Information-technology
Promotion
Agency, Japan

脆弱性対策の動向と
効果的な収集に向けて

独立行政法人 情報処理推進機構 (IPA)
セキュリティセンター

2019年03月28日

セミナー概要

脆弱性を悪用したセキュリティインシデントに対応するため、対策の基盤を実現する技術仕様も、いろいろな取組みが進められています。本セミナーでは、脆弱性対策の動向として、脆弱性の動向、米国政府の取り組み、IPAの取り組みを紹介するとともに、脆弱性対策基盤の実現する技術仕様を紹介します。

1. 情報収集に役立つ技術仕様
2. 自動化(機械化)処理基盤の潮流
3. JVN脆弱性対策機械処理基盤

情報収集に役立つ技術仕様

CVE
脆弱性に番号を割当て、一意に識別することを可能にする

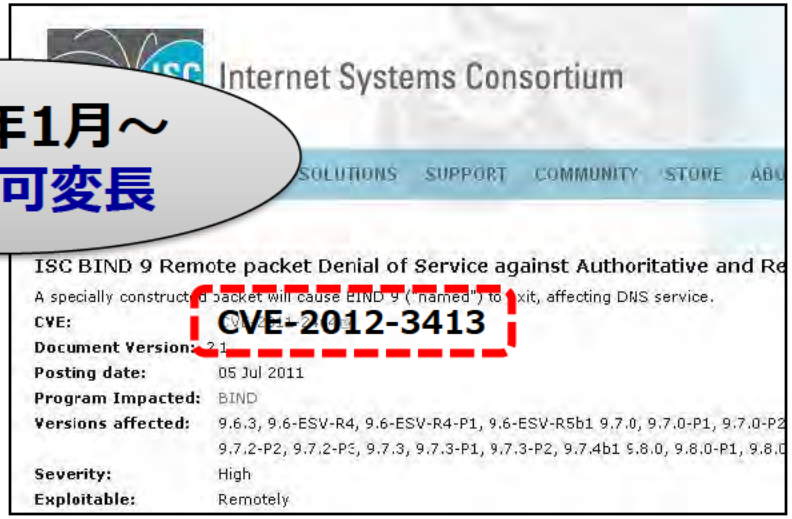
- Common Vulnerabilities and Exposures
  (共通脆弱性識別子)
- プログラム上のセキュリティ問題に一意の番号(CVE識別番号)を付与して管理

CVE識別番号の構成

西暦	連番

CVE-2014-1000
CVE-2014-10000
CVE-2014-100000
CVE-2014-1000000

2014年1月～
連番は可変長

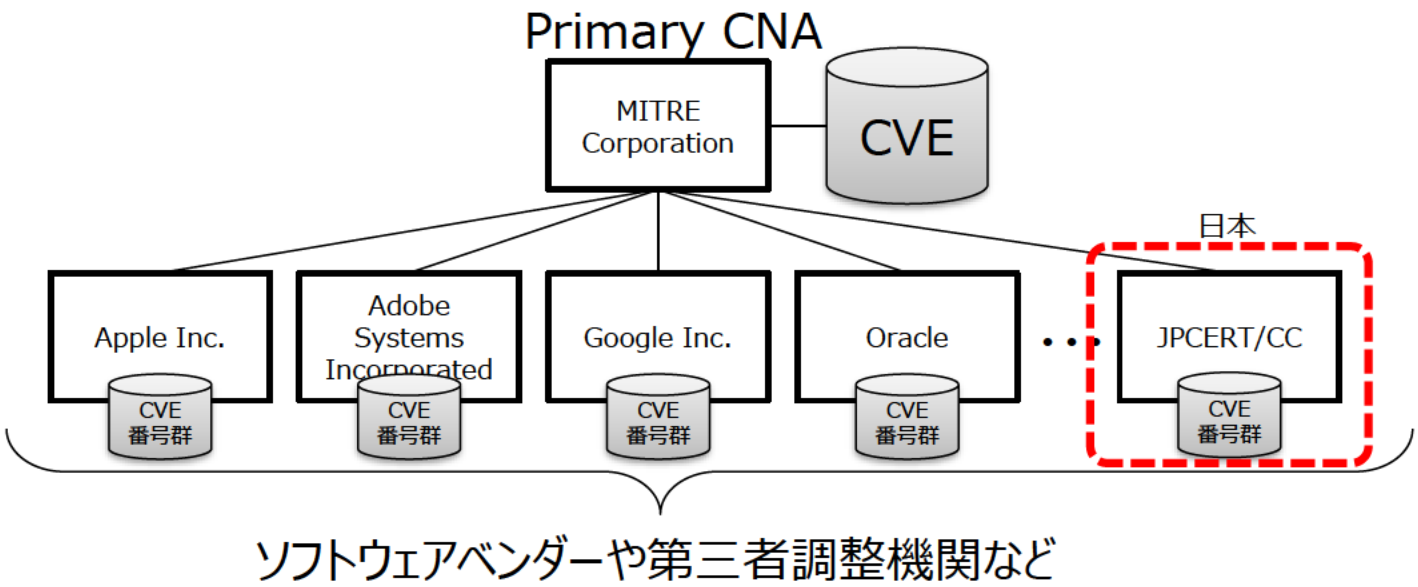


公表されている脆弱性に割り当てられた識別番号で、脆弱性を一意に特定することを可能となる

[URL] 共通脆弱性識別子CVE概説
https://www.ipa.go.jp/security/vuln/CVE.html
CVE - Common Vulnerabilities and Exposures (CVE)
https://cve.mitre.org/

CVE
採番方法の仕組み

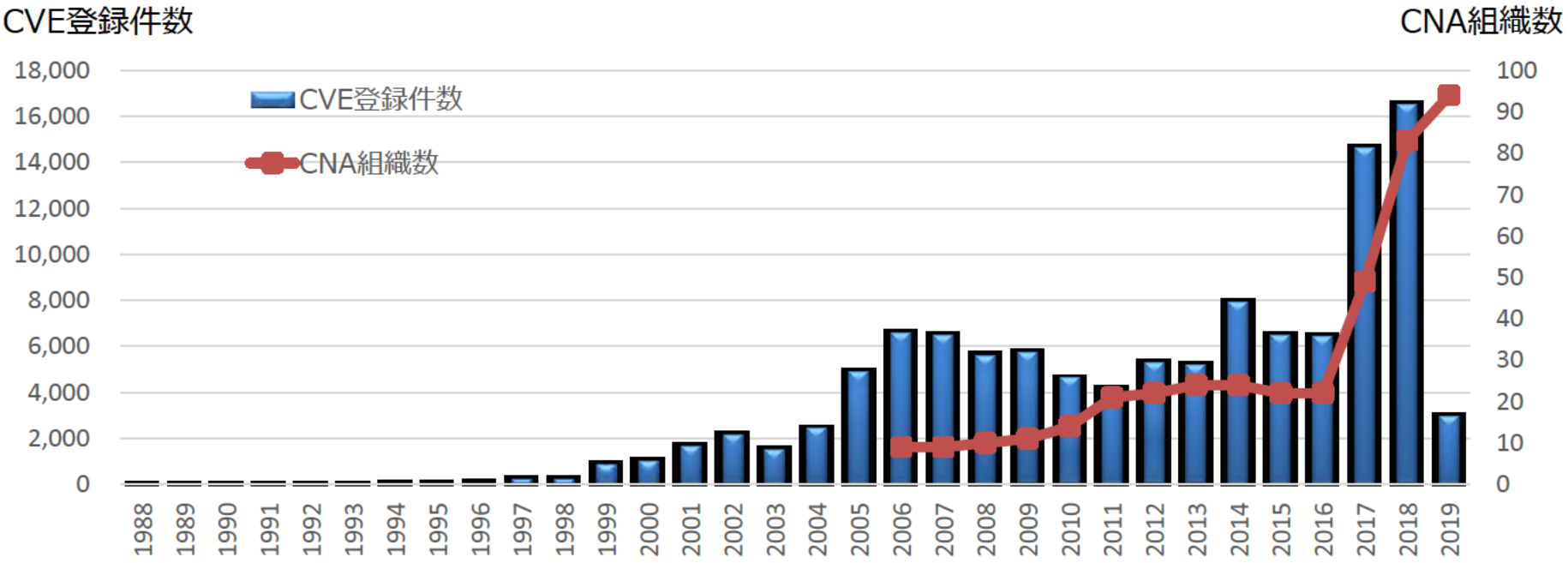
- 新規脆弱性毎に米国MITREに申請、または
- 米国MITREに認定されたCNA(CVE Numbering Authority、CVE 採番機関)から割り当て



[URL] Submit a CVE Request
https://cveform.mitre.org/
CVE Numbering Authorities
https://cve.mitre.org/cve/cna.html

CVE
CVE登録件数の推移

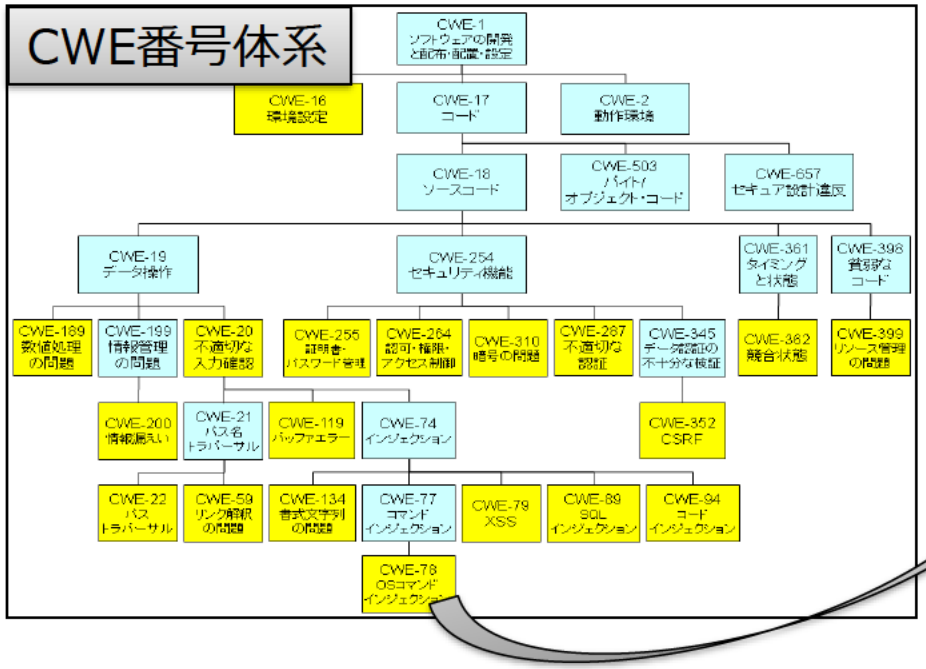
- 米国NVD(National Vulnerability Database)に登録されている脆弱性情報(=CVE登録件数)は2017年から急増



[URL] Search Vulnerability Database
https://nvd.nist.gov/vuln/search

CWE
脆弱性のタイプを体系立てて分類した指標

- Common Weakness Enumeration
  (共通脆弱性タイプ一覧)
- 脆弱性を種別毎に分類



[URL] 共通脆弱性タイプ一覧CWE概説
https://www.ipa.go.jp/security/vuln/CWE.html
CWE - Common Weakness Enumeration
https://cwe.mitre.org/

CWE
脆弱性のタイプを体系立てて分類した指標

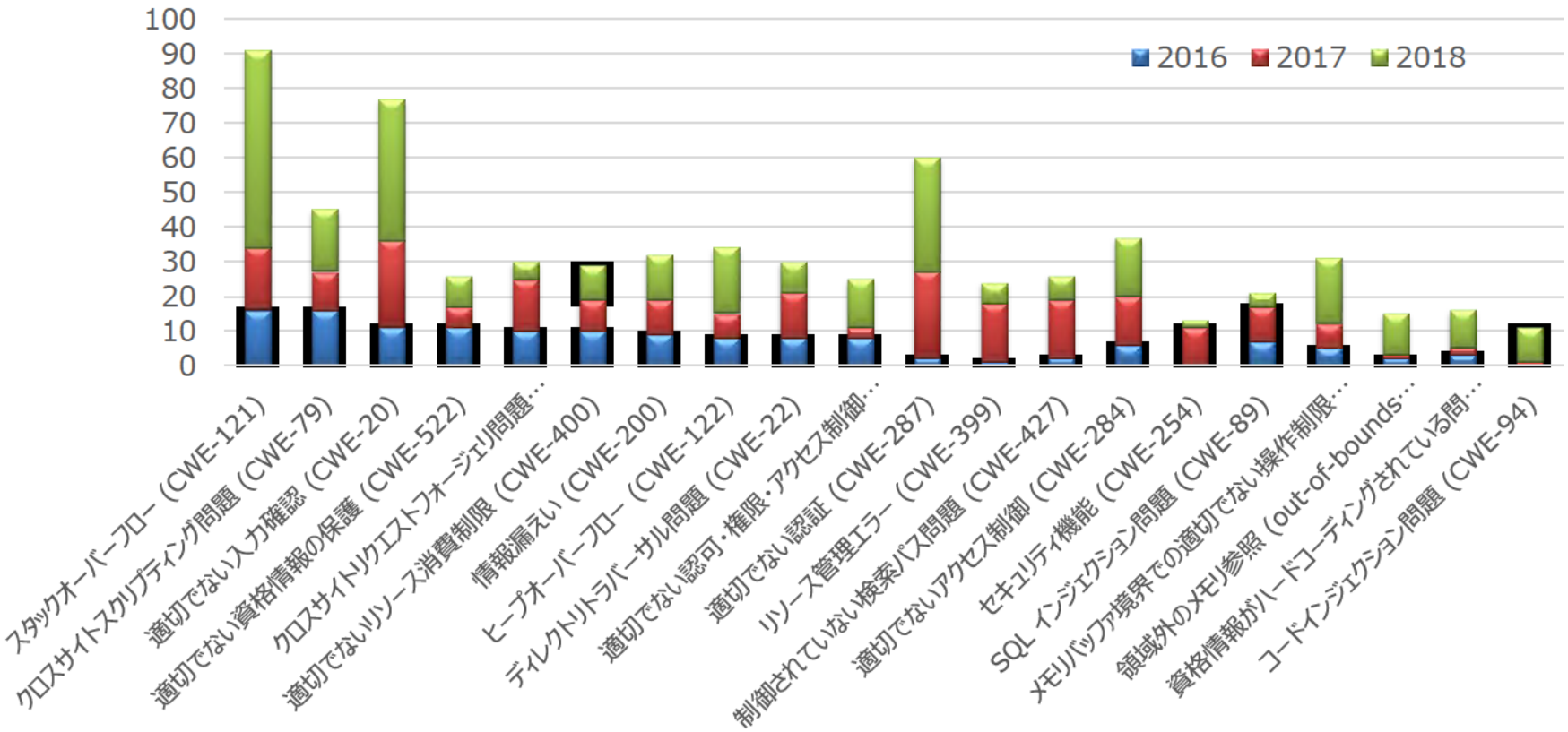
- 主な脆弱性タイプとCWE識別子番号

ID	概要
CWE-16	環境設定
CWE-20	不適切な入力確認
CWE-22	パス・トラバーサル
CWE-59	リンク解釈の問題
CWE-78	OSコマンドインジェクション
CWE-79	クロスサイトスクリプティング
CWE-89	SQLインジェクション
CWE-94	コード・インジェクション
CWE-119	バッファエラー
CWE-134	書式文字列の問題

ID	概要
CWE-189	数値処理の問題
CWE-200	情報漏えい
CWE-255	証明書・パスワードの管理
CWE-264	認可・権限・アクセス制御
CWE-287	不適切な認証
CWE-310	暗号の問題
CWE-352	クロスサイトリクエストフォージェリ
CWE-362	競合状態
CWE-399	リソース管理の問題

CWE
脆弱性のタイプを体系立てて分類した指標

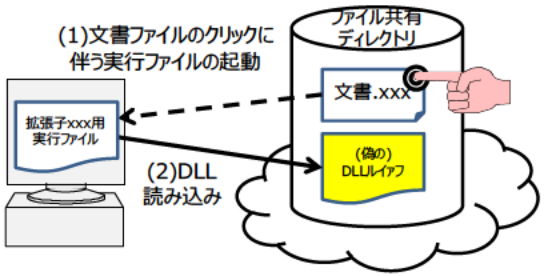
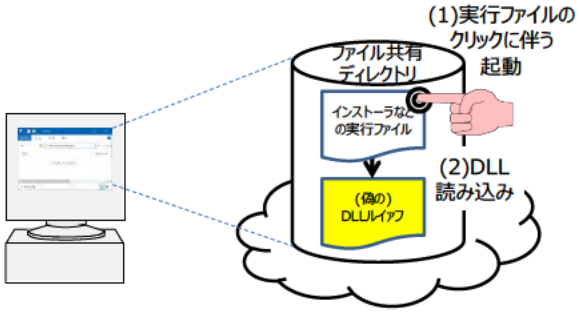
- ICS-CERT発行のアドバイザリでのCWE出現数



[URL] ICS-CERT
https://ics-cert.us-cert.gov/

CWE
脆弱性のタイプを体系立てて分類した指標

- ICS-CERT発行のアドバイザリでのCWE出現数
- 制御されていない検索パス問題 (CWE-427) … DLL読み込み問題
  アプリケーションが本来読み込むべきではないDLL/EXEファイルを読み込んでしまう状況が発生することに起因する。

分類	概要	
2010年に報告されたDLL問題・・・カレントディレクトリ型	データファイルと同一のディレクトリに置かれた(偽の)DLLファイルを、システムディレクトリにある(正規の)DLLファイルよりも優先して読み込んでしまう問題	
2016年以降に報告されたDLL問題・・・アプリケーションディレクトリ型	アプリケーション起動時にアプリケーションと同一のディレクトリに置かれた(偽の)DLLファイルを、システムディレクトリにある(正規の)DLLファイルよりも優先して読み込んでしまう問題	

CWE
脆弱性のタイプを体系立てて分類した指標

- ICS-CERT発行のアドバイザリでのCWE出現数
- 制御されていない検索パス問題 (CWE-427) … DLL読み込み問題
  アプリケーションが本来読み込むべきではないDLL/EXEファイルを読み込んでしまう状況が発生することに起因する。

分類	概要	対策
2010年に報告されたDLL問題・・・カレントディレクトリ型	データファイルと同一のディレクトリに置かれた(偽の)DLLファイルを、システムディレクトリにある(正規の)DLLファイルよりも優先して読み込んでしまう問題	カレントディレクトリに置かれた(偽の)DLLファイルを読み込まないように制御する。
2016年以降に報告されたDLL問題・・・アプリケーションディレクトリ型	アプリケーション起動時にアプリケーションと同一のディレクトリに置かれた(偽の)DLLファイルを、システムディレクトリにある(正規の)DLLファイルよりも優先して読み込んでしまう問題	アプリケーションと同一のディレクトリに置かれた(偽の)DLLファイルを読み込まないように制御する。ただし、アプリケーションだけで制御できない場合もあるため、ユーザ側での対応も必要となる。

CVSS
脆弱性の深刻度を数値で表現する

- Common Vulnerability Scoring System (共通脆弱性評価システム)
- 脆弱性の深刻度を0.0～10.0のスコアで評価

CVE#	Component	Protocol	Package and/or Privilege Required	Remote Exploit without Auth.?	CVSS VERSION 2.0 RISK (see Risk Matrix Definitions)							Supported Versions Affected	Notes
					Base Score	Access Vector	Access Complexity	Authen-tication	Confiden-tiality	Integrity	Avail-ability		
CVE-2016-0451	Oracle GoldenGate	Oracle Golden Gate	None	Yes	10.0	Network	Low	None	Complete	Complete	Complete	11.2, 12.1.2	See Note 1
CVE-2016-0452	Oracle	Oracle	None	Yes	10.0	Network	Low	None	Complete	Complete	Complete	11.2, 12.1.2	See

CVSS値

CVE番号

CVE#	Component	Package and/or Pr Required		Remote Exploit without Auth.?	CVSS VERSION 3.0 RISK (see Risk Matrix Definitions)									Supported Versions Affected	Notes
					Base Score	Attack Vector	Attack Complex	Privs Req'd	User Interact	Scope	Confid-entiality	Inte-grity	Avail-ability		
CVE-2019-2444		Logon	Local Logon	No	8.2	Local	Low	Low	Required	Changed	High	High	High	12.2.0.1, 18c	
CVE-2019-2406	Core RDBMS	Create Session, Execute Catalog Role	Oracle Net	No	7.2	Network	Low	High	None	Un-changed	High	High	High	12.1.0.2, 12.2.0.1, 18c	
CVE-2019-2547	Java VM	Create Session, Create Procedure	Multiple	No	3.5	Network	Low	Low	Required	Un-changed	None	None	Low	11.2.0.4, 12.1.0.2, 12.2.0.1, 18c	

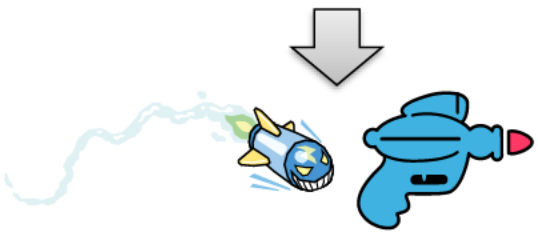
[URL] Critical Patch Updates, Security Alerts and Bulletins
https://www.oracle.com/technetwork/topics/security/alerts-086861.html

CVSS
攻撃状況やシステムの重要度を加味した評価

- 脆弱性の深刻度を表す評価

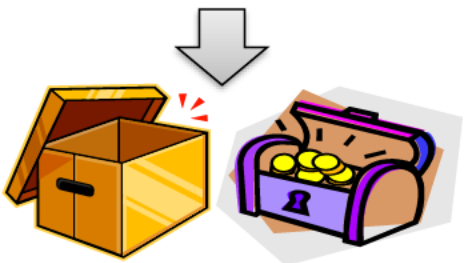
＝ 「技術的な特性」 × 「脅威の大きさ」 × 「情報資産の価値」

＝ 「基本評価基準」 × 「現状評価基準」 × 「環境評価基準」



何が引き起こされるのか?

既に攻撃されている?
対策パッチは出ている?



システムの重要度は?

「バッファオーバフロー」×「攻撃観測なし」×「内部システム」
＝深刻度 低

「クロスサイトスクリプティング」×「攻撃観測あり」×「外部システム」
＝深刻度 高

CVSS
CVSSv3の特徴

- コンポーネント単位で評価する手法の採用
  攻撃の難易度を評価する項目と、攻撃による影響を評価する項目を分けて評価する手法を採用。攻撃の難易度を評価する項目については、攻撃者がソフトウェアの脆弱性を悪用して攻撃できる対象(以降、コンポーネント)を範囲とする。
- 脆弱性の影響範囲拡大の加味
  脆弱性の影響範囲拡大を加味するため、基本評価基準にスコープという評価項目を導入。脆弱性の影響がコンポーネントに留まる場合、スコープ＝変更なし。影響がコンポーネント以外にも広がる場合、スコープ＝変更あり。
- 基本評価基準の細分化
  『必要な特権レベル』、『ユーザ関与レベル』を導入。
- 環境評価基準のアプローチの変更
  攻撃の難易度を評価する項目、攻撃による影響を評価する項目を実状に合わせて再評価するという評価手法を採用。

[URL] 共通脆弱性評価システムCVSS v3概説
https://www.ipa.go.jp/security/vuln/CVSSv3.html
Common Vulnerability Scoring System v3.0: Specification Document
https://www.first.org/cvss/specification-document

CVSS
CVSSv3の特徴

- 基本評価基準の変更点

CVSSv2	
攻撃の難易度	どこから攻撃可能であるか 攻撃元区分 (AV: Access Vector)
	攻撃する際に必要な条件の複雑さ 攻撃条件の複雑さ (AC: Access Complexity)
	攻撃するために認証が必要であるか 攻撃前認証要否 (Au: Authentication)
攻撃による影響	機密情報が漏えいする可能性 機密性への影響 (C: Confidentiality Impact)
	情報が改ざんされる可能性 完全性への影響 (I: Integrity Impact)
	業務が遅延・停止する可能性 可用性への影響 (A: Availability Impact)

CVSSv3	
攻撃の難易度	どこから攻撃可能であるか 攻撃元区分 (AV: Attack Vector)
	攻撃する際に必要な条件の複雑さ 攻撃条件の複雑さ (AC: Attack Complexity)
	攻撃する際に必要な特権レベル 必要な特権レベル (PR: Privileges Required)
	攻撃する際に必要なユーザ関与レベル ユーザ関与レベル (UI: User Interaction)
攻撃による影響	攻撃による影響の想定範囲 影響の想定範囲 (S: Scope)
	機密情報が漏えいする可能性 機密性への影響 (C: Confidentiality Impact)
	情報が改ざんされる可能性 完全性への影響 (I: Integrity Impact)
	業務が遅延・停止する可能性 可用性への影響 (A: Availability Impact)

CVSS
CVSSv3の特徴

- 現状評価基準の変更点

CVSSv2		CVSSv3
攻撃手法が実際に利用可能であるか 攻撃可能性 (E: Exploitability)	→	攻撃手法が実際に利用可能であるか 攻撃可能性 (E: Exploitability)
対策がどの程度利用可能であるか 利用可能な対策のレベル (RL: Remediation Level)	→	対策がどの程度利用可能であるか 利用可能な対策のレベル (RL: Remediation Level)
脆弱性情報の信頼性は 脆弱性情報の信頼性 (RC: Report Confidence)	→	脆弱性情報の信頼性は 脆弱性情報の信頼性 (RC: Report Confidence)

CVSS
CVSSv3の特徴

- 環境評価基準の変更点

CVSSv2
システムからの二次的被害の可能性 二次的被害 (CDP: Collateral Damage Potential)
システムの影響範囲 システム範囲 (TD: Target Distribution)
システムにおける機密性の重要度 機密性の要求度 (CR: Confidentiality Requirement)
システムにおける完全性の重要度 完全性の要求度 (IR: Integrity Requirement)
システムにおける可用性の重要度 可用性の要求度 (AR: Availability Impact)

CVSSv3	
緩和策後の再評価	どこから攻撃可能であるか 攻撃元区分 (MAV: Modified Attack Vector)
	攻撃する際に必要な条件の複雑さ 攻撃条件の複雑さ (MAC: Modified Attack Complexity)
	攻撃する際に必要な特権レベル 必要な特権レベル (MPR: Modified Privileges Required)
	攻撃する際に必要なユーザ関与レベル ユーザ関与レベル (MUI: Modified User Interaction)
	攻撃による影響の想定範囲 影響の想定範囲 (MS: Modified Scope)
	機密情報が漏えいする可能性 機密性への影響 (MC: Modified Confidentiality Impact)
	情報が改ざんされる可能性 完全性への影響 (MI: Modified Integrity Impact)
	業務が遅延・停止する可能性 可用性への影響 (MA: Modified Availability Impact)
システムにおける機密性の重要度 機密性の要求度 (CR: Confidentiality Requirement)	
システムにおける完全性の重要度 完全性の要求度 (IR: Integrity Requirement)	
システムにおける可用性の重要度 可用性の要求度 (AR: Availability Impact)	

VDO、Vulntology
脆弱性を体系化して記述する

- Vulnerability Description Ontology (脆弱性記述の概念体系)
- 脆弱性記述の体系化

項目	説明
Vulnerability	脆弱性
Sector of Interest	分野
Known Chain	関連する脆弱性
Provenance	情報源
Scenario	攻撃シナリオ
Type	脆弱性種別
Product	製品
Attack Theater	攻撃者
Barrier	攻撃を阻害する要因

項目	説明
Context	コンポーネント種別
Application Type	アプリケーション種別
Entity Role	エンティティの役割
Mitigation	緩和要因
Impact Method	脆弱性の攻撃方法
Logical Impact	論理的な影響
Physical Impact	物理的な影響
Scope	スコープ
Criticality	重要度

[URL] Vulnerability Description Ontology (VDO): a Framework for Characterizing Vulnerabilities
https://csrc.nist.gov/publications/detail/nistir/8138/draft

VDO、VuIntology
脆弱性を体系化して記述する

- CVSS基本評価基準の詳細化

CVSSv3	
攻撃の難易度	どこから攻撃可能であるか 攻撃元区分 (AV: Attack Vector)
	攻撃する際に必要な条件の複雑さ 攻撃条件の複雑さ (AC: Attack Complexity)
	攻撃する際に必要な特権レベル 必要な特権レベル (PR: Privileges Required)
	攻撃する際に必要なユーザ関与レベル ユーザ関与レベル (UI: User Interaction)
攻撃による影響	攻撃による影響の想定範囲 影響の想定範囲 (S: Scope)
	機密情報が漏えいする可能性 機密性への影響 (C: Confidentiality Impact)
	情報が改ざんされる可能性 完全性への影響 (I: Integrity Impact)
	業務が遅延・停止する可能性 可用性への影響 (A: Availability Impact)

CVSSv3
物理
ローカル
隣接ネットワーク
ネットワーク

Attack Theater:Limited Remote
Cellular
Wireless
Bluetooth
Infrared
Line-of-Sight

Attack Theater:Remote
Internet
Intranet
Local Network

CVSSv3
高
低
不要

Barrier:Privilege Level
Anonymous
User
Privileged
Administrator
Generic Trust

Context
Hypervisor
Firmware
Host OS
Guest OS
Application
Channel
Physical Hardware

CVSSv3
高
低
なし

Logical Impact:Service Interrupt Type
Shutdown
Reboot
Hang
Panic
Unrecoverable

CPE
製品を識別する

IPA

- Common Platform Enumeration
  (共通プラットフォーム一覧)
- 情報システムを構成するハードウェア、ソフトウェアの名称を、プログラムで(機械)処理しやすい形式で記述するための仕様

IPAが提供するMyJVN

IPAが提供するマイ・ジェイ・ブイ・エヌ

情報処理推進機構が提供するMyJVN

アイ・ピー・エーが提供するMyJVN

情報処理推進機構が提供するマイ・ジェイ・ブイ・エヌ

cpe:/a:ipa:myjvn

cpe:/{種別}:{ベンダ}:{製品}:{バージョン}:{アップデート}:{エディション}:{言語}

種別：h=ハードウェア、o=OS、a=アプリケーション

CPE
v2.2とv2.3

- CPE v2.2とv2.3には互換性がある
- MyJVN APIでは、CPE v2.2をサポート

v2.3

cpe:2.3:{種別}:{ベンダ}:{製品}:{バージョン}:{アップデート}:{エディション}:{言語}
:{sw_edition}:{target_sw}:{target_hw}:{other}

v2.2

基本形
cpe:/{種別}:{ベンダ}:{製品}:{バージョン}:{アップデート}:{エディション}:{言語}

拡張形
cpe:/{種別}:{ベンダ}:{製品}:{バージョン}:{アップデート}
:~{エディション}~{sw_edition}~{target_sw}~{target_hw}~{other}:{言語}

CPE
MyJVN APIでのCPEサポート仕様

- MyJVN APIでのベンダ、製品記述の仕様
- ベンダ
  - cpe:/:{ベンダ}

    cpe:/:oracle

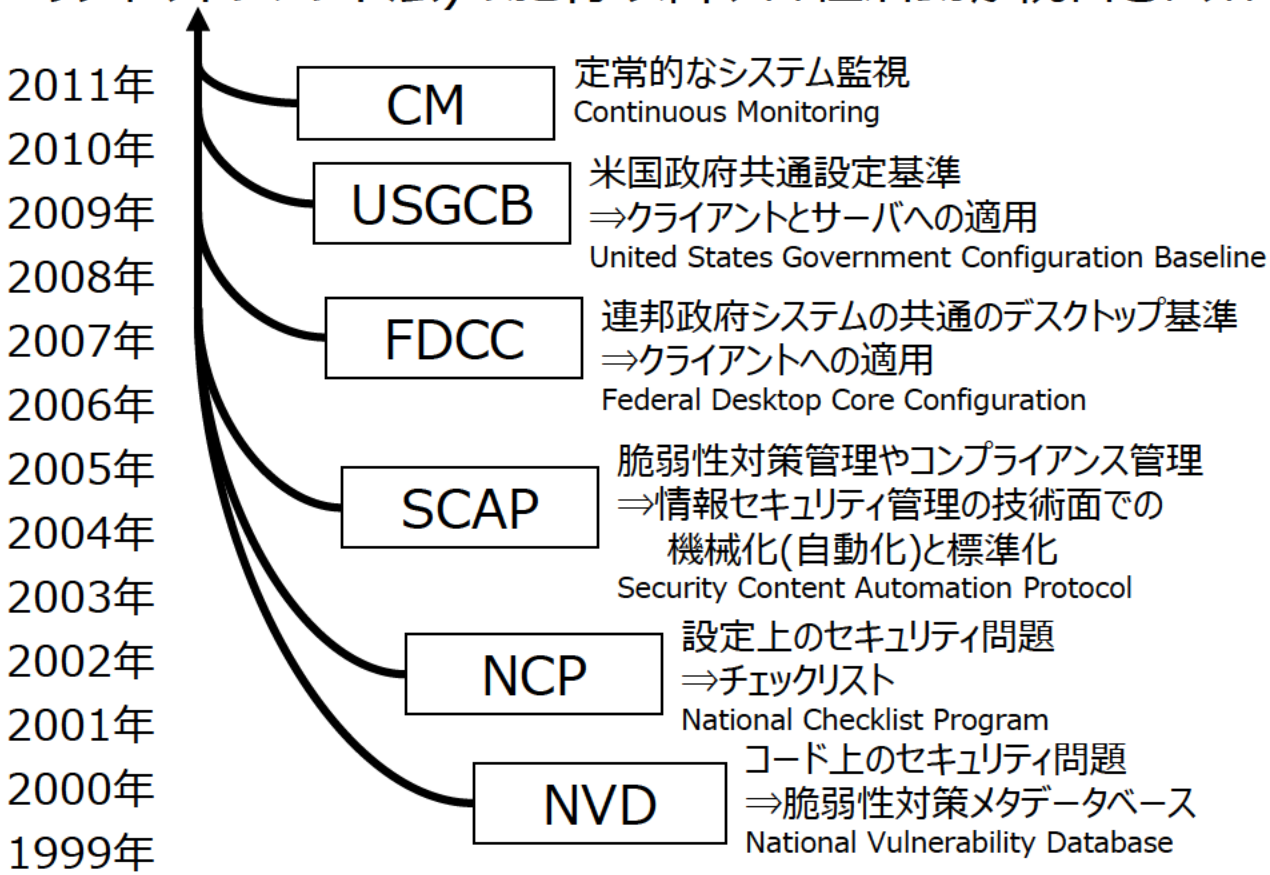
- 製品
  - cpe:/{種別}:{ベンダ}:{製品}

    cpe:/o:oracle:solaris
    cpe:/a:oracle:jre

自動化(機械化)処理基盤の潮流

自動化(機械化)処理基盤の潮流
米国における取り組み：脆弱性対策

- 1999年のiCAT(脆弱性対策メタデータベース)構築からはじまり、2002年のFISMA(Federal Information Security Management Act：連邦情報セキュリティマネジメント法)の施行以降、各種活動が統合され始めた。



自動化(機械化)処理基盤の潮流
セキュリティ設定共通化手順SCAP

IPA

- 稼働する情報システムが、抽象レベルで記載された情報システムセキュリティの規格やガイドラインに沿っているかを手作業で確認することは難しいため、機械処理により実現すべきであることから始まった。

【 課題 】

セキュリティ設定に関する作業を手作業で行なうと、設定ミスや設定者のセキュリティ知識の程度や判断の相違などによりセキュリティ要件を損なう可能性大

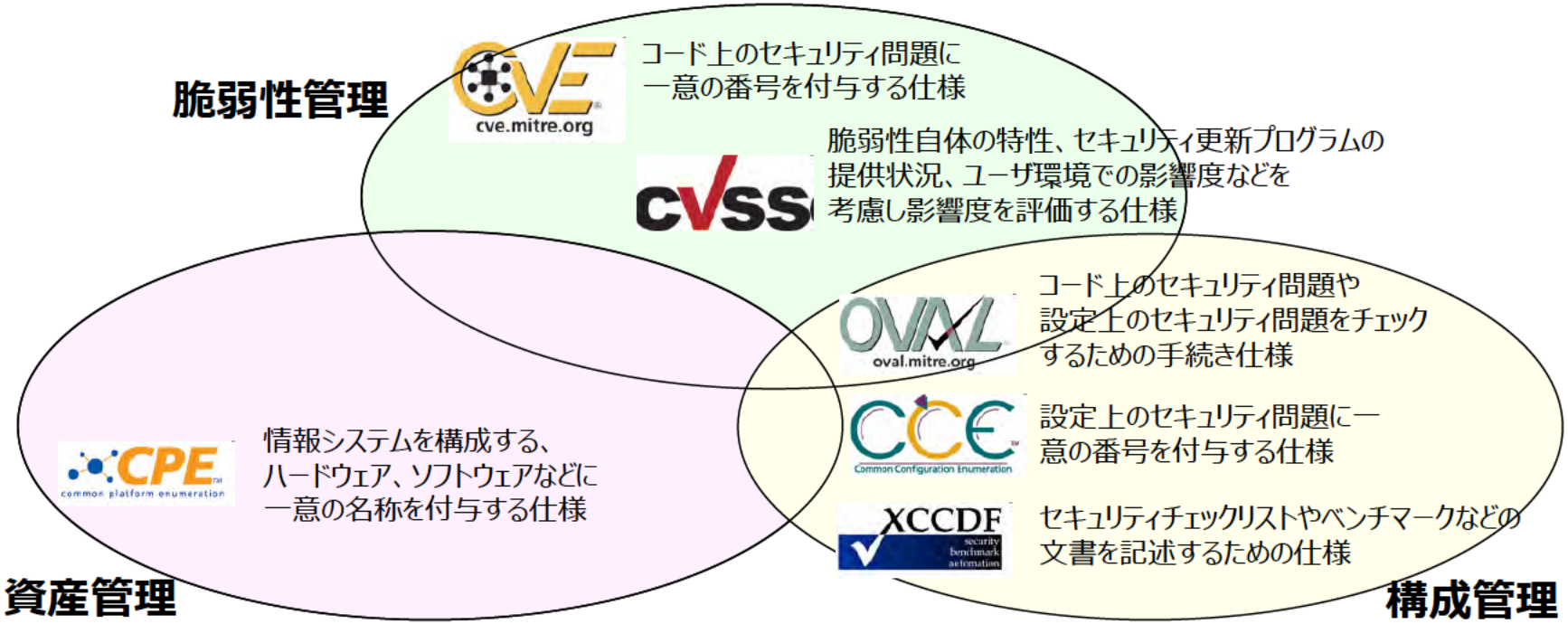
⇩

【 解決策 】

作業の機械化(自動化)による対処
⇒ **SCAP(Security Content Automation Protocol)**

自動化(機械化)処理基盤の潮流
セキュリティ設定共通化手順SCAP

- 脆弱性管理、コンプライアンス管理の一部を機械化(自動化)することにより、情報システムに対するセキュリティ対策の負荷軽減と情報セキュリティ施策の推進の両立を目的とした6つの仕様から構成されている。

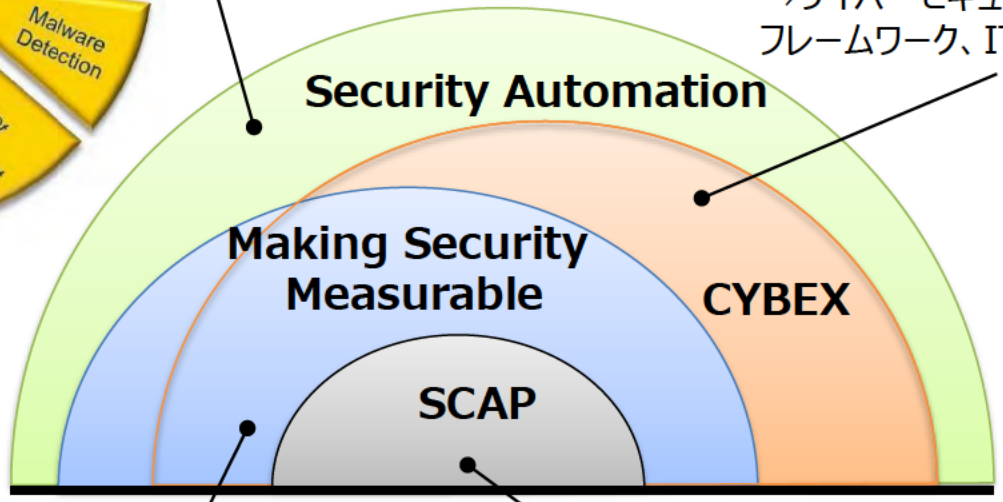


自動化(機械化)処理基盤の潮流

自動化(機械化)処理基盤に関連する仕様

Security Automation
⇒米NISTがSP800-137（Information Security Continuous Monitoring for Federal Information Systems and Organizations）で想定する対象の仕様群

X.1500
⇒サイバーセキュリティ情報交換フレームワーク、ITU-Tで規定する仕様群



Making Security Measurable
⇒米MITRE社で開発中の仕様群

SCAP
⇒Security Content Automation Protocol：セキュリティ設定共通化手順、米NISTがFDCC、USGCBで使用する仕様群

自動化(機械化)処理基盤の潮流
NCP(National Checklist Program)

- NIST SP800-70rev4で規定された、
『セキュリティ設定のガイド』の開発ならびに共有支援プログラム
- 米国連邦政府内部で使用される(使用される可能性のある)
コンピュータハードウエアまたはソフトウエア、システムに関連する
セキュリティリスクを最小限に抑えるための設定とオプション選択を規定したチェックリストとリポジトリを整備している。
- NCPチェックリストリポジトリには、559件(単一製品のバージョン違い含む)の『セキュリティ設定のガイド』が登録されている。

*)セキュリティ設定ガイドは、セキュリティ設定チェックリスト、ロックダウンガイド、セキュリティ強化ガイド、ベンチマーク等とも呼ばれる。

[URL] National Checklist Program for IT Products: Guidelines for Checklist Users and Developers
https://csrc.nist.gov/publications/detail/sp/800-70/rev-4/final
National Checklist Program Repository
https://nvd.nist.gov/ncp/repository

自動化(機械化)処理基盤の潮流
セキュリティ設定共通化手順SCAP

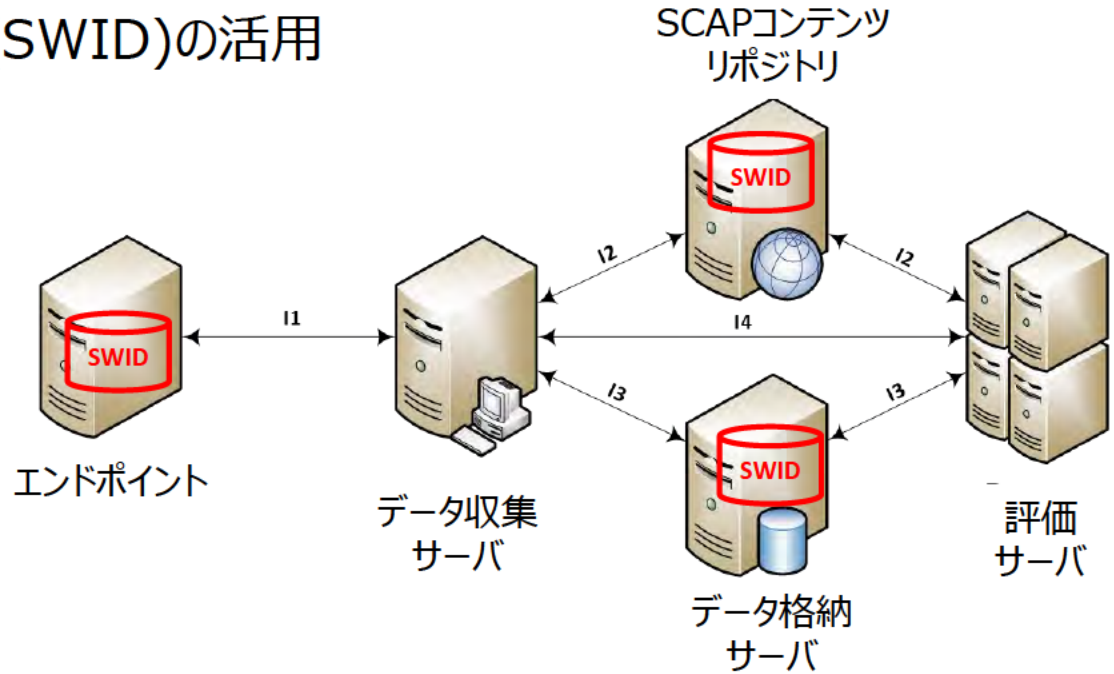
- 脆弱性対策を進める3つの視点(仕様、コード、設定)

脆弱性の分類	事例	チェック方法
仕様の脆弱性	認証の仕組みがない	机上レビュー 手動検査(ペネトレーション検査)
コードの脆弱性	パスワードの値をチェックしていない パスワードがハードコーディングされている	ホワイトボックス型 ソースコード検査
		ブラックボックス型 未知の脆弱性 ⇒ファジング検査
		ブラックボックス型 既知の脆弱性 ⇒脆弱性検査
設定の脆弱性	アカウントとパスワードが同じ	セキュリティ設定検査 (ハードニング検査)

自動化(機械化)処理基盤の潮流

新たな動き　SCAP 2.0

- 2018年12月、NSA/NIST/DHSが検討中のSCAP v2の概要を公開
- SCAP v1では、自動化(機械化)に必要となる識別(CCE、CPE、CVE)、収集(OVAL)、評価(XCCDF)のための機能を開発
- SCAP v2では、エンドポイントのハードウェア/ソフトウェア資産管理、脆弱性管理、設定管理のためのアーキテクチャ、インターフェース、データモデル、プロトコルを開発
- ソフトウェア識別タグ(SWID)の活用



自動化(機械化)処理基盤の潮流

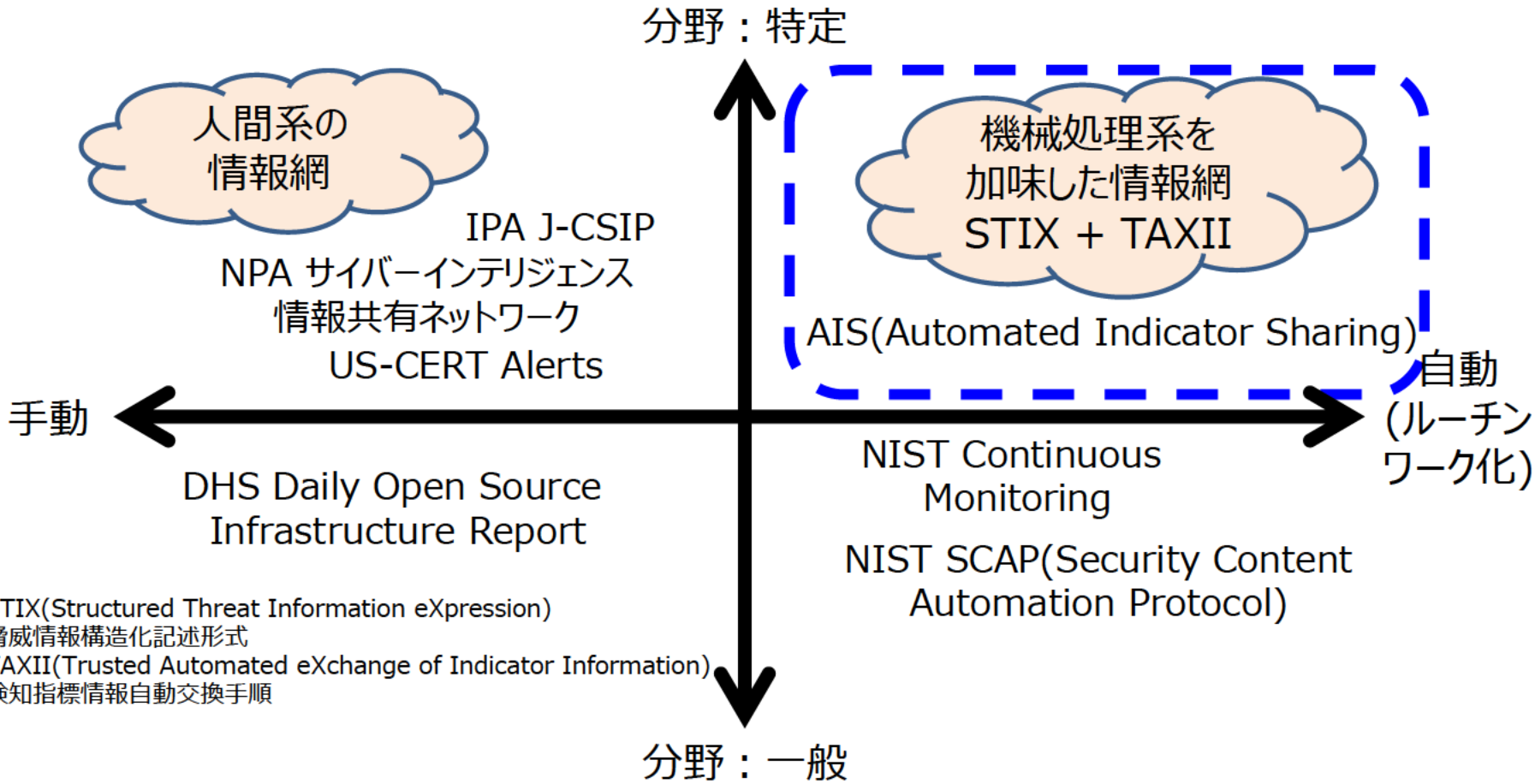
新たな動き　Software Component Transparency

- 米国商務省電気通信情報局(National Telecommunications and Information Administration)が主体となって推進中で、2018年7月に、初回打合せが開催された。
- ソフトウェアを構成するコンポーネントの透明性(Software Component Transparency)をあげることを目的とする。
- SBOM(ソフトウエア部品表)というアイディアについて、活発に議論されている。
  - よく使用されている例: 食品の成分表のようなもの
  - 使用コンポーネントの「透明性」の確保

[URL]　NTIA Software Component Transparency
https://www.ntia.doc.gov/SoftwareTransparency

自動化(機械化)処理基盤の潮流
米国における取り組み：情報共有

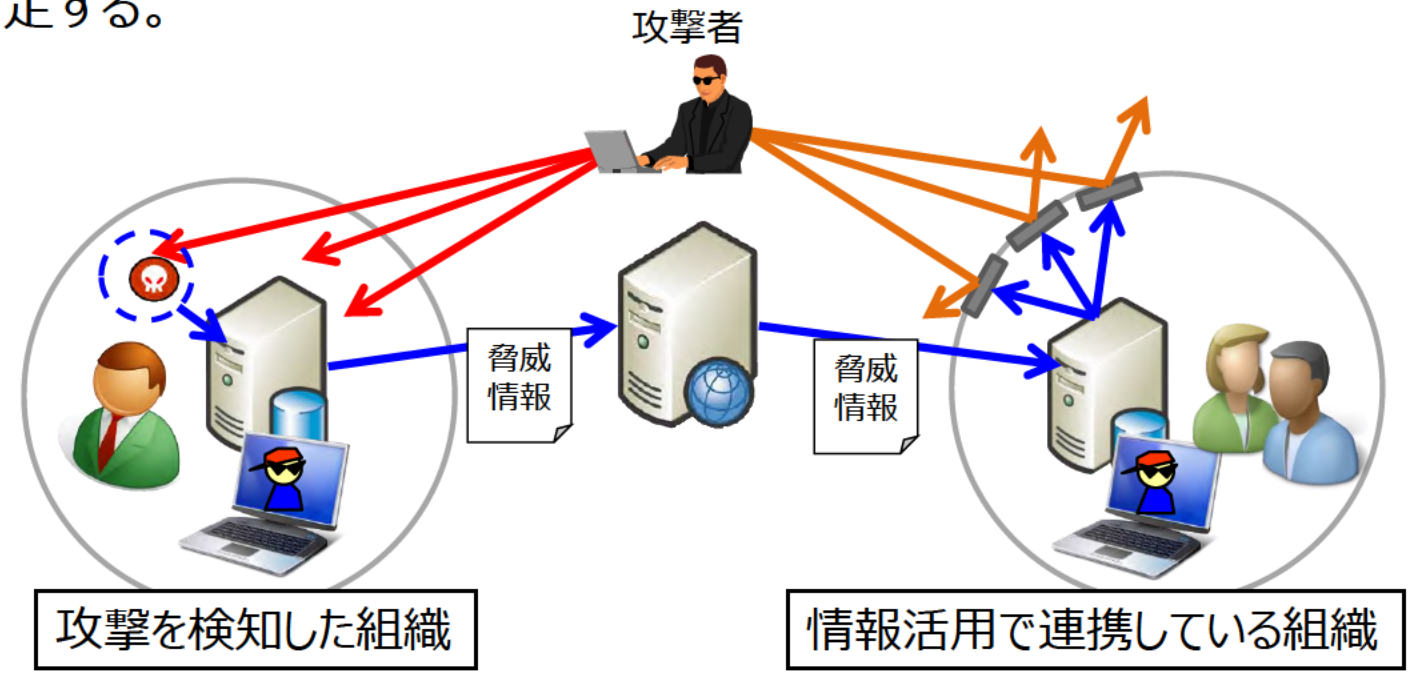
- 2012年頃から機械処理系を想定した大量の脅威情報が流通する仕組みが検討され始める。



STIX(Structured Threat Information eXpression)
脅威情報構造化記述形式
TAXII(Trusted Automated eXchange of Indicator Information)
検知指標情報自動交換手順

攻撃のモデル化と対処
多層防御としての(情報活用＋対策)

- 多層防御としての(情報活用＋対策)連携
  - 事前措置：組織にサイバー攻撃が行われる前に(入手タイミング)、組織にない情報を利用して(カバー率)、サイバー攻撃対策につなげる。
  - 事後措置：組織にない情報を利用してサイバー攻撃による影響有無を特定する。



攻撃のモデル化と対処
サイバー攻撃スピードへの追従

- 認知から対策までの時間短縮



アラートの観測　脅威の把握　対策の決定　脅威への対処

Manual(手動)

最悪のケース　11時間　45分

ベストケース　10分　45分

Automated(ルーチンワーク化)

最悪のケース　10分

ベストケース　1秒

対策までの時間の98%を削減可

トリアージ能力10000倍向上

60秒

30秒

[URL] Financial Sector Use Case: Cyber Threat Intelligence Processes & Sources
https://www.oasis-open.org/events/sites/oasis-open.org.events/files/Carlson.pdf

攻撃のモデル化と対処
TLP(Traffic Light Protocol)

- 情報活用のための情報共有レベル区分
- 情報発行元毎に、区分を定義している(区分の定義は必ずしも一致しない場合があるので注意が必要)。

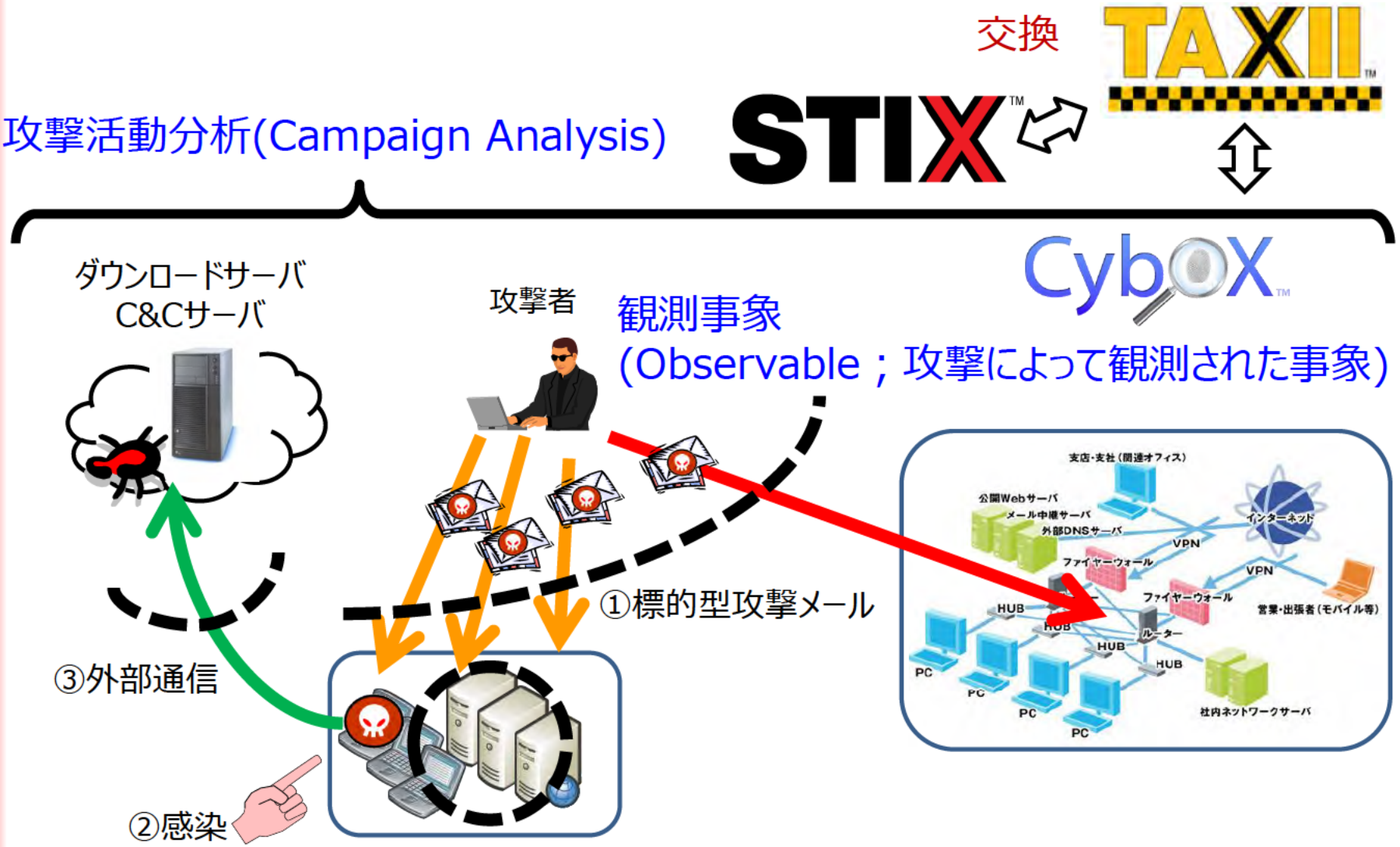
区分	US-CERT/FIRST.org	金融ISAC	会員区分に基づく例
TLP:RED	非公開、関係者限り	特定のグループ(会議参加者等)の受信者のみとし、当該グループ外への転送を禁ずる	宛先限り(ただし、受信者の上長や責任者への説明・報告は可)
TLP:AMBER	公開制限、関係者組織限り	金融ISAC会員限りで共有する	会員のみ
TLP:GREEN	公開制限、コミュニティ限り	金融 ISAC会員及び予め理事会が定める業界団体等の外部組織との共有を行うことが可能	会員に加えて、会員のグループ関係会社や外部委託先を含めた範囲で共有できる
TLP:WHITE	公開制限なし	公知の情報として扱う ※著作権法その他の法令を遵守する	公開可能(ただし著作権法その他の法令は遵守する必要がある)

[URL] https://www.us-cert.gov/tlp
https://www.first.org/tlp/
http://f-isac.jp/pdf/F-ISACjpn_Management_Rules.pdf

攻撃のモデル化と対処
脅威表現の標準化

交換

攻撃活動分析(Campaign Analysis)



攻撃のモデル化と対処
CybOX (サイバー攻撃観測記述形式)

- MITREが中心となり仕様策定を進めてきたもので、サイバー攻撃活動によって観測された事象を記述するためのXML仕様である。MandiantのOpenIOC (侵害を受けたシステムの痕跡(Indicator of Compromise)を記述する仕様)を踏まえた仕様となっている。
- システム(Windows、UNIX)内部状態、ネットワーク通信、アプリケーション動作など、観測できる事象を記録する。

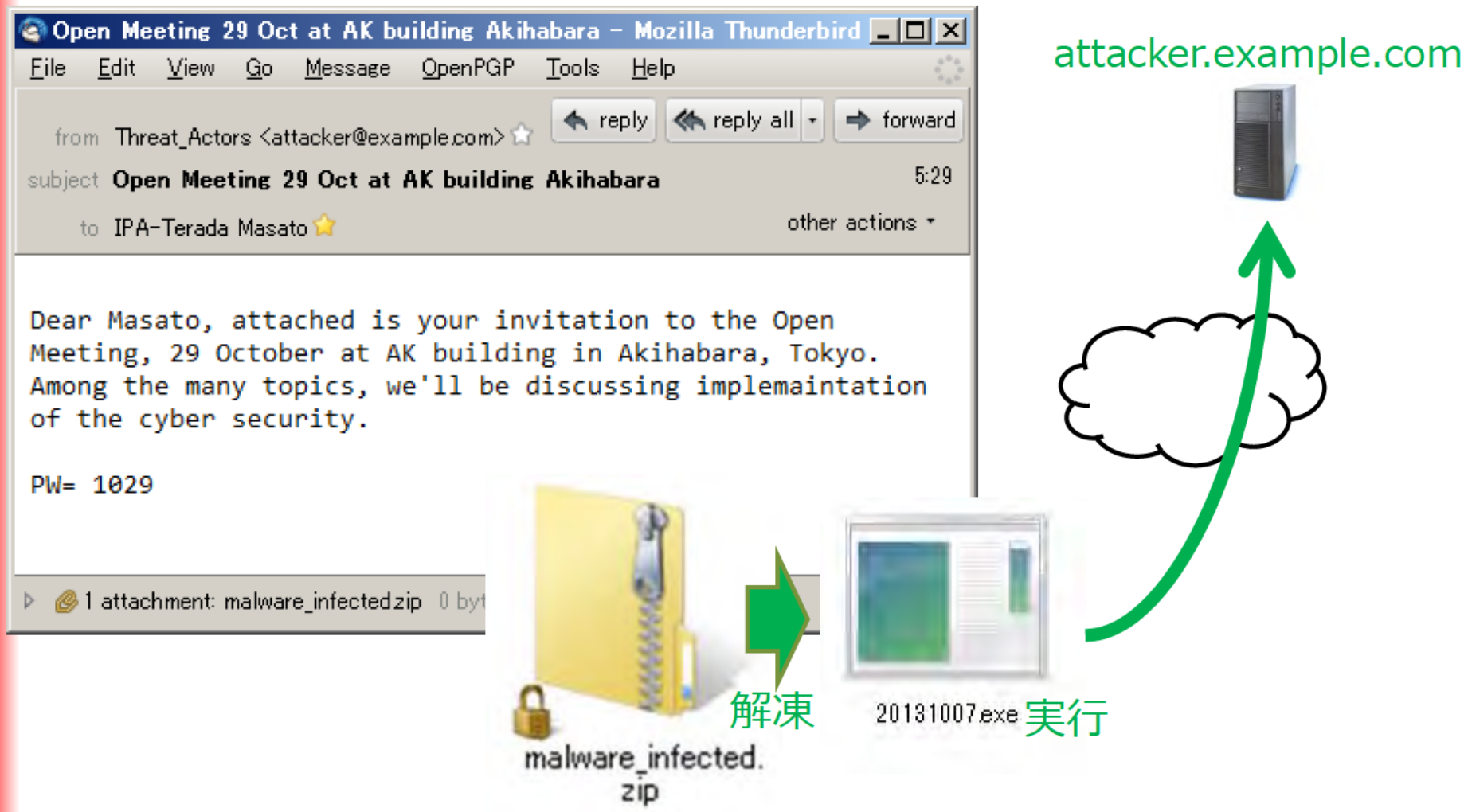
Common
API
Account
Address
Artifact
Code
Device
Disk
DomainName
File
HostName
:

Windows
Win File
Win Mutex
Win Process
Win Registry Key
Win Service
Win Task
Win User Account
Win Volume
:

Network
Network Connection
Network Flow
Network Packet
Network Route
Network Socket
Port
Socket Address
:

攻撃のモデル化と対処

標的型攻撃メールの観測事象(Observable)記述例



攻撃のモデル化と対処

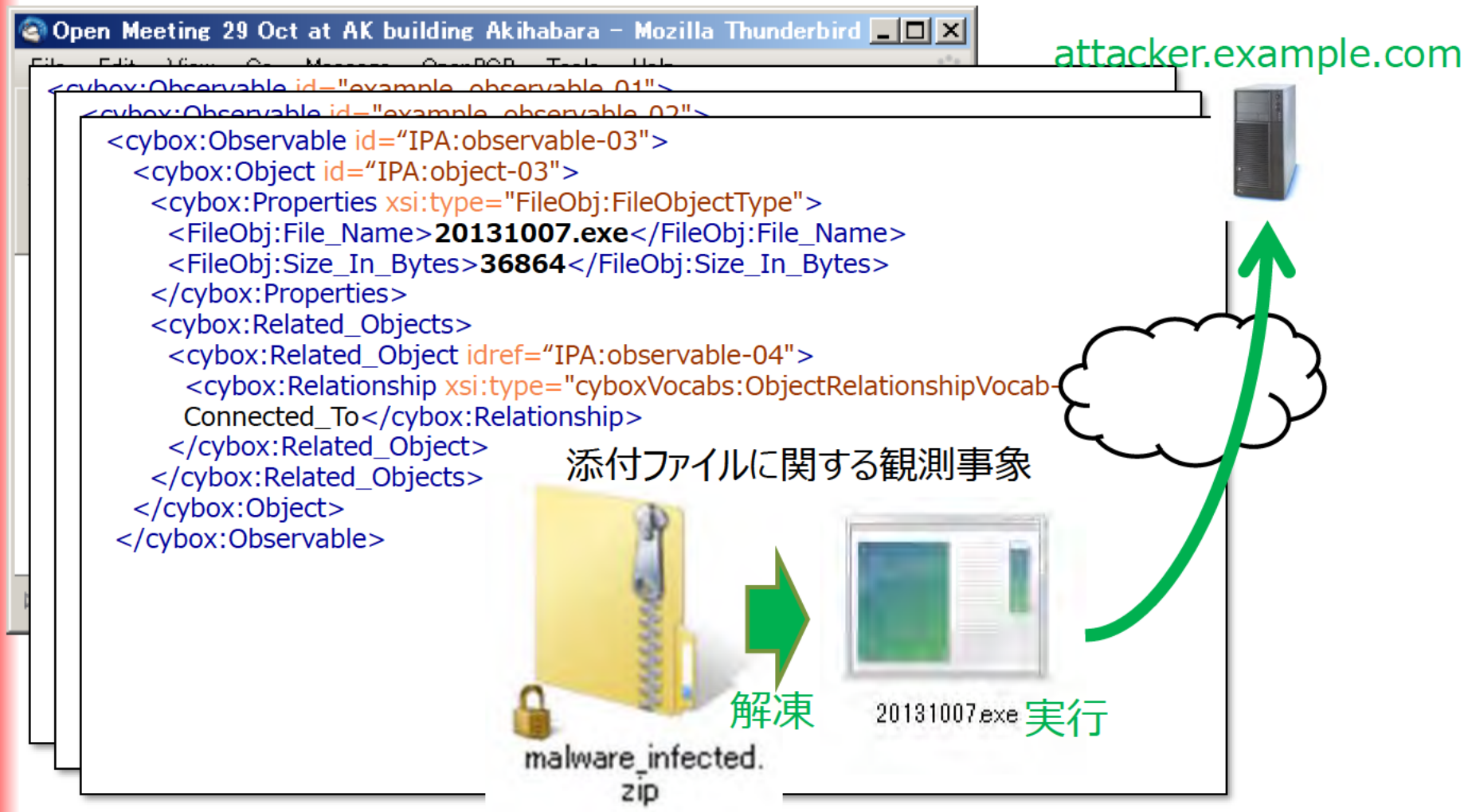
標的型攻撃メールの観測事象(Observable)記述例

攻撃のモデル化と対処

標的型攻撃メールの観測事象(Observable)記述例

攻撃のモデル化と対処

標的型攻撃メールの観測事象(Observable)記述例



攻撃のモデル化と対処

標的型攻撃メールの観測事象(Observable)記述例

Open Meeting 29 Oct at AK building Akihabara - Mozilla Thunderbird

```
<cybox:Observable id="IPA:observable-04">
  <cybox:Object id="IPA:object-04">
    <cybox:Properties xsi:type="DomainNameObj:DomainNameObjectType"
      type="FQDN">
      <DomainNameObj:Value condition="Equals">attacker.example.com
      </DomainNameObj:Value>
    </cybox:Properties>
  </cybox:Object>
</cybox:Observable>
```

attacker.example.com

不正接続先に関する観測事象

malware_infected.zip

解凍

20131007.exe

実行

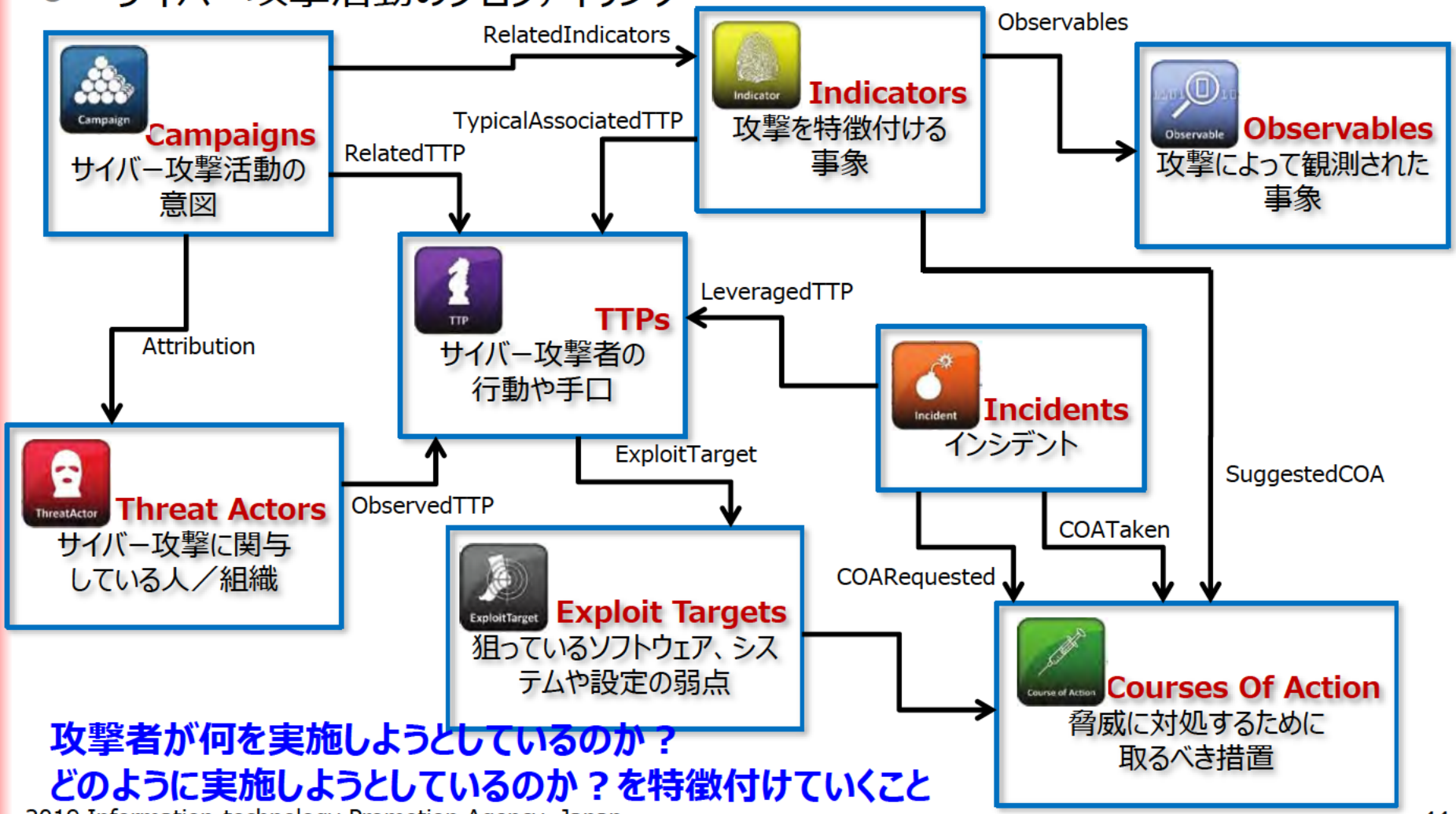
攻撃のモデル化と対処
STIX(脅威情報構造化記述形式)

- MITREが中心となり仕様策定を進めてきたもので、サイバー空間における脅威やサイバー攻撃の分析、サイバー攻撃を特徴付ける事象の特定、サイバー攻撃活動の管理、サイバー攻撃に関する情報の共有などを目的としたXML仕様である。
- ８つの情報群から構成され、これらの情報群を関連づけて脅威情報を表現する。

情報群	脅威情報
サイバー攻撃活動 (Campaigns)	サイバー攻撃活動における意図や攻撃活動の状態など
攻撃者 (Threat Actors)	サイバー攻撃に関与している人または組織
攻撃手口 (TTPs)	サイバー攻撃者の行動や手口、攻撃のパターン
検知指標 (Indicators)	検知に有効なサイバー攻撃を特徴づける指標
観測事象 (Observables)	サイバー攻撃によって観測された事象
インシデント (Incidents)	サイバー攻撃によって発生した事案
対処措置 (Courses Of Action)	脅威に対処するために取るべき措置
攻撃対象 (Exploit Targets)	攻撃の対象となりうるソフトウェアやシステムの弱点

攻撃のモデル化と対処
STIX(脅威情報構造化記述形式)

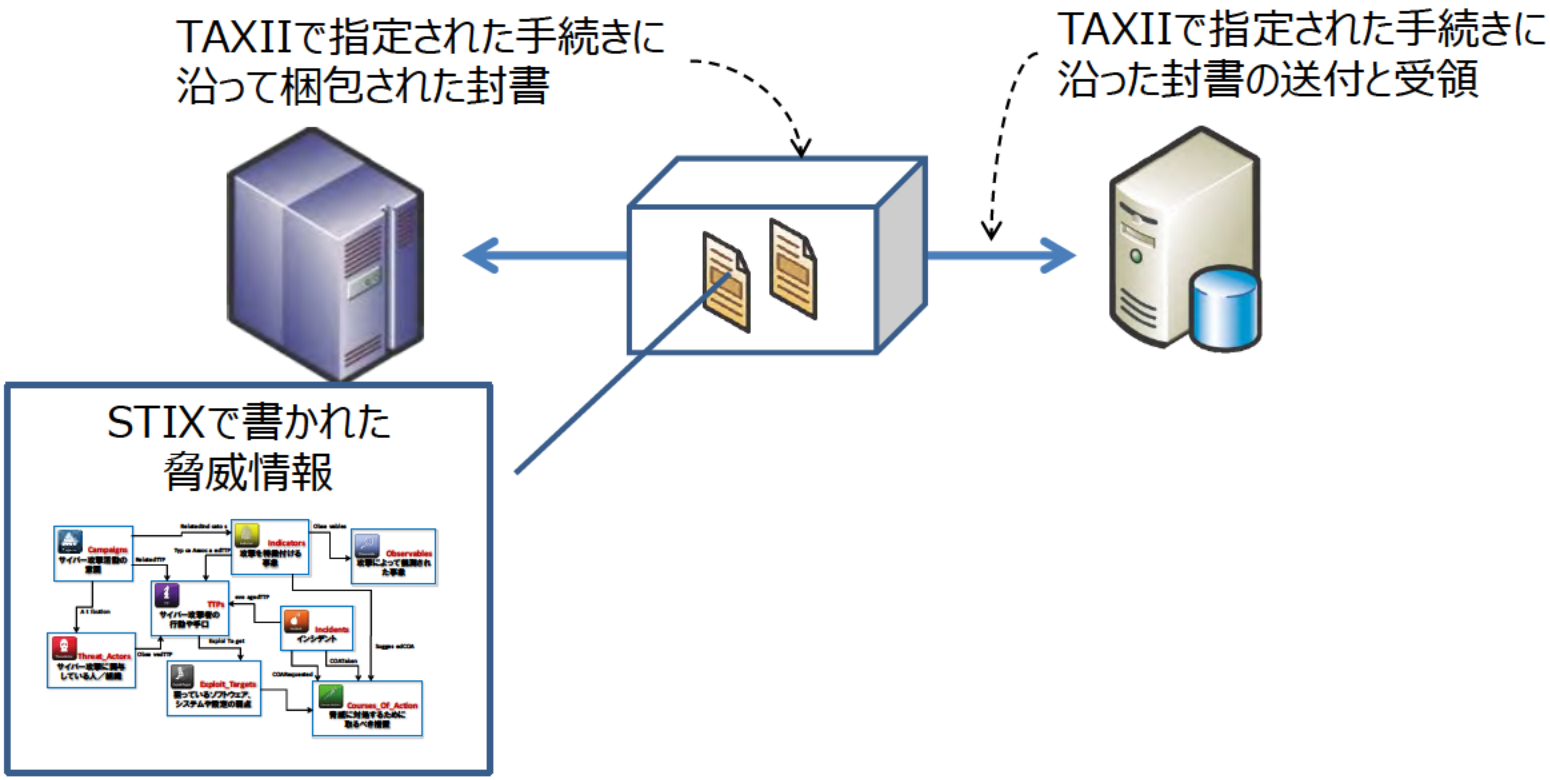
- サイバー攻撃活動のプロファイリング



攻撃者が何を実施しようとしているのか？
どのように実施しようとしているのか？を特徴付けていくこと

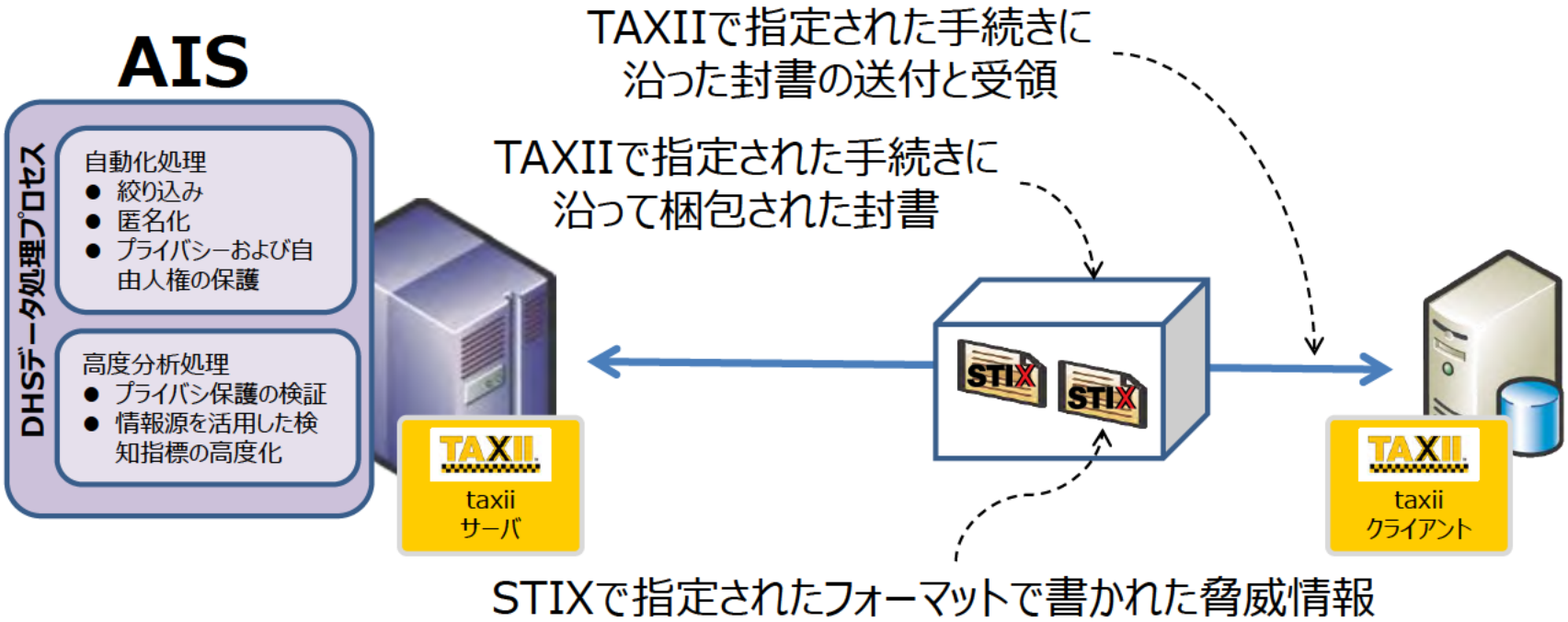
攻撃のモデル化と対処
TAXII(検知指標情報自動交換手順)

- DHSが主導し、MITREが中心となり仕様策定を進めてきたもので、サイバー攻撃を検知するために使用できる指標(Indicators)を交換するためのサービス、プロトコルならびにフォーマットを規定した仕様である。
- STIXで書かれた脅威情報をTAXIIという手順を使って交換する。



攻撃のモデル化と対処
米国AIS(Automated Indicator Sharing)

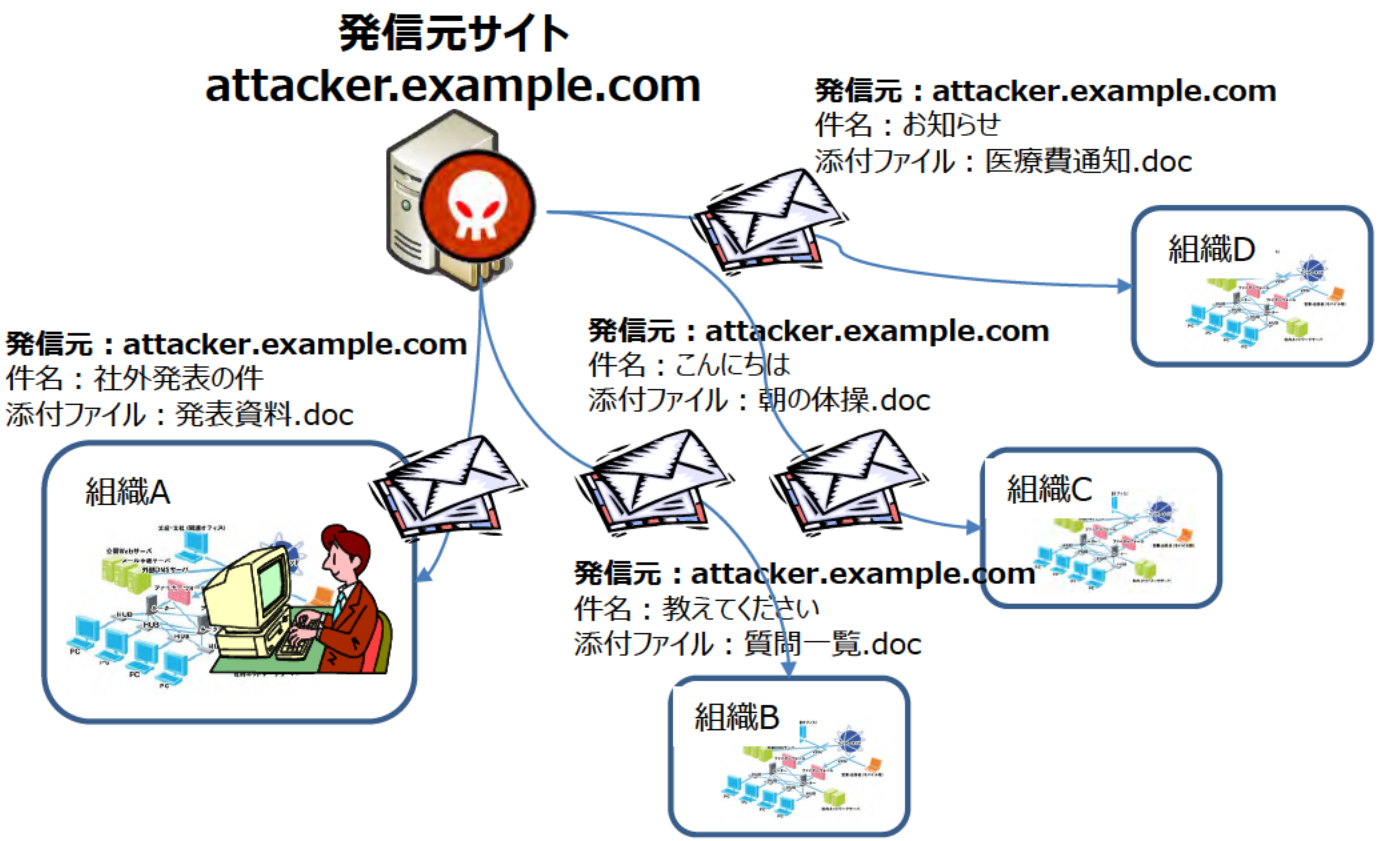
- 米国政府が提供する官民連携の情報共有基盤
- 2015年サイバーセキュリティ法により、2016年3月からDHSの下で活動を開始した。攻撃指令サーバのドメインやIPアドレス、マルウェアのハッシュ値などの検知指標(インディケータ)を共有する情報基盤である。



[URL] Automated Indicator Sharing (AIS)
https://www.dhs.gov/cisa/automated-indicator-sharing-ais

攻撃のモデル化と対処
検知指標(インディケータ)

- 検知に有効なサイバー攻撃を特徴づける指標



発信元：attacker.example.com
攻撃活動の共通項はインディケータとして使いやすい。

攻撃のモデル化と対処

標的型攻撃メールの観測事象(Observable)記述例の場合

- 不正接続先を検知指標(インディケータ)とする場合には、該当する観測事象をポイントする。

```
<stix:Indicators>
 <stix:Indicator xsi:type="indicator:IndicatorType" id="IPA:indicator-6928">
  <indicator:Title>標的型攻撃メール攻撃活動を検知する(検知指標に対するタイトル)</indicator:Title>
  <!-- 検知指標のタイプ -->
  <indicator:Type xsi:type="stixVocabs:IndicatorTypeVocab-1.1">Domain Watchlist</indicator:Type>
  <indicator:Description>malware_infected.zipの接続先(検知指標の説明)</indicator:Description>
  <!-- 検知指標に関連する観測事象 -->
  <indicator:Observable>
    <cybox:Observable idref="IPA:observable-2313"/>
  </indicator:Observable>
 </stix:Indicator>
</stix:Indicators>
```

```
<cybox:Observable id="IPA:observable-2313">
  <cybox:Title>不正接続先(観測事象に対するタイトル)</cybox:Title>
  <cybox:Object id="IPA:object-4392">
    <cybox:Properties xsi:type="DomainNameObj:DomainNameObjectType" type="FQDN">
      <DomainNameObj:Value condition="Equals">attacker.example.com
      </DomainNameObj:Value>
    </cybox:Properties>
  </cybox:Object>
 </cybox:Observable>
```

不正接続先に関する観測事象

攻撃のモデル化と対処
STIXバージョン1と2との違い

項目	STIXバージョン1	STIXバージョン2
リリース日	Ver1.2(MITRE版2015年5月) Ver1.2.1(OASIS版2016年5月)	Ver2.0(OASIS版2017年10月)
仕様の特徴	脅威情報記述の構造化を重視	脅威情報記述の実用性を重視
	脅威情報を構造化する視点から開発されているため、大きく8つの情報群を用意している。個々の情報群でさらに詳細記述できる。	実用的な脅威情報表記の視点から開発しているため、頻繁に使用する表記を情報群として用意している。特に、Ver1.2系の攻撃手口(TTP)、攻撃対象(Exploit Target)を細分化している。
記述言語	XML	JSON
検知指標(Indicator)の記述	不正なファイルハッシュ値を記述する際に、CybOXという別な仕様と組み合わせて記述していた。 STIX v1.2 Indicator CybOX md5=xxxx	不正なファイルハッシュ値を記述する際に、STIXの仕様だけで記述できるようにした。 STIX v2.0 Indicator md5=xxx

攻撃のモデル化と対処

STIXバージョン1と2との違い

項目	STIXバージョン1	STIXバージョン2
仕様の特徴	1.サイバー攻撃活動(Campaign) 2.攻撃者(Threat Actor) └攻撃者の名前、組織名(Identity) 3.検知指標(Indicator) 4.観測事象(Observable) 5.インシデント(Incident) 6.対処措置(Courses Of Action) 7.攻撃手口(TTP) ├攻撃パタン(Attack Pattern) ├マルウェア(Malware) └攻撃で用いたツール(Tool) 8.攻撃対象(Exploit Target) └脆弱性(Vulnerability)	1.サイバー攻撃活動(Campaign) 2.攻撃者(Threat Actor) 3.攻撃者の名前、組織名(Identity) 4.検知指標(Indicator) 5.観測事象(Observed Data) 6.対処措置(Courses Of Action) 7.攻撃パタン(Attack Pattern) 8.マルウェア(Malware) 9.攻撃で用いたツール(Tool) 10.侵害(Intrusion Set) 11.脆弱性(Vulnerability) 12.レポート(Report)

攻撃のモデル化と対処
STIXバージョン1と2との違い

- STIXバージョン1でのハッシュ値の記述

```
<stix:Indicators>
<stix:Indicator>
 <indicator:Title>Poison Ivy Malware</indicator:Title>
 <indicator:Type>File Hash Watchlist</indicator:Type>
 <indicator:Description>This file is part of Poison Ivy</indicator:Description>
 <indicator:Observable>
  <cybox:Object>
  <cybox:Properties xsi:type="FileObj:FileObjectType">
   <FileObj:Hashes>
    <cyboxCommon:Hash>
    <cyboxCommon:Type xsi:type="cyboxVocabs:HashNameVocab-1.0">MD5</cyboxCommon:Type>
    <cyboxCommon:Simple_Hash_Value condition="Equals">3773a88f65a5e780c8dff9cdc3a056f3
    </cyboxCommon:Simple_Hash_Value>
    </cyboxCommon:Hash>
  </FileObj:Hashes>
  </cybox:Properties>
  </cybox:Object>
 </indicator:Observable>
</stix:Indicator>
</stix:Indicators>
```

攻撃のモデル化と対処
STIXバージョン1と2との違い

- STIXバージョン2でのハッシュ値の記述


```
{
  "type": "indicator",
  "id": "indicator--8e2e2d2b-17d4-4cbf-938f-98ee46b3cd3f",
  "created_by_ref": "identity--f431f809-377b-45e0-aa1c-6a4751cae5ff",
  "created": "2016-04-06T20:03:48.000Z",
  "modified": "2016-04-06T20:03:48.000Z",
  "labels": ["malicious-activity"],
  "name": "Poison Ivy Malware",
  "description": "This file is part of Poison Ivy",
  "pattern": "[ file.hashes.MD5 = '3773a88f65a5e780c8dff9cdc3a056f3' ]",
  "valid_from": "2016-01-01T00:00:00Z"
 },
```

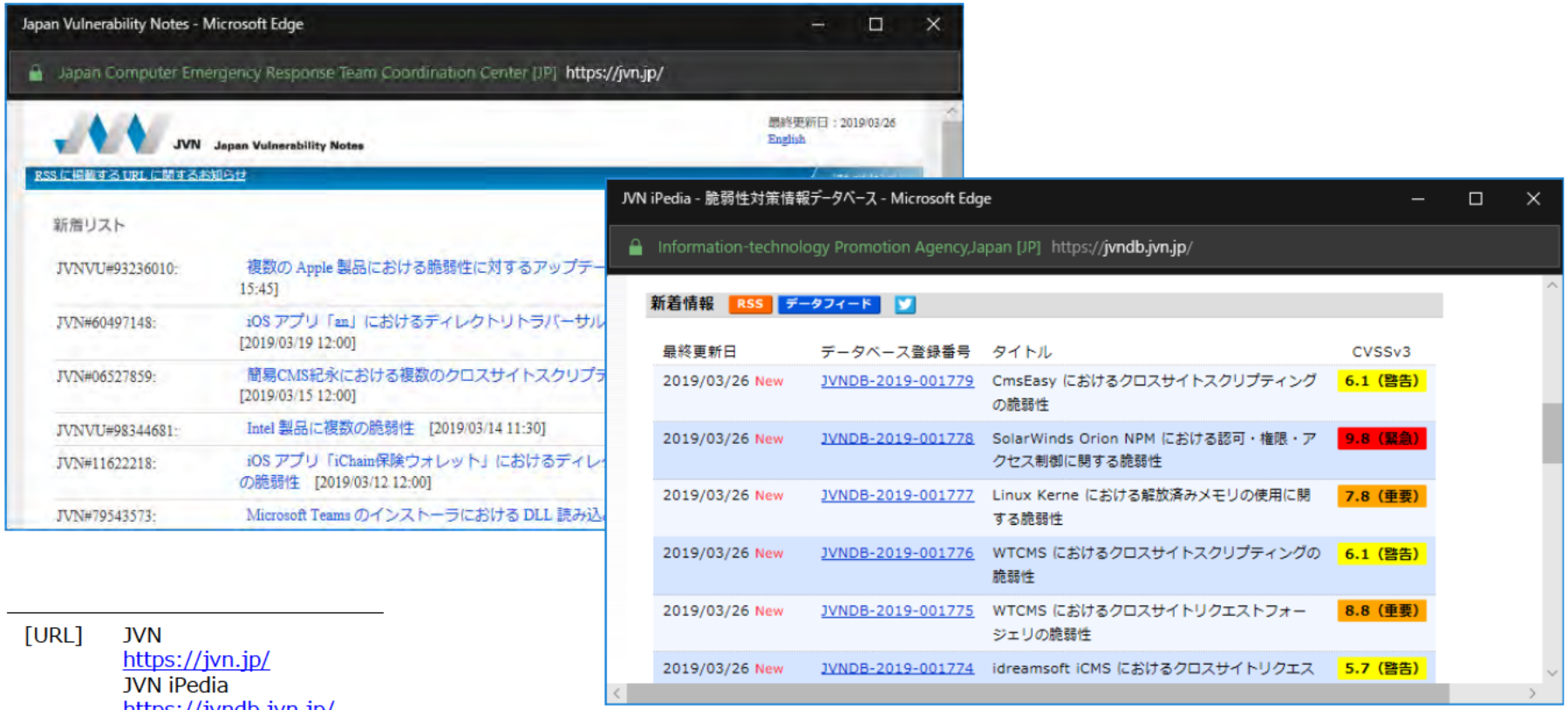
攻撃のモデル化と対処
TAXIIバージョン1と2との違い

項目	TAXIIバージョン1	TAXIIバージョン2
リリース日	Ver1.1(DHS版2014年1月) Ver1.1.1(OASIS版2016年5月)	Ver2.0(OASIS版2017年10月)
仕様の特徴	多様性を重視しており、Web APIも特別な仕様を採用	実用性と汎用性を重視し、Web APIも一般的な仕様を採用
	HTTP/HTTPS+XML XMLデータとして、STIXだけではなく、CAP(Common Alerting Protocol)と呼ばれる緊急時の警報情報を取り扱うことができる。	HTTPS+JSON
	パラメタをTAXIIで規定されたXMLで送付する	パラメタをREST APIを使って送付する

JVN脆弱性対策機械処理基盤

JVN脆弱性対策機械処理基盤
JVN

- JVN は、“Japan Vulnerability Notes” の略
- 日本で使用されているソフトウェアなどの脆弱性関連情報とその対策情報を提供し、情報セキュリティ対策に資することを目的とする脆弱性対策情報サイト

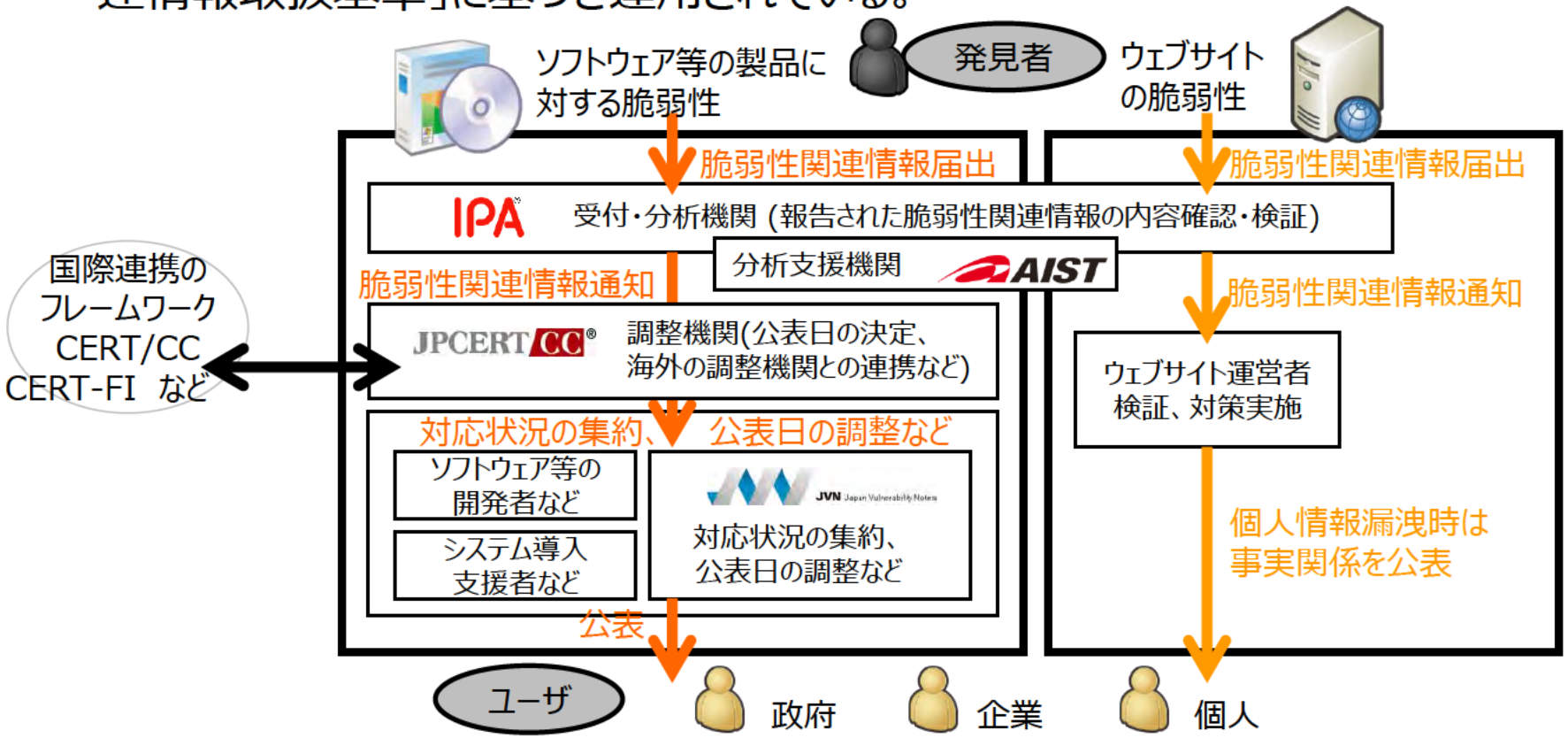


[URL] JVN
https://jvn.jp/
JVN iPedia
https://jvndb.jvn.jp/

JVN脆弱性対策機械処理基盤

情報セキュリティ早期警戒パートナーシップ

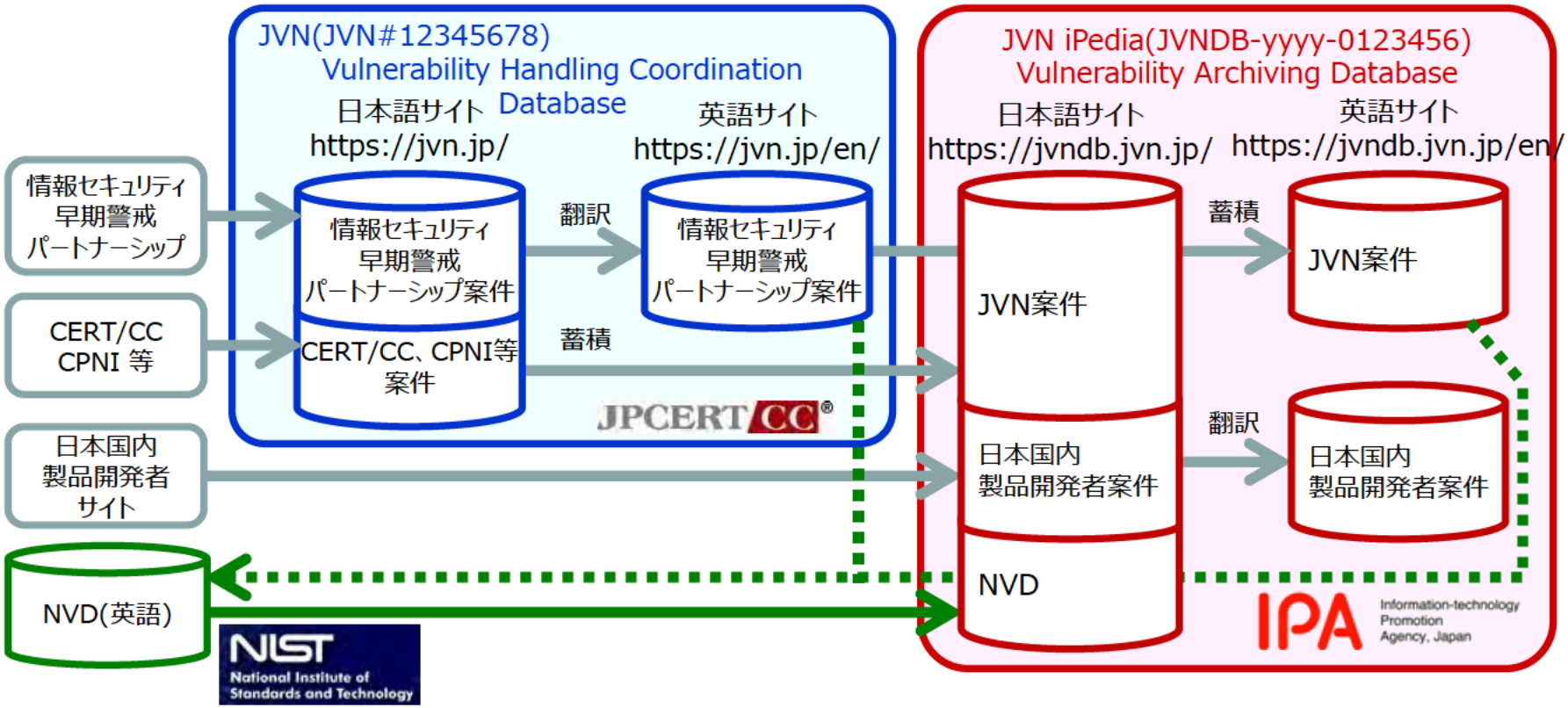
- ソフトウエア等の製品やウェブサイトに見つかった脆弱性に関する情報を受け付け、製品開発者に修正を促すフレームワーク
- 2004年7月8日施行の脆弱性関連情報の取扱い「ソフトウエア等脆弱性関連情報取扱基準」に基づき運用されている。



JVN脆弱性対策機械処理基盤

JVNは2つのDBから構成されている

- 脆弱性対策情報ポータルサイトJVN(製品開発者と調整した脆弱性対策情報をタイムリーに公開)と、脆弱性対策情報データベースJVN iPedia(国内で利用されている製品を対象にした脆弱性対策情報を広く蓄積)から構成



JVN脆弱性対策機械処理基盤
対策実施サービスに繋げるための仕組み

IPA

- ＝(JVN＋JVN iPedia)× MyJVN
- ＝(国際性＋地域性)× 利活用基盤

MyJVN

バージョンチェッカ

セキュリティ設定チェッカ

脆弱性対策情報収集ツール

JVN Japan Vulnerability Notes

情報セキュリティ早期警戒パートナーシップに報告された脆弱性対策情報

JVN iPedia

国内外で報告された脆弱性に対する国内製品開発者の脆弱性対策情報

共通脆弱性識別子CVEが付与されている脆弱性対策情報

国内外で報告された全ての脆弱性対策情報

MyJVN
JVNとJVN iPediaに登録されている脆弱性対策情報を対策実施に直結したサービスに繋げるための仕組みを提供する

JVN iPedia
国内で利用されている製品を対象にした脆弱性対策情報を広く蓄積する

JVN
製品開発者と調整した脆弱性対策情報をタイムリーに公開する

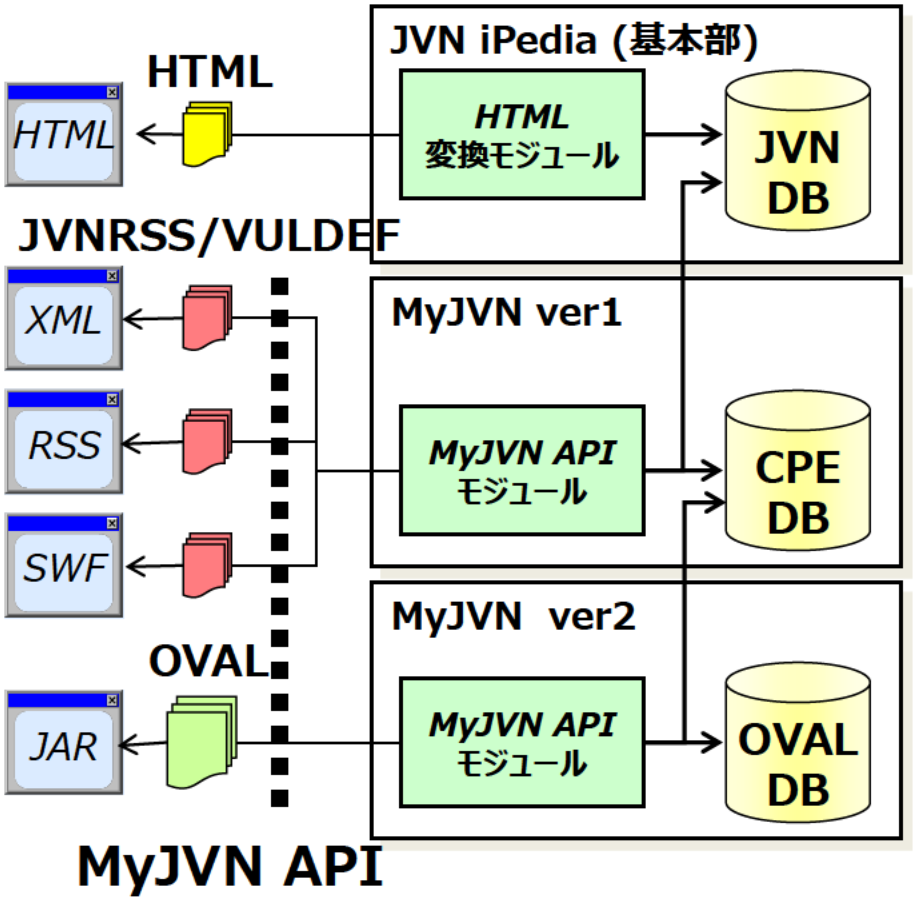
MyJVN API
http://jvndb.jvn.jp/apis/

- JVN iPediaを活用し、新たなサービスを準備できる環境を整備する。

JVN iPediaの情報を、Webを通じて利用するためのソフトウェアインタフェース
⇒ユーザ側でのツール開発も可能

フィルタリング型情報提供
⇒ MyJVN脆弱性対策情報収集ツール
フィルタリング収集ツール (mjcheck3)
脆弱性対策情報ダッシュボード (mjdashboard)

検査データ提供
⇒ MyJVNバージョンチェッカ
オンラインバージョンチェック (.NET Framework版)



MyJVN API
http://jvndb.jvn.jp/apis/

IPA

分類	名称	概要
フィルタリング型情報提供	製品提供者一覧取得 getVendorList	フィルタリング条件に当てはまるベンダ名(製品開発者)リストを取得します。
	製品一覧取得 getProductList	フィルタリング条件に当てはまる製品名リストを取得します。
	脆弱性対策概要情報一覧取得 getVulnOverviewList	フィルタリング条件に当てはまる脆弱性対策の概要情報リストを取得します。
	脆弱性対策詳細情報取得 getVulnDetailInfo	フィルタリング条件に当てはまる脆弱性対策の詳細情報を取得します。
	注意警戒情報一覧取得 getAlertList	注意警戒情報リストを取得します。
検査データ提供	OVAL定義データ一覧の取得 getOvalList	フィルタリング条件に当てはまる OVAL 定義データリストを取得します。
	OVAL定義データの取得 getOvalData	OVAL 定義データを取得します。
	XCCDFチェックリストデータ一覧の取得 getXccdfList	XCCDF チェックリストデータリストを取得します。
	XCCDFチェックリストデータの取得 getXccdfData	XCCDF チェックリストデータを取得します。
その他	統計データの取得 getStatistics	脆弱性対策情報を、脆弱性統計情報)、CVSS統計情報、CWE統計情報で集計したデータを取得します。

MyJVN API
フィルタリング型情報提供

- 脆弱性対策情報のためのXMLフォーマット
- 情報の記述粒度による使い分けを考慮した概要記述向けと詳細記述向けXMLフォーマットを用意

概要フォーマット
JVNRSS 2.0
= RSS1.0+mod_sec

MyJVN API
脆弱性対策概要情報一覧取得
getVulnOverviewList
応答フォーマット

- Title
- Overview
- Affected System
- Impact
- Solution
- Exploit
- Reference

MyJVN API
脆弱性対策詳細情報取得
getVulnDetailInfo
応答フォーマット

詳細フォーマット
VULDEF

MyJVN API
フィルタリング型情報提供

- 脆弱性対策概要情報一覧取得
  getVulnOverviewList

  - リクエスト形式
    https://jvndb.jvn.jp/myjvn?method=getVulnOverviewList&cpeName=cpe:/a:ipa:ilogscanner&rangeDatePublic=n&rangeDatePublished=n&rangeDateFirstPublished=n&feed=hnd
    注)MyJVN APIではCPEのバージョンを未サポート。
    CPEにバージョンを含めると検索にヒットしない。

  - レスポンス形式
    - JVNRSS 2.0= RSS1.0+mod_sec
    - https://jvndb.jvn.jp/schema/jvnrss_3.2.xsd
    - https://jvndb.jvn.jp/schema/mod_sec_3.0.xsd
      - TLP(US-CERTの定義を使用)、CVSSv2、CVSSv3をサポート

MyJVN API
フィルタリング型情報提供

- 脆弱性対策詳細情報取得
  getVulnDetailInfo
  - リクエスト形式
    https://jvndb.jvn.jp/myjvn?method=getVulnDetailInfo&vulnId=JVNDB-2009-000004&feed=hnd
  - レスポンス形式
    - VULDEF: The VULnerability Data publication and Exchange Format data model
    - https://jvndb.jvn.jp/schema/vuldef_3.2.xsd
      - Historyフィールドに更新履歴を記載
    - https://jvndb.jvn.jp/schema/mod_sec_3.0.xsd
      - TLP(US-CERTの定義を使用)、CVSSv2、CVSSv3をサポート

MyJVN API
フィルタリング型情報提供

- 注意警戒情報一覧取得
  getAlertList
  - リクエスト形式
    https://jvndb.jvn.jp/myjvn?method=getAlertList&feed=hnd
  - レスポンス形式
    - XML：Atom+mod_sec
      - https://jvndb.jvn.jp/schema/atom.xsd
      - https://jvndb.jvn.jp/schema/mod_sec_3.0.xsd
        - グループ化記述用フィールドをサポート
    - JSON
      - https://jvndb.jvn.jp/schema/getalert1.json

MyJVN API
フィルタリング型情報提供

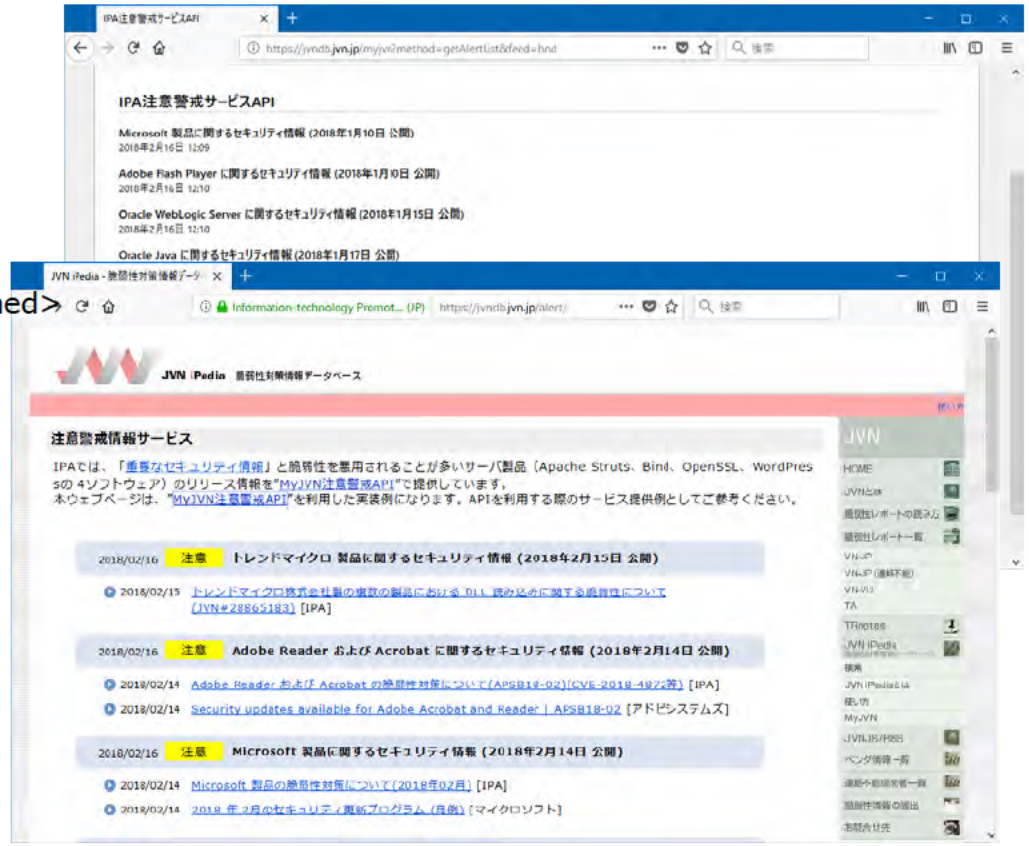
- 注意警戒情報一覧取得
  getAlertList

パラメタ名	名称	パラメタ値	必須	デフォルト
method	メソッド名	getAlertList (固定)	○	–
startItem	エントリ開始位置	整数 (半角数字) 1～応答エントリ数	–	1
maxCountItem	エントリ(<entry>)取得件数	整数 (半角数字) 1～50 (getAlertListエントリ上限値)	–	50
datePublished	更新日年	半角整数4桁	–	動作時の年
dateFirstPublished	発行日年	半角整数4桁	–	
cpeName	CPE 製品名 (cpe:/{part}:{vendor}:{product})	cpe:/{part}:{vendor}:{product} {part}フィールド … "h" | "o" | "a" | "*" {vendor}:{product}フィールド … CPE 製品名 ――――――――――――――― ・ワイルドカード "*" 指定可 ・アスキー文字 ・大文字／小文字区別なし ・複数指定時は "+" で連結 ・URL エンコードされたエスケープシーケンス	–	–
feed	フィードフォーマット名	フィードフォーマット(=APIバージョン)を示す名称	○	–
ft	応答フォーマット	xmlまたはjson	–	xml

MyJVN API
フィルタリング型情報提供

- 注意警戒情報一覧取得
  getAlertList
  - 脆弱性対策の動きの見える化の試みとして、関連する情報をグループ化しながら追記

```
<entry>
  <title>注意警戒のタイトル</title>
  <id>注意警戒の識別子</id>
  <summary>注意警戒の概要</summary>
  <link href="https://www.ipa.go.jp/" />
  <published>2017-07-13T08:29:29-04:00</published>
  <updated>2017-07-03T12:29:29Z</updated>
  <category term=" /Critical" label=" 緊急" />
  <sec:items>
    <sec:item>
      <sec:title>関連情報のタイトル</sec:title>
      <sec:identifier>関連情報の識別子</sec:identifier>
      <sec:summary>関連情報の概要</sec:summary>
      <sec:link href=" 関連情報の概要のURL" />
    </sec:item>
    sec:item ノードを繰り返します。
  </sec:items>
</entry>
```



自動化(機械化)処理基盤の潮流

IT資産管理の規格：ISO/IEC 19770シリーズ

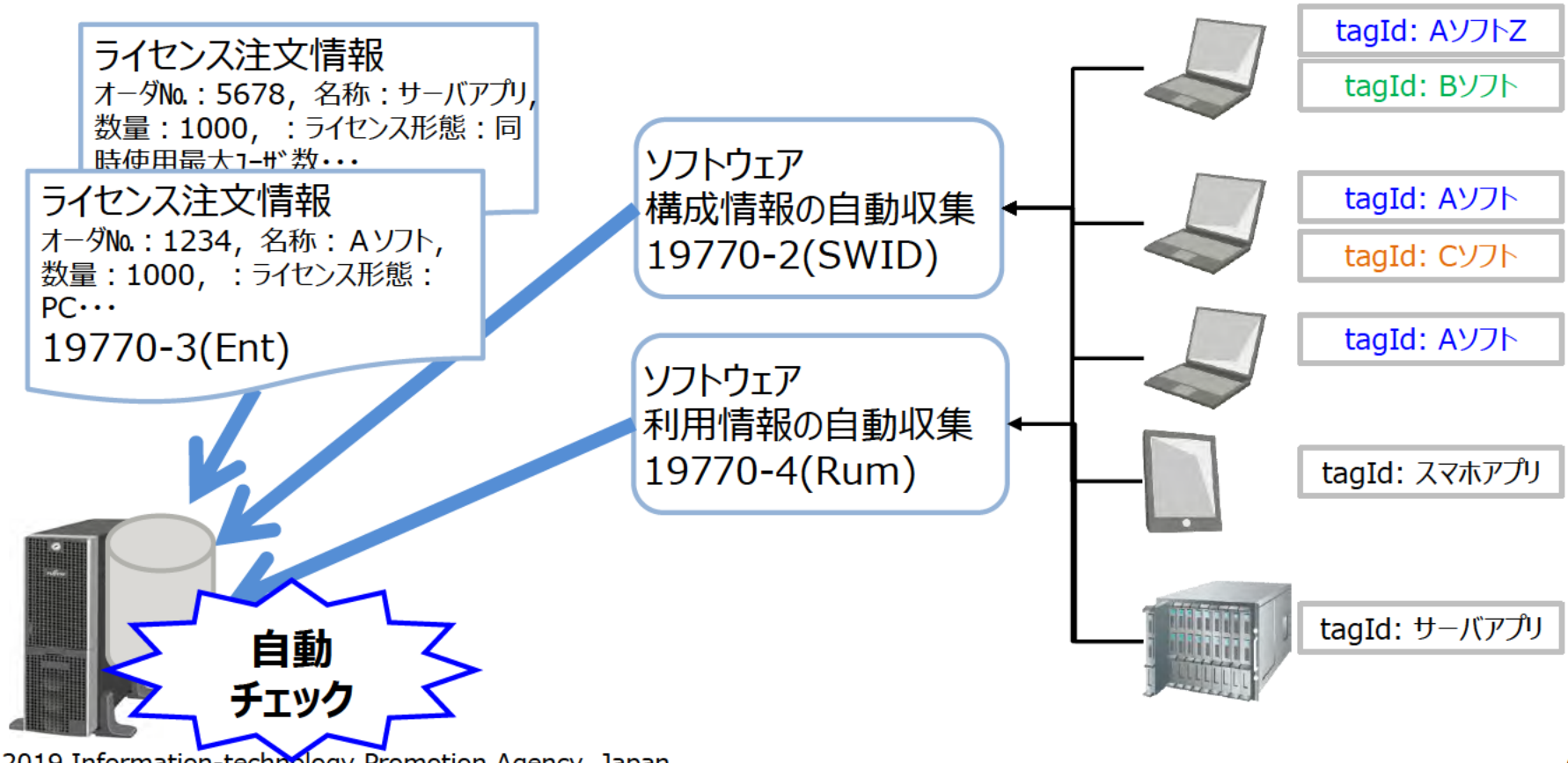
IPA

規格名		概要	状況
ISO/IEC 19770-1:2012 Processes and tiered assessment of conformance 価評性合適的階段び及スセロプ		ソフトウェア資産管理(SAM)のための統合されたプロセス群のベースラインを定める国際規格。2012年の改訂で、付加的な導入，アセスメント及び認識を可能にする「段階」という考え方が取り入れられている。また、ISO/IEC 20000と緊密な整合がとられており、ISO/IEC 20000のITサービスマネジメントを支援することを意図している。	
ISO/IEC 19770-2:2009 Software identification tag ソフトウェア識別タグ (SWID)		ソフトウェアの導入状況を把握するために、導入されたソフトウェアを識別するためのタグの規格である。	改訂版は、2015年9月30日にIS発行。改訂版については、2018年に1月にJIS化(JIS X 0164-2:2018)
ISO/IEC 19770-2:2015 Software identification tag ソフトウェア識別タグ (SWID)	IS	ISO/IEC 19770-2:2009のタグ必須の多くが任意となった。また、パッチ対応属性(delta)、ハッシュ値属性 (sha1, sha256など) がサポートされるようになった。	
ISO/IEC 19770-3 Entitlement schema 権利スキーマ	IS	導入されているソフトウェアのライセンス情報を記述するためのタグの規格である。	2016年4月15日にIS発行 2019年3月にJIS化(JIS X 0164-3:2019)
ISO/IEC 19770-4 Resource Utilization Measurement 資源利用測定	DIS	IT資産の使用に伴うリソースの消費に関する規格である。	2017年9月14日にIS発行
ISO/IEC 19770-5:2015 Overview and vocabulary	IS	IT資産管理のコンセプトや原理を概説し、ISO/IEC 19770シリーズで用いられる用語の定義や、各規格の関連について説明した規格である。	2015年7月31日にIS発行
NISTIR 8060 Guidelines for the Creation of Interoperable Software Identification (SWID) Tags		ISO/IEC 19770-2「ソフトウェア識別タグ」の生成に関するガイドライン	2016年4月発行
NISTIR 8085 Forming Common Platform Enumeration (CPE) Names from Software Identification (SWID) Tags.		ISO/IEC 19770-2「ソフトウェア識別タグ」からX.1528「製品識別子」の生成に関する指針	2015年12月ドラフト発行

自動化(機械化)処理基盤の潮流

IT資産管理の規格：ISO/IEC 19770シリーズ

- 各種情報を集約し、自動チェックするなどの基盤整備に利用可能
- 権利情報は19770-3、ソフトウェア識別情報は19770-2、リソースの利用状況は19770-4で情報構造が標準化される。



自動化(機械化)処理基盤の潮流
IT資産管理の規格：ISO/IEC 19770シリーズ

IPA

- ISO/IEC 19770-2:2015 Software identification tag
  ソフトウェア識別タグ(SWID)
- 組織名_プロダクト名.swidtagに、ソフトウェア識別に関する情報を記述

要素			説明
ルート属性(SoftwareIdentity)			ソフトウェア識別についてのルート属性を記述する。
子要素	組織情報(Entity)	タグを生成した組織は必須 他は選択	このSWIDタグに対して責任のある組織の情報を記述する。
	リンク情報(Link)	選択	他のファイルの参照関係を記述する。関係するファイルやダウンロード元、脆弱性データベース、使用権なども定義できる。
	メタ情報(Meta)	選択	このSWIDに関する任意の情報を記述する。
	ソフトウェアの本来情報(Payload)	選択	インストールされるファイルについて本来の情報を記述する。
	ソフトウェアの実際情報(Evidence)	選択	SWIDタグが見つからないソフトウェアのシステム検査結果を記述する。
	署名情報(Signature)	選択	このSWIDタグに対して責任のある組織の署名情報を記述する。

自動化(機械化)処理基盤の潮流
IT資産管理の規格：ISO/IEC 19770シリーズ

- ISO/IEC 19770-2:2015 Software identification tag
  ソフトウェア識別タグ(SWID)の例

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SoftwareIdentity
   xmlns="http://standards.iso.org/iso/19770/-2/2014/schema.xsd"
   xmlns:SHA256="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"
   name="Fabrikam Writer 2013"
   tagId="{GUID}FabrikamWriter-2013"
   version="12.1.1"
   versionScheme="multipartnumeric">
  <Entity name="Fabrikam Corporation" regid="fabrikam.com" role="softwareCreator licensor tagCreator" />
<!-- also requires the C++ runtime -->
  <Link rel="requires" href="swidpath:/SoftwareIdentity/Meta[@persistentId='EAB2F277-0920-4F8C-B466-
6FE0CA7D59DF']“ ownership="shared" />
  <Payload>
    <File name="writer.exe" size="1024000" location="../bin/"
      SHA256:hash="c4128c2dc663ae7a6b6bc6787594057396e6b3f569cd50fd5ddb4d1bbafd2b6a" />
    <File name="writer.dll" size="524012" location="../bin/"
      SHA256:hash="96e6b3f569cd50fd5ddb4d1bbafd2b6ac4128c2dc663ae7a6b6bc67875940573" />
  </Payload>
</SoftwareIdentity>
```

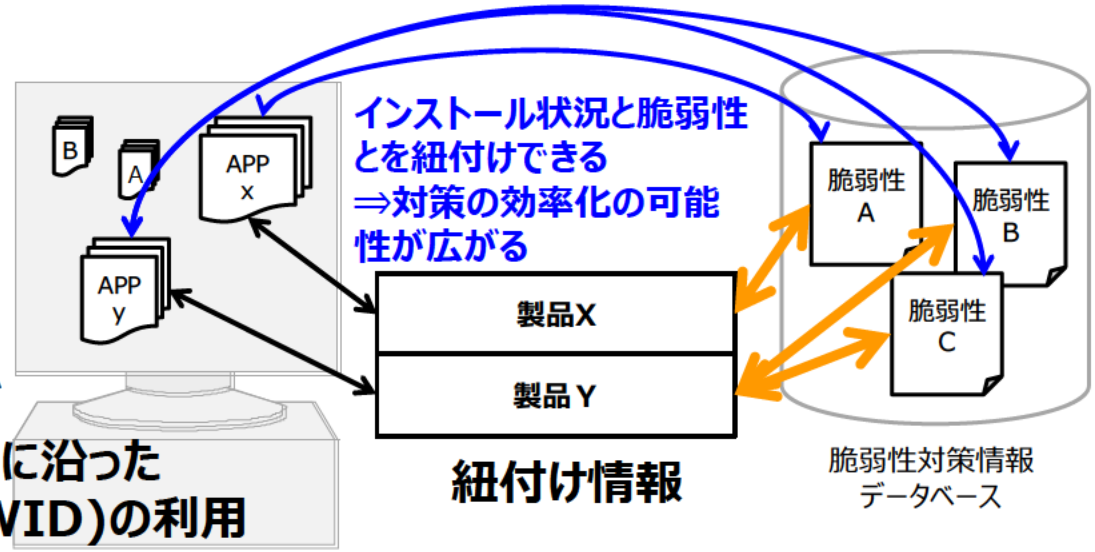
自動化(機械化)処理基盤の潮流
IT資産管理と脆弱性管理の連携

IPA

- インストール状況と脆弱性との紐付け

- 多くの場合、インストール状況と脆弱性との紐付けを人手で実施している(IT資産管理と脆弱性管理とが連携できているわけではない)。

- もし、インストール状況を把握できる情報と脆弱性対策情報サイトとを紐付け[橙色の線]できると、、、



紐付けるための実現方法のひとつが NISTIR 8060、NISTIR 8085に沿ったソフトウェア識別タグ19770-2(SWID)の利用

自動化(機械化)処理基盤の潮流
IT資産管理と脆弱性管理の連携

IPA

- インストール状況と脆弱性との紐付け：国内での取り組み
- ～JVN iPediaの脆弱性対策情報と
  ソフトウェア資産管理情報のデータ連携に着手～


プレス発表　組織で使用するソフトウェアの脆弱性対策の効率・効果的な徹底に向けた調査報告書を公開

～JVN iPedia[注1]の脆弱性対策情報とソフトウェア資産管理情報のデータ連携に着手～

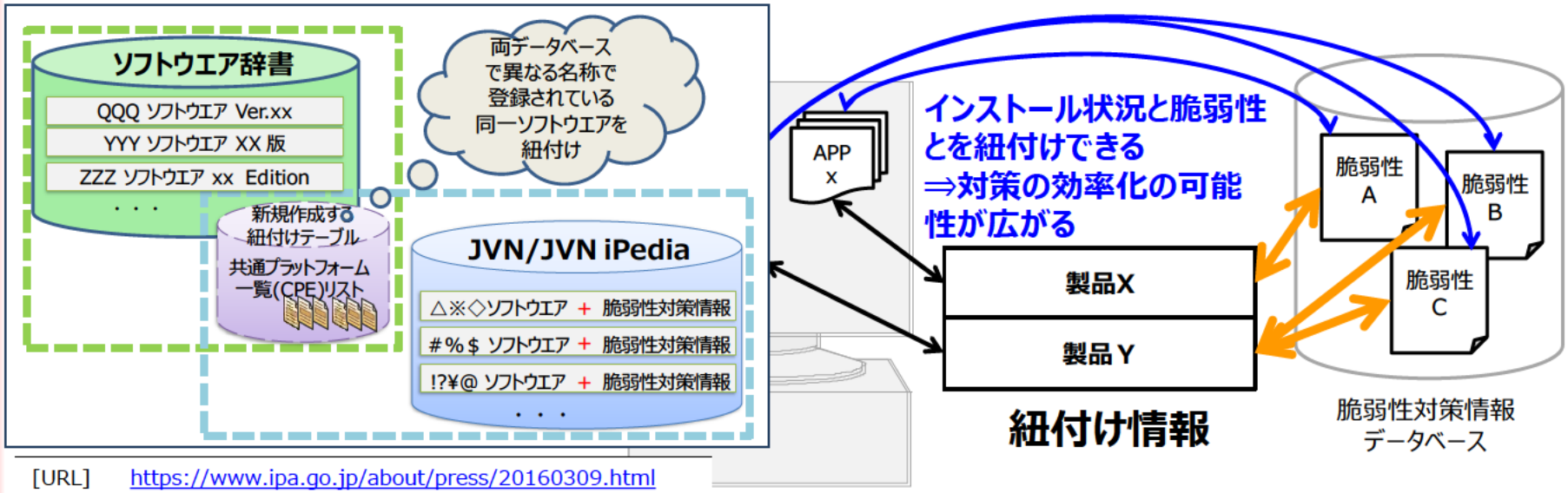
2016年3月9日
独立行政法人情報処理推進機構
一般社団法人ソフトウェア資産管理評価認定協会

IPA（独立行政法人情報処理推進機構、理事長：富田 達夫）は、組織で使用するソフトウェアの脆弱性対策の効率・効果的な徹底を目指した「ソフトウェア脆弱管理に向けた分析事業」報告書を3月9日（水）に公開しました。これを受け、一般社団法人ソフトウェア資産管理評価認定協会（理事長：高橋快男　以後、SAMAC[注2]）は2016年4月以降、脆弱性対策情報とソフトウェア資産管理のデータ連携に向けた紐付けテーブルの作成に着手します。

URL：http://www.ipa.go.jp/sec/reports/20160309.html

ソフトウェアは今やパソコン、スマホだけでなく、家電、自動車などあらゆる機器に組み込まれ、便利な機能の実現や、新たな価値を生み出しています。その一方でソフトウェアに潜む脆弱性は、組み込まれた製品を意図せぬ攻撃の標的にし、利用者にもその影響を及ぼします。また、その攻撃では多くの場合、ソフトウェアの脆弱性が悪用されています。

ソフトウェアの管理と安全な使用の一元的な管理を実現可能にするため、2014年6月にIPAではJVN iPediaが持つ脆弱性対策情報とSAMAC(一般社団法人 IT資産管理評価認定協会)が持つ“ソフトウェア辞書”とのデータ連携について検討に着手しました。



[URL]　https://www.ipa.go.jp/about/press/20160309.html

自動化(機械化)処理基盤の潮流

IT資産管理と脆弱性管理の連携

- インストール状況と脆弱性との紐付け：国内での取り組み
- SAMACソフトウエア辞書
  SAMAC(一般社団法人ソフトウエア資産管理評価認定協会)が保守提供しているインストール状況を把握できるデータベース

ソフトウェア名	ベンダ名	エイリアス	バージョン	エディション	種別
Adobe Flash Player 10 ActiveX	ADOBE SYSTEMS	Flash Player	10	ActiveX	フリーウェア
Realtek High Definition Audio Driver	Realtek Semiconductor	High Definition Audio Driver	-	-	ドライバー・ユーティリティー等
Microsoft .NET Framework 3.5 SP1	Microsoft	.NET Framework	3	-	フリーウェア
IP Messenger for Win32	白水 啓章	IP Messenger	32	-	フリーウェア
Microsoft Office Personal 2007	Microsoft	Office	2007	Personal	有償ソフトウェア
JUSTSYSTEMｱﾌﾟﾘｹｰｼｮﾝの追加と削除	JUSTSYSTEMS	アプリケーションの追加と削除	-	-	ドライバー・ユーティリティー等
Google Toolbar for Internet Explorer	Google	Google Toolbar	-	-	フリーウェア
Intel(R) Graphics Media Accelerator Driver	Intel	Graphics Media Accelerator Driver	-	-	ドライバー・ユーティリティー等

- インベントリ収集ツールで収集可能な[プログラムの追加と削除]に表示されているインストール名称をベースに作成
- ソフトウエア辞書に登録されている項目は、ベンダ名、ソフトウエア名、エディション、バージョン、ソフトウエア種別(有償ソフトウエア・フリーウェア、HOTFIX、ドライバ・ユーティリティ等)

自動化(機械化)処理基盤の潮流

IT資産管理と脆弱性管理の連携

IPA

- インストール状況と脆弱性との紐付け：国内での取り組み
- 製品識別子CPEを用いた製品の紐付け
  Common Platform Enumeration(共通プラットフォーム一覧)
  情報システムを構成するハードウェア、ソフトウエアの名称を、プログラムで(機械)処理しやすい形式で記述するための仕様

脆弱性対策情報データベースJVN iPediaが提供するMyJVN APIを利用することで、脆弱性対策情報に紐づく製品識別子CPEを取得可能

cpe:/a:ipa:myjvn

cpe:/{種別}:{ベンダ}:{製品}:{バージョン}

:{アップデート}:{エディション}:{言語}

種別：h=ハードウェア、o=OS、a=アプリケーション

自動化(機械化)処理基盤の潮流

IT資産管理と脆弱性管理の連携

- インストール状況と脆弱性との紐付け：国内での取り組み
- 製品識別子CPEを用いた製品の紐付け
  インストール状況を把握できるSAMACソフトウエア辞書に連携用項目に製品識別子CPEを用いた製品を追記

SAMAC ソフトウエア辞書

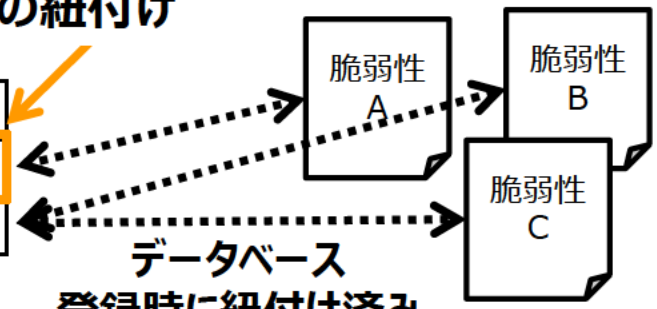
SAMACソフトウエア辞書の既存登録項目(9項目)				連携用項目(1項目)
sw_id	sw_vendor	sw_name	その他項目	CPE
・・・	・・・	Adobe Acrobat 8.2.0 Professional	・・・	cpe:/a:adobe:acrobat
・・・	・・・	Adobe Acrobat 8.2.0 Standard	・・・	cpe:/a:adobe:acrobat
・・・	・・・	Adobe Acrobat 8.2.1 - CPSID_50570	・・・	cpe:/a:adobe:acrobat
・・・	・・・	Adobe Acrobat 8.2.1 Professional	・・・	cpe:/a:adobe:acrobat
・・・	・・・	Adobe Acrobat 8.2.1 Standard	・・・	cpe:/a:adobe:acrobat
・・・	・・・	Adobe Acrobat 9.3.0 - CPSID_52073	・・・	cpe:/a:adobe:acrobat

ソフトウエアの脆弱性データベース

製品識別子CPEを用いた製品の紐付け

ソフトウエア名	CPE
Adobe Acrobat	cpe:/a:adobe:acrobat
製品Y	cpe:/a:y:yyy

JVN製品データベース



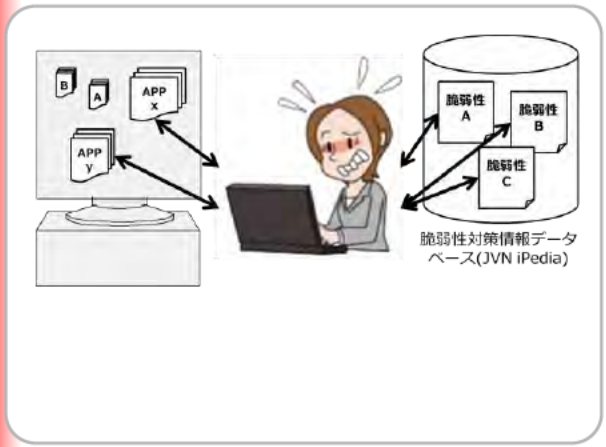
データベース 登録時に紐付け済み

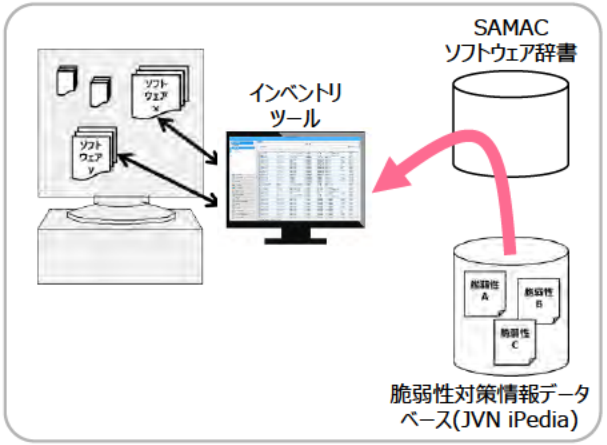
自動化(機械化)処理基盤の潮流

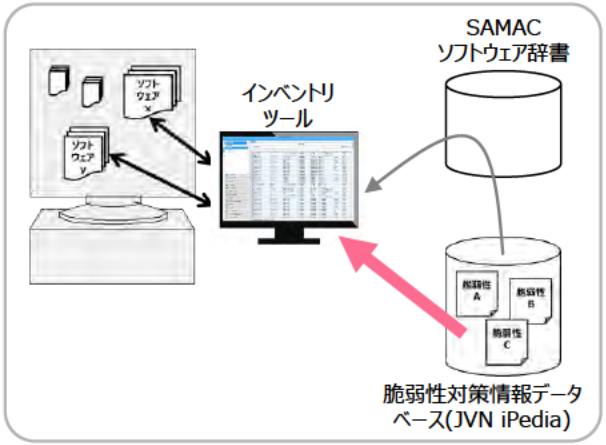
IT資産管理と脆弱性管理の連携

IPA

現在	① 現在の取組み『JVN iPediaの活用』	② 将来的な連携『JVN iPedia SWID連携』







人が脆弱性情報を検索し、各ソフトウェアを確認する。	SAMACソフトウェア辞書に脆弱性対策情報を連携する。	ソフトウェア識別タグが正式化すると直接インベントリツールと連携できる。
『全部なんて調べきれない！』	『辞書登録にあるものは、ツールの中で視覚的に分かる！』	『辞書登録に関わらず、SWIDがあればツールの中で視覚的に分かる！』
『いつの間にか脆弱性のあるソフトが入ってた！』	『でも直近の新しい情報は反映が遅くなってしまう･･･。』	『かなりリアルタイムに情報が反映される！』

脆弱性対策の動向と
効果的な収集に向けて

独立行政法人 情報処理推進機構(IPA)
セキュリティセンター
2019年03月28日