SCADA (2 件), POS 端末 (2 件), RFID/スマートカード, 自動車, 家電, MDM ソフトウェア, キーボックス
2015 年	ICS/SCADA (4 件), ネットワークスイッチ (2 件), 自動車 (2 件), RFID/スマートカード, ZigBee プロトコル, ライフル

が，データの改ざんが行われている時点でログも改ざん可能になっており，インテグリティチェックに使う値が正しく得られる保証がない．このように，SIEM システムでは検出できない攻撃もあることに注意が必要である．

S4: 無線通信プロトコルの脆弱性攻撃

2.2 で述べたように，IoT デバイス独自の通信プロトコルは攻撃される可能性が高い．IEEE 802.11 の WEP プロトコルの脆弱性が広く知られているが，ZigBee にも攻撃事例が存在する [*8]．

3.4 IoT デバイスならではの脅威シナリオ

S5: ハードウェアを盗んで秘密情報を盗む

まず IoT デバイスを 1 台物理的に盗み，デバイス内のデータを解析して，価値のある情報（個人情報）や設定情報（データセンター接続用認証情報）などを盗み出す．デバイス管理によって認証情報を適切に管理できている場合は，IoT デバイスと連携するサービスの側のログに現れる認証情報から，攻撃の起点となったデバイスを特定することは可能である．

S6: 失効した証明書を使った中間者攻撃

SSL/TLS の失効証明書を更新できなかったり，できても大量のデバイスに対して失効証明書の配付が間に合わない場合，DNS spoofing 等で接続先を悪意ある URL にリダイレクトすることできれば，不正に入手した（しかし本来は失効しているはずの）証明書を提示してデバイスをだまし，中間者攻撃によってデータの漏洩・改ざんが可能になる．シナリオ S3 と同様に，この攻撃も SIEM システムで検出するのは困難である．

[*8] ZigBee Exploited the Good the Bad and the Ugly (Black Hat USA 2015)

4. 関連研究

セキュリティを含めた IoT 全般の考慮事項については文献 [11] が詳しい．また，IoT セキュリティの一般的な survey については．本文中で触れた [7] のほかにも，全般的なサーベイとしては [3] が，IoT-A[12] など既存の IoT Architecture の評価がついている文献 [2] が挙げられる．本論文では SIEM システムとの連携に着目して脅威シナリオを検討したが，[13] では，IoT デバイス全般に対する脅威のユースケースおよび脅威モデルについて検討されている．SIEM に関する概説としては，文献 [1] が挙げられる．文献 [14] では agent-based/agent-less/order-based の 3 種の SIEM システムの比較を行っている．一般的 IT 機器に対して SIEM システムを適用するための脅威シナリオについては，網羅的な分析例 [15] や，実際に特定の脅威を想定した上で SIEM システムを実装した実例 [16] が参考になる．大量の IoT デバイスから送られる大量のログを処理するにあたっては，IoT データをデバイスに近い場所で事前処理するしくみ [17] の利用も検討できる．IoT デバイス向けプロトコルに関して，センサーネットワーク向け近距離無線通信プロトコル [5]，MQTT で OAuth を行う実装 [18] といった独自プロトコルや，一般的な機器・オペレーティングシステム向けの機能（例えばアクセス制御を組み込み OS 向けに実装した例 [19]，簡易版 HTTP である CoAP で認証を行う手法 [20] など）を IoT 向けのコンポーネントへ移植する際は，脅威分析を行った上で，文献 [21] の手法等を用いてセキュリティ要件を明確にすることが求められる．脅威モデルに関する一般的な知識に関しては文献 [22] に詳しい．今回の脅威モデル分析では悪意ある第三者を想定したが，IoT デバイスは多くのユーザが利用するため，その中には悪意ある内部攻撃者が存在する可能性もあり得

る．文献 [23] ではそのような状況における脅威を分類・列挙している．

5. まとめ

本論文では，実際の攻撃例に基づいて，IoT デバイスに対する脅威を検討し，SIEM システムでそれらの脅威を検出するための要件について検討を行った．今後はこの脅威に基づき，具体的なログを用いてルールが正しく脅威を検出できることを示したい．また，SIEM システムが IoT デバイスの特徴のひとつであるスケーラビリティへの対応を確認するため，大量の IoT デバイスが接続された場合にも，適切な量の警報をユーザに示すことができることを検証したい．

参考文献

[1] Bhatt, S., Manadhata, P. K. and Zomlot, L.: The Operational Role of Security Information and Event Management Systems, *IEEE Security Privacy*, Vol. 12, No. 5, pp. 35–41 (online), DOI: 10.1109/MSP.2014.103 (2014).

[2] Vasilomanolakis, E., Daubert, J., Luthra, M., Gazis, V., Wiesmaier, A. and Kikiras, P.: On the security and privacy of Internet of Things architectures and systems, *SIoT '15: Proceeding of International Workshop on Secure Internet of Things* (2015).

[3] Atzori, L., Iera, A. and Morabito, G.: The Internet of Things: A survey, *Computer Networks*, Vol. 54, No. 15, pp. 2787 – 2805 (online), DOI: http://dx.doi.org/10.1016/j.comnet.2010.05.010 (2010).

[4] Grau, A. and Kurisu, W.: IoT Security for Gateways and Edge Devices. `https://www.mentor.com/embedded-software/multimedia/iot-security-for-gateways-and-edge-devices`.

[5] Rosner, D., Tataroiu, R., Gheorghe, L. and Tilimpea, R.: UNCHAIN - Ubiquitous Wireless Network Communication Architecture for Ambient Intelligence and Health Scenarios, *SIoT '14: Proceeding of International Workshop on Secure Internet of Things*, pp. 44–51 (online), DOI: 10.1109/SIoT.2014.12 (2014).

[6] Zenger, C. T., Chur, M. J., Posielek, J. F., Paar, C. and Wunder, G.: A Novel Key Generating Architecture for Wireless Low-Resource Devices, *SIoT '14: Proceeding of International Workshop on Secure Internet of Things*, pp. 26–34 (online), DOI: 10.1109/SIoT.2014.7 (2014).

[7] Roman, R., Najera, P. and Lopez, J.: Securing the Internet of Things, *Computer*, Vol. 44, No. 9, pp. 51–58 (online), DOI: 10.1109/MC.2011.291 (2011).

[8] Niemietz, M., Somorovsky, J., Mainka, C. and Schwenk, J.: Not so Smart: On Smart TV Apps, *SIoT '15: Proceeding of International Workshop on Secure Internet of Things* (2015).

[9] Cluley, G.: Hacked Shopping Mall CCTV Cameras Are Launching DDoS Attacks. `http://www.tripwire.com/state-of-security/security-data-protection/hacked-shopping-mall-cctv-cameras-are-launching-ddos-attacks/`.

[10] Donohue, B.: Urban surveillance camera systems lacking security. `https://blog.kaspersky.com/urban-surveillance-not-secure/8901/`.

[11] Xu, L. D., He, W. and Li, S.: Internet of Things in Industries: A Survey, *IEEE Transactions on Industrial Informatics*, Vol. 10, No. 4, pp. 2233–2243 (online), DOI: 10.1109/TII.2014.2300753 (2014).

[12] IoT-A Consortium: IoT-A - Internet of Things Architecture. `http://www.iot-a.eu/`.

[13] Atamli, A. W. and Martin, A.: Threat-Based Security Analysis for the Internet of Things, pp. 35–43 (online), DOI: 10.1109/SIoT.2014.10 (2014).

[14] Kufel, L.: Security Event Monitoring in a Distributed Systems Environment, *IEEE Security Privacy*, Vol. 11, No. 1, pp. 36–43 (online), DOI: 10.1109/MSP.2012.61 (2013).

[15] 榊原裕之，居城秀明，河内清人：ログ分析によるサイバー攻撃検知の効果について，研究報告マルチメディア通信と分散処理（DPS），Vol. 2016-DPS-166, No. 9, pp. 1–8 (2016).

[16] 大谷尚通，北野美紗，重田真義：企業内ネットワークの通信ログを用いたサイバー攻撃検知システム，CSS '13: コンピュータセキュリティシンポジウム (2013).

[17] 中村優吾，諏訪博彦，荒川　豊，山口弘純，安本慶一：多様な IoT データストリームをクラウドレスで分散処理するミドルウェアの設計，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），Vol. 2015-MBL-77, No. 22, pp. 1–8 (2015).

[18] Fremantle, P., Aziz, B., Kopecky, J. and Scott, P.: Federated Identity and Access Management for the Internet of Things, *SIoT '14: Proceeding of International Workshop on Secure Internet of Things* (2014).

[19] Mituca, A., Moin, A. H. and Prehofer, C.: Access Control for Apps Running on Constrained Devices in the Internet of Things, *SIoT '14: Proceeding of International Workshop on Secure Internet of Things* (2014).

[20] Nguyen, H. V. and Iacono, L. L.: REST-ful CoAP Message Authentication, *SIoT '15: Proceeding of International Workshop on Secure Internet of Things*, pp. 35–43 (online), DOI: 10.1109/SIOT.2015.8 (2015).

[21] 金子朋子，高橋雄志，勅使河原可海，田中英彦：CC-Case を用いた IoT セキュリティ認証方法の提案，研究報告コンピュータセキュリティ（CSEC），Vol. 2016-CSEC-72, No. 14, pp. 1–8 (2016).

[22] Shostack, A.: *Threat Modeling: Designing for Security*, Wiley (2014).

[23] Nurse, J. R. C., Erola, A., Agrafiotis, I., Goldsmith, M. and Creese, S.: Smart Insiders: Exploring the Threat from Insiders Using the Internet-of-Things, *SIoT '15: Proceeding of International Workshop on Secure Internet of Things*, pp. 5–14 (online), DOI: 10.1109/SIOT.2015.10 (2015).

[24] Kotenko, I., Shorov, A., Chechulin, A. and Novikova, E.: Dynamical Attack Simulation for Security Information and Event Management, *Information Fusion and Geographic Information Systems*, Springer Berlin Heidelberg, pp. 219–234 (2014).

[25] Yen, T.-F., Oprea, A., Onarlioglu, K., Leetham, T., Robertson, W., Juels, A. and Kirda, E.: Beehive: Large-scale Log Analysis for Detecting Suspicious Activity in Enterprise Networks, *ACSAC '13: Proceedings of the 29th Annual Computer Security Applications Conference*, pp. 199–208 (online), DOI: 10.1145/2523649.2523670 (2013).