



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  
寺田真敏  
2026年03月26日

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、いろいろな取り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、IPAの取り組みを紹介するとともに、脆弱性対策基盤を実現する技術仕様を紹介します。

1. トピック[1]
2. 情報収集に役立つ技術仕様
3. JVN脆弱性対策機械処理基盤
4. トピック[2]

## トピック[1]

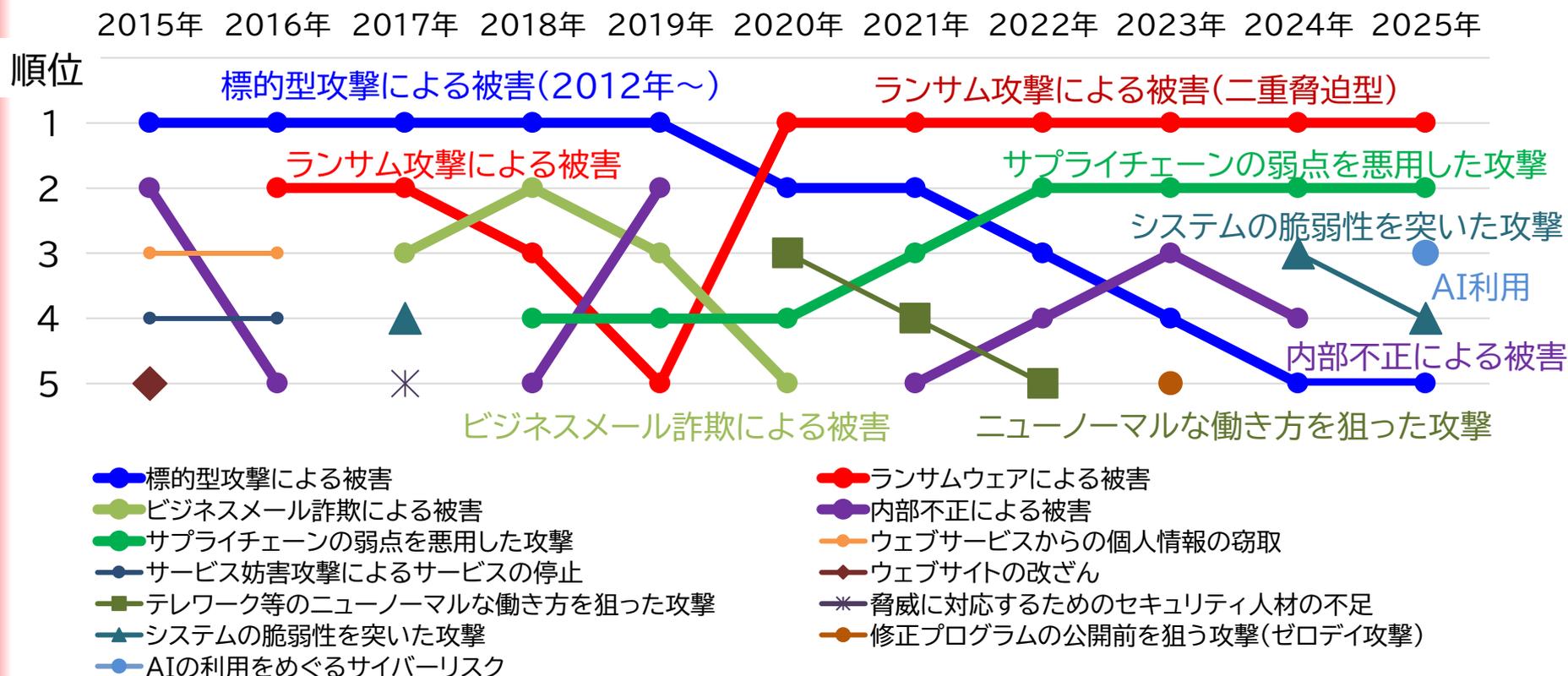


## 情報セキュリティ10大脅威

# 情報セキュリティ10大脅威 2026



## ● 社会的に影響が大きいと考える情報セキュリティにおける事案から選出



[出典] 情報セキュリティ10大脅威 2026  
<https://www.ipa.go.jp/security/10threats/10threats2026.html>

# 情報セキュリティ10大脅威 2026



- 社会的に影響が大きいと考える情報セキュリティにおける事案から選出

個人
インターネット上のサービスからの個人情報の窃取
インターネット上のサービスへの不正ログイン
インターネットバンキングの不正利用
クレジットカード情報の不正利用
サポート詐欺(偽警告)による金銭被害
スマホ決済の不正利用
ネット上の誹謗・中傷・デマ
フィッシングによる個人情報等の詐取
不正アプリによるスマートフォン利用者への被害
メールやSNS等を使った脅迫・詐欺の手口による金銭要求

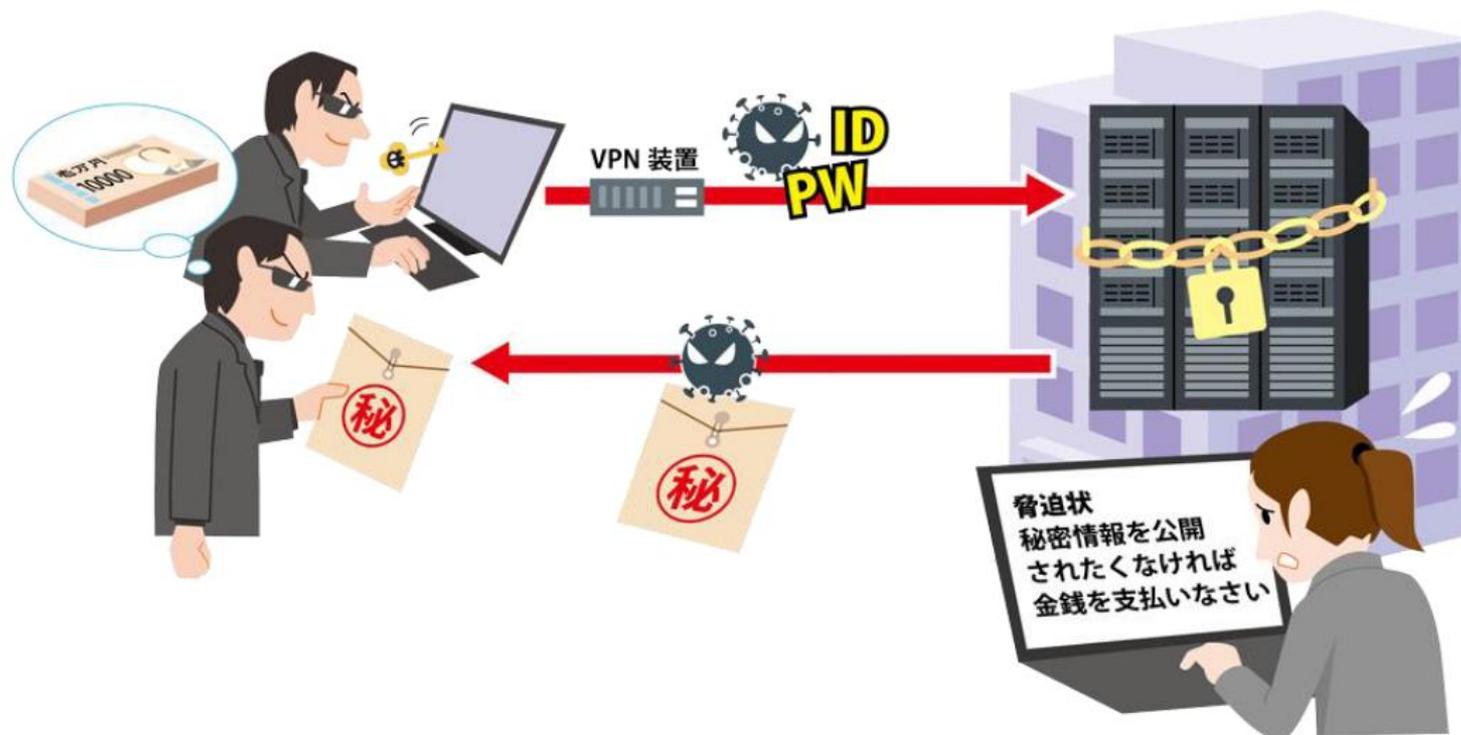
順位	組織
1位	ランサム攻撃による被害
2位	サプライチェーンや委託先を狙った攻撃
3位	AIの利用をめぐるサイバーリスク
4位	システムの脆弱性を悪用した攻撃
5位	機密情報を狙った標的型攻撃
6位	地政学的リスクに起因するサイバー攻撃(情報戦を含む)
7位	内部不正による情報漏えい等
8位	リモートワーク等の環境や仕組みを狙った攻撃
9位	DDoS攻撃(分散型サービス妨害攻撃)
10位	ビジネスメール詐欺

[出典] 情報セキュリティ10大脅威 2026  
<https://www.ipa.go.jp/security/10threats/10threats2026.html>

# 情報セキュリティ10大脅威 2026

## ランサム攻撃による被害

- 変わらず続く脅威、リスクを見つけて対策を
  - 二重脅迫: データの暗号化、窃取情報の暴露
  - 四重脅迫: データの暗号化、窃取情報の暴露、DDoS攻撃(予告)、攻撃を受けていることの暴露(顧客やビジネスパートナーからの信用失墜)



# ランサム攻撃による被害 報告件数の推移

## ● 企業・団体等におけるランサムウェア被害の報告件数の推移



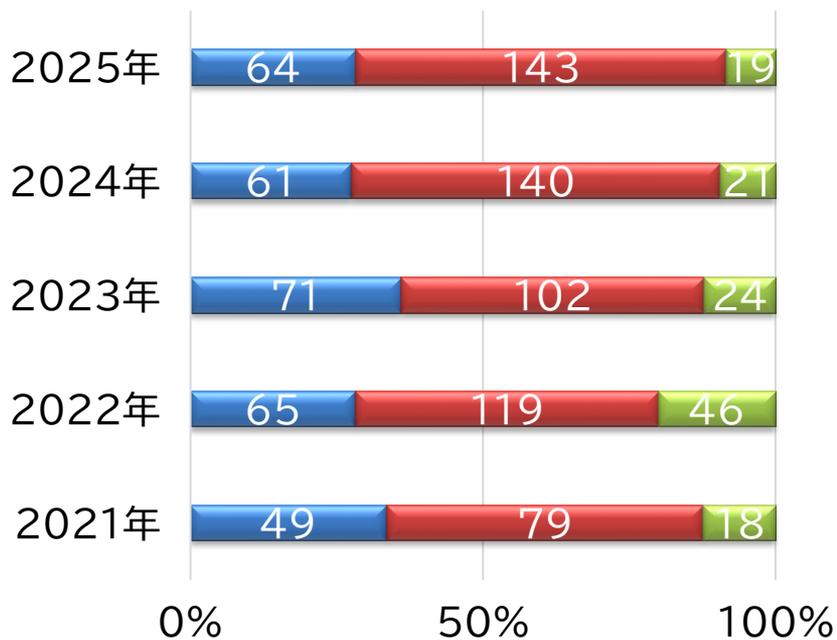
## ● 2021年～2025年のランサムウェア被害の特徴

- 二重恐喝(ダブルエクストーション)による被害が多くを占める
- 暗号資産による金銭の要求が多くを占める
- 企業・団体等の規模や業種を問わず被害が発生
- 感染経路は、テレワークにも利用される機器等の脆弱性や漏えい・強度の弱い認証情報等を利用して侵入したと考えられるものが大半を占めている

# ランサム攻撃による被害 被害の状況

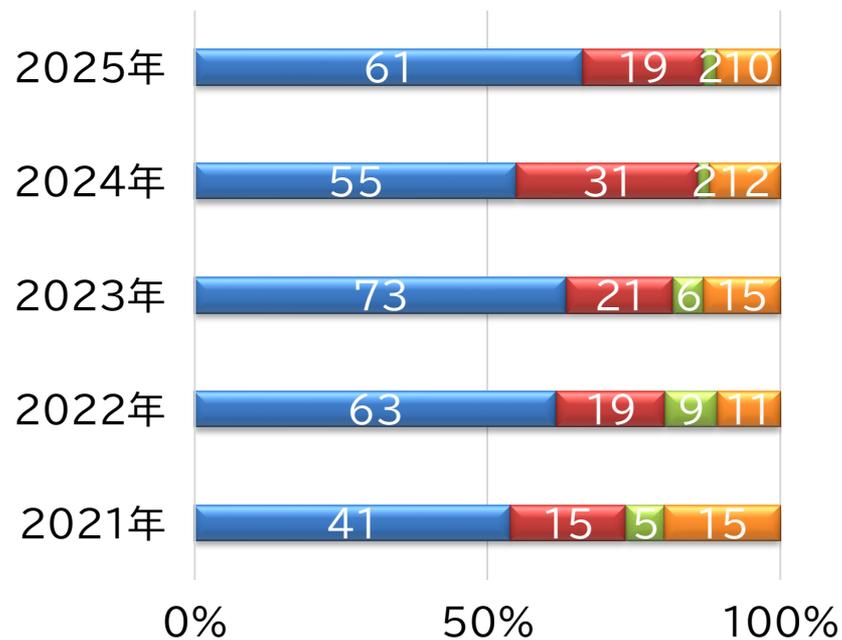
## ● 被害企業・団体等の規模別

- 大企業
- 中小企業
- 団体等



## ● 手口別

- VPN機器からの侵入
- リモートデスクトップからの侵入
- 不審メールやその添付ファイル
- その他

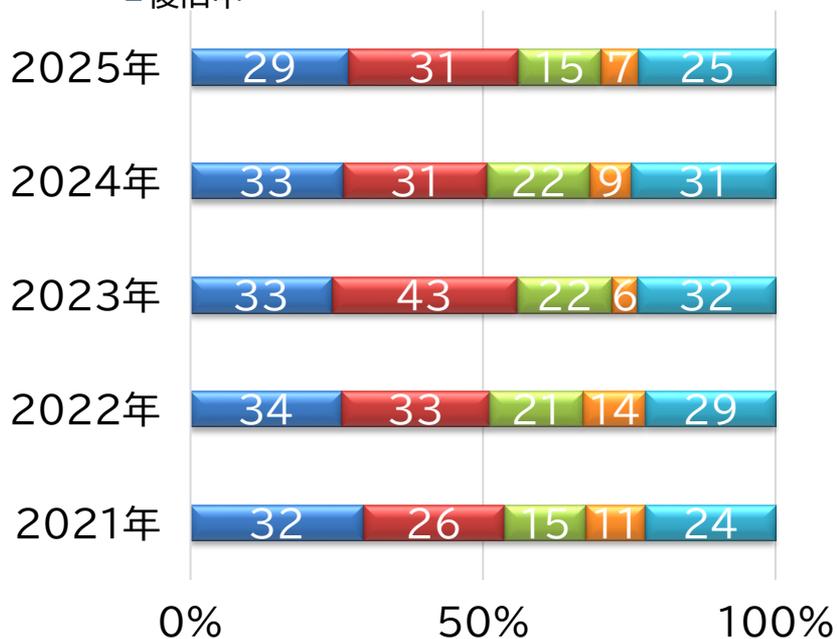


[出典] 令和7年におけるサイバー空間をめぐる脅威の情勢等について  
[https://www.npa.go.jp/publications/statistics/cybersecurity/data/R7/R07\\_cyber\\_jousei.pdf](https://www.npa.go.jp/publications/statistics/cybersecurity/data/R7/R07_cyber_jousei.pdf)

# ランサム攻撃による被害 被害の状況

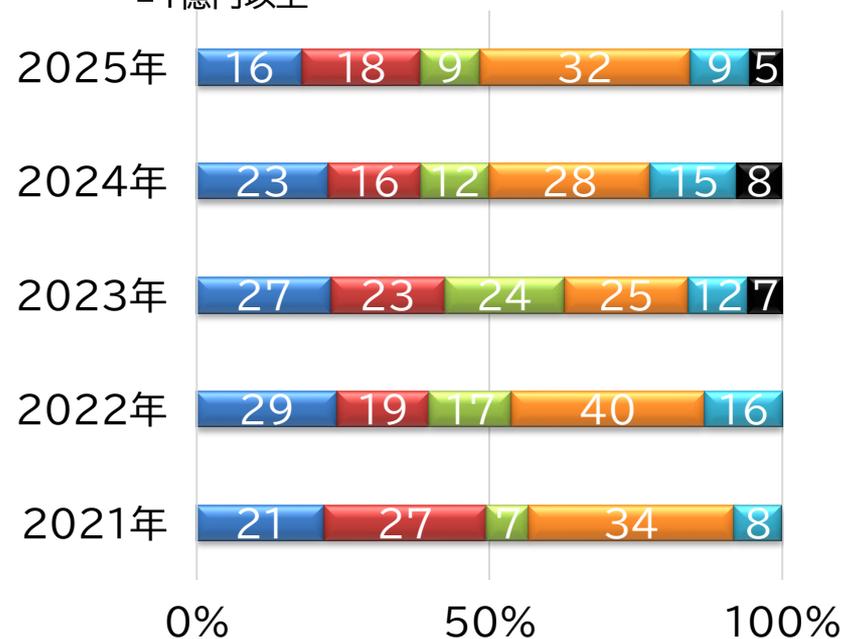
## ● 復旧に要した期間

- 即時～1週間
- 1週間～1ヶ月
- 1ヶ月～2ヶ月
- 2ヶ月以上
- 復旧中



## ● 調査・復旧費用の総額

- 100万円未満
- 100万円以上～500万円未満
- 500万円以上～1,000万円未満
- 1,000万円以上～5,000万円未満
- 5,000万円以上～1億円未満
- 1億円以上



[出典] 令和7年におけるサイバー空間をめぐる脅威の情勢等について  
[https://www.npa.go.jp/publications/statistics/cybersecurity/data/R7/R07\\_cyber\\_jousei.pdf](https://www.npa.go.jp/publications/statistics/cybersecurity/data/R7/R07_cyber_jousei.pdf)

# ランサム攻撃による被害

## 脅迫手段の高度化

- ランサムウェアは、「ランサム(Ransom=身代金)」と「ウェア(Software)」をつなげた造語で、パソコン内のファイルを人質にとる不正プログラムの総称である。

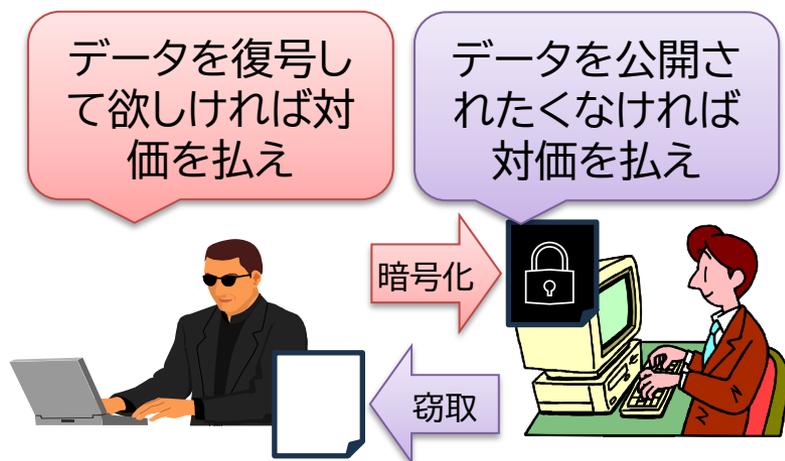
年代	区分	脅迫の概要
2013年	データの暗号化	感染したコンピュータ内のファイルやハードディスクなどのデータを暗号化し、復号したければと脅迫して金銭を要求する。
2019年	窃取情報の暴露 二重脅迫	感染したコンピュータ内からデータを窃取し、窃取データを公開すると脅迫して金銭を要求する。
2020年	DDoS攻撃 三重脅迫	交渉を開始するまで、DDoS(Distributed Denial of Service;通信過負荷状態を発生させる)攻撃を使ってさらにプレッシャーをかける。
2022年	攻撃を受けている ことの暴露 四重脅迫	顧客やビジネスパートナーからの信用失墜を狙い、窃取データにある連絡先に状況を通知することでさらにプレッシャーをかける。

# ランサム攻撃による被害 脅迫手段の高度化

- ノーウェアランサム攻撃 = 窃取情報の暴露
  - 組織内ネットワークへの侵入には、テレワークにも利用される機器等の脆弱性や強度の弱い認証情報等を利用
  - さらに侵入後には、窃取したデータを公開されたくなければと脅迫して金銭を要求

二重脅迫型ランサムウェア攻撃  
= データの暗号化 + 窃取情報の暴露

ノーウェアランサム攻撃  
= 窃取情報の暴露



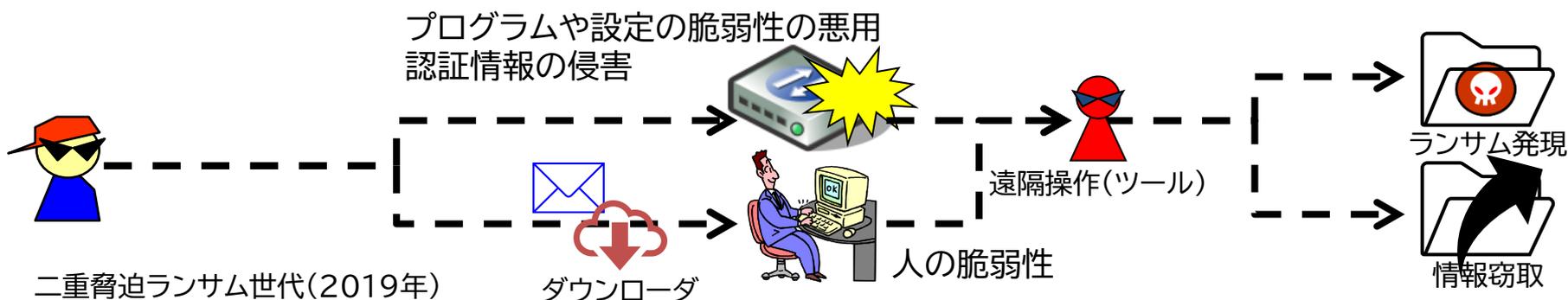
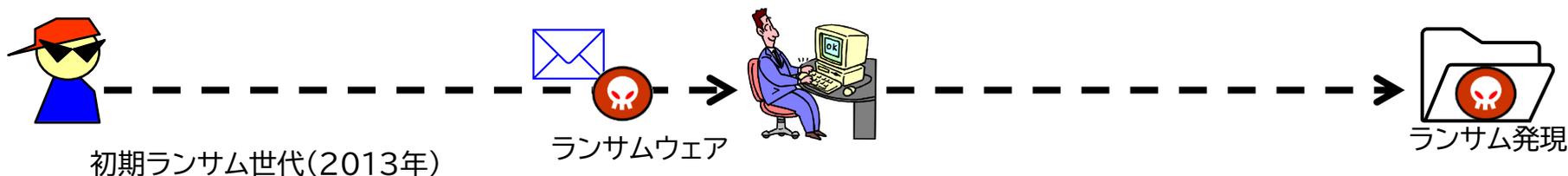
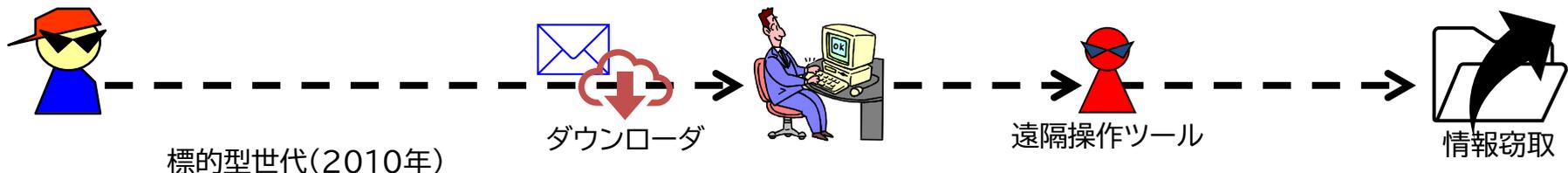
## トピック[1]



補助資料  
～ランサム攻撃～

# ランサム攻撃

標的型 → 初期ランサム → 二重脅迫ランサム世代



# ランサム攻撃

## 攻撃者側の体制

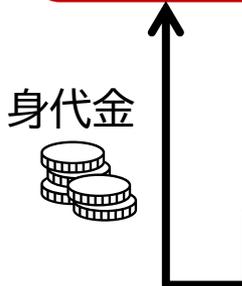
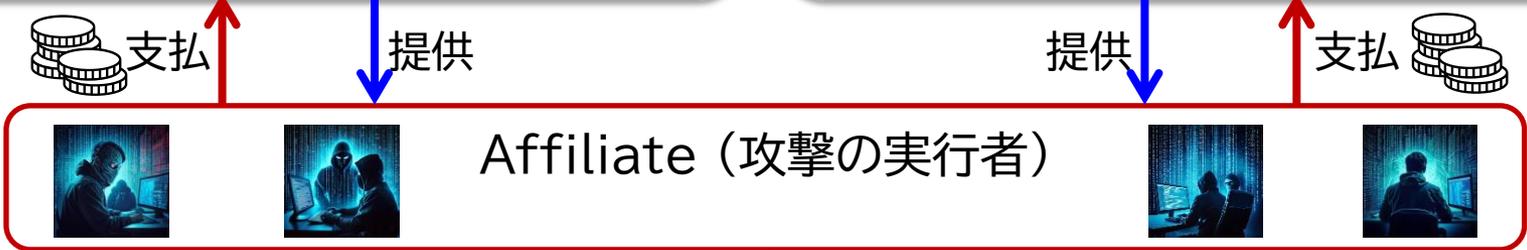


**RaaS (Ransom as a Service)**  
ランサム攻撃環境の提供

- ランサムウェア
- リークサイト
- 被害者との連絡手段
- 攻撃方法

**IAB (Initial Access Broker)**  
ログイン情報の売買

- 認証情報
- 攻撃ツール



**被害者・被害組織**

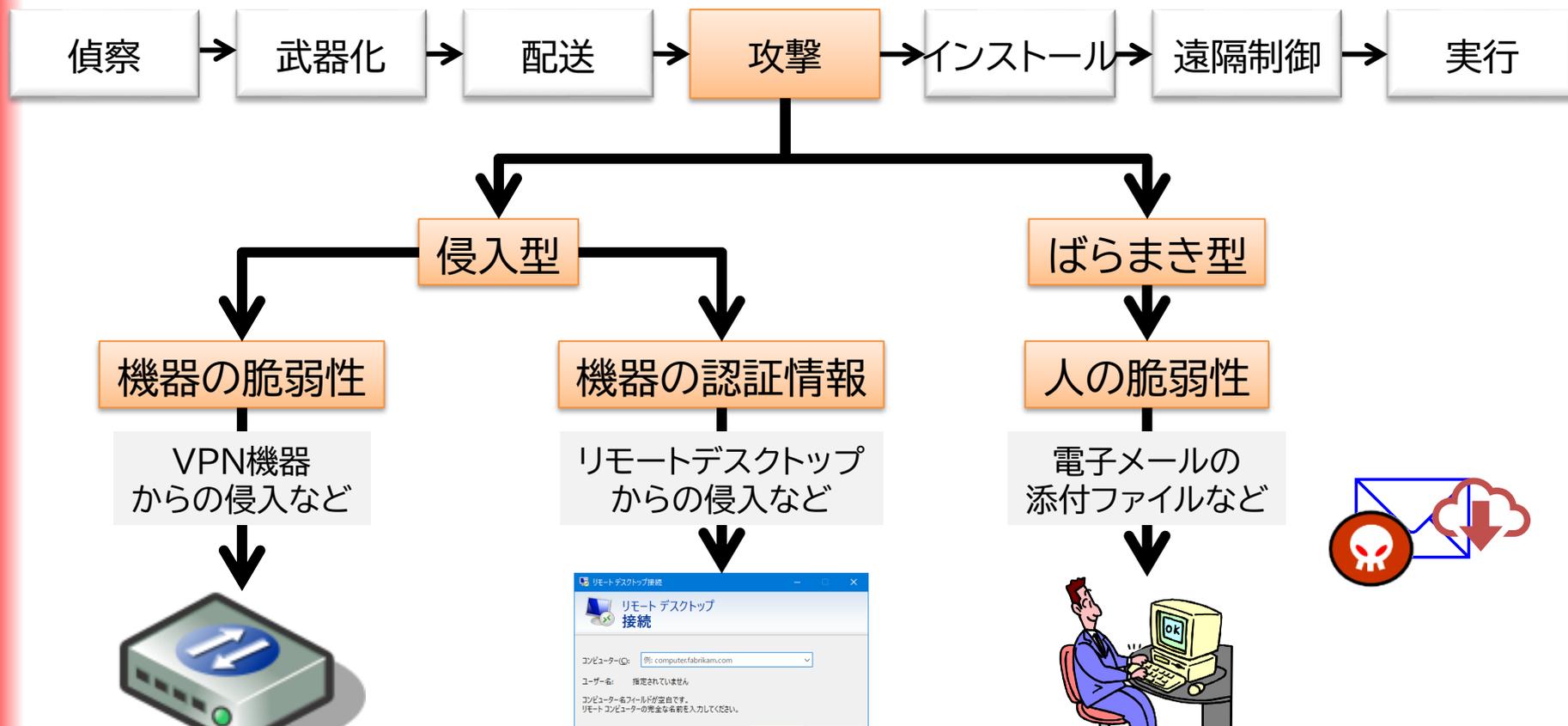
- データの暗号化 → 事業継続妨害
- 情報窃取

- 攻撃**
- データの暗号化
  - 窃取情報の暴露
  - DDoS攻撃
  - 攻撃を受けていることの暴露
  - ノーウェアランサム攻撃

# ランサム攻撃 侵入経路



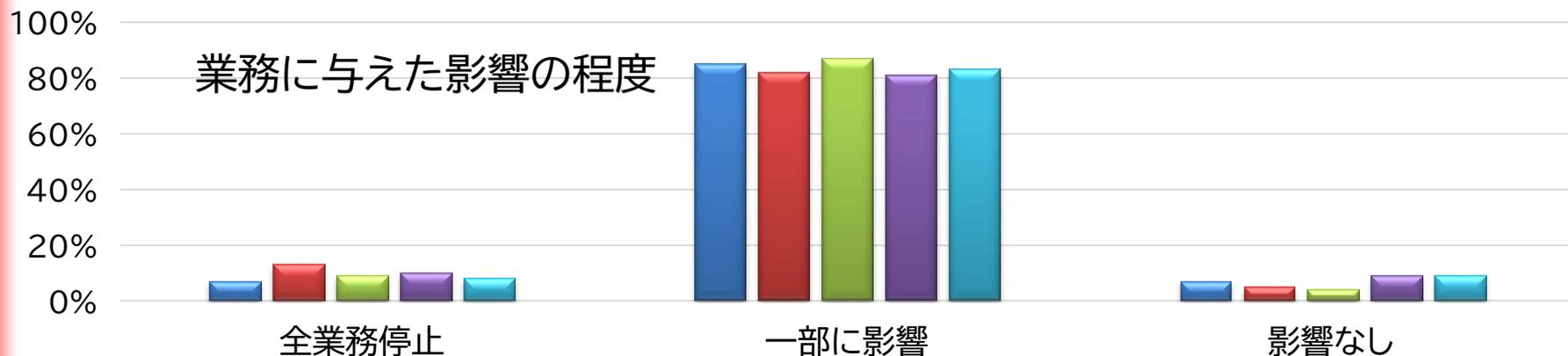
# ランサム攻撃 侵入経路



ソフォス「ランサムウェアの現状 2024年版」の報告によると、パッチが適用されていない脆弱性の悪用から始まった攻撃を受けた組織は、認証情報の侵害で始まった攻撃を受けた組織よりも、さらに深刻な被害に

# ランサム攻撃

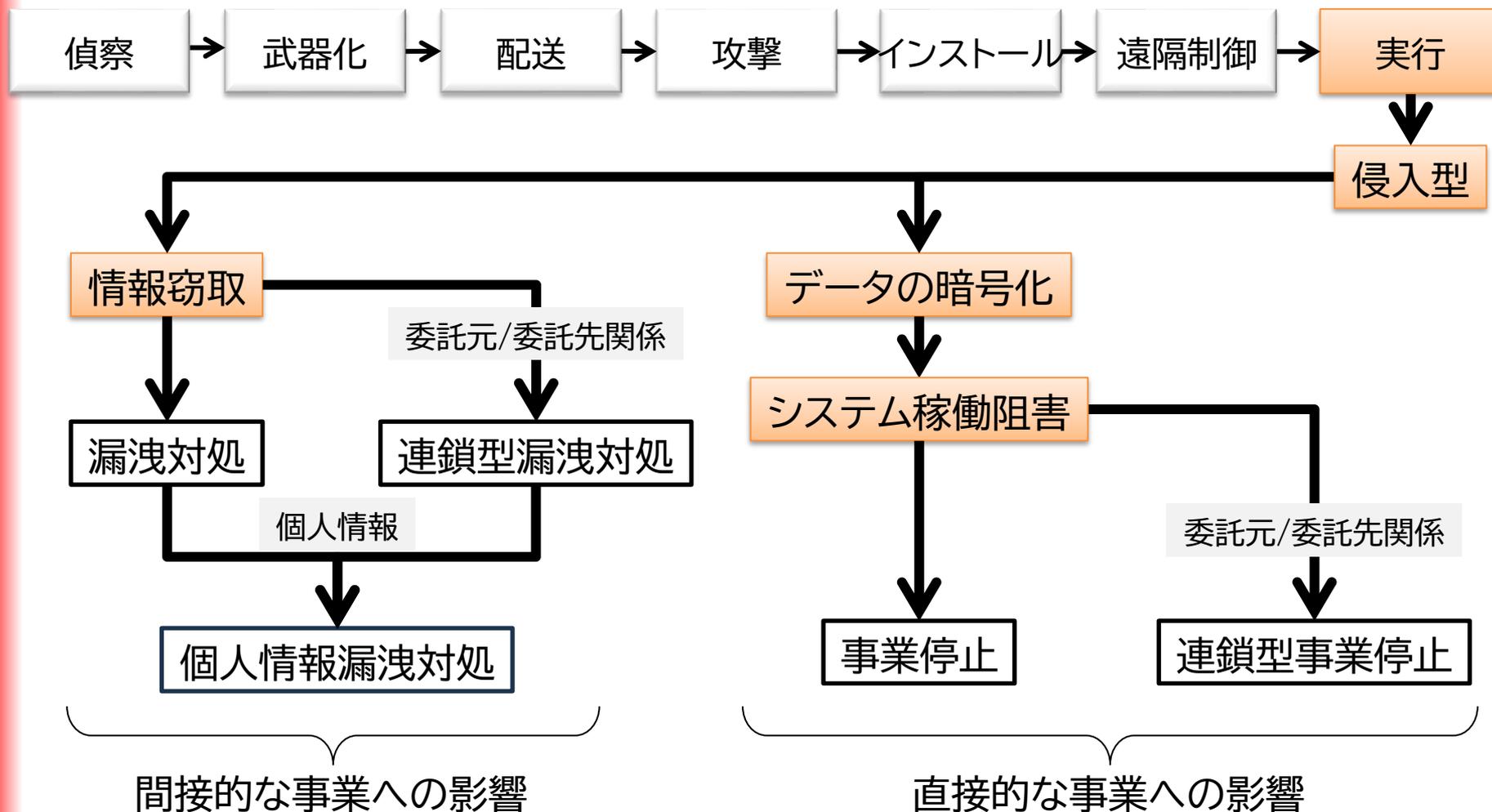
復旧に要した期間、業務に与えた影響の程度



[出典] [R07 cyber jousei.pdf](#)

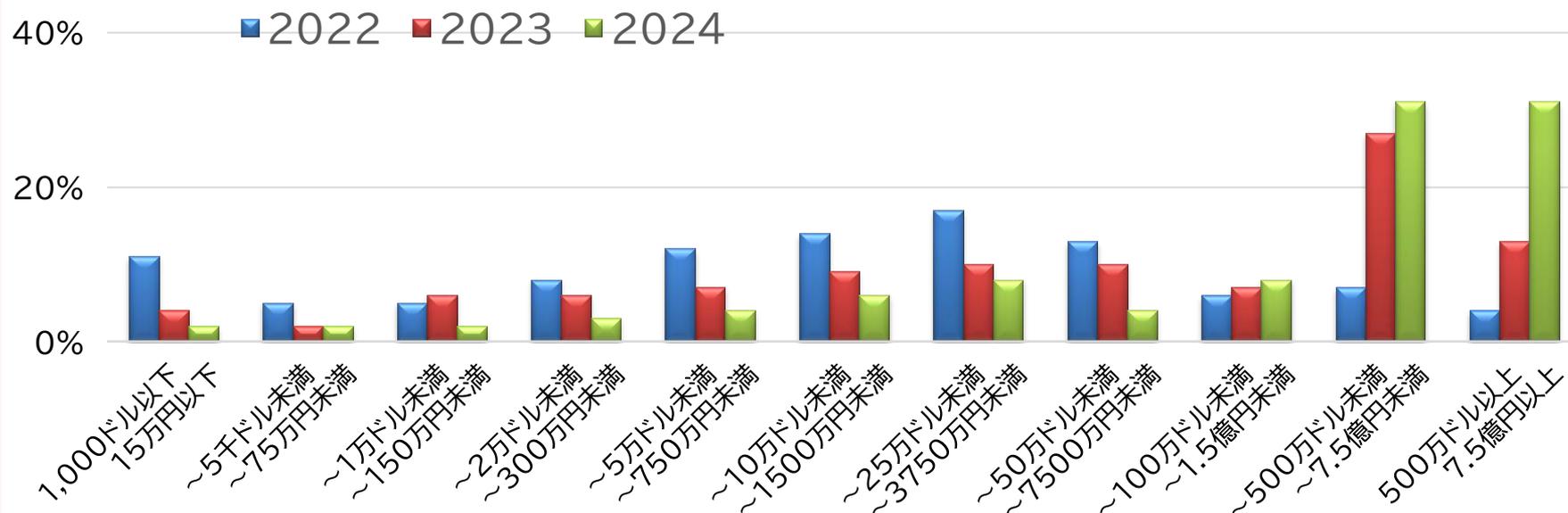
# ランサム攻撃

## 侵入に伴う被害の発生形態



# ランサム攻撃

## ランサムウェア身代金の支払額



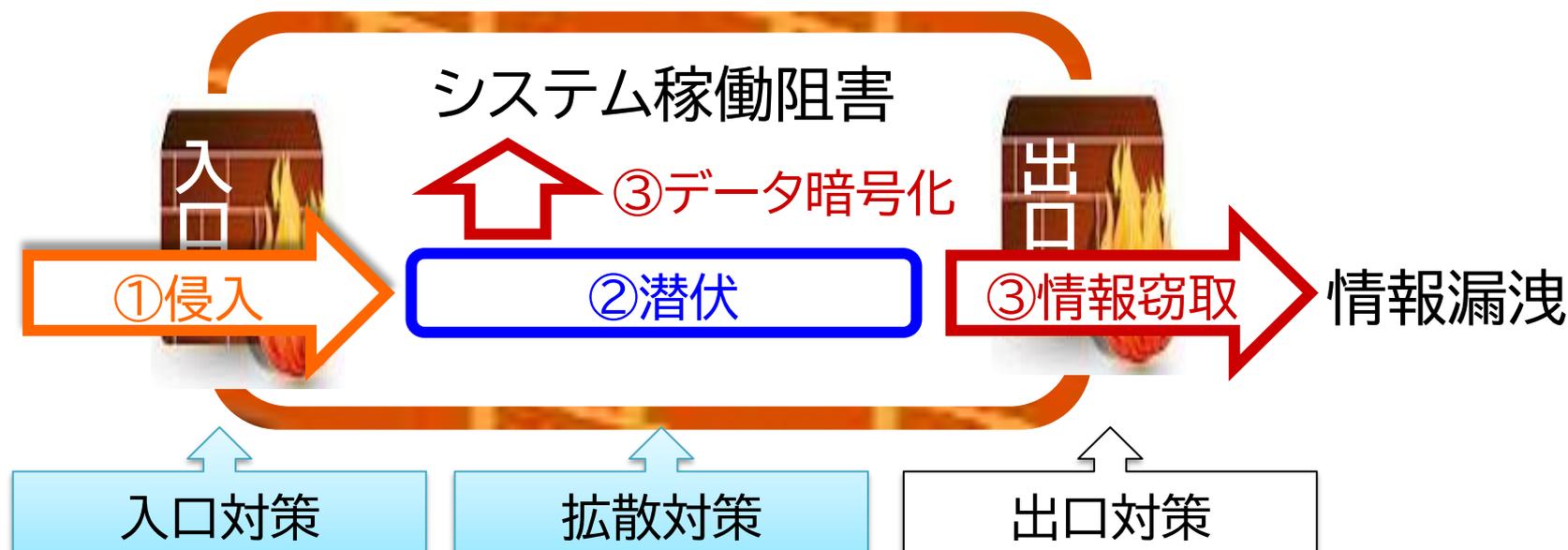
- 2025年の調査によれば、2024年から、
  - 身代金要求額の平均(中央値):200万ドルから132万ドルへ(34%減少)
  - 身代金支払額の平均(中央値):200万ドルから100万ドルへ(50%減少)
  - 身代金を支払う割合:56%から49%へ(過去6年間で2番目に高い割合)

[出典] [sophos-state-of-ransomware-2024-wpia.pdf](#)  
[sophos-state-of-ransomware-2025-ja.pdf](#)

# ランサム攻撃 対策アプローチ



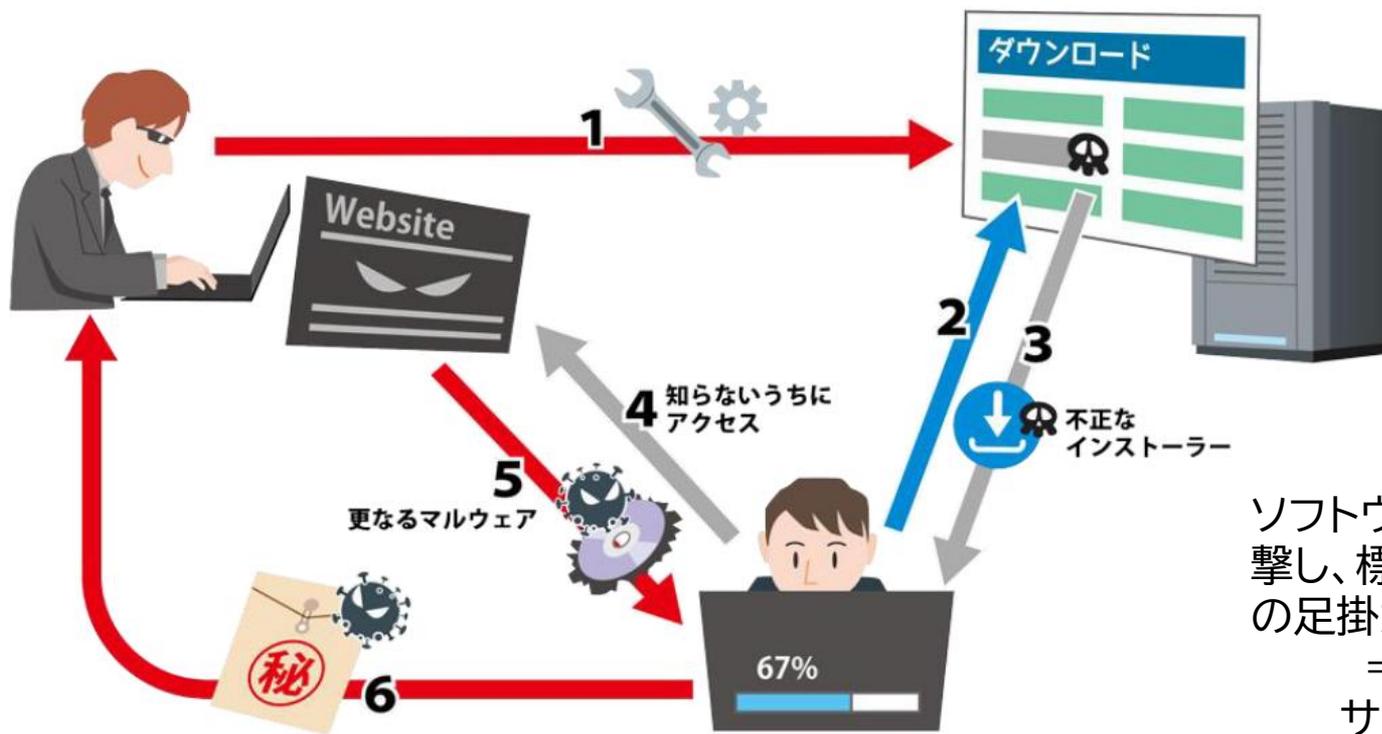
- 攻撃を防げず、侵入を許してしまう可能性は高まっている。
  - 機器の脆弱性・・・様々なソフトウェアの脆弱性が狙われる。
  - 機器の認証情報・・・認証情報を悪用されてしまう。
  - 人の脆弱性・・・不審な添付ファイルを開いたり、リンクをクリックしてしまう。



# 情報セキュリティ10大脅威 2026

## サプライチェーンや委託先を狙った攻撃

- 委託先が狙われる！ 関係組織も視野に入れたリスク管理を
  - 取引先や委託先が保有する機密情報を狙う
  - ソフトウェア開発元や企業システムの運用・監視等を請け負う事業者等を攻撃し、標的を攻撃するための足掛かりとする



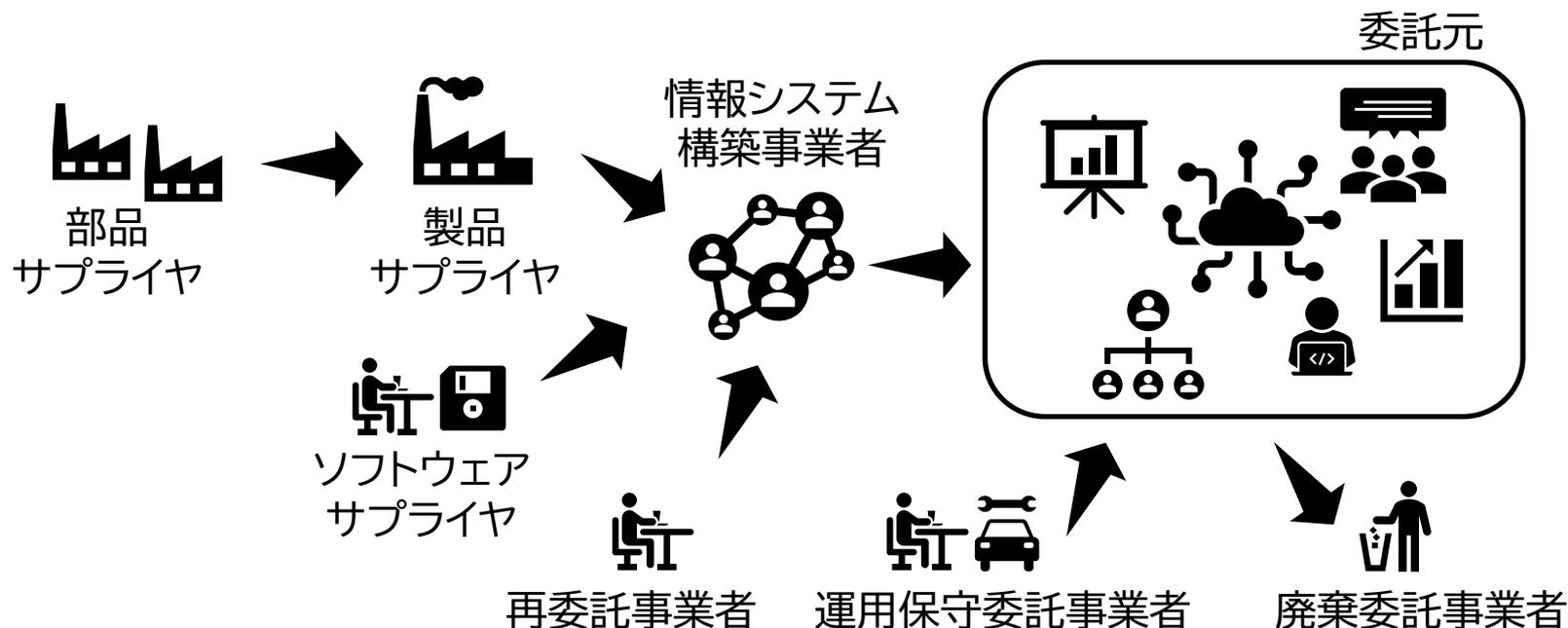
ソフトウェア開発元等を攻撃し、標的を攻撃するための足掛かりとする

⇒ ソフトウェア  
サプライチェーン

# サプライチェーンや委託先を狙った攻撃

## サプライチェーンとは

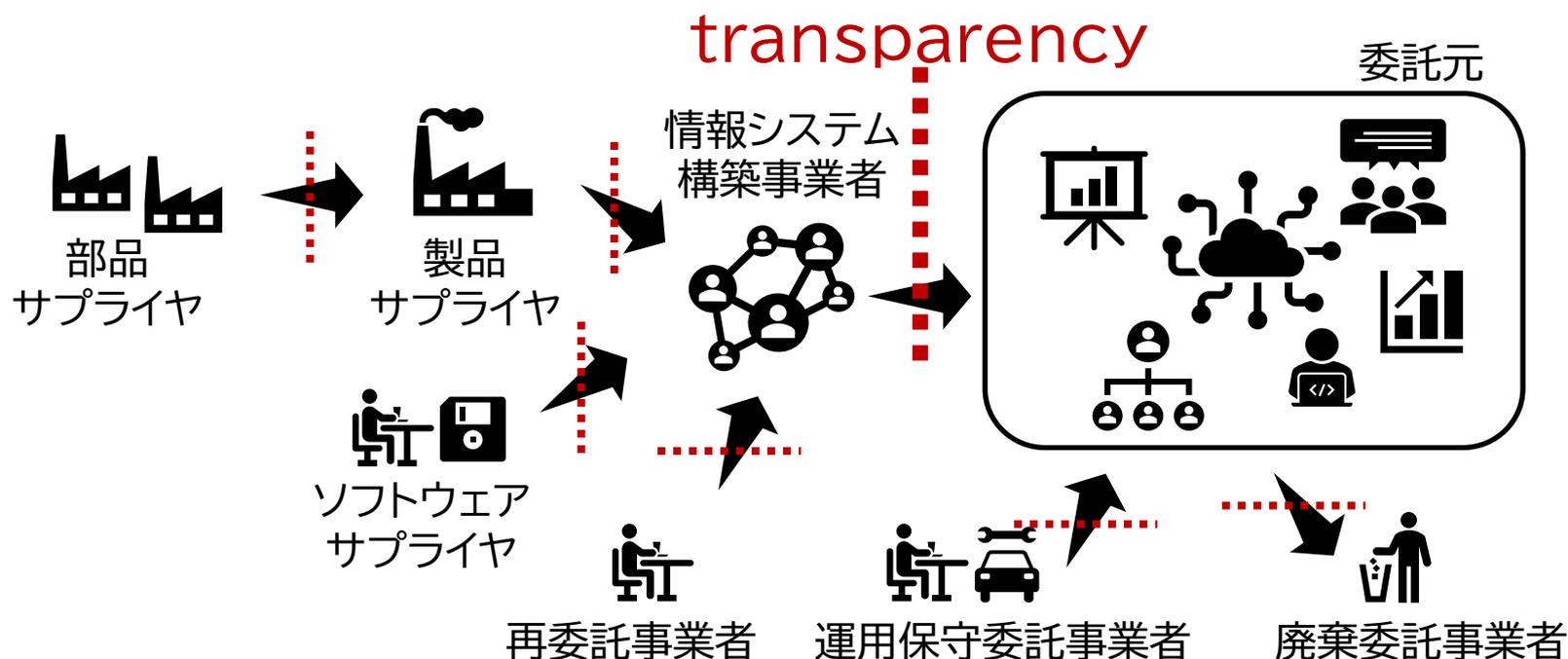
- ITにおけるシステム開発やサービス提供/利用に関する連鎖
- ビジネスパートナーや委託先企業も含めたサプライチェーン全体でのセキュリティ対策の必要性の高まり
  - サイバーセキュリティ経営ガイドラインv2.0 (2017年)
  - 米国立標準技術研究所(NIST) サイバーセキュリティフレームワークv1.1 (2018年)



# サプライチェーンや委託先を狙った攻撃

## サプライチェーンとは

- (ソフトウェア) サプライチェーンのセキュリティにおいては、
  - 「見える」セキュリティ、「見せる」セキュリティが求められている。
    - 委託元の視点 「見える」セキュリティ
    - 情報システム構築事業者の視点 「見せる」セキュリティ
  - 共通の仕様や言語でコミュニケーションする必要がある。



# セキュア・バイ・デザインの原則



## ● 2023年以降急速に国際署名文書が増えている

年月日	イベント
2026年03月05日	AI・機械学習のサプライチェーンリスクと緩和策
2025年10月23日	最新の防御可能なアーキテクチャのための基礎
2025年09月04日	サイバーセキュリティのためのソフトウェア部品表(SBOM)の共有ビジョン
2025年08月26日	暗号鍵及びシークレットの管理-実践者向けガイダンス
2025年05月27日	SIEM 及び SOAR プラットフォームに関するガイダンス
2025年02月04日	エッジデバイスのための緩和戦略
2024年12月04日	通信インフラの可視性向上と堅牢化に関するガイダンス
2024年10月02日	OT サイバーセキュリティの原則
2024年08月22日	イベントログと脅威検知のベストプラクティス
2024年05月15日	人権保護や民主主義の推進に関与する組織や個人のためのガイダンス
2024年01月24日	AI使用に関するガイダンス
2023年11月28日	セキュアAIシステム開発ガイドライン
2023年10月16日	セキュア・バイ・デザインの原則とアプローチに関するガイダンス

[出典] 国家サイバー統括室 新着情報  
<https://www.cyber.go.jp/news/list/index.html>

■:米CISA ■:豪ACSC

# セキュア・バイ・デザインの原則

セキュア・バイ・デザインの原則とアプローチに関するガイダンス



報道資料



令和5年10月17日

内閣官房内閣サイバーセキュリティセンター (NISC)

セキュアバイデザイン・セキュアバイデフォルト原則について

- 2023年10月16日、CISA(米国サイバーセキュリティ・インフラセキュリティ庁)が、日本のNISC(内閣サイバーセキュリティセンター)などのサイバーセキュリティ組織と共同で公開
  - セキュア・バイ・デザインの原則とアプローチに関するガイダンス  
Secure-by-Design - Shifting the Balance of Cybersecurity Risk:  
Principles and Approaches for Secure by Design Software

# セキュア・バイ・デザインの原則

## セキュア・バイ・デザインの原則とアプローチに関するガイダンス



- セキュア・バイ・デザイン

悪意ある者が端末機器、データ、インフラに不正にアクセスできないように、設計・開発段階から、合理的に保護されている形で製品を作ること。そのために、セキュリティ対策を後から「追加」するのではなく、製品設計や開発プロセスにあらかじめ組み込んでおくこと。

- セキュア・バイ・デフォルト

顧客にリスクを生じさせることのない初期状態で製品を出荷すること。そのために、基本的なセキュリティが確保できているというセキュアな初期設定を基準とすること。

- セキュア・バイ・デザインの原則とアプローチに関するガイダンス

数百の個人、企業、非営利団体から受けたフィードバックを受け、特にソフトウェア作成業者に対する3つの原則、

原則1:顧客のセキュリティの結果に責任を持つ

原則2:徹底した透明性と説明責任を負う

原則3:トップ主導での実施

を具体的に説明し、ソフトウェア作成業者がこれらの原則を顧客や一般の人々に対し、どのように示すことができるかに焦点を当てた提言となっている。

# セキュア・バイ・デザインの原則

## 原則1:顧客のセキュリティの結果に責任を持つ



### ● セキュア・バイ・デフォルトのプラクティス

推奨事項	説明
初期パスワードを無くす	<p>デフォルトパスワードで運用し続けることができないようにすること。安易なパスワードを設定できないようにすること。</p> <p>多くのサイバー攻撃の原因である「脆弱なパスワード」の利用については、そもそも脆弱な状態にできることが問題であり、設定できないようにしておくことが求められている。</p>
実地テストの実施	<p>顧客先の実際の利用環境や利用方法を踏まえたテストを実施すること。特に、セキュリティに重点を置いたユーザ調査を実施するなど、製品セキュリティが現場でどのように機能してるかを把握し、改善すべき箇所を特定して改良すること。</p> <p>セキュリティ強化ガイドを正しく実施しているか、製品の既存のセキュリティ機能は期待されたとおり機能するか、それらの機能は実際の攻撃を防げるか、被害の可能性をより低減できる機能は何か、などを確認することが求められている。</p>
セキュリティ強化ガイドのサイズを小さくする	<p>セキュリティ強化ガイドではなく、セキュリティ対策の実施プロセスを簡素化するためのツールや自動化手段を提供することで、製品が導入された顧客環境のセキュリティ確保を考えること。</p> <p>強化ガイドに沿って、管理者がセキュリティ対策を実行することに頼らないことが求められている。</p>

# セキュア・バイ・デザインの原則

## 原則1:顧客のセキュリティの結果に責任を持つ



### ● セキュア・バイ・デフォルトのプラクティス(続き)

推奨事項	説明
安全でないレガシー機能の使用を積極的にやめさせる	アップグレードは、後方互換性よりもセキュリティ確保を優先すること。 リスクを軽減するというアップグレードの方向に積極的に誘導することが求められている。
注意喚起アラートを導入する	ユーザや管理者がセキュリティ面で危険な使い方や状態にある場合には、適時かつ繰り返し警告すること。 製品の仕様として危険な使い方にある場合には、見える化することが求められている。
セキュアな構成テンプレートを作成する	組織のリスク選好度に応じて、安全な構成を設定できるテンプレートを予め作成しておくこと。 安全な環境を利用しやすくすることが求められている。

# セキュア・バイ・デザインの原則

## 原則1:顧客のセキュリティの結果に責任を持つ



### ● セキュアな製品開発のプラクティス

推奨事項	説明
SDLCフレームワークへの適合性を文書にする	<p>採用しているSDLC(セキュアソフトウェア開発ライフサイクル)の枠組みと適合性を文書として示すこと。</p> <p>安全なソフトウェア開発を実施していることが求められている。米国内では、NISTのSSDF(セキュアソフトウェア開発フレームワーク)の利用を推奨している。</p>
サイバーパフォーマンス目標または同等の適合性を文書にする	<p>CISA(米国サイバーセキュリティ・インフラセキュリティ庁)クロスセクタサイバーパフォーマンス目標やNIST(米国標準技術研究所)サイバーセキュリティフレームワークなどとの適合性を文書として示すこと。</p> <p>攻撃者から企業と開発環境を保護していることが求められている。</p>
脆弱性管理	<p>脆弱性の根本原因を排除することを考えた脆弱性管理をすること。</p> <p>ソフトウェアの品質計画、品質管理、品質改善、品質測定に関する措置を講じることが求められている。</p>
オープンソースのソフトウェアを責任持って利用する	<p>オープンソースのソフトウェアを利活用する場合、セキュリティ確保に責任を持つこと。</p> <p>使用するオープンソースの開発や維持を支援することが求められている。</p>

# セキュア・バイ・デザインの原則

## 原則1:顧客のセキュリティの結果に責任を持つ



### ● セキュアな製品開発のプラクティス(続き)

推奨事項	説明
開発者のためのセキュアデフォルトを提供する	開発者に提供する環境はデフォルトの使い方が安全になるようにすること。
セキュリティを理解するソフトウェア開発者を育成する	自組織のセキュリティ開発者をセキュアコーディングのベストプラクティスで訓練し、セキュリティを理解させること。
SIEMとSOARとの統合をテストする	セキュリティインシデント調査において役立つログを出力するために、SIEM(セキュリティインシデントイベント管理)やSOAR(セキュリティ統合調整・自動化・対応機能)と連携したテストを実施すること。  インシデント対応チームが実際のセキュリティインシデント調査あるいはセキュリティインシデントが疑われる調査において役立つログを出力することが求められている。
ゼロトラストアーキテクチャと整合させる	製品導入ガイドがNIST(米国標準技術研究所)やCISA(米国サイバーセキュリティ・インフラセキュリティ庁)のゼロトラストアーキテクチャのモデルと整合していること。  顧客の環境にゼロトラストアーキテクチャの原則を取り入れることを求めている。

# セキュア・バイ・デザインの原則

## 原則1:顧客のセキュリティの結果に責任を持つ



### ● ビジネスプラクティス

推奨事項	説明
追加費用なしでログ記録機能を提供する	追加料金なしでセキュリティ関連ログを生成し記録できるようにすること。さらに、デフォルトでセキュリティ関連ログに記録すること。
隠れた負担をなくす	セキュリティは価格設定のあるオプションではなく、顧客の権利とみなし、セキュリティやプライバシー機能、または統合について、追加料金を課さないこと。  SSO(シングルサインオン)やMFA(多要素認証)の導入など強固なセキュリティ態勢の実現を妨げている要因を排除することが求められている。
オープンスタンダードを採用する	オープンスタンダードが利用できる場合、独自仕様のプロトコルを使用せずにオープンスタンダードを利用して実装すること。  仕様についても、セキュアな品質であること、セキュアな品質を維持できることが求められている。
アップグレードツールを提供する	安全なネットワーク接続など、最新かつより安全な機能導入などにあたっては、不確実性やリスクの軽減を支援するツールを提供すること。  最新バージョンの導入などのアップグレードをしやすい環境が求められている。

# セキュア・バイ・デザインの原則

## 原則2: 徹底的な透明性と説明責任を果たす



### ● セキュア・バイ・デフォルトのプラクティス

推奨事項	説明
セキュリティ関連の総合的な統計や傾向を公表する	(良い慣習が見える化するため)顧客や管理者のMFA(多要素認証)の採用状況やレガシーなプロトコルの使用などを示すこと。  強固なセキュリティ態勢の実現を妨げている要因を排除するための企業努力の見える化が求められている。
パッチ適用の統計を公表する	製品の最新バージョンを使用している顧客の割合を公表し、更新時の容易さや信頼性への努力を示すこと。  同上
未使用の管理者特権のデータを公表する	顧客ベースに対する過剰な権限付与に関する集計情報と、攻撃を減らすための顧客の行動変容を促す対応や製品の変更を示すこと。  同上

# セキュア・バイ・デザインの原則

## 原則2: 徹底的な透明性と説明責任を果たす



### ● セキュアな製品開発のプラクティス

推奨事項	説明
内部セキュリティ管理の確立	<p>SaaS事業者は強固なセキュリティ態勢の実現を促す内部管理の統計を公表すること。</p> <p>FIDO(Fast ID Online)認証の導入など強固なセキュリティ態勢の実現が求められている。</p>
高レベルな脅威モデルを公表する	<p>セキュアバイデザインに基づく製品開発にあたっては、何を誰から守ろうとしているか記述した脅威モデルの作成から始めること。</p> <p>セキュリティリスク分析が求められている。</p>
安全なSDLCの自己検証の証明を公表する	<p>各製品について、採用しているSDLC(セキュアソフトウェア開発ライフサイクル)の枠組みと適合性を自己検証の証明を通して示すこと。</p> <p>セキュアバイデザインを実践しているという根拠を示すことが求められている。</p>
脆弱性の透明性を活用する	<p>製品の脆弱性については、必要事項を記載したCVE(共通脆弱性識別子)エントリーとして公開すること。</p> <p>脆弱性に対してはCVEを付与し、グローバルで一意に管理できるようにすることが求められている。</p>

# セキュア・バイ・デザインの原則

## 原則2: 徹底的な透明性と説明責任を果たす



### ● セキュアな製品開発のプラクティス(続き)

推奨事項	説明
SBOMの公表	<p>サプライチェーンを掌握するため、サプライヤにデータを要求してSBOM(ソフトウェア部品表)を作成・維持し、顧客やユーザに提供すること。</p> <p>購入の決定、運用の支援、悪意のあるサプライチェーン攻撃の検出などにあたり、製品をコンポーネントレベルで管理できるようにすることが求められている。</p>
脆弱性開示ポリシーの公表	<p>脆弱性開示ポリシーを公表すること。</p> <p>発見された脆弱性の根本原因を分析して、可能な限り脆弱性の属性全体を排除する措置を講ずることが求められている。</p>

# セキュア・バイ・デザインの原則

## 原則2: 徹底的な透明性と説明責任を果たす

### ● ビジネスプラクティス

推奨事項	説明
セキュアバイデザインの保証人となる経営層幹部を指名し公表する	<p>セキュアバイデザインのプログラムを監督するトップ経営幹部を指名すること。</p> <p>製品セキュリティをトップレベルの経営に関わる問題として取り組むことが求められている。</p>
セキュアバイデザインのロードマップを公表する	<p>セキュリティ改善プログラムのロードマップを作成し、セキュリティを改善するためにSDLC(セキュアソフトウェア開発ライフサイクル)に加えた変更を記録として残すこと。</p> <p>品質改善の取組の場合と同様に、セキュリティ改善プログラムにおいても、計画、管理、改善という取組、実践しているという根拠を示すことが求められている。</p>
メモリ安全性のロードマップを公表する	<p>既存製品についてはメモリセーフな言語に移行すること。新たな製品はメモリセーフな言語を使用すること。</p> <p>メモリの安全性に起因する脆弱性を排除することが求められている。</p>
結果を公表する	<p>セキュアバイデザインの具現化にあたり実施したSDLC(セキュアソフトウェア開発ライフサイクル)の良い事例(成功)、悪い事例(失敗)を記録として残し、活用すること。</p> <p>セキュアバイデザインを実践しているという根拠を示すことが求められている。</p>

# セキュア・バイ・デザインの原則

## 原則3: トップ主導での実施



推奨事項	説明
企業の財務報告にセキュアバイデザインのプログラムの詳細を含める	顧客のセキュリティと企業の財務成果を結びつけていることを示すこと。
自組織の取締役会に対し定期的に報告を行う	セキュリティ態勢に関する情報に加え、製品セキュリティおよびそれが顧客のセキュリティに及ぼす影響に関する情報も含めること。
セキュアバイデザイン担当取締役の権限を強化する	顧客のセキュリティ向上を達成するために必要となる製品投資に関する権限を与えること。
企業内部で意味あるインセンティブを作る	顧客のセキュリティ向上を、評価する仕組みを作ること。
セキュアバイデザイン委員会を設置する	セキュアバイデザインに向けて活動するための仕組みを作ること。
顧客協議会を設立し、発展させる	顧客のセキュリティ向上に向けて、協働するための仕組みを作り活用すること。

# セキュア・バイ・デザインの原則

## Secure by Design Alerts

- CISA(米国サイバーセキュリティ・インフラセキュリティ庁)が発行する「顧客の安全に影響を与える脆弱性や製品の欠陥を排除するために、最初から製品にセキュリティを組み込むよう求めている注意喚起」

FACT SHEET

### Secure by Design Alert: Eliminating OS Command Injection Vulnerabilities



Publish Date  
July 10, 2024

RELATED TOPICS: [CYBERSECURITY BEST PRACTICES](#)

#### Malicious Actors Use OS Command Injection Vulnerabilities to Compromise Systems

Operating system (OS) command injection vulnerabilities are a preventable class of vulnerability in software products. Software manufacturers can eliminate them at the source by taking a secure by design approach. Despite this fact, OS command injection vulnerabilities continue to surface, allowing adversaries to exploit them to cause harm. CISA and FBI are releasing this Secure by Design Alert in response to recent well-publicized threat actor campaigns that exploited OS command injection defects in network edge devices (CVE-2024-20399<sup>1</sup>, CVE-2024-3400<sup>2</sup>, CVE-2024-21887<sup>3</sup>) to target and compromise users. These vulnerabilities allowed unauthenticated malicious actors to remotely execute code on network edge devices.

Cisco NX-OS、Paloalto PAN-OS、Ivanti の「OSコマンドインジェクション」脆弱性をきっかけとして、どのような実装をすべきか、セキュア・バイ・デザイン原則とどう関係しているかを記載している。

# セキュア・バイ・デザインの原則

## Secure by Design Pledge (自主宣言)



- 2024年5月のRSAカンファレンスで発表されたCISAの施策
- Secure by Designを宣言させ、7つの具体的な目標に対し、署名から1年以内に取り組みや達成状況を測定可能な形で公表すること

事項	ゴール
MFA(多要素認証)	1年以内に、製品全体で多要素認証の使用を増加させるために講じた措置を示すこと。
デフォルトパスワード	1年以内に、製品全体でデフォルトパスワードを削減に向けた進捗を示すこと。
脆弱性のクラス全体を削減	1年以内に、製品全体で1つ以上の脆弱性クラスを削減させるために講じた措置を示すこと。
セキュリティパッチ	1年以内に、顧客によるセキュリティパッチのインストールを増加させるために講じた措置を示すこと。
VDP(脆弱性開示ポリシー)	1年以内に、脆弱性開示ポリシーを公開すること。
CVE(共通脆弱性識別子)	1年以内に、脆弱性報告の透明性を示すこと。 CWE(共通脆弱性タイプ一覧)、CPE(共通プラットフォーム一覧)など脆弱性対策に必要な項目を記載した情報を発信すること。
侵入の証拠	1年以内に、製品に影響を及ぼす侵害の証拠を顧客が収集できる状況を向上させること。

# セキュア・バイ・デザインの原則

## EO14144(大統領令14144)の改正



- EO14144(サイバーセキュリティ改善のための追加措置)の改正
  - セキュアソフトウェア開発ガイドラインへの準拠を証明する義務を撤廃
  - NISTに対し、業界パートナーと協力し、セキュアソフトウェア開発に関する実践的なガイダンスを策定し、関連フレームワークを更新するよう指示

年月日	イベント
～2025年12月01日	<p>SSDFの暫定更新版(ソフトウェアの安全かつ信頼性の高い開発と提供、並びにソフトウェア自体のセキュリティに関する実践、手順、管理策、及び実装例を含む)の作成</p> <p>2025年12月17日 SSDF Ver 1.2 パブリックコメント版リリース <a href="https://csrc.nist.gov/News/2025/draft-ssdf-version-1-2">https://csrc.nist.gov/News/2025/draft-ssdf-version-1-2</a></p>
～2025年09月02日	<p>NIST-SP800-53:情報システムおよび組織のためのセキュリティとプライバシー管理を更新し、パッチとアップデートを安全かつ確実に展開する方法に関するガイダンスを提供</p>
～2025年08月01日	<p>NIST-SP800-218:セキュアソフトウェア開発フレームワーク(SSDF)に基づく安全なソフトウェア開発、セキュリティ、運用慣行の実装を示すガイダンスを策定</p> <p>2025年07月30日 DevSecOps Practices (NIST Special Publication (SP) 1800-44)プロジェクトの概要版リリース <a href="https://www.nist.gov/news-events/news/2025/07/nist-consortium-and-draft-guidelines-aim-improve-security-software">https://www.nist.gov/news-events/news/2025/07/nist-consortium-and-draft-guidelines-aim-improve-security-software</a></p>

## 情報収集に役立つ技術仕様



# 脆弱性対策情報の収集

## 脆弱性対策に役立つキーワード

- 脆弱性対策情報、注意喚起、ニュース記事等でも使用されているキーワード



・・・脆弱性を一意に識別する番号



・・・脆弱性の深刻度を評価する指標



・・・脆弱性の種別を体系的に分類



・・・製品を一意に識別する仕様

# CVE(共通脆弱性識別子) Common Vulnerabilities and Exposures

- プログラム上のセキュリティ問題に一意的番号(CVE識別番号)を付与して管理



CVE識別番号の構成

西暦	連番
2014	1000
2014	10000
2014	100000
2014	1000000

2014年1月～  
連番は可変長  
それ以前は4桁

Internet Systems Consortium

SOLUTIONS SUPPORT COMMUNITY STORE ABOUT

Remote packet Denial of Service against Authoritative and Recursive Name Servers

A specially constructed packet will cause BIND 9 ("named") to exit, affecting DNS service.

**CVE: CVE-2012-3413**

Document Version: 2.1

Posting date: 05 Jul 2011

Program Impacted: BIND

Versions affected: 9.6.3, 9.6-ESV-R4, 9.6-ESV-R4-P1, 9.6-ESV-R5b1 9.7.0, 9.7.0-P1, 9.7.0-P2, 9.7.2-P2, 9.7.2-P3, 9.7.3, 9.7.3-P1, 9.7.3-P2, 9.7.4b1 9.8.0, 9.8.0-P1, 9.8.0-P2

Severity: High

Exploitable: Remotely

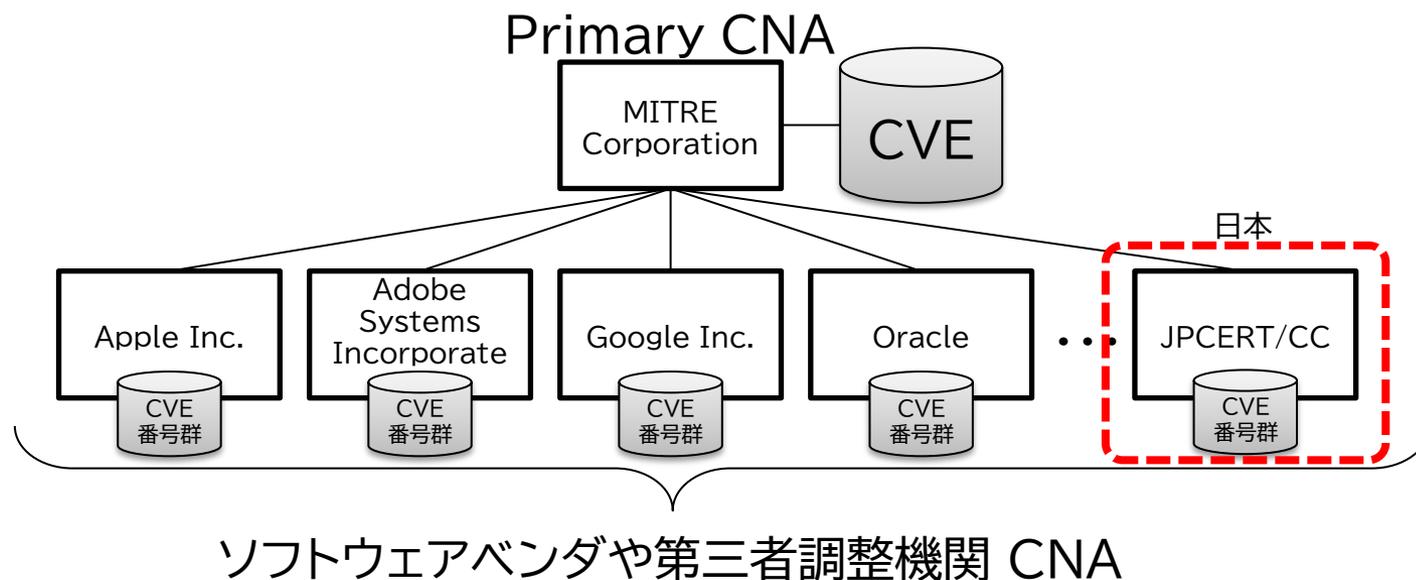
公表されている脆弱性に割り当てられた識別番号で、脆弱性を一意に特定することを可能となる

[出典] 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://www.cve.org/>

# CVE(共通脆弱性識別子) Common Vulnerabilities and Exposures



- 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採番機関、CVE Numbering Authority)から割り当て



[出典] Submit a CVE Request  
<https://cveform.mitre.org/>  
CVE Numbering Authorities (CNAs)  
<https://www.cve.org/ProgramOrganization/CNAs>

# CVSS(共通脆弱性評価システム) Common Vulnerability Scoring System

- 攻撃状況やシステムの重要度を加味して脆弱性の深刻度を表す

$$\begin{aligned} &= \text{「技術的な特性」} \times \text{「脅威の大きさ」} \times \text{「情報資産の価値」} \\ &\quad \text{「基本評価基準」} \times \text{「脅威評価基準」} \times \text{「環境評価基準」} \end{aligned}$$



何が引き起こされる?

SQLインジェクション  
技術面で危険度高

クロスサイトスクリプティング  
技術面で危険度中



既に攻撃されている?

攻撃観測  
なし

攻撃観測  
あり



システムの重要度は?

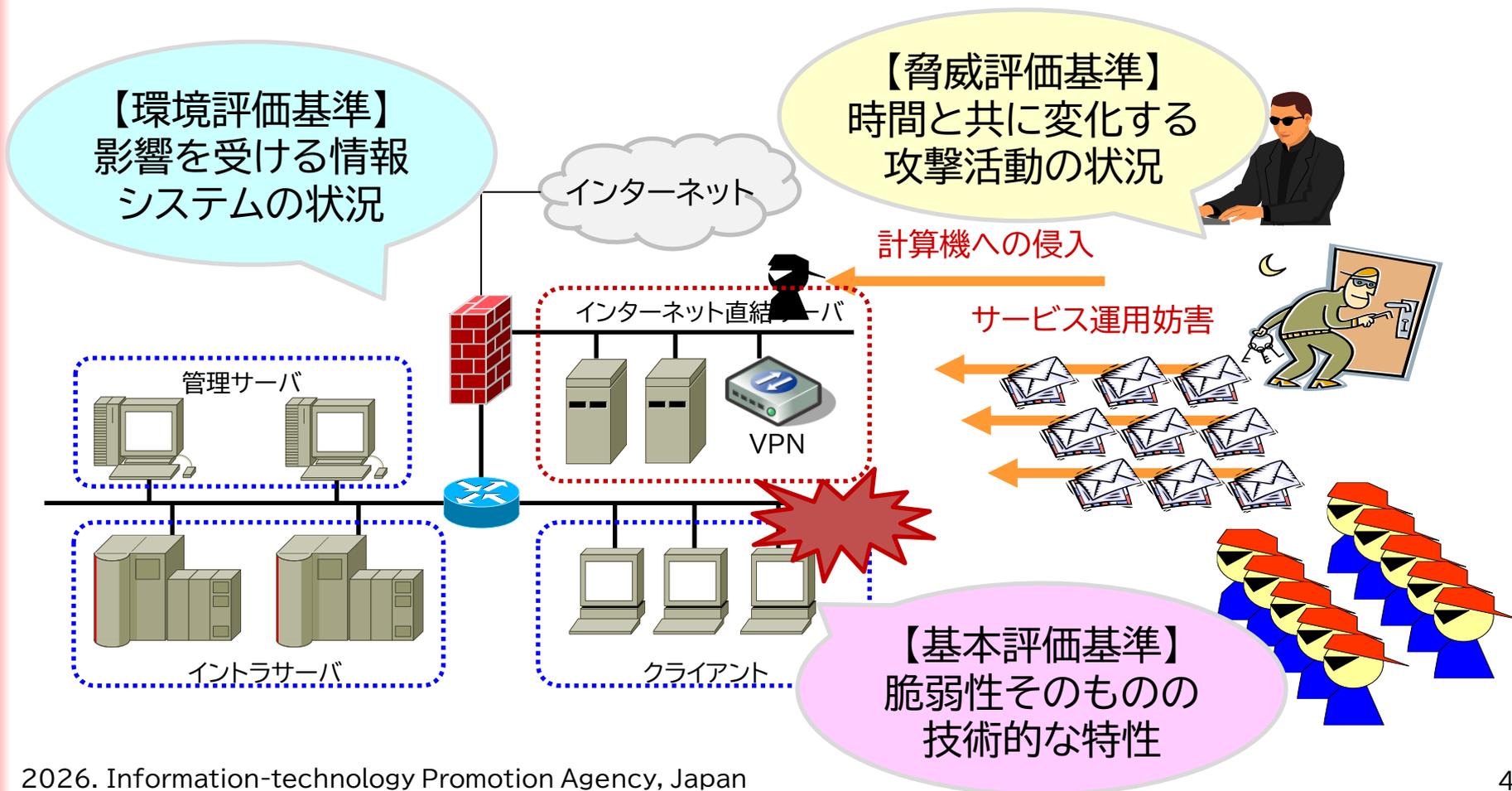
内部システム → 深刻度 低

外部システム → 深刻度 高

[出典] 共通脆弱性評価システムCVSS概説  
<https://www.ipa.go.jp/security/vuln/CVSS.html>  
Common Vulnerability Scoring System SIG  
<https://www.first.org/cvss/>

# CVSS (共通脆弱性評価システム) Common Vulnerability Scoring System

- 攻撃状況やシステムの重要度を加味して脆弱性の深刻度を表す

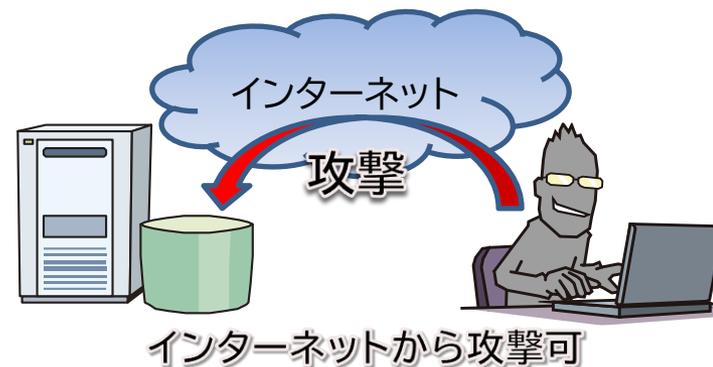


# CVSS(共通脆弱性評価システム) Common Vulnerability Scoring System

- 基本評価基準・・・脆弱性の技術的な特性を評価
  - 攻撃の難易度



OR



- 攻撃による影響



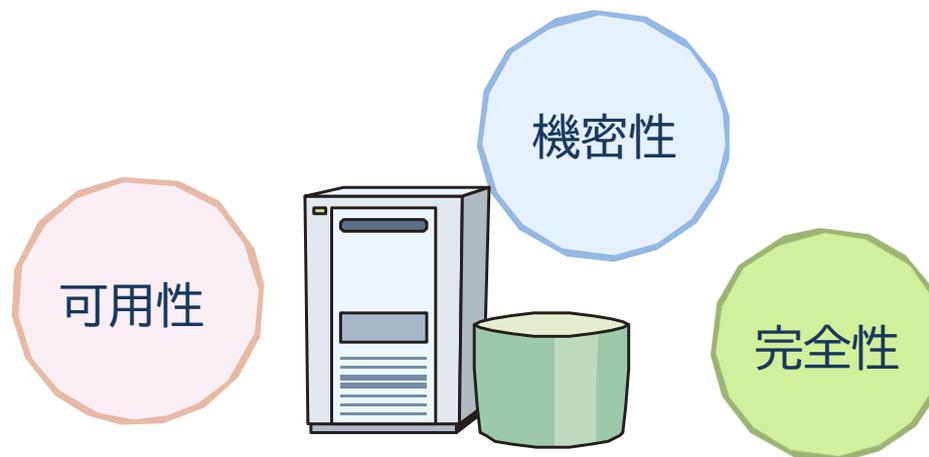
# CVSS(共通脆弱性評価システム) Common Vulnerability Scoring System

- 脅威評価基準・・・脆弱性を取り巻く状況进行评估
  - 実際に攻撃が行われている
  - 攻撃コードが一般に公開(攻撃の予兆)



# CVSS(共通脆弱性評価システム) Common Vulnerability Scoring System

- 環境評価基準・・・システムにおける問題の大きさを評価
  - CVSSv3 / CVSSv4
    - 対象システムのセキュリティ要求度を評価  
(機密性、完全性、可用性を評価)
    - 環境条件を加味した基本評価の再評価



# CVSS(共通脆弱性評価システム)

開発



- CVSSは、情報システムの脆弱性に対するオープンで包括的、汎用的な評価手法の確立と普及を目指し、NIAC(米国家インフラストラクチャ諮問委員会、National Infrastructure Advisory Council)のプロジェクトで原案が作成された。
  - 2004年10月 原案の作成
- その後、FIRST(Forum of Incident Response and Security Teams)のCVSS-SIG(Special Interest Group)によって仕様改善と適用推進が行われている。
  - 2005年6月 CVSS v1.0
  - 2007年6月 CVSS v2.0
  - 2015年6月 CVSS v3.0
  - 2019年6月 CVSS v3.1
  - 2023年11月 CVSS v4.0

# CVSS(共通脆弱性評価システム) スコアのレベル分けの変遷

- CVSSスコア(0.0~10.0)のレベル分け

CVSSv1	CVSSv2	CVSSv3 & v3.1	CVSSv4
	<div style="background-color: red; color: white; padding: 2px;">危険</div> 7.0~10.0 <div style="background-color: orange; color: white; padding: 2px;">警告</div> 4.0~6.9 <div style="background-color: yellow; color: black; padding: 2px;">注意</div> 0.0~3.9		<div style="background-color: red; color: white; padding: 2px;">緊急</div> 9.0~10.0 <div style="background-color: orange; color: white; padding: 2px;">重要</div> 7.0~8.9 <div style="background-color: yellow; color: black; padding: 2px;">警告</div> 4.0~6.9 <div style="background-color: #f0e68c; color: black; padding: 2px;">注意</div> 0.1~3.9 なし 0
<p>基本値</p>			
<p>現状値</p>			
<p>環境値</p>	<p>4,837,212</p>		<p>10,077,696</p>

# CVSS(共通脆弱性評価システム) v1.0からv2.0への変更点

- CVSSスコア算出の計算式を改善

CVSSv1	CVSSv2	CVSSv3 & v3.1	CVSSv4
--------	--------	---------------	--------

スコアの分布に  
偏りがあった

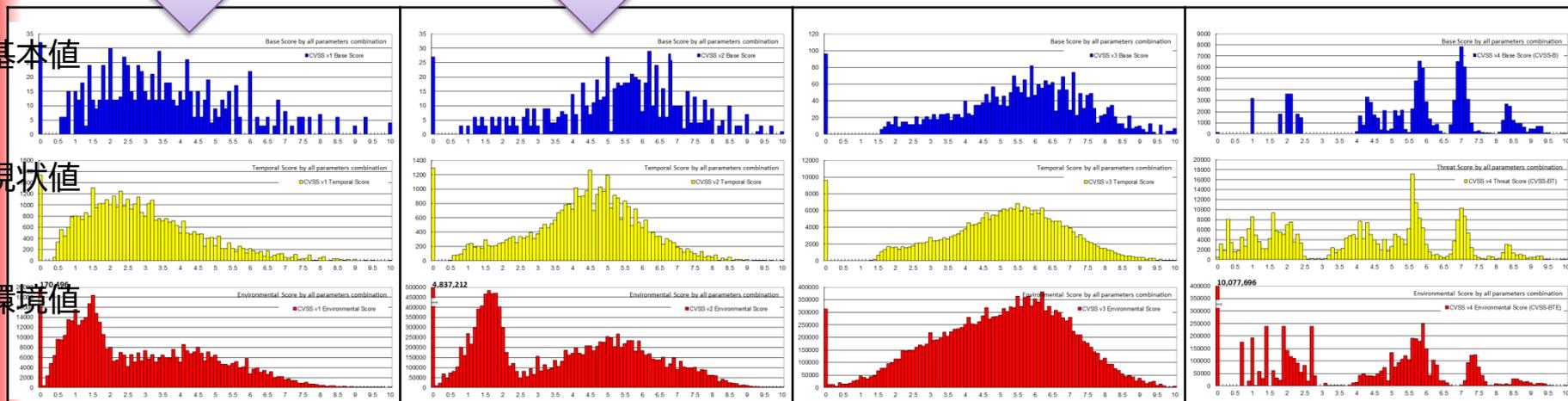
スコアの分布の  
偏りを是正した



基本値

現状値

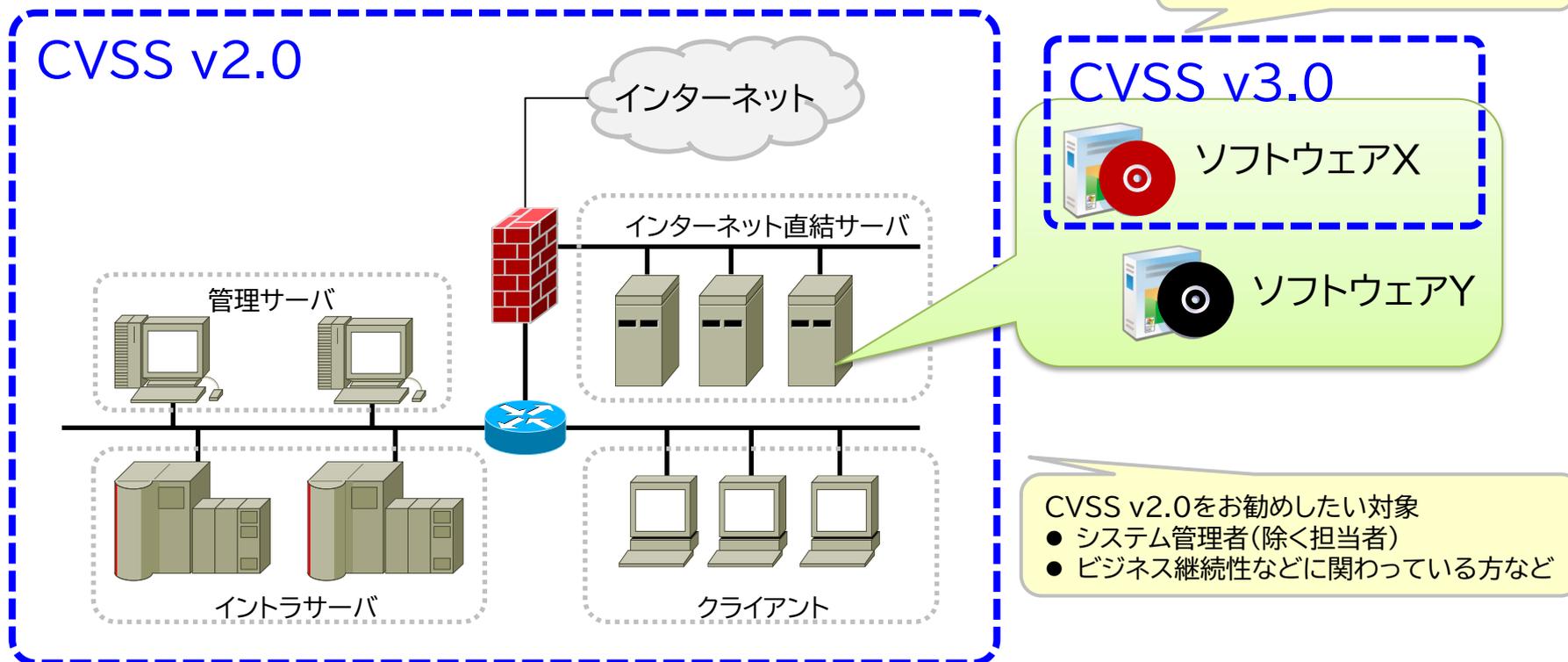
環境値



# CVSS(共通脆弱性評価システム)

## v2.0からv3.0への変更点

- 脆弱性の影響の捉え方を変更
  - CVSS v2.0:大局的(マクロ)に評価するアプローチ
  - CVSS v3.0:局所的(ミクロ)に評価するアプローチ



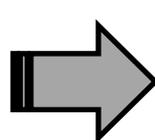
# CVSS(共通脆弱性評価システム)

v3.xの良い点

- 対策前後の数値化の活用  
[例] 脆弱性のあるアプリケーションへのアクセスをネットワークファイアウォールで制御した場合



攻撃者



脆弱性のある  
コンポーネント

脆弱性による影響の  
広がりを評価  
●スコープ=変更なし

脆弱性を取り巻く状況の評価

- 攻撃される可能性(E)
- 利用可能な対策のレベル(RL)
- 脆弱性情報の信頼性(RC)

攻撃の難易度を評価

- 攻撃元区分=ネットワーク
- 攻撃条件の複雑さ=低
- 必要な特権レベル=不要
- ユーザ関与レベル=不要

攻撃による影響を評価

- 機密性への影響=高
- 完全性への影響=なし
- 可用性への影響=なし

セキュリティ要求度

- 機密性の要求(CR)
- 完全性の要求度(IR)
- 可用性の要求度(AR)

CVSS環境評価値

対策前=7.5

# CVSS(共通脆弱性評価システム)

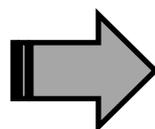
v3.xの良い点

- 対策前後の数値化の活用

[例] 脆弱性のあるアプリケーションへのアクセスをネットワークファイアウォールで制御した場合



攻撃者



脆弱性のある  
コンポーネント

脆弱性による影響の  
広がりを再評価  
●スコープ=変更なし

脆弱性を取り巻く状況进行评估  
●攻撃される可能性(E)  
●利用可能な対策のレベル(RL)  
●脆弱性情報の信頼性(RC)

攻撃の難易度を再評価  
●攻撃元区分=隣接  
●攻撃条件の複雑さ=低  
●必要な特権レベル=不要  
●ユーザ関与レベル=不要

攻撃による影響を再評価  
●機密性への影響=高  
●完全性への影響=なし  
●可用性への影響=なし

セキュリティ要求度  
●機密性の要求(CR)  
●完全性の要求度(IR)  
●可用性の要求度(AR)

CVSS環境評価値

対策後=6.5

# CVSS(共通脆弱性評価システム)

## v3.0からv3.1への変更点



- CVSSスコア算出のための指針の明確化
  - CVSSスコアのブレを少なくする。
- v3.0計算式の不具合の修正
  - 言語によって小数計算に誤差が生じる問題
  - 環境評価の再評価において、評価が逆転してしまう問題
- 独自の評価基準を追加できるよう拡張
  - CVSS利用の適用範囲を拡大する。

# CVSS(共通脆弱性評価システム)

## v3.1からv4への変更点



- 脆弱性そのものの技術的な特性を評価する基本評価基準を改良
  - 攻撃の難易度
    - 脆弱性攻撃の前提条件に関する評価項目の追加
    - ユーザ関与レベル(UI)の細分化(不要、受動的、能動的)
  - 攻撃による影響
    - 影響の想定範囲(S)を、脆弱なコンポーネントへの影響、他のコンポーネントへの影響に細分化
- 脆弱性を取り巻く状況を評価する現状評価基準を改良
  - 現状評価基準から脅威評価基準へ名称変更
  - 利用可能な対策のレベル(RL)、脆弱性情報の信頼性(RC)を削除
  - 攻撃可能性(E)を、攻撃の成熟度に変更

# CVSS(共通脆弱性評価システム)

## v3.1からv4への変更点

- システムにおける問題の大きさを評価する環境評価基準を改良
  - 影響の想定範囲(S)を、脆弱なコンポーネントへの影響、他のコンポーネントへの影響に細分化して再評価
  - 他のコンポーネントへの影響の再評価において安全性を考慮
- 補助評価基準の新設
  - 安全性、攻撃の自動化可能性、情報の緊急度、回復の手段、攻撃に利用可能な資源、対処するための労力
- CVSSスコア算出アプローチの変更
  - CVSS v1～v3.1:計算式から算出
  - CVSS v4:270個のスコア区分に振り分けた後、微調整して算出 (MacroVectors and Interpolation)



CVSS v2、CVSS v3の考え方を抑えておくことが重要です。

# CVSS(共通脆弱性評価システム) v4.0

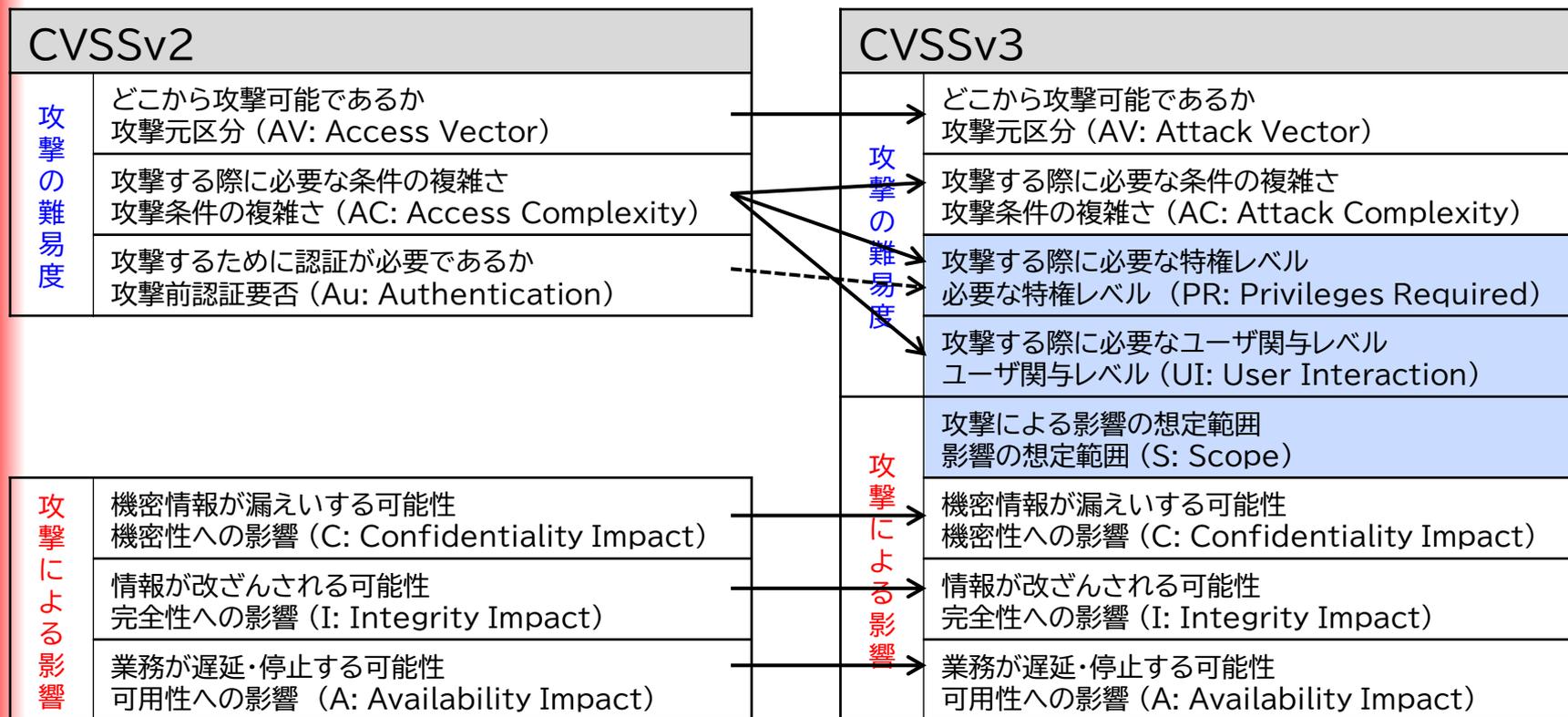


- 基本評価基準、脅威評価基準、環境評価基準、補助評価基準の4つから構成
- CVSSスコア算出に使用するのは、基本評価基準、脅威評価基準、環境評価基準の3つ

基本評価基準	脅威評価基準	環境評価基準	補助評価基準
攻撃の難易度 <ul style="list-style-type: none"> <li>● 攻撃元区分</li> <li>● 攻撃条件の複雑さ</li> <li>● 脆弱性攻撃の前提条件</li> <li>● 必要な特権レベル</li> <li>● ユーザ関与レベル</li> </ul>	<ul style="list-style-type: none"> <li>● 攻撃の成熟度</li> </ul>	対策後の基本評価の再評価 <ul style="list-style-type: none"> <li>● 攻撃元区分</li> <li>● 攻撃条件の複雑さ</li> <li>● 脆弱性攻撃の前提条件</li> <li>● 必要な特権レベル</li> <li>● ユーザ関与レベル</li> <li>● 脆弱なシステムへの影響(機密性)</li> <li>● 脆弱なシステムへの影響(完全性)</li> <li>● 脆弱なシステムへの影響(可用性)</li> <li>● 他のシステムへの影響(機密性)</li> <li>● 他のシステムへの影響(完全性)</li> <li>● 他のシステムへの影響(可用性)</li> </ul>	<ul style="list-style-type: none"> <li>● 安全性</li> <li>● 攻撃の自動化可能性</li> <li>● 情報の緊急度</li> <li>● 回復の手段</li> <li>● 攻撃に利用可能な資源</li> <li>● 対処するための労力</li> </ul>
攻撃による影響 <ul style="list-style-type: none"> <li>● 脆弱なシステムへの影響(機密性)</li> <li>● 脆弱なシステムへの影響(完全性)</li> <li>● 脆弱なシステムへの影響(可用性)</li> <li>● 他のシステムへの影響(機密性)</li> <li>● 他のシステムへの影響(完全性)</li> <li>● 他のシステムへの影響(可用性)</li> </ul>		<ul style="list-style-type: none"> <li>● 機密性の要求</li> <li>● 完全性の要求度</li> <li>● 可用性の要求度</li> </ul>	

# CVSS(共通脆弱性評価システム) v4.0

基本評価基準:v2からv3への変更点



# CVSS(共通脆弱性評価システム) v4.0

基本評価基準:v3からv4への変更点



CVSSv3		CVSSv4	
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)	→	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前提条件の有無 脆弱性攻撃の前提条件 (AT: Attack Requirements)
	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)		脆弱なコンポーネントへの影響 ／他のコンポーネントへの影響
	機密情報が漏えいする可能性 機密性への影響 (C: Confidentiality Impact)	→	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)	→	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)	→	業務が遅延・停止する可能性 可用性への影響 (A: Availability Impact)

# CVSS(共通脆弱性評価システム) v4.0

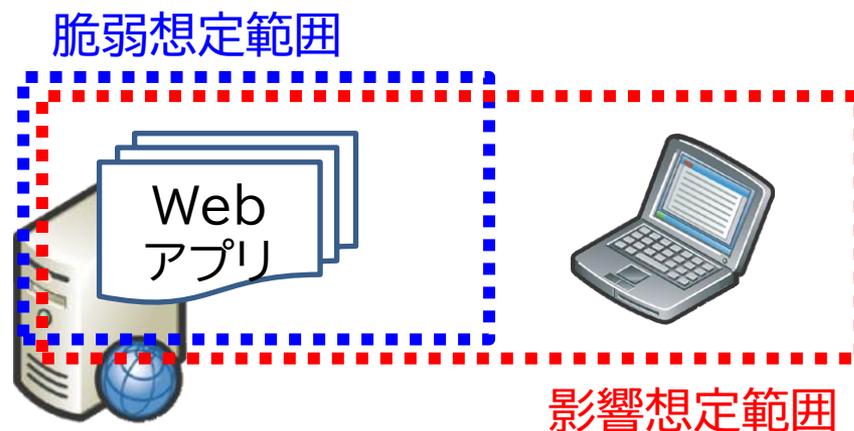
基本評価基準:脆弱性による二次的な影響波及の表記方法

- 脆弱性による二次的な影響波及の表記方法 (v3)
  - スコープ
    - 脆弱想定範囲(Vulnerable Component):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(コンポーネント)範囲
    - 影響想定範囲(Impacted Component):脆弱性を悪用された場合に及ぶ影響範囲

スコープ変更なし(S:U)



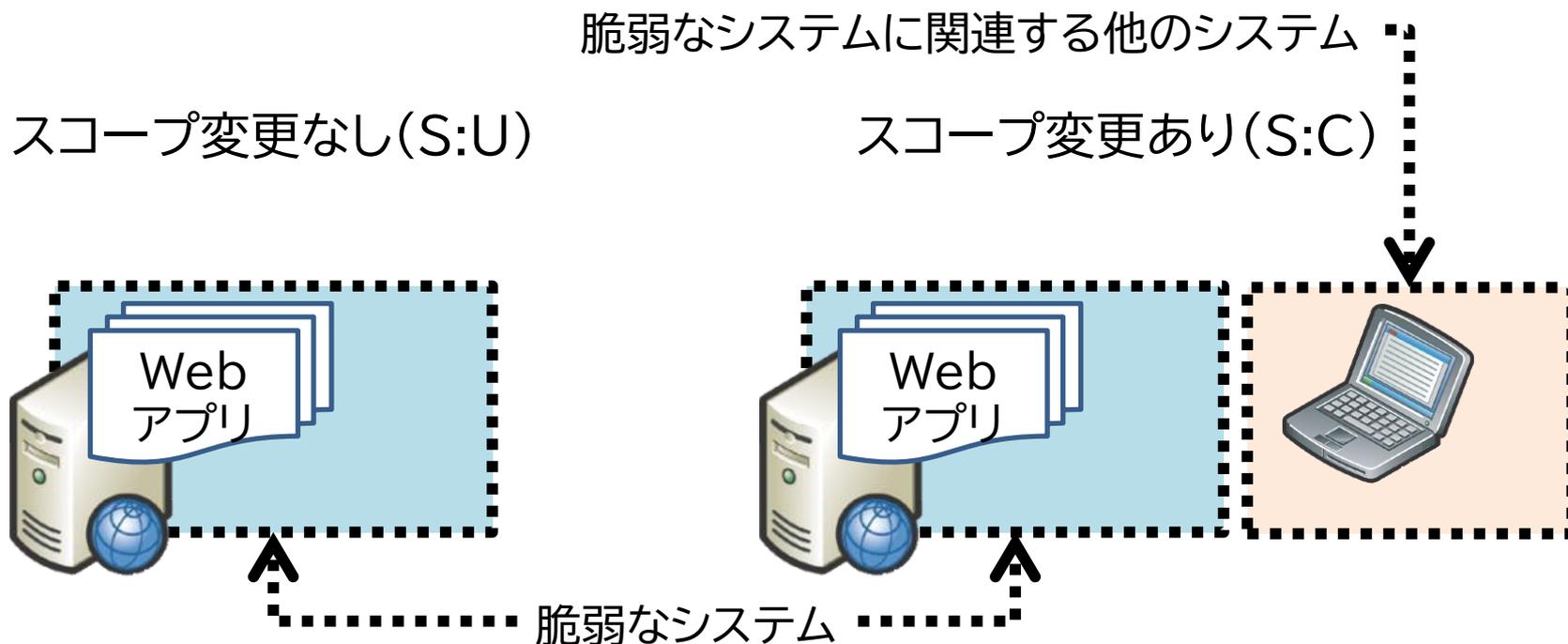
スコープ変更あり(S:C)



# CVSS(共通脆弱性評価システム) v4.0

基本評価基準:脆弱性による二次的な影響波及の表記方法

- 脆弱性による二次的な影響波及の表記方法 (v4)
  - 脆弱なシステムへの影響/他のシステムへの影響
    - 脆弱なシステムを対象とした影響をCIAで表記
    - 脆弱なシステムに関連する他のシステムへの影響をCIAで表記

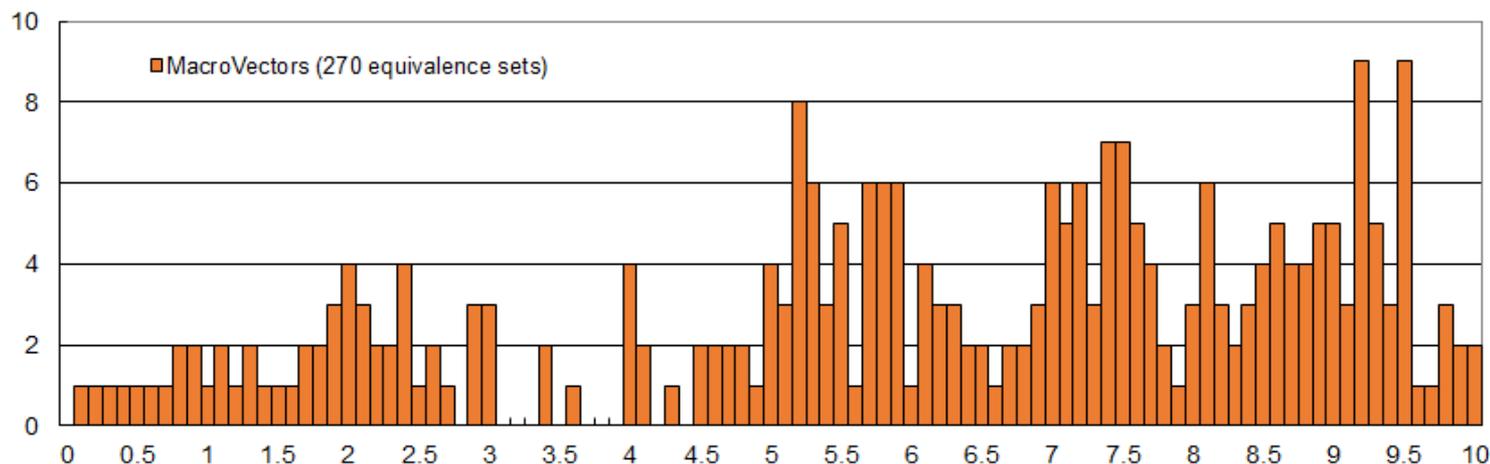


# CVSS(共通脆弱性評価システム) v4.0

スコアの算出方法:v4への変更点



- CVSS v1~v3.1:計算式から算出
- CVSS v4:270個のスコア区分に振り分けた後、微調整して算出
  - CVSS環境値(CVSS-BTE:基本評価+脅威評価+環境評価)の組合せ、約1500万件を元に270個のスコア区分を作成
  - パラメタ値の組合せの特徴から、270個のスコア区分のいずれかに振り分け
  - スコア区分に付与されているCVSSスコア値に対して、CVSSスコア区分値とスコア値が低くなる組合せとの差を元に、微調整をして算出

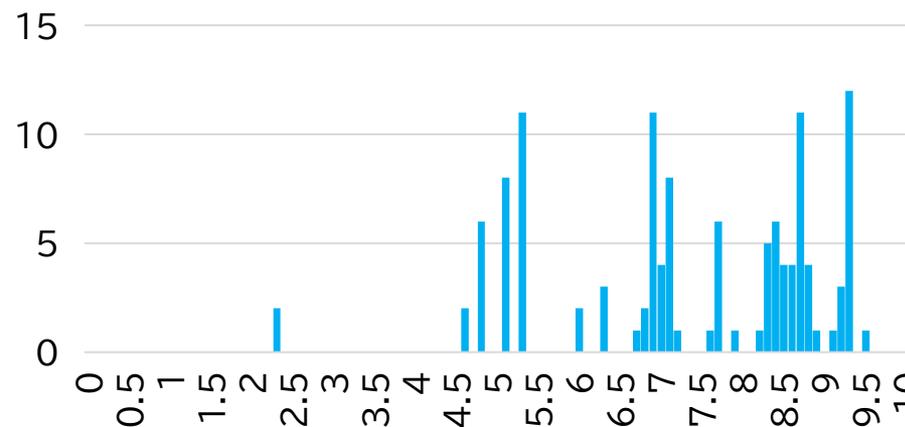
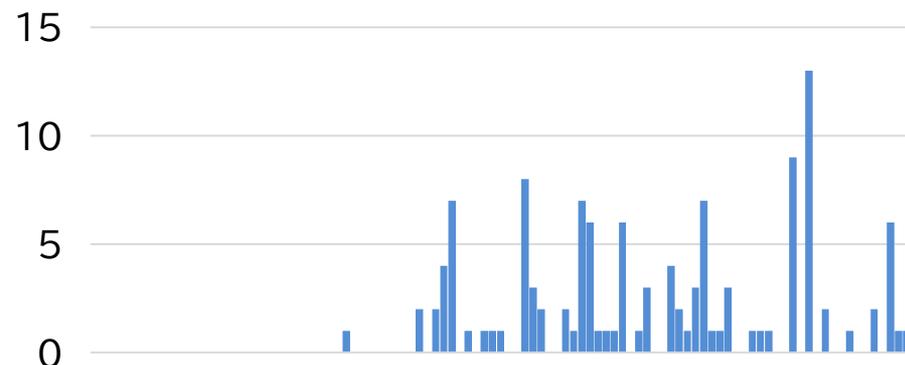
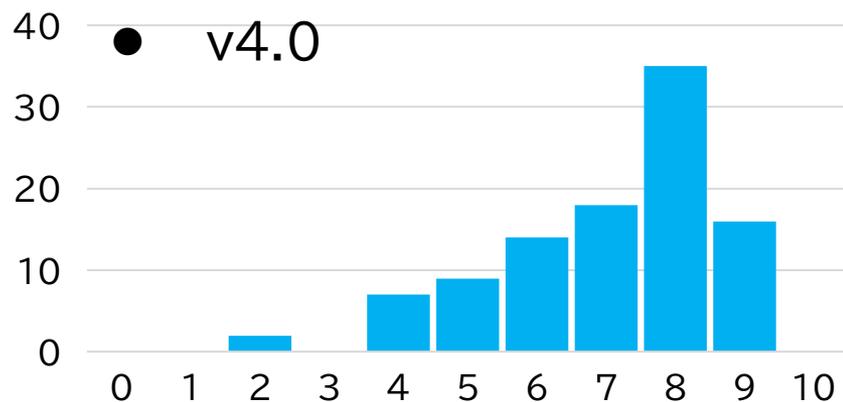
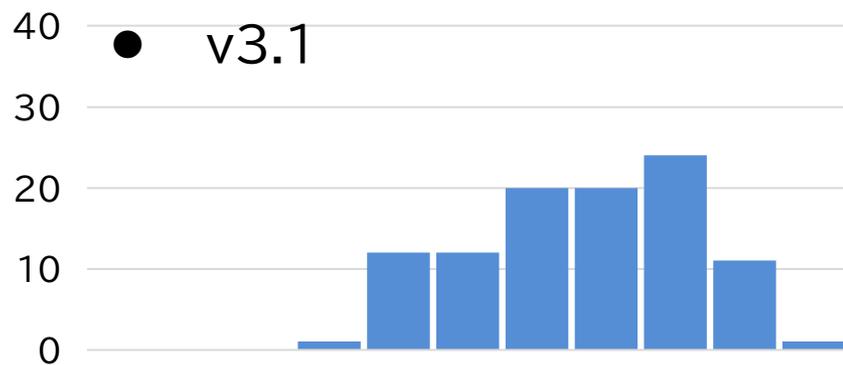


# CVSS(共通脆弱性評価システム) v4.0

スコアの算出方法:v4への変更点



## ● 基本評価値の比較(事例122件による比較)



# CWE(共通脆弱性タイプ一覧)

## Common Weakness Enumeration



- 脆弱性を種別毎に分類
  - 2024 CWE トップ 25
    - スコア: NVD(米国脆弱性対策データベース)での出現頻度とCVSS(共通脆弱性評価システム)の深刻度から算出
    - 掲載数: KEV(悪用された既知の脆弱性一覧)での出現数

#	CWE	名称	スコア	掲載数
1	<a href="#">CWE-79</a>	クロスサイトスクリプティング	56.92	3
2	<a href="#">CWE-787</a>	領域外メモリへの書き出し	45.20	18
3	<a href="#">CWE-89</a>	SQL インジェクション	35.88	4
4	<a href="#">CWE-352</a>	クロスサイトリクエストフォージェリ	19.57	0
5	<a href="#">CWE-22</a>	ディレクトリトラバーサル	12.74	4

[出典] 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(共通脆弱性タイプ一覧)

## Common Weakness Enumeration



- 脆弱性を種別毎に分類
  - 2025 CWE トップ 25
    - スコア: NVD(米国脆弱性対策データベース)での出現頻度とCVSS(共通脆弱性評価システム)の深刻度から算出
    - 掲載数: KEV(悪用された既知の脆弱性一覧)での出現数

#	CWE	名称	スコア	掲載数
1	<a href="#">CWE-79</a>	クロスサイトスクリプティング	60.38	7
2	<a href="#">CWE-89</a>	SQL インジェクション	28.72	4
3	<a href="#">CWE-352</a>	クロスサイトリクエストフォージェリ	13.64	0
4	<a href="#">CWE-862</a>	認可の欠落	13.28	0
5	<a href="#">CWE-787</a>	領域外メモリへの書き出し	12.68	12

[出典] 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(共通脆弱性タイプ一覧)

## Common Weakness Enumeration



- 脆弱性を種別毎に分類
  - 2025 CWE トップ 25

#	CWE	名称	スコア	掲載数
6	<a href="#">CWE-22</a>	ディレクトリトラバーサル	8.99	10
7	<a href="#">CWE-416</a>	メモリの解放後使用	8.47	14
8	<a href="#">CWE-125</a>	領域外のメモリ参照	7.88	3
9	<a href="#">CWE-78</a>	OSコマンドインジェクション	7.85	20
10	<a href="#">CWE-94</a>	コードインジェクション	7.57	7
11	<a href="#">CWE-120</a>	バッファオーバーフロー	6.96	0
12	<a href="#">CWE-434</a>	適切でないアップロードファイル制限	6.87	4
13	<a href="#">CWE-476</a>	NULL ポインタ逆参照	6.41	0
14	<a href="#">CWE-121</a>	スタックオーバーフロー	5.75	4
15	<a href="#">CWE-502</a>	信頼できないデータの逆シリアル化	5.23	11

# CWE(共通脆弱性タイプ一覧)

## Common Weakness Enumeration

- 脆弱性を種別毎に分類
  - 2025 CWE トップ 25



#	CWE	名称	スコア	掲載数
16	<a href="#">CWE-122</a>	ヒープオーバーフロー	5.21	6
17	<a href="#">CWE-863</a>	適切でない認可	4.14	4
18	<a href="#">CWE-20</a>	適切でない入力確認	4.09	2
19	<a href="#">CWE-284</a>	適切でないアクセス制御	4.07	1
20	<a href="#">CWE-200</a>	権限のないユーザへの機密情報の漏洩	4.01	1
21	<a href="#">CWE-306</a>	重要な機能に対する認証の欠如	3.47	11
22	<a href="#">CWE-918</a>	サーバサイドリクエストフォージェリ	3.36	0
23	<a href="#">CWE-77</a>	コマンドインジェクション	3.15	2
24	<a href="#">CWE-639</a>	ユーザ制御キーによる認可の迂回	2.62	0
25	<a href="#">CWE-770</a>	制限やスロットリングなしのリソース割当	2.54	0

# CPE(共通プラットフォーム一覧) Common Platform Enumeration



- 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≡製品識別子

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 v2.2

**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(共通プラットフォーム一覧)

## Common Platform Enumeration



- その他の製品識別子

- CPE 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}:{その他}  
cpe:2.3:a:company\_a:application:1.1:\*:\*:\*:\*:\*:\*

- purl(package URL)

pkg:{タイプ}/{名前空間}/{パッケージ名}@{バージョン}?{修飾子}#{パス}  
pkg:maven/org.company\_b/browser@2.1

- SWID(ソフトウェア識別子、Software identification tag)

ISO19770-2で標準化されたソフトウェア識別タグの仕様  
グローバルなユニークIDで16バイトのUUID or タグ生成者によって定義された値

- TEI(Transparency Exchange Identifier)

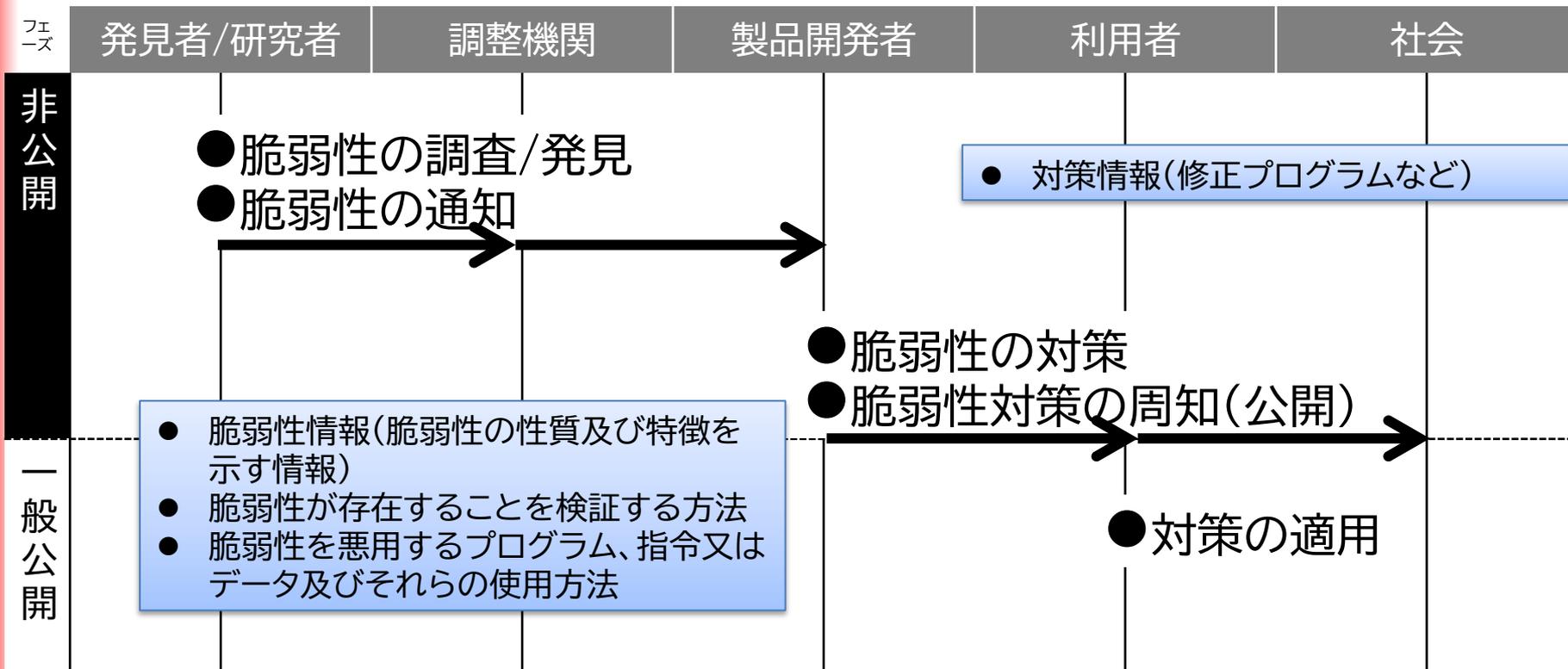
OWASPが提案する複数の製品識別子を統合する仕様  
urn:tei:{TEI type}:{DNS domain}:{製品識別子}  
urn:tei:uuid:products.example.com:d4d9f54a-abcf-11ee-...

- フルディスクロージャ(Full Disclosure)(1993年)  
セキュリティに関するひとつの考え方であり、「真にセキュアなシステムとは、プロトコル、ソースコードなどすべての視点でオープンレビューに耐えうること」「脆弱性に関する詳細情報はすべてのユーザが利用できること」としている。
- 脆弱性開示ポリシー(Vulnerability Disclosure Policy)
  - 脆弱性情報の取り扱いに関する考え方
  - 発見者、調整機関、製品開発ベンダなどが、自身の立場での考え方をまとめたものであり、全体としての整合性が取れているわけではない。

# 脆弱性(ぜいじゃくせい)の開示

脆弱性ハンドリングとは

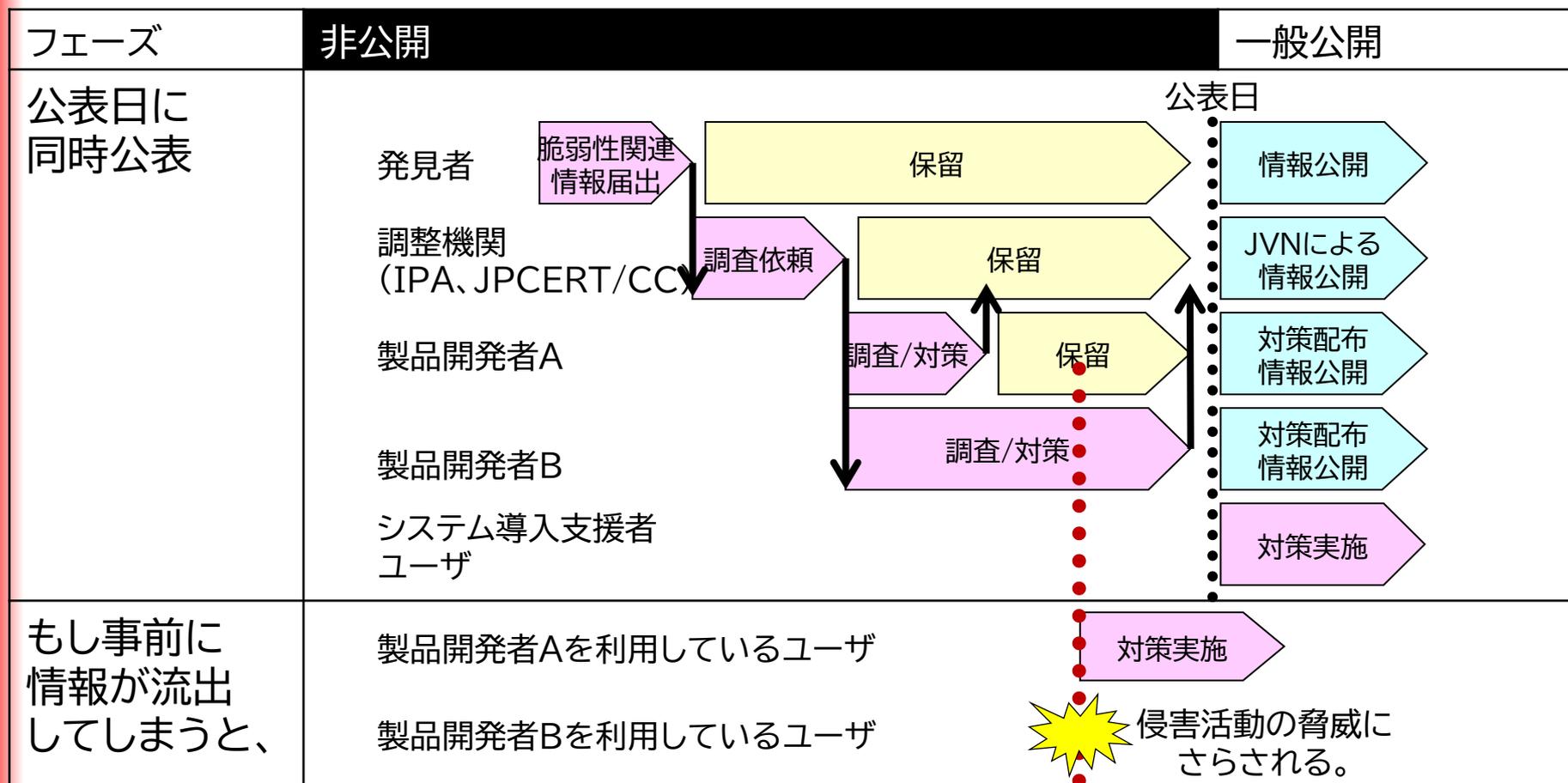
- サイバー攻撃による被害発生を抑制するために、関係者が推進する関連情報の適切な流通活動
- 脆弱性の発見、通知、対策、周知(公開)、適用までの一連のプロセス



# 脆弱性(ぜいじゃくせい)の開示

## 公開日一致の原則

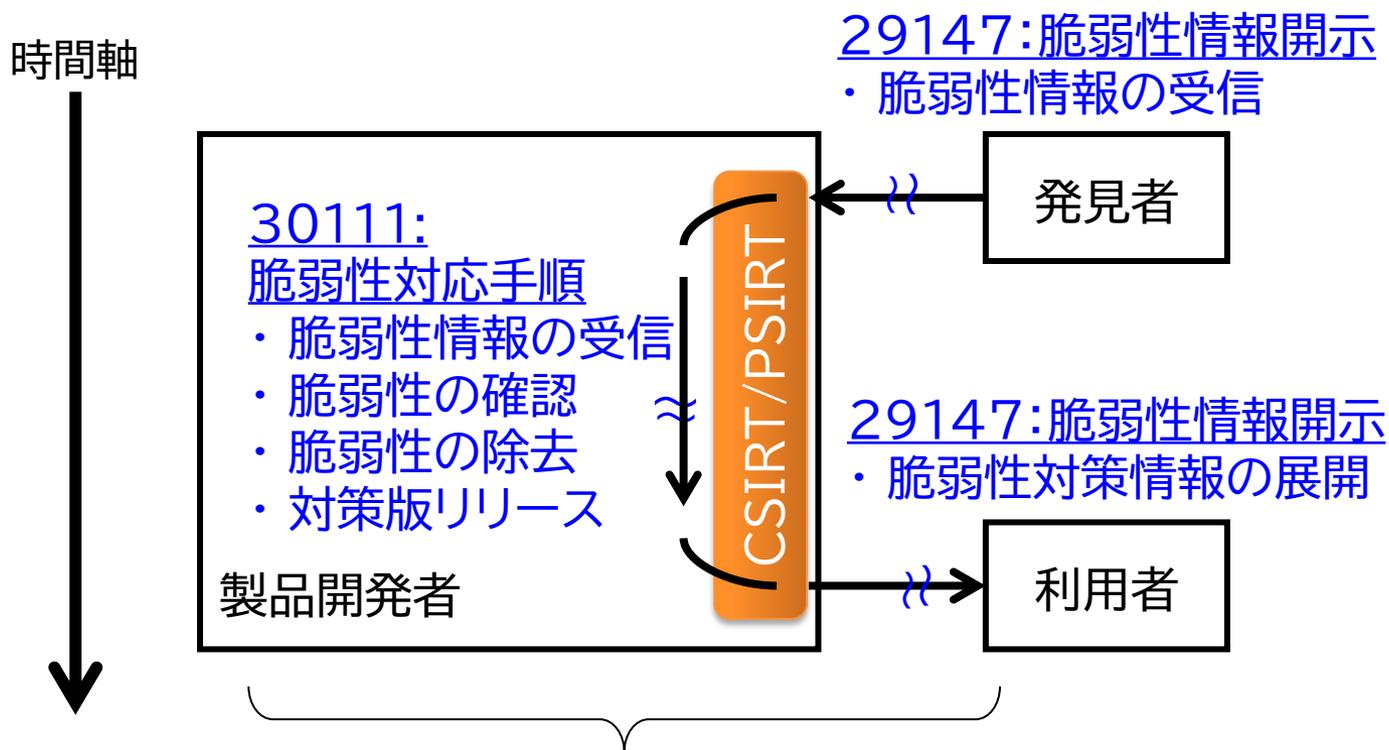
- 該当製品の対策を揃えつつ、格差のない対策環境を提供する考え方



# 脆弱性(ぜいじゃくせい)の開示

## 国際標準ISO/IEC

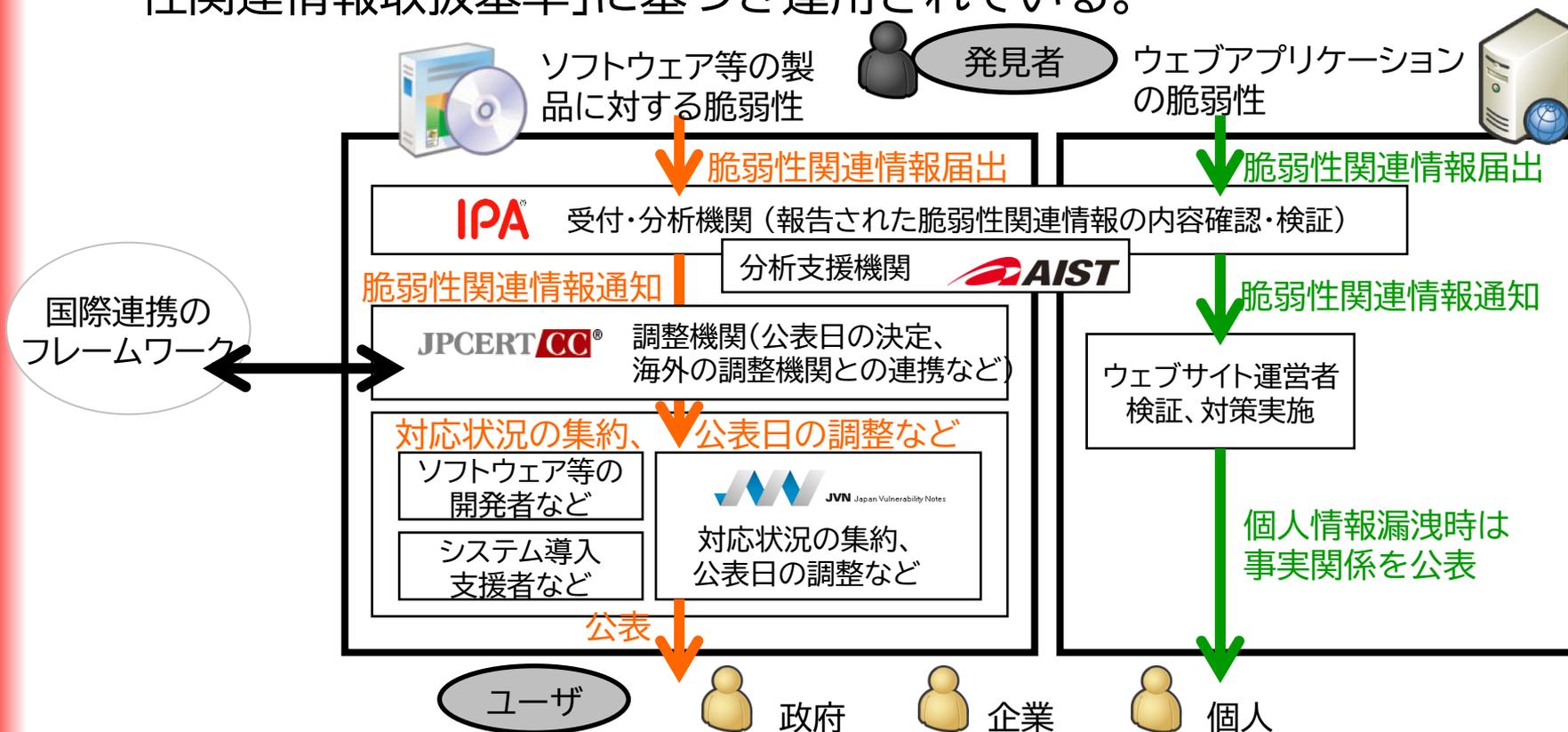
- ISO/IEC 29147、30111 (2008年～2014年にかけて初版標準化)



製品開発者の脆弱性開示ポリシー(Vulnerability Disclosure Policy)  
＝脆弱性情報開示、脆弱性対応手順の考え方をまとめたもの

# 脆弱性(ぜいじゃくせい)の開示 情報セキュリティ早期警戒パートナーシップ

- ソフトウェア等の製品やウェブアプリケーション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「ソフトウェア等脆弱性関連情報取扱基準」に基づき運用されている。

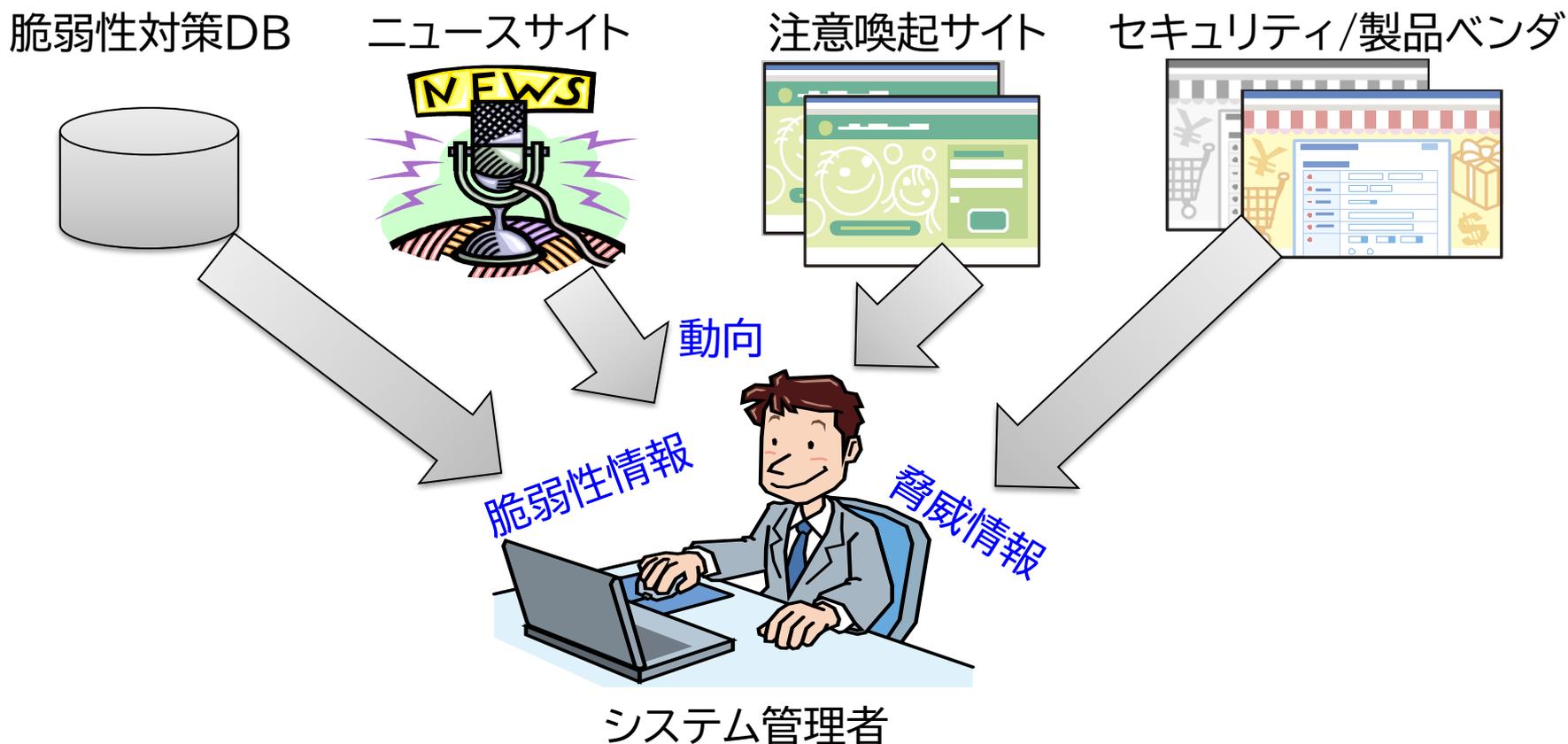


## JVN脆弱性対策機械処理基盤



# 脆弱性関連情報の収集 防御としての(情報活用+対策)

- 脆弱性対策の判断要素となる情報を収集し、自組織の対策に役立てる。

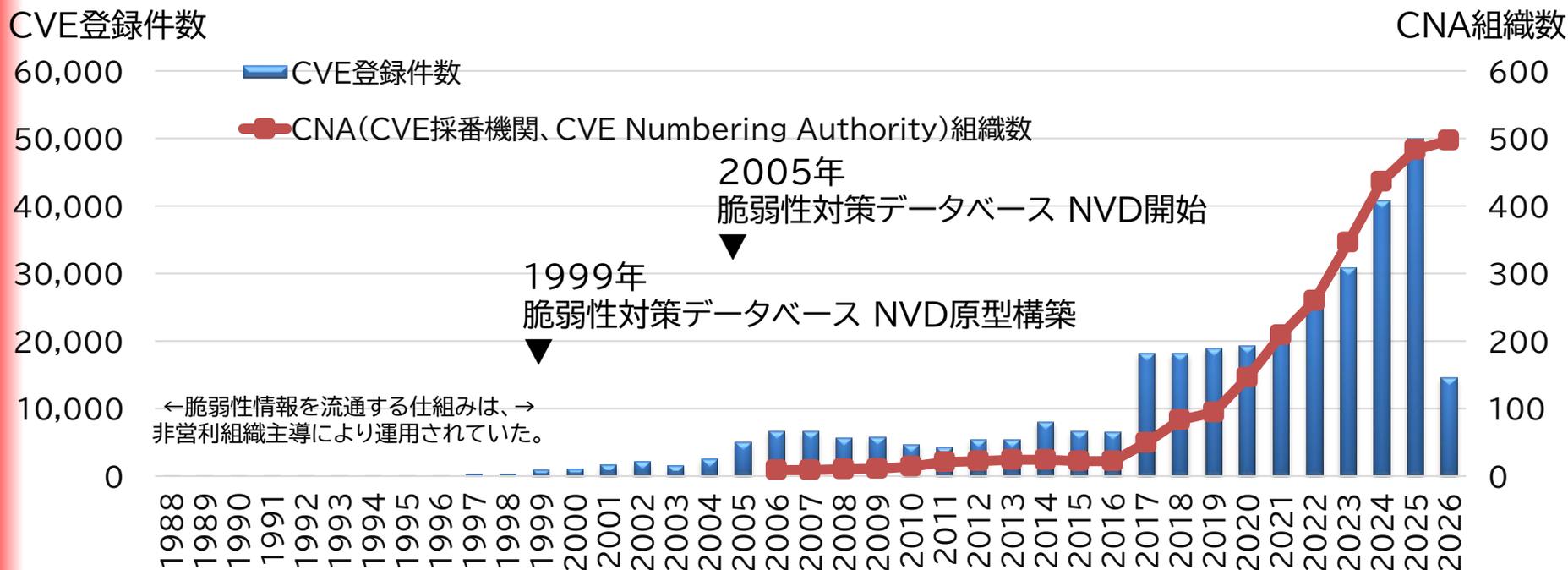


# 脆弱性関連情報の収集

【 課題 】 脆弱性情報の件数は増加傾向



- NVD(米国脆弱性対策データベース、National Vulnerability Database)に登録されている件数は？
  - 脆弱性を分担協力して登録する組織(脆弱性登録組織)の増加と共に増加している。



# 脆弱性関連情報の収集

【課題】インストール状況と脆弱性との紐付けは人手で実施

- 重要なセキュリティ情報が発信されるたびに、手作業でIT資産一覧表を検索して、影響範囲の調査をしていませんか？
  - 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（資産管理と脆弱性対策とが連携できていない）。



# 脆弱性関連情報の収集

【課題】カスタムアプリ管理は整備途上

- 重要なセキュリティ情報が発信されたときに、カスタムアプリ(SIで開発したアプリケーションなど)の影響範囲を調査していますか？
  - 多くの場合、カスタムアプリ(SIで開発したアプリケーションなど)は、資産管理や脆弱性管理の対象に含まれていない。

カスタムアプリ  
Ex. 在庫管理アプリ



## 重要なセキュリティ情報



最新情報	重要なセキュリティ情報	脆弱性対策情報 (VFN)	他機関からの情報
2017年1月11日	Microsoft 製品の脆弱性対策について(2017年1月)		
2017年1月11日	Adobe Flash Player の脆弱性対策について(APS817-02)(CVE-2017-2338等)		
2017年1月11日	Adobe Acrobat の脆弱性対策について(APS817-01)(CVE-2017-2339)		
2016年12月22日	「SKYSEA Client View」において任意のコード実行可能な脆弱性について (CVE-2016-7892)		
2016年12月14日	Adobe Flash Player の脆弱性対策について(APS816-30)(CVE-2016-7892等)		

# 脆弱性関連情報の収集

脆弱性対策作業のルーチンワーク化(コンピュータ処理)

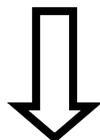
## 【課題】

- 脆弱性情報の件数は増加傾向
- インストール状況と脆弱性との紐付けは人手で実施
- カスタムアプリ管理は整備途上

セキュリティ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

### ● 識別子の共通化

- プログラムの脆弱性を一意に識別する。
- 情報システムを構成するハードウェア、ソフトウェアなどの製品を一意に識別する。
- 脆弱性の種別を識別する。



### ● 評価指標の共通化

- 脆弱性の深刻度を評価する。

## 【解決策】

- ルーチンワーク化(コンピュータ処理)による対処

「ルーチンワーク化」の部分は、一般的にMachine Readableと言われている。直訳すると、機械可読であるが、手順化してコンピュータに処理させることを意味する。

# 脆弱性関連情報の収集

脆弱性対策データベースに記載されていること

- 脆弱性対策データベースに記載されていることは、、、
  - 概要
  - 影響を受けるシステム
  - 詳細情報
  - 想定される影響
  - 対策方法
  - ベンダ情報
  - 参考情報

- 概要、詳細情報、想定される影響に記載される内容は、セキュリティ専門用語が多く、内容を把握することも、なかなか難しい状況にある。
- さらに、情報システムだけでなく、医療機器、制御系システムなど対象分野が広がりつつあることもあり、分野毎の専門用語も増え始めている。

# 脆弱性対策データベース

## 識別子の共通化

- 脆弱性対策作業のルーチンワーク化(機械化/手順化)

- 識別子の共通化

- プログラムの脆弱性を一意に識別する。

- CVE(共通脆弱性識別子)

[形式] CVE-西暦-連番

[例] CVE-2017-0145 (WannaCry(2017年)が悪用した脆弱性)



- 情報システムを構成するハードウェア、ソフトウェアなどの製品を一意に識別する。

- CPE(共通プラットフォーム一覧)

[形式] cpe:/{種別}:{ベンダ名}:{製品名}

[例] cpe:/o:microsoft:windows\_7

マイクロソフト ウィンドウズ 7と言っても表記方法は様々

Microsoft Windows 7、マイクロソフト Windows 7など



# 脆弱性対策データベース

## 評価指標の共通化

- 脆弱性対策作業のルーチンワーク化(機械化/手順化)

- 識別子の共通化

- 脆弱性の種別を識別する。

- CWE(共通脆弱性タイプ一覧)

[形式] CWE-番号

[例] CWE-78 (OSコマンドインジェクション)



- 評価指標の共通化

- 脆弱性の深刻度を評価する。

- CVSS(共通脆弱性評価システム)

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



何が起きるのか?



既に攻撃は発生している?  
対策は出ている?



システムの重要度は?

# 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

Japan Vulnerability Notes - Microsoft Edge  
Japan Computer Emergency Response Team Coordination Center [JP] https://jvn.jp/

最終更新日: 2019/03/26  
English

RSS に掲載する URL に関するお知らせ

新着リスト

JVNVU#93236010:	複数の Apple 製品における脆弱性に対するアップデート [2019/03/19 15:45]
JVNV#60497148:	iOS アプリ「an」におけるディレクトリトラバーサル [2019/03/19 12:00]
JVNV#06527859:	簡易CMS紀永における複数のクロスサイトスクリプティング [2019/03/15 12:00]
JVNVU#98344681:	Intel 製品に複数の脆弱性 [2019/03/14 11:30]
JVNV#11622218:	iOS アプリ「iChain保険ウォレット」におけるディレクトリトラバーサルの脆弱性 [2019/03/12 12:00]
JVNV#79543573:	Microsoft Teams のインストーラにおける DLL 読み込みの脆弱性 [2019/03/12 12:00]

JVN iPedia - 脆弱性対策情報データベース - Microsoft Edge  
Information-technology Promotion Agency, Japan [JP] https://jvndb.jvn.jp/

新着情報 RSS データフィード

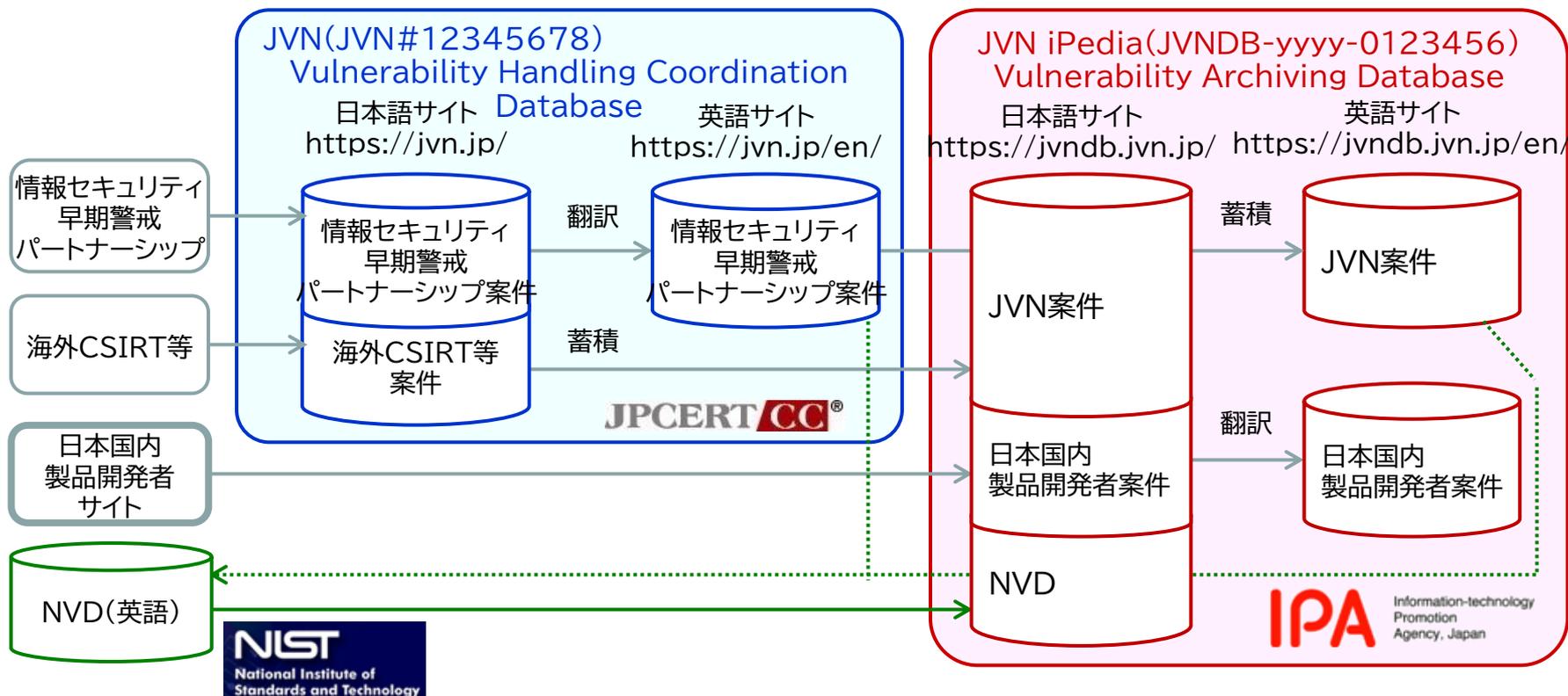
最終更新日	データベース登録番号	タイトル	CVSSv3
2019/03/26 New	<a href="#">JVNDDB-2019-001779</a>	CmsEasy におけるクロスサイトスクリプティングの脆弱性	6.1 (警告)
2019/03/26 New	<a href="#">JVNDDB-2019-001778</a>	SolarWinds Orion NPM における認可・権限・アクセス制御に関する脆弱性	9.8 (緊急)
2019/03/26 New	<a href="#">JVNDDB-2019-001777</a>	Linux Kerne における解放済みメモリの使用に関する脆弱性	7.8 (重要)
2019/03/26 New	<a href="#">JVNDDB-2019-001776</a>	WTCMS におけるクロスサイトスクリプティングの脆弱性	6.1 (警告)
2019/03/26 New	<a href="#">JVNDDB-2019-001775</a>	WTCMS におけるクロスサイトリクエストフォージェリの脆弱性	8.8 (重要)
2019/03/26 New	<a href="#">JVNDDB-2019-001774</a>	idreamsoft iCMS におけるクロスサイトリクエスト	5.7 (警告)

[出典] JVN  
<https://jvn.jp/>  
JVN iPedia  
<https://jvndb.jvn.jp/>

# 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

- 効率的な脆弱性対策を目指すことのできるグローバルな利活用基盤  
= (国際性:JVN + 地域性:JVN iPedia) × 利活用基盤(MyJVN)



バージョン  
チェック

セキュリティ設定  
チェック

脆弱性対策  
情報収集ツール



## MyJVN

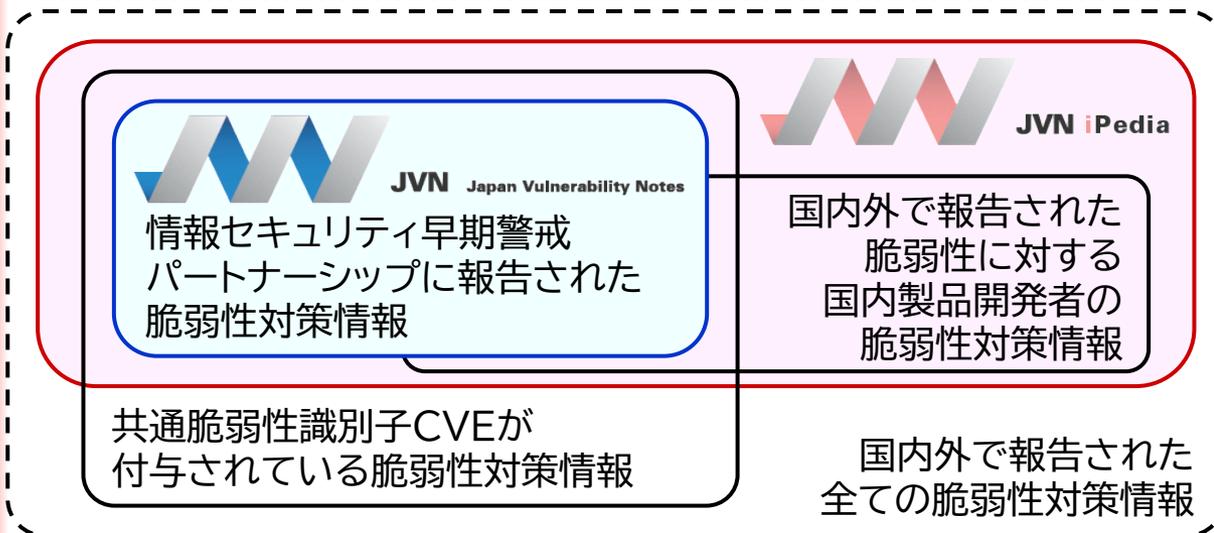
JVNとJVN iPediaに登録されている脆弱性対策情報を対策実施に直結したサービスに繋げる仕組みを提供する

## 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

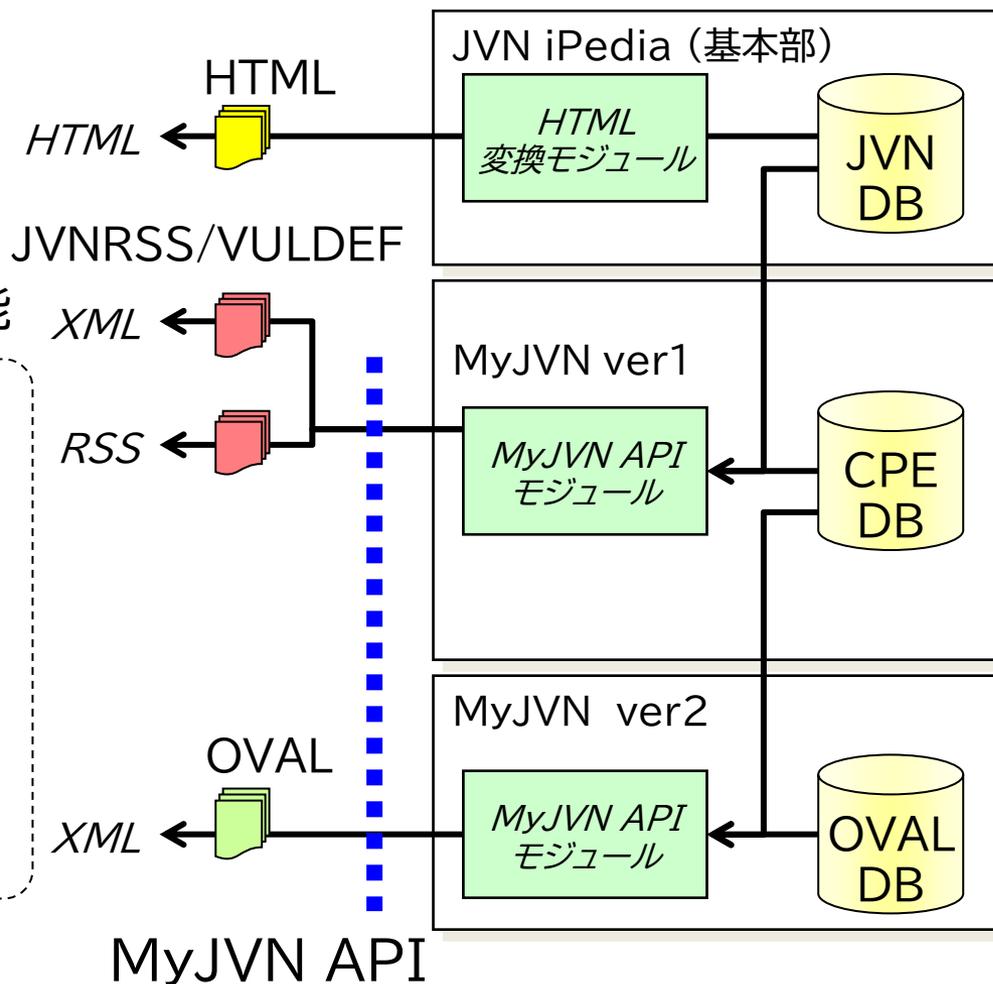
<http://jvndb.jvn.jp/apis/>

- 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 (mjcheck4)  
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  
バージョンチェック (.NET Framework版)



# 機械処理可能な情報活用基盤の整備 望ましい環境に向けて

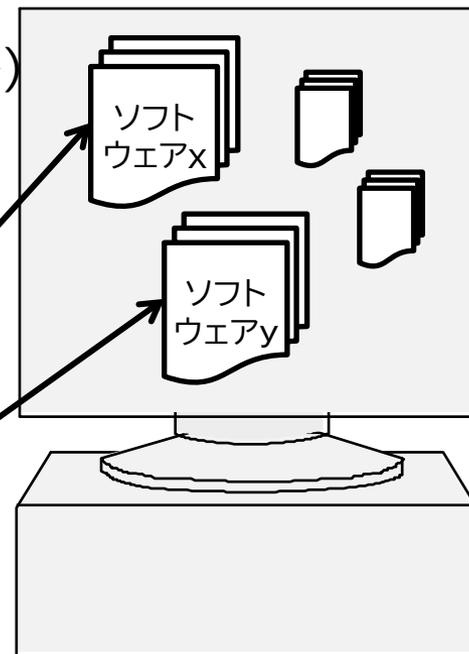
- 製品やカスタムアプリ(SIで開発した業務アプリケーションなど)に関わらず、情報連携により、人が介在しなくても良い脆弱性対策環境(脆弱性の影響有無の判定など)が実現できる。

重要なセキュリティ情報  
(MyJVN)



最新情報	重要なセキュリティ情報	脆弱性対策情報 [VFN]	自組織からの情報
2017年1月11日	重要	Microsoft 製品の脆弱性対策について(2017年1月)	
2017年1月11日	重要	Adobe Flash Player の脆弱性対策について(APS17-02)(CVE-2017-2025)	
2017年1月11日	重要	Adobe Reader および Acrobat の脆弱性対策について(APS17-01)(CVE-2017-2025)	
2016年12月22日	重要	SCVSA Client View において任意のコードが実行可能な脆弱性について(CVE-2016-7892)	
2016年12月14日	重要	Adobe Flash Player の脆弱性対策について(APS16-39)(CVE-2016-7892)	

IT資産一覧表  
(IT資産管理ツール)



# 機械処理可能な情報活用基盤の整備 望ましい環境に向けて



- グローバルな利活用基盤整備にあたってはNVDとの連携を考慮



# 機械処理可能な情報活用基盤の整備

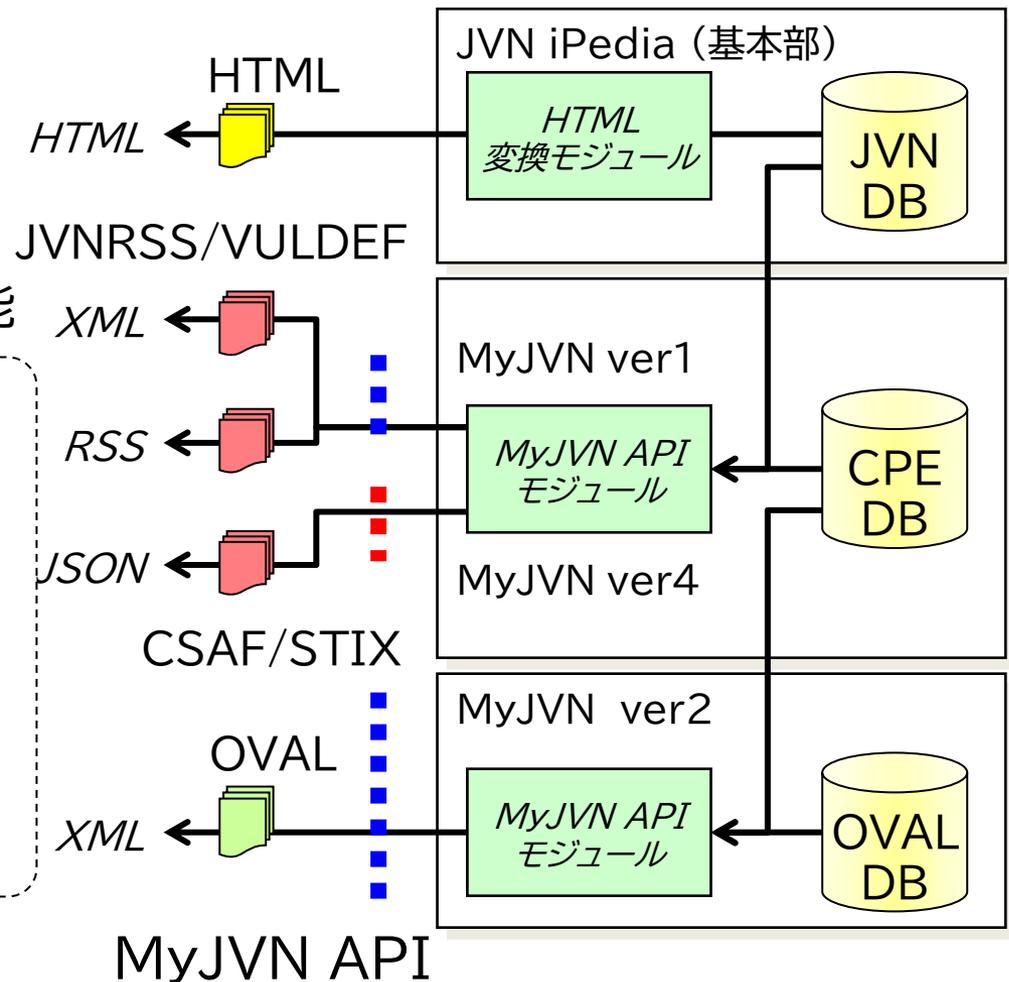
## JVN脆弱性対策機械処理基盤 ver4

- JVN iPediaを活用した環境

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 (mjcheck5)

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



# 機械処理可能な情報活用基盤の整備

## ポイント① MyJVN REST APIの拡張



- JSON形式のMyJVN REST APIの提供
  - 利用者自身ならびに開発者らがJVNを活用し、脅威情報と脆弱性情報を資産管理と連携することのできる自動化フレームワークの整備につなげるため、JVNに登録されている脆弱性対策情報を使用するためのWebサービスAPIを規定
- リクエストURLの基本構成
  - <https://jvndb.jvn.jp/myjvn?method=メソッド&パラメタ>

API 名	メソッド名	概要	Ver3.2 HND	Ver4.0 OKA
注意警戒情報一覧の取得	getAlertList	注意警戒情報一覧を取得します。	XML JSON	JSON
ベンダー一覧の取得	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。	XML	JSON
製品一覧の取得	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。	XML	JSON
脆弱性対策概要情報一覧の取得	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。	XML (JVNRSS)	JSON
脆弱性対策詳細情報の取得	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。	XML (VULDEF)	JSON (CSAF/STIX)
脆弱性対策情報統計データの取得	getStatistics	脆弱性対策情報を、登録件数、深刻度、脆弱性種別で集計したデータを取得します。	XML	JSON

# 機械処理可能な情報活用基盤の整備

## ポイント② JVN製品識別子の導入



- JVN製品識別子
  - MyJVN製品辞書で使用する管理用製品識別子
    - CPEとの互換性を考慮した形式
      - 主として{ベンダ}:{製品}を使用することで表記ゆれを防ぐ
    - バージョン範囲指定などの記述可とするため、構造型形式を採用
      - 構造型(purl, CPE)
      - 非構造型(ハッシュ、UUID)

### 【形式】

`jvnpid:1.0::{ベンダ}:{製品}:{バージョン}`

`cpe:2.3:{タイプ}:{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}:{その他}`

### 【例】

`jvnpid:1.0::sysbom.sample.jp:bomgen:0.46.3`

# 機械処理可能な情報活用基盤の整備

## ポイント② JVN製品識別子の導入



- JVN製品識別子
  - JVN製品識別子を用いた、多様な製品識別子を関連付けるための製品辞書の整備を通して、グローバルな連携を可能としつつ、製品識別子の付与されていない(国内)製品に識別子を付与し、その製品識別子を流通させることを実現する。

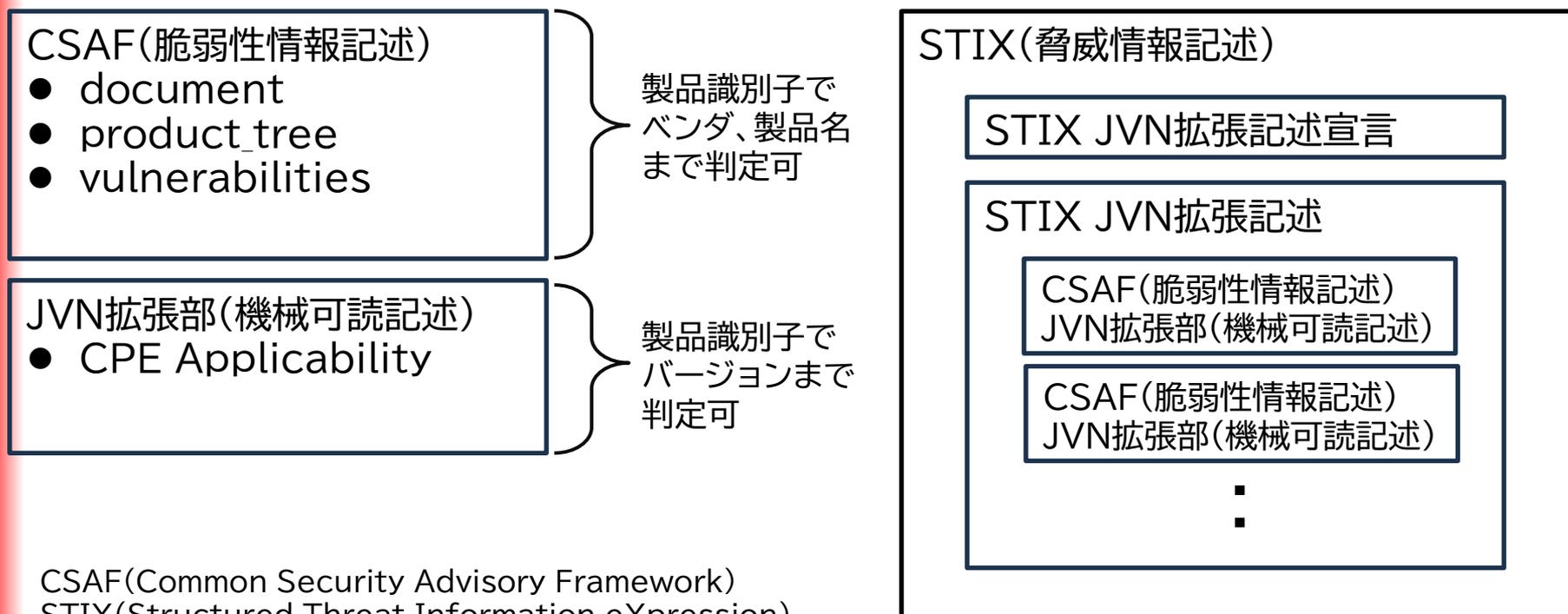
製品	JVN製品識別子
	製品名(日本語 英語)
	<ul style="list-style-type: none"><li>● CPE</li><li>● SWID</li><li>● UUID</li><li>● ベンダ固有のID</li><li>● SPDX識別子</li><li>● purl</li><li>● ハッシュ値</li></ul>

```
{
  "products": [
    {
      "product_id": "jvnpid:1.0::ipa.go.jp:myjvn_api",
      "pid": 99999999991001,
      "pname": "マイジエイブイエヌ API",
      "pname_i18n": { "ja": "マイジエイブイエヌ API", "en": "MyJVN API" },
      "product_ids": [
        {
          "cpe": "cpe:2.3:a:ipa.go.jp:myjvn_api:*:*:*:*:*:*",
          "id_refs": [
            { "nameType": "purl",
              "value": "pkg:rpm/ipa.go.jp/myjvn_api" },
            { "nameType": "swid",
              "value": "swid:ipa.go.jp+myjvn_api" }
          ]
        }
      ]
    }
  ]
}
```

# 機械処理可能な情報活用基盤の整備

## ポイント③ 統合配信のためのJSON形式の導入

- 脅威/脆弱性情報の統合配信のためのJSON形式
  - 脅威情報と脆弱性情報の記述形式としてCSAF、STIXを採用
  - 脆弱性対策詳細情報取得(getVulnDetailInfo)応答フォーマット
    - CSAF 2.1をベース
    - STIXでは、CSAFを内包



CSAF(Common Security Advisory Framework)  
STIX(Structured Threat Information eXpression)

# 機械処理可能な情報活用基盤の整備 ポイント③ 統合配信のためのJSON形式の導入

- 脅威/脆弱性情報の統合配信のためのJSON形式
  - 脅威情報と脆弱性情報の記述形式としてCSAF、STIXを採用
  - 脆弱性対策詳細情報取得(getVulnDetailInfo)応答フォーマット
    - CSAF 2.1をベース
    - STIXでは、CSAFを内包

製品識別子でベンダ、製品名まで判定可

CSAF(脆弱性情報記述)

- document
- product\_tree
- vulnerabilities

JVN拡張部(機械可読記述)

- CPE Applicability

```
{
  "product": {
    "product_id": "jvnpid:1.0::ipa.go.jp:myjvn_api",
    "name": "マイジェイブイエヌ API",
    "product_identification_helper": {
      "cpe": "cpe:2.3:a:ipa.go.jp:myjvn_api:*:*:*:*:*:*",
      "id_refs": [
        {
          "nameType": "swid",
          "value": "swid:ipa.go.jp+myjvn_api"
        }
      ]
    }
  },
  "category": "product_name",
  "name": "マイジェイブイエヌ API"
}
```

# 機械処理可能な情報活用基盤の整備

## ポイント③ 統合配信のためのJSON形式の導入

- 脅威/脆弱性情報の統合配信のためのJSON形式
  - 脅威情報と脆弱性情報の記述形式としてCSAF、STIXを採用
  - 脆弱性対策詳細情報取得(getVulnDetailInfo)応答フォーマット
    - CSAF 2.1をベース
    - STIXでは、CSAFを内包

製品識別子でバージョンまで判定可

CSAF(脆弱性情報記述)

- document
- product\_tree
- vulnerabilities

JVN拡張部(機械可読記述)

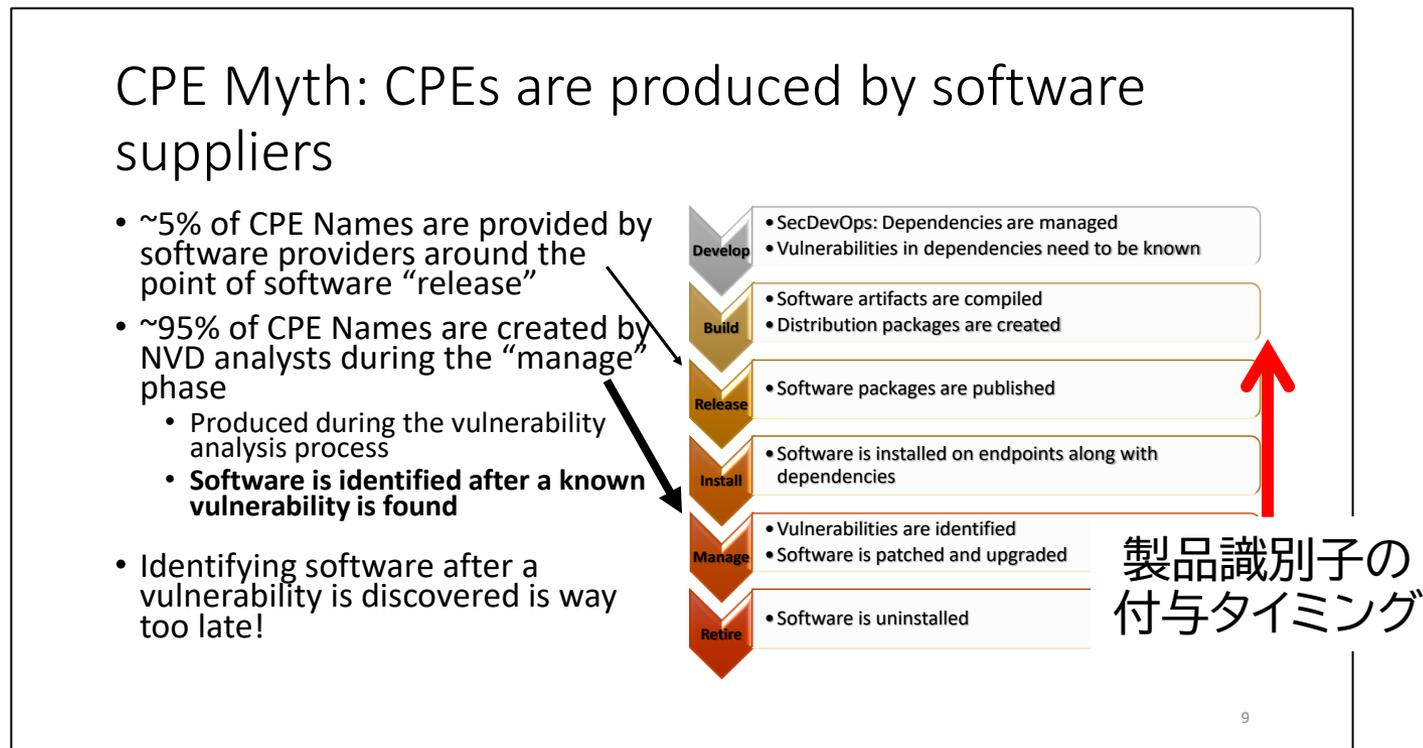
- CPE Applicability

```
{
  "cpeApplicability": [
    {
      "nodes": [
        {
          "operator": "OR",
          "negate": false,
          "cpeMatch": [
            {
              "vulnerable": true,
              "criteria": "jvnpid:1.0::ipa.go.jp:myjvn_api:8.1.2"
            },
            {
              "vulnerable": true,
              "criteria": "cpe:2.3:a:ipa.go.jp:myjvn_api:*:*:*:*:*:*",
              "versionStartIncluding": "0.0.0",
              "versionEndExcluding": "1.0.6"
            }
          ]
        }
      ]
    }
  ]
}
```

# 機械処理可能な情報活用基盤の整備

管理された製品識別子を提供するために、、、

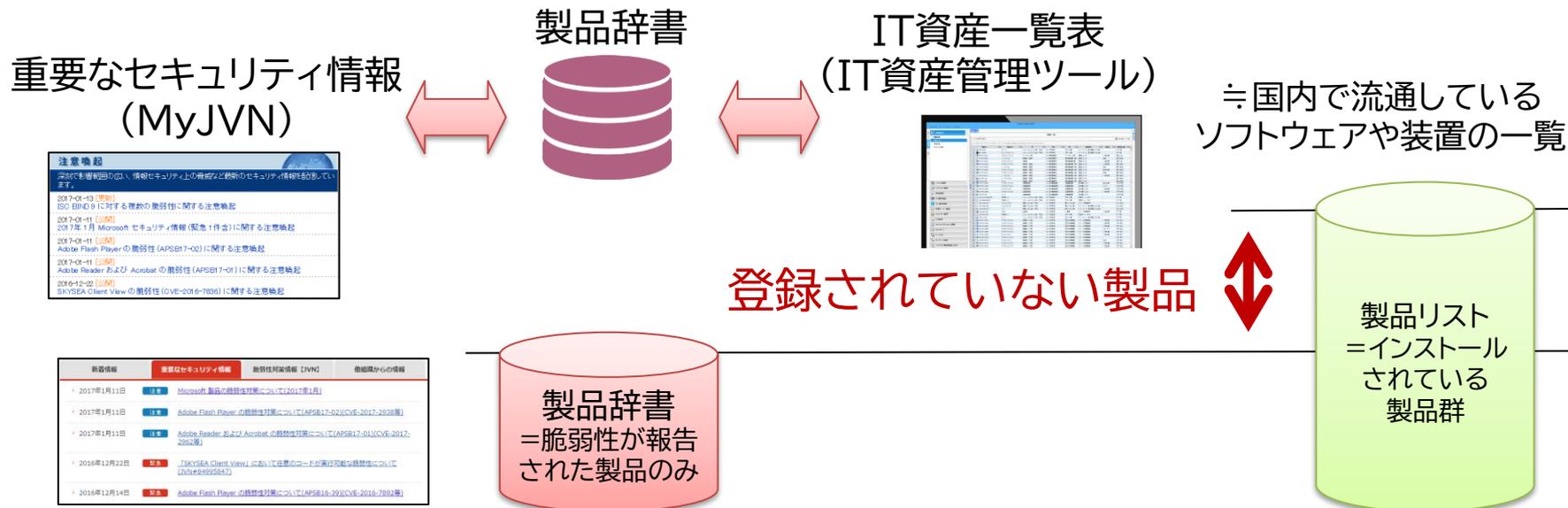
- 資産管理と脆弱性対策の融合のきっかけ
  - 製品識別子(CPE)付与の95%は、脆弱性が発見されてから、、、  
≡ リファレンス(製品辞書)に登録されるのは、脆弱性が発見されてから、、



# 機械処理可能な情報活用基盤の整備

管理された製品識別子を提供するために、..

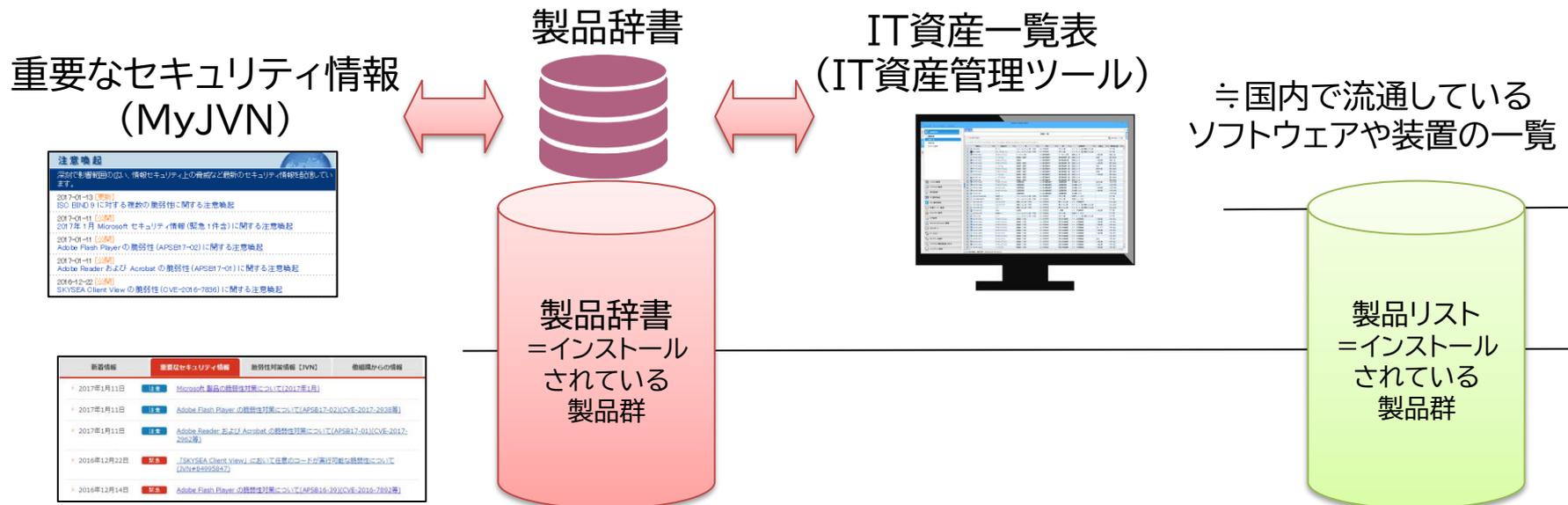
- 【短期的】製品識別という課題：現状
  - リファレンス(製品辞書)に登録されているソフトウェア名は、脆弱性が報告された製品のためのソフトウェア名から構成



# 機械処理可能な情報活用基盤の整備

管理された製品識別子を提供するために、..

- 【短期的】製品識別という課題：課題が解決された望ましい姿
  - リファレンス(製品辞書)に登録されているソフトウェア名は、国内で流通している製品のソフトウェア名から構成



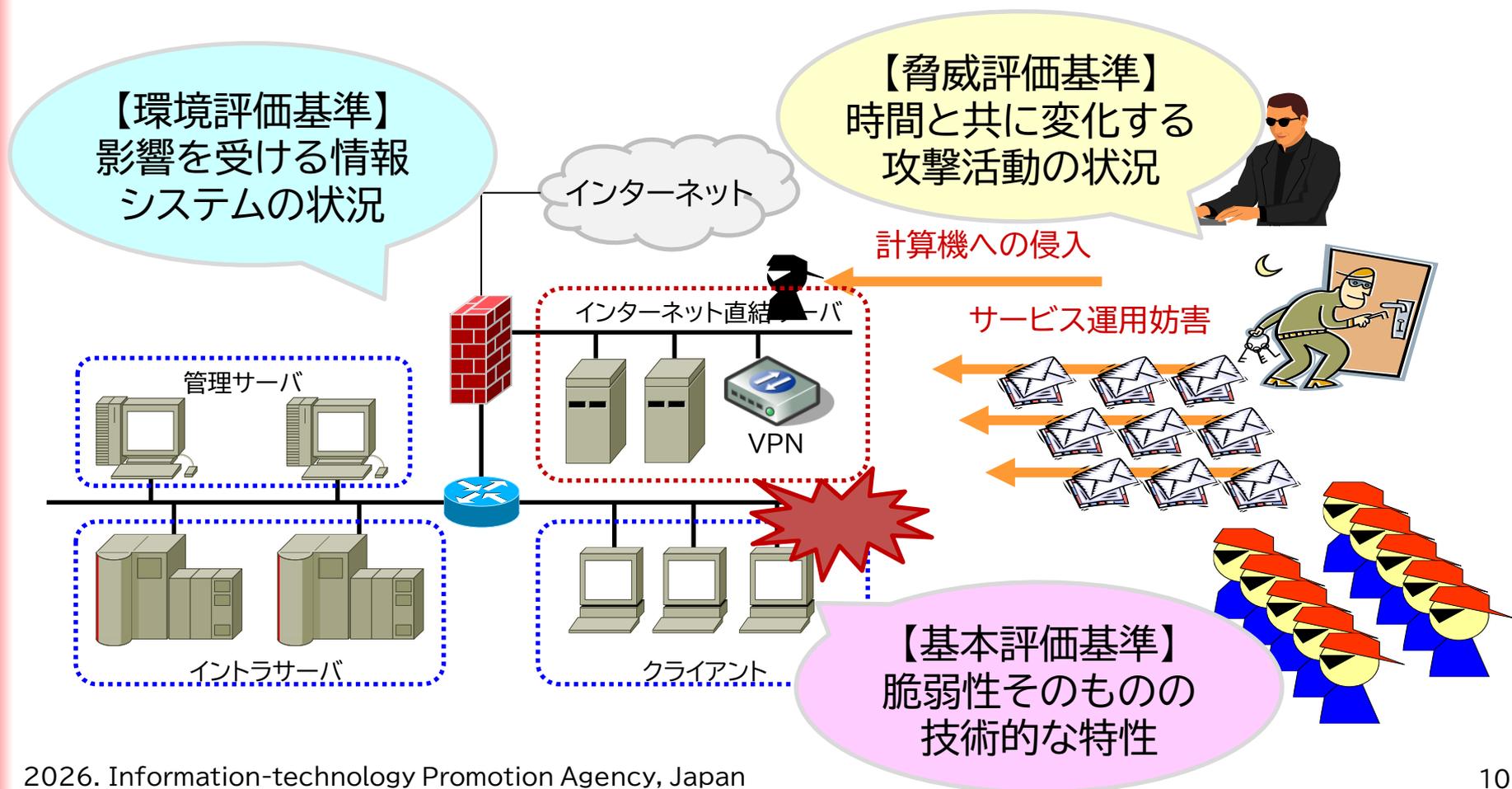
## トピック[2]



KEV、EPSS、SSVC

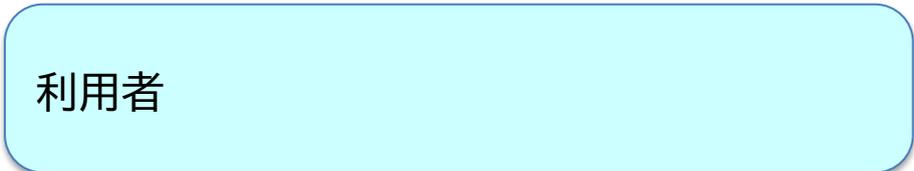
# CVSS(共通脆弱性評価システム) Common Vulnerability Scoring System

- 攻撃状況やシステムの重要度を加味して脆弱性の深刻度を表す



# CVSS(共通脆弱性評価システム) Common Vulnerability Scoring System

- 各項目を評価するための情報提供者は？

項目		関係性
基本評価基準	脆弱性そのものの 技術的な特性	 
脅威評価基準	時間と共に変化する攻撃 活動の状況	
環境評価基準	影響を受ける情報 システムの状況	

# CVSS(共通脆弱性評価システム) Common Vulnerability Scoring System

- 各項目を評価するための情報提供者は？

項目		関係性	
基本評価基準	脆弱性そのものの技術的な特性	<p>NVD JVN/JVN iPedia MyJVN</p>	<p>製品ベンダ セキュリティベンダ</p>
脅威評価基準	時間と共に変化する攻撃活動の状況	<p>現状を踏まえて KEV(悪用された既知の脆弱性一覧)</p>	<p>将来を踏まえて EPSS(脆弱性悪用予測評価システム)</p>
環境評価基準	影響を受ける情報システムの状況		<p>SSVC(役割に応じた脆弱性対応分類)</p>

- 脆弱性対策を支援するための情報ならびに手法

用語	概要
KEV Known Exploited Vulnerabilities catalog 悪用された既知の脆弱性一覧	米国サイバーセキュリティ・インフラセキュリティ庁(CISA)が公開している悪用が観測された既知の脆弱性の一覧
EPSS Exploit Prediction Scoring System 脆弱性悪用予測評価システム	脆弱性関連情報と脆弱性悪用の試みに関連するデータを用いて、機械学習を行い、その結果から今後30日間における悪用可能性を予測
SSVC Stakeholder-Specific Vulnerability Categorization 役割に応じた脆弱性対応分類	判断分岐点が設定された決定木を辿ることで、取るべき脆弱性対応の優先度を提示

# KEV(悪用された既知の脆弱性一覧)

## Known Exploited Vulnerabilities catalog



CISA(米国サイバーセキュリティ・インフラセキュリティ庁)が、2021年11月3日から公開している悪用が観測された既知の脆弱性の一覧

- 用途
  - 悪用されている既知の脆弱性に対処する。
- KEV掲載の条件
  - CVE(共通脆弱性識別子)が割り当てられている。
  - 脆弱性が悪用されているという信頼できる証拠がある。
  - 脆弱性に対処する方法がある。

# KEV(悪用された既知の脆弱性一覧)

## Known Exploited Vulnerabilities catalog



- 掲載情報
  - CVE番号 (cveID)
  - ベンダー/プロジェクト名 (vendorProject)
  - 製品名 (product)
  - 脆弱性名 (vulnerabilityName)
  - 追加日 (dateAdded)
  - 概要 (shortDescription)
  - 取るべきアクション (requiredAction)
  - 対応期日 (dueDate)
  - 既知のランサム攻撃での使用 (knownRansomwareCampaignUse)

# KEV(悪用された既知の脆弱性一覧)

## Known Exploited Vulnerabilities catalog



CISCO | ADAPTIVE SECURITY APPLIANCE (ASA) AND FIREPOWER THREAT DEFENSE (FTD)

### [CVE-2020-3259](#)

#### Cisco ASA and FTD Information Disclosure Vulnerability

Cisco Adaptive Security Appliance (ASA) and Firepower Threat Defense (FTD) contain an information disclosure vulnerability. An attacker could retrieve memory contents on an affected device, which could lead to the disclosure of confidential information due to a buffer tracking issue when the software parses invalid URLs that are requested from the web services interface. This vulnerability affects only specific AnyConnect and WebVPN configurations.

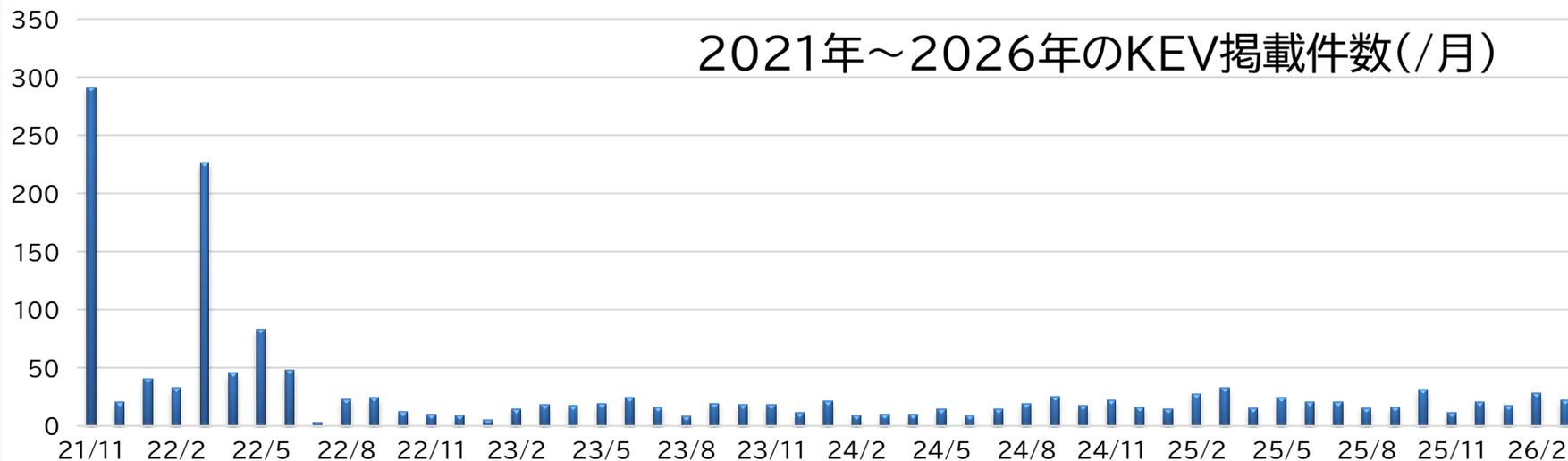
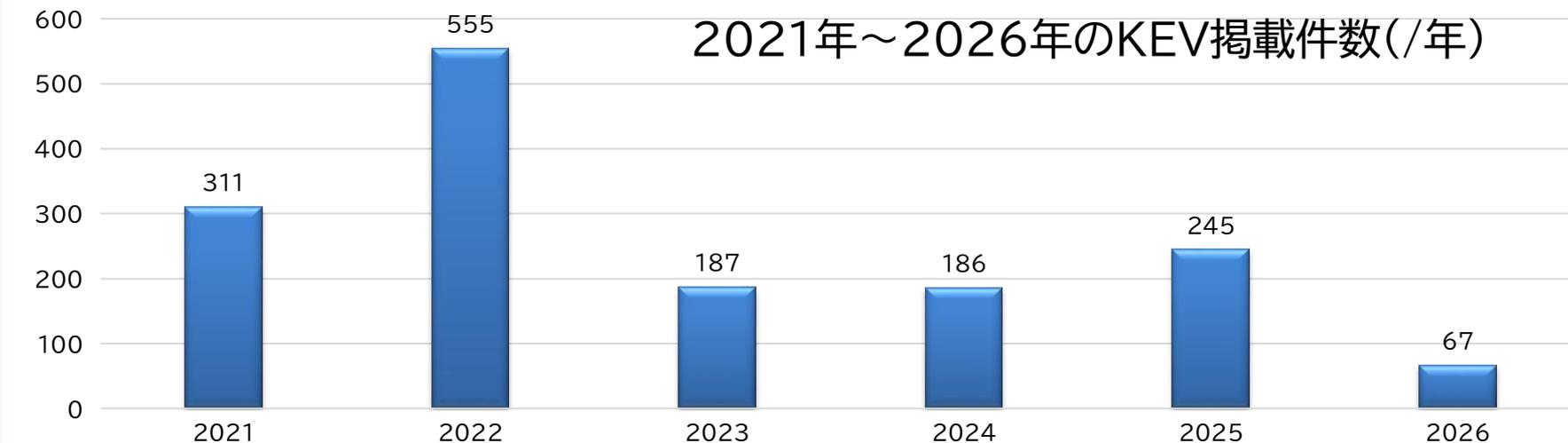
- **Action:** Apply mitigations per vendor instructions or discontinue use of the product if mitigations are unavailable.
- **Known To Be Used in Ransomware Campaigns?:** Known
- **Date Added:** 2024-02-15
- **Due Date:** 2024-03-07

[Resources and Notes +](#)

JVNDB-2020-005198  
CVE-2020-3259  
Cisco Adaptive Security Appliance  
および Cisco Firepower Threat  
Defense ソフトウェアにおける脆弱性  
2020年06月09日

<https://tools.cisco.com/security/center/content/CiscoSecurityAdvisory/cisco-sa-asaftd-info-disclose-9eJtycMB>

# KEV(悪用された既知の脆弱性一覧) Known Exploited Vulnerabilities catalog



# EPSS(脆弱性悪用予測評価システム)

## Exploit Prediction Scoring System



脆弱性関連情報と脆弱性悪用の試みに関連するデータを用いて、機械学習を行い、その結果から今後30日間における悪用可能性を予測

- 用途
  - 今後、既知の脆弱性が悪用される可能性を踏まえて対処する。
- 悪用可能性
  - EPSSスコア
    - 今後30日間に悪用される確率
    - 0～100% の確率で表記
  - EPSSパーセンタイル
    - 悪用される可能性を相対的に表現
    - EPSSスコアを小さい順で並べたとき、あるスコアがデータの小さい方から見て何%の位置にあるかを表す

# EPSS(脆弱性悪用予測評価システム)

## Exploit Prediction Scoring System

- 仕組み
  - 脆弱性関連情報  
ベンダ情報、CVEが公開されてからの日数、脆弱性の説明で使用されている用語、脆弱性の種別、CVSSスコア、掲載されているCVE情報、攻撃コードの公開など)
  - 脆弱性悪用の試みに関連するデータ  
データパートナーから提供されるハニーポット、IDS/IPSセンサーなどのデータで、実際に脆弱性悪用が試みられた証拠



# EPSS(脆弱性悪用予測評価システム)

## Exploit Prediction Scoring System



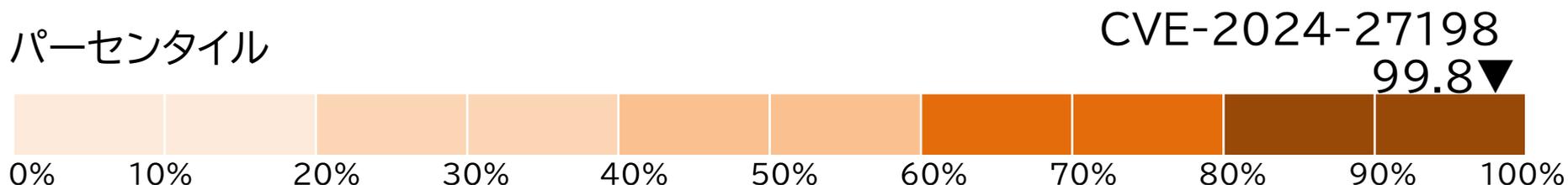
- 悪用可能性
  - EPSSスコア(今後30日間に悪用される確率を0~100%で表記)
  - EPSSパーセンタイル(EPSSスコアを小さい順で並べたとき、あるスコアがデータの小さい方から見て何%の位置にあるかを表す)

### EPSSスコア

We selected the 48 highest rated CVEs published in the last 30 days. They are shown here with the CVE and EPSS score.

CVE-2024-27198 97.2%	CVE-2024-23225 0.1%	CVE-2024-21795 0.1%	CVE-2024-24794	CVE-2023-43552	CVE-2023-47534
CVE-2024-1709 93.5%	CVE-2024-26000 0.1%	CVE-2024-21812 0.1%	JVNDB-2024-002919 CVE-2024-27198 JetBrains の TeamCity における脆弱性 2024年03月07日		
CVE-2024-27199 0.9%	CVE-2024-23296 0.1%	CVE-2024-22097 0.1%			
CVE-2024-1212 0.7%	CVE-2024-27743 0.1%	CVE-2024-23305 0.1%			
CVE-2024-26492 0.1%	CVE-2024-27744 0.1%	CVE-2024-23310 0.1%			

### パーセンタイル



# EPSS(脆弱性悪用予測評価システム)

## Exploit Prediction Scoring System

- 情報発信の形態
  - EPSSスコアを、CSVやJSON(API経由)などで提供
  - 例 [https://epss.cyentia.com/epss\\_scores-current.csv.gz](https://epss.cyentia.com/epss_scores-current.csv.gz) (最新データ)  
[https://epss.cyentia.com/epss\\_scores-2024-11-30.csv.gz](https://epss.cyentia.com/epss_scores-2024-11-30.csv.gz)
- 悪用可能性
  - EPSSスコア(今後30日間に悪用される確率を0~100%で表記)
  - EPSSパーセンタイル(EPSSスコアを小さい順で並べたとき、あるスコアがデータの小さい方から見て何%の位置にあるかを表す)

EPSSスコア ↓

↓ パーセンタイル

	A	B	C	D
1	#model_version:v2 score_date:2024-11-30T00:00:00+0000			
2	cve	epss	percentile	
246575	CVE-2024-27193	0.00043	0.10425	
246576	CVE-2024-27194	0.00043	0.10425	
246577	CVE-2024-27195	0.00043	0.10425	
246578	CVE-2024-27196	0.00043	0.10425	
246579	CVE-2024-27197	0.00043	0.10425	
246580	CVE-2024-27198	0.97238	0.9988	
246581	CVE-2024-27199	0.0099	0.84158	
246582	CVE-2024-27200	0.00043	0.10425	
246583	CVE-2024-27201	0.00045	0.15556	
246584	CVE-2024-27202	0.00043	0.10425	
246585	CVE-2024-27204	0.00043	0.10425	
246586	CVE-2024-27205	0.00043	0.10425	

# EPSS(脆弱性悪用予測評価システム)

## Exploit Prediction Scoring System



- 情報発信の形態
  - EPSSスコアを、CSVやJSON(API経由)などで提供
  - 例 <https://api.first.org/data/v1/epss?cve=CVE-2024-27198> (最新データ)  
<https://api.first.org/data/v1/epss?cve=CVE-2024-27198&date=2024-03-19>
- 悪用可能性
  - EPSSスコア(今後30日間に悪用される確率を0~100%で表記)
  - EPSSパーセンタイル(EPSSスコアを小さい順で並べたとき、あるスコアがデータの小さい方から見て何%の位置にあるかを表す)

```
1 {
2   "status": "OK",
3   "status-code": 200,
4   "version": "1.0",
5   "access": "public",
6   "total": 1,
7   "offset": 0,
8   "limit": 100,
9   "data": [
10    {
11      "cve": "CVE-2024-27198",
12      "epss": "0.972090000", ← EPSSスコア
13      "percentile": "0.998110000", ← パーセンタイル
14      "date": "2024-03-19"
15    }
16  ]
17 }
```

# SSVC(役割に応じた脆弱性対応分類)

## Stakeholder-Specific Vulnerability Categorization



判断分岐点が設定された決定木を辿ることで、取るべき脆弱性対応の優先度を提示

- 用途
  - 役割(=対象者)に応じて、脆弱性対応の優先度を分類する。
- 対象者(=役割)
  - デプロイヤー(修正プログラム利用側)
  - サプライヤー(修正プログラム提供側)
  - コーディネーター(脆弱性ハンドリング調整役)[脆弱性情報公開]
  - コーディネーター(脆弱性ハンドリング調整役)[脆弱性トリアージ]

# SSVC(役割に応じた脆弱性対応分類)

## Stakeholder-Specific Vulnerability Categorization



- デプロイヤー(修正プログラム利用側)  
Deployer Patch Application Priority
  - 判断分岐点
    - 脆弱性の悪用状況(Exploitation): 未報告、実証可能、攻撃可能
    - システムの露出状況(System Exposure): 閉域的、限定的、オープン
    - 攻撃の自動化可能性(Automatable): なし、あり
    - 影響の大きさ(Human Impact): 低、中、高、極めて高
  - 脆弱性対応の優先度
    - Immediate: 迅速な対応
    - Out-of-cycle: 定期メンテナンス以外の早い時期での対応
    - Scheduled: 定期メンテナンス時に対応
    - Defer: 現時点では対応不要

# SSVC(役割に応じた脆弱性対応分類)

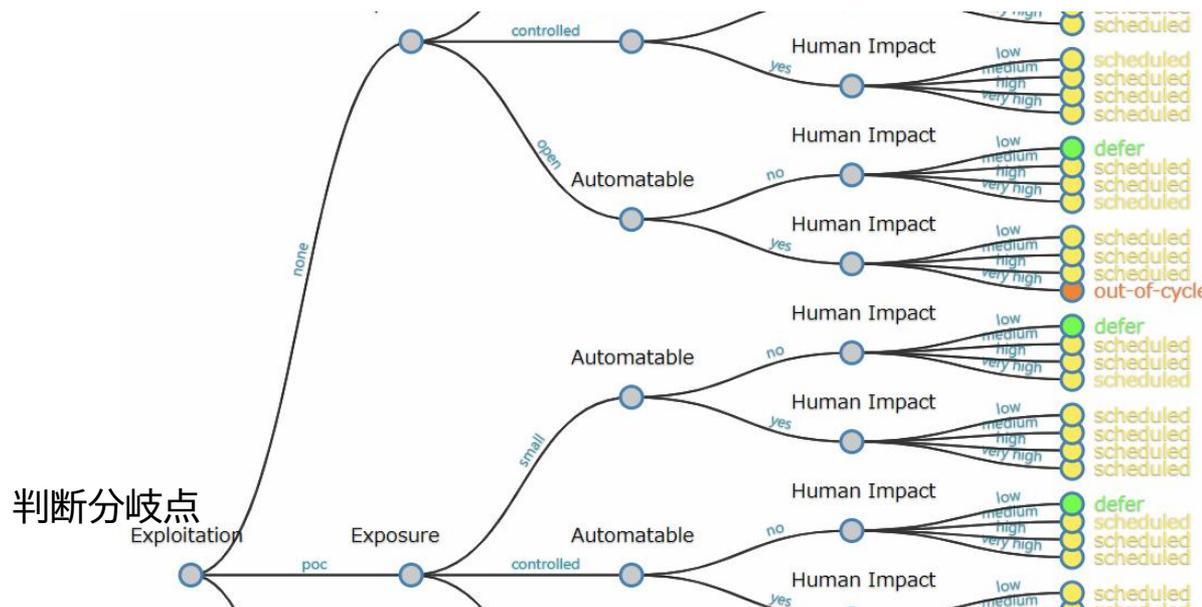
## Stakeholder-Specific Vulnerability Categorization



- デプロイヤー(修正プログラム利用側)  
Deployer Patch Application Priority

- 判断分岐点

- 脆弱性の悪用状況(Exploitation): 未報告、実証可能、攻撃可能
- システムの露出状況(Exposure): 閉域的、限定的、オープン
- 攻撃の自動化可能性(Automatable): なし、あり
- 影響の大きさ(Human Impact): 低、中、高、極めて高



脆弱性対応の優先度  
計72件の選択肢

# SSVC(役割に応じた脆弱性対応分類)

## Stakeholder-Specific Vulnerability Categorization

- デプロイヤー(修正プログラム利用側)

Calculator - SSVC: Stakeholder- x

certcc.github.io/SSVC/ssvc-calc/

SSVC: Stakeholder-Specific Vulnerability Categorization

Use public CVE AP ...

CERTCC/SSVC v2025.9.5 ☆177 ♪43

Dryad SSVC Calculator

Sample Decision Models: **Deployer Patch Application Priority (1.0.0)** Multiple CVEs Export

CVE/Identifier Use public CVE API data Check All

Exploitation	System Exposure	Automatable	Human Impact	Defer, Scheduled, Out-of-Cycle, Immediate
<input checked="" type="checkbox"/> None	<input checked="" type="checkbox"/> Small	<input checked="" type="checkbox"/> No	<input checked="" type="checkbox"/> Low	Defer
<input checked="" type="checkbox"/> Public PoC	<input checked="" type="checkbox"/> Controlled	<input checked="" type="checkbox"/> Yes	<input checked="" type="checkbox"/> Medium	Scheduled
<input checked="" type="checkbox"/> Active	<input checked="" type="checkbox"/> Open		<input checked="" type="checkbox"/> High	Out-of-Cycle
			<input checked="" type="checkbox"/> Very High	Immediate

Hide help text

Human Impact

Low Medium High Very High

Defer Defer Scheduled Scheduled

Automatable

Human Impact

Low Medium High Very High

Defer Scheduled Scheduled Scheduled

Human Impact

Low Medium High Very High

Defer Scheduled Scheduled Scheduled

System Exposure Automatable

No Yes No

# SSVC(役割に応じた脆弱性対応分類)

## Stakeholder-Specific Vulnerability Categorization



- サプライヤー(修正プログラム提供側)  
Supplier Patch Development Priority
  - 判断分岐点
    - 脆弱性の悪用状況(Exploitation): 未報告、実証可能、攻撃可能
    - 攻撃の実効性(Utility): 手間、効率的、極めて効率的
    - 技術的な影響(Technical Impact): 部分的、全面的
    - 安全性全般への影響(Public-Safety Impact): 最小、重大
  - 脆弱性対応の優先度
    - Immediate: 迅速な対応
    - Out-of-cycle: 定期メンテナンス以外の早い時期での対応
    - Scheduled: 定期メンテナンス時に対応
    - Defer: 現時点では対応不要

# SSVC(役割に応じた脆弱性対応分類)

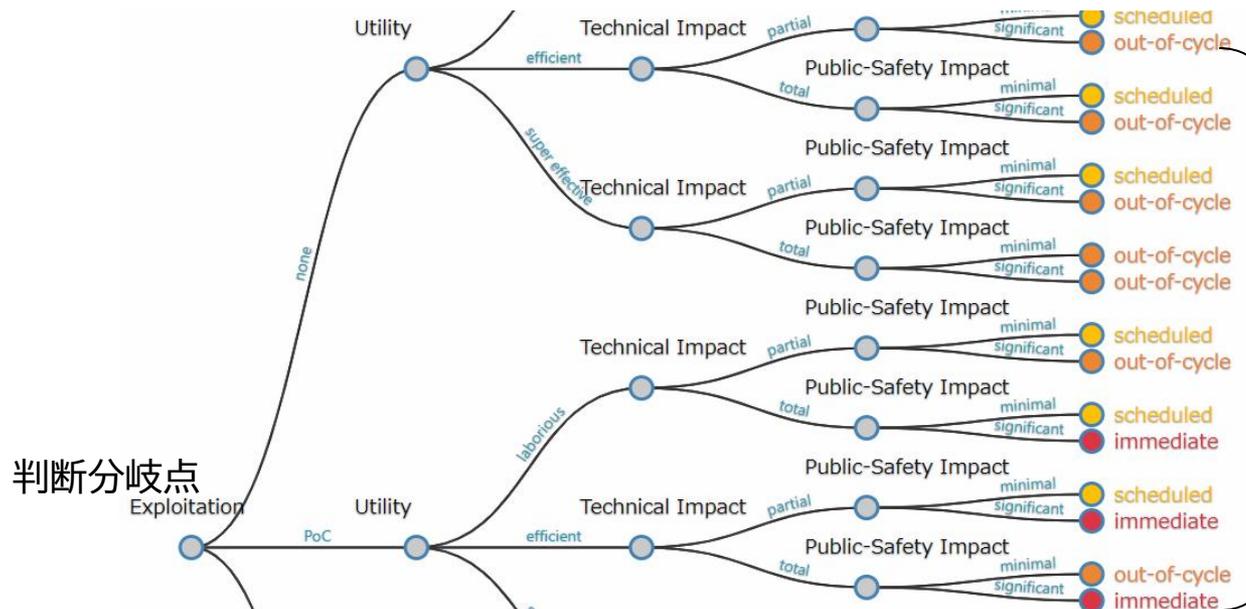
## Stakeholder-Specific Vulnerability Categorization



- サプライヤー(修正プログラム提供側)  
Supplier Patch Development Priority

- 判断分岐点

- 脆弱性の悪用状況(Exploitation): 未報告、実証可能、攻撃可能
- 攻撃の実効性(Utility): 手間、効率的、極めて効率的
- 技術的な影響(Technical Impact): 部分的、全面的
- 安全性全般への影響(Public-Safety Impact): 最小、重大



脆弱性対応の優先度  
計36件の選択肢

# SSVC(役割に応じた脆弱性対応分類)

## Stakeholder-Specific Vulnerability Categorization



- コーディネーター(脆弱性ハンドリング調整役)[脆弱性情報公開]  
Coordinator Publish Decision Table
  - 判断分岐点
    - サプライヤーの関与(Supplier involvement): 対応済、調整中、応答なし
    - 脆弱性の悪用状況(Exploitation): 未報告、実証可能、攻撃可能
    - 情報の位置付け(Value added): 初版、改訂版、付録
  - 脆弱性対応の優先度
    - publish: 公開
    - don't publish: 公開しない

# SSVC(役割に応じた脆弱性対応分類)

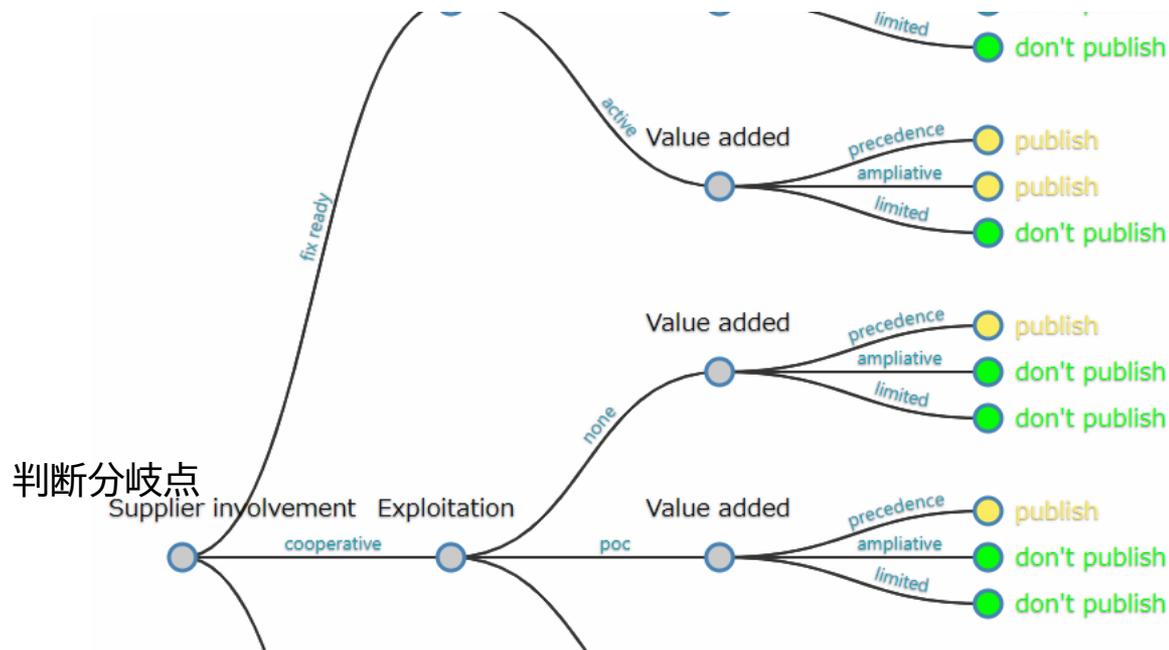
## Stakeholder-Specific Vulnerability Categorization



- コーディネーター(脆弱性ハンドリング調整役)[脆弱性情報公開]  
Coordinator Publish Decision Table

- 判断分岐点

- サプライヤーの関与(Supplier involvement): 対応済、調整中、応答なし
- 脆弱性の悪用状況(Exploitation): 未報告、実証可能、攻撃可能
- 情報の位置付け(Value added): 初版、改訂版、付録



脆弱性対応の優先度  
計27件の選択肢

# SSVC(役割に応じた脆弱性対応分類)

## Stakeholder-Specific Vulnerability Categorization



- コーディネーター(脆弱性ハンドリング調整役)[脆弱性トリアージ]  
Coordinator Triage
  - 判断分岐点
    - 情報の公開(Report Public): されている、されていない
    - 報告者の報告調整(Supplier Contacted): 良い、悪い
    - 報告の信頼性(Report Credibility): あり、なし
    - 対応関係者の数(Supplier Cardinality): シングル、マルチ
    - 対応関係者の関与(Supplier Engagement): 積極的、応答なし
    - 攻撃の実効性(Utility): 手間、効率的、極めて効率的
    - 社会安全への影響(Public Safety Impact): 最小、重要
  - 脆弱性対応の優先度
    - decline: 対応せず
    - track: 様子見
    - coordinate: 対応する

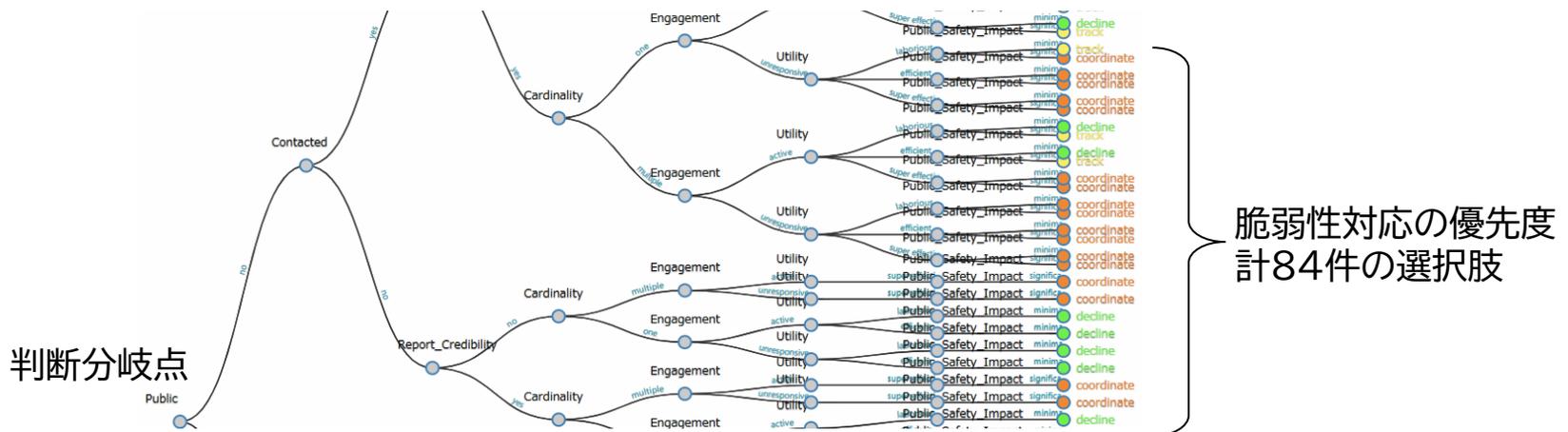
# SSVC(役割に応じた脆弱性対応分類)

## Stakeholder-Specific Vulnerability Categorization

- コーディネーター(脆弱性ハンドリング調整役)[脆弱性トリアージ]  
Coordinator Triage

- 判断分岐点

- 情報の公開(Report Public): されている、されていない
- 報告者の報告調整(Supplier Contacted): 良い、悪い
- 報告の信頼性(Report Credibility): あり、なし
- 対応関係者の数(Supplier Cardinality): シングル、マルチ
- 対応関係者の関与(Supplier Engagement): 積極的、応答なし
- 攻撃の実効性(Utility): 手間、効率的、極めて効率的
- 社会安全への影響(Public Safety Impact): 最小、重要



# 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  
2026年03月26日