

システム管理とセキュリティ

enPiT-Proセキュリティ : ProSecセキュリティリテラシー サイバーセキュリティ
立命館大学情報理工学部 上原哲太郎

Futurize.

きみの意志が、未来。



「システム」とは？

- システム
 - 機能の異なる多くの要素（技術・部品）が密接に体系化された組み合わせ、全体として多くの機能を発揮する集合体
 - 「システム工学」 Systems Engineering
 - システムの設計・制御・運用管理・効率効果などを研究する学問
- システムは本質的に複雑系

情報とは何か？情報システムとは何か？

- 情報

コンピュータシステムや磁気媒体等に保存されているデータのみならず、紙に印刷されたものやコンピュータシステムに入力される前のメモおよび職員の会話や個人の記憶も含む



- 情報システム

ハードウェア、ソフトウェアのみならず、それらを適切に運用・管理するために必要なすべての人や物を含む

(JIS X 5080などによる定義)



情報システムとは

- **システム**とは、いくつかの異なる要素が連携を取りながら、いくつかの機能を発揮するもの
- **情報のライフサイクル**とは、情報の生成・蓄積・変換および伝達を言う
- **情報管理**とは、情報のライフサイクル全体または一部を、効率よく、正確に、かつ安全に行うこと。
- **情報システム**とは、情報の管理をコンピュータによって行うシステム




情報システムのセキュリティとは

• 情報システムのライフサイクル



- この全てをセキュアに保つことといえる
- プロジェクトマネジメント・ソフトウェア工学
 - ・ サービスサイエンス



DXが進めばセキュリティの重点は機密性より可用性にシフトする

• 情報のセキュリティ = CIA

機密性
Confidentiality
(情報漏洩防止)

完全性
Integrity
(処理誤り防止)

可用性
Availability
(処理停止防止)



機密性ばかりが話題になるが
企業のDXが進むほど
可用性が事業継続上重要に

6



攻撃者は「スキのある組織」を狙う組織の大小はあまり関係ない

愉快犯→思想犯

技術誇示目的

思想信条の表現

目的を持つ外部犯

怨恨

金銭目的

破壊工作・諜報

内部の事故

ミス・ポリシー違反

目的を持つ内部犯

怨恨

金銭目的

最近はメインは「金銭」「諜報」の模様

一般的なシステム調達の流れ

事業要件定義

- 経営層が事業の要件を定義
- 予算の大枠を決めシステム化を決定

業務要件定義

- 業務部門が業務のシステム化を決定
- システム化の範囲と要件を定義

RFP

- システム化に関する提案を募集

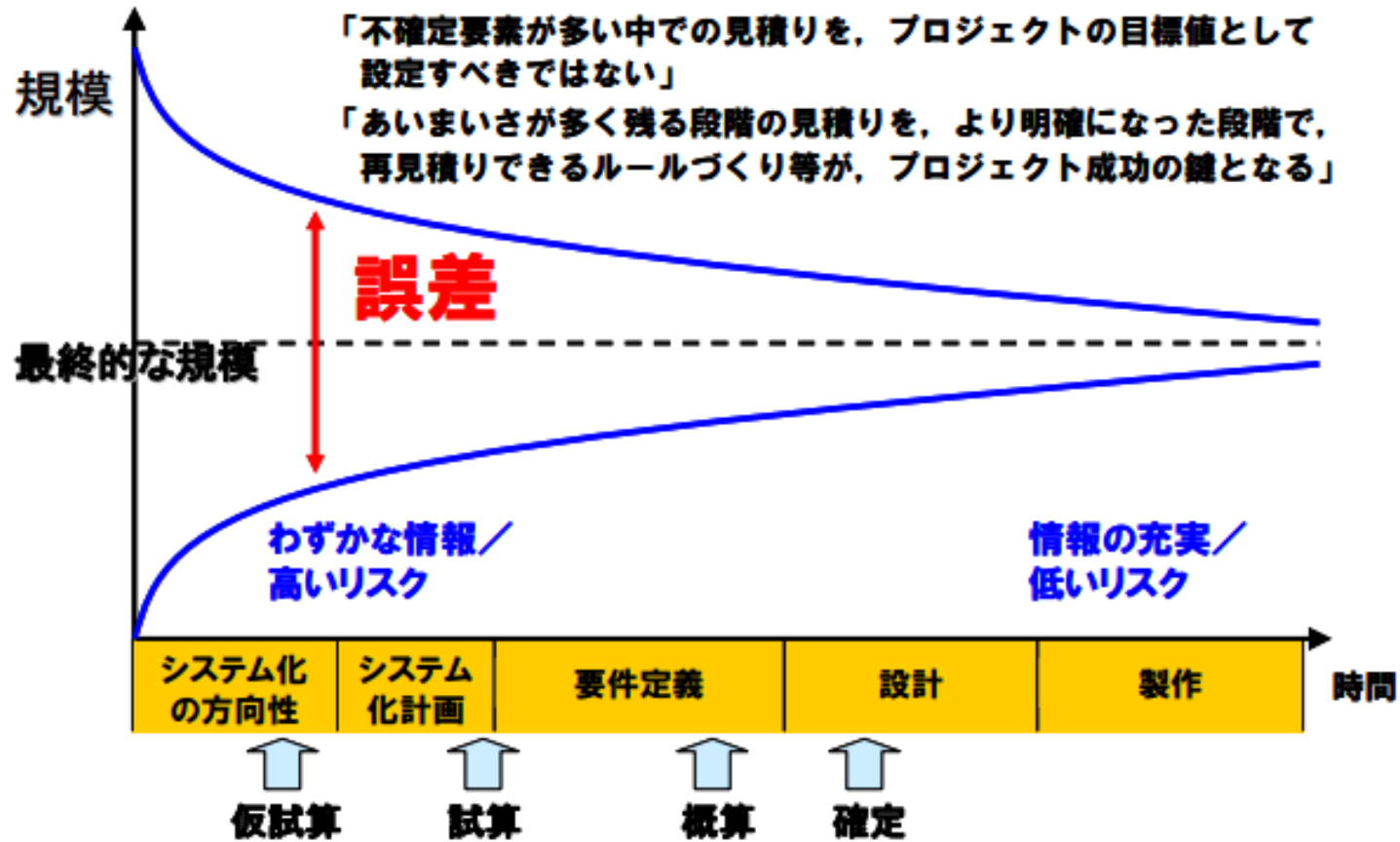
システム要件定義

- 発注者のシステム部門と受注者が共同でシステムの要件を定義

発注開発

- 実際の開発に入る

最近は何次にも渡るRFP (多段階見積もり)



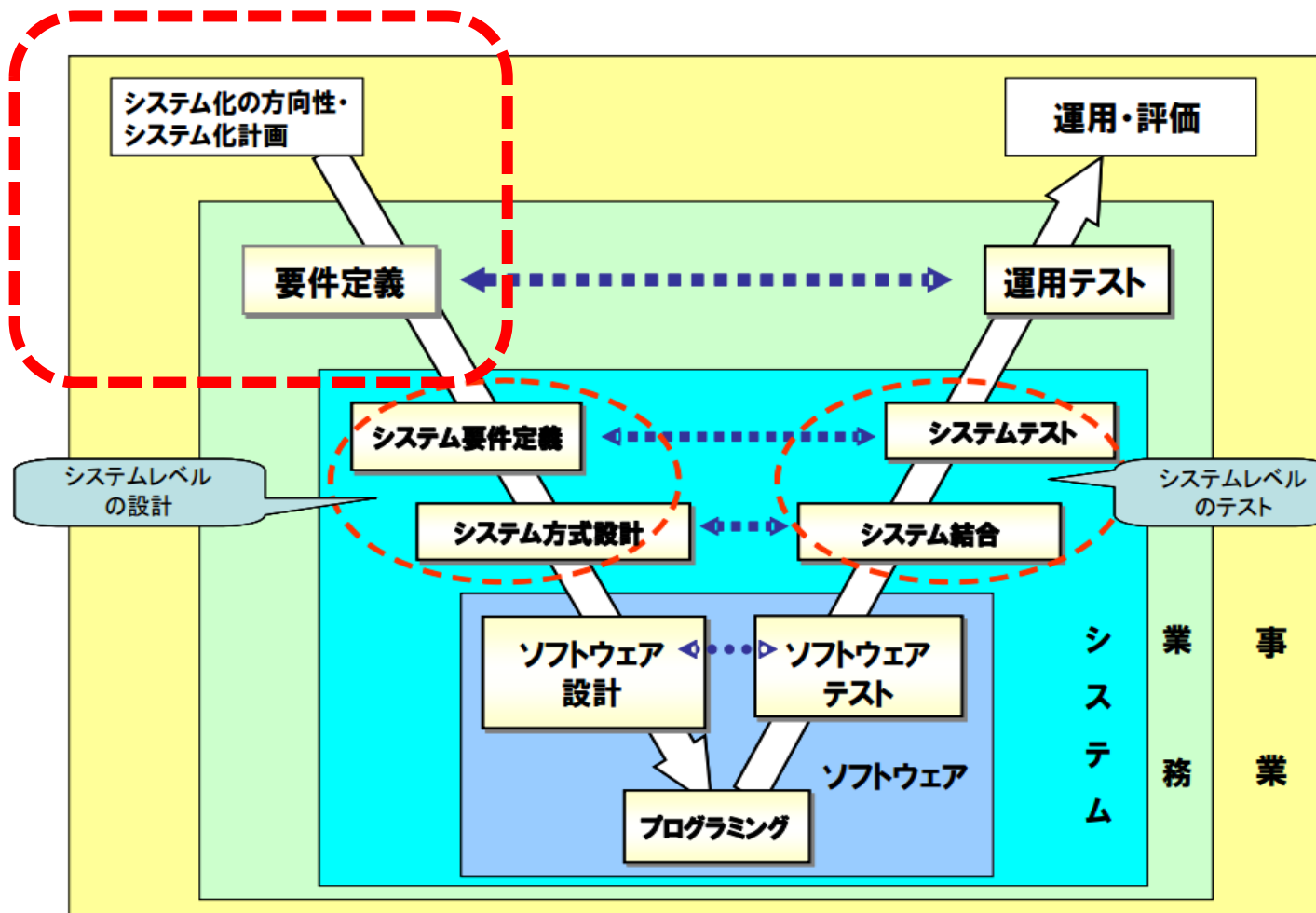
※SEC BOOKS「経営者が参画する要求品質の確保 ～超上流から攻めるIT化の勘どころ～（第2版）」より引用、一部改修

IPAが提唱する「超上流工程」と多段階見積もり

IPAが提唱するV字型モデル システムのSecure by Design化

超上流工程
ここでできるだけ
セキュアにしたい

最近
Shift Left
シフトレフト
とも言われる



政府調達における Secure by Design

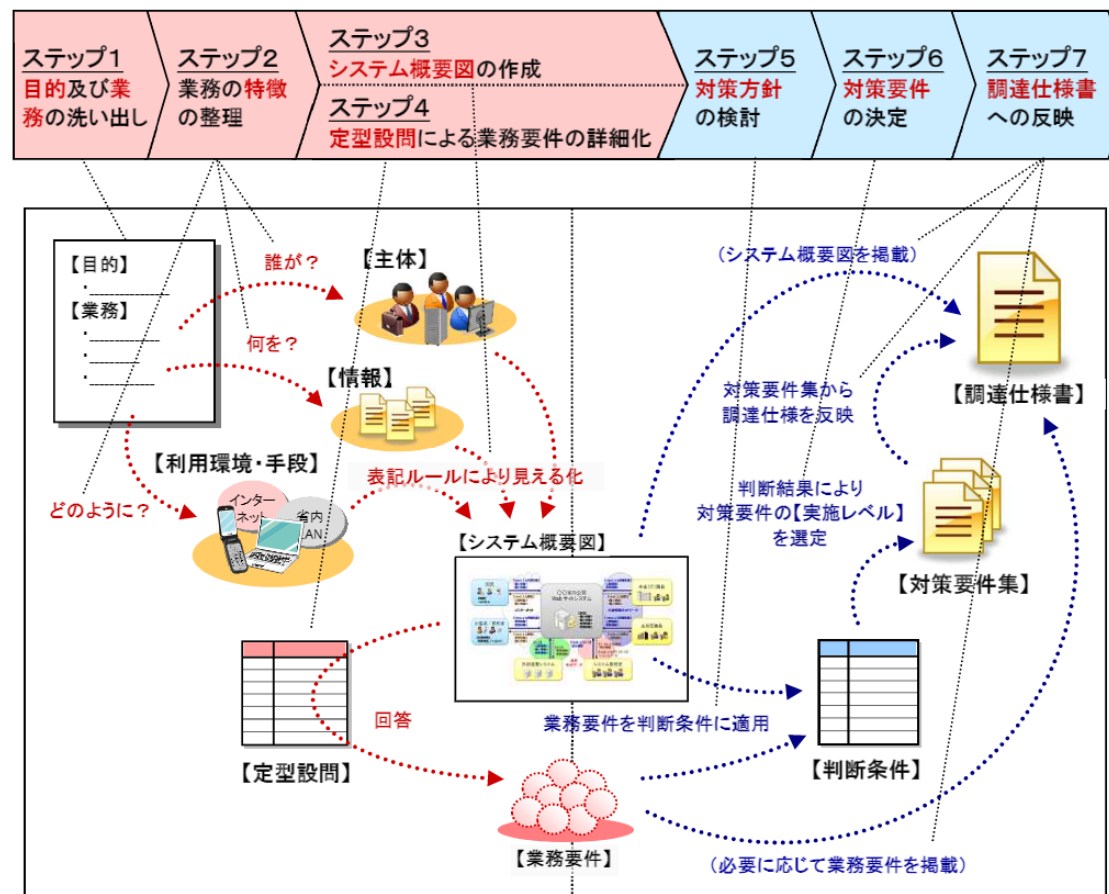
情報システムに係る政府調達における セキュリティ要件策定マニュアル


2022 年 7 月 29 日

内閣官房 内閣サイバーセキュリティセンター

(1) 業務要件の検討 [※ 他の方法による代替可]

(2) セキュリティ要件の策定





実システムは複数のシステムの組合せなので…

- パッケージの組合せ＋カスタマイズかフルスクラッチ開発か？
 - 要求をどこまで実装に擦り合わせられるか？
 - 多くの相反する要求への解を求める
 - コスト（開発と運用保守）
 - パフォーマンス（処理性能と使い勝手）
 - セキュリティ
-
- 簡単に解は出ない



苦しむ自治体情報システム

動かないコンピュータ

+ 特集をフォロー

他人の住民票が誤発行される謎バグの真相、富士通Japanの「稚拙」設計に専門家も驚く

鈴木 慶太 日経クロステック／日経コンピュータ

2023.04.21

有料会員限定



全3273文字

PR

急速に進むマルチクラウド化で直面する、新たな「サイロ化」問題を解決に導く100%に近い保守パーツの遵守率をあるサーバーベンダーが実現 その理由は？
IT／製造／建設分野の製品・サービス選択支援情報サイト：日経クロステックActive

2023年3月、横浜市のコンビニの証明書交付サービスでトラブルが発生した。住民が住民票の写しの交付を申請したところ、別人のものが発行されたのだ。原因は富士通Japanが手掛けるサービスの不具合だった。利用が増えて負荷が高まり、潜在的なバグが表面化した。国がマイナンバーカード普及に力を注ぐ中、冷や水を浴びせる結果となった。

「個人情報漏洩にも当たる事案で大変重要な問題であり、遺憾に思っている」――。河野太郎デジタル相は2023年3月31日の閣議後記者会見において、厳しい口調でこう述べた。河野氏が言及したのは、横浜市内で発生したコンビニの証明書交付

地方公共団体の情報システムの標準化に向けた取組

- 住民記録システムなど、地方公共団体が基本的な事務を処理するための情報システム（基幹系情報システム）は、事務の処理の大半が法令で定められているが、地方公共団体が利便性等の観点から個別に機能のカスタマイズ等を行っており、その結果、
 - ・ 維持管理や制度改正時の改修等において**地方公共団体は個別対応を余儀なくされ、負担が大きい**
 - ・ **情報システムの差異の調整が負担**となり、クラウドによる共同利用が円滑に進まない
 - ・ 住民サービスを向上させる最適な取組みを、**迅速に全国へ普及させることが難しい** 等の課題が生じている。
- こうした課題を解決するため、**地方公共団体の情報システムの標準化を推進することが必要。**

「経済財政運営と改革の基本方針2020」(令和2年7月17日閣議決定)

第3章「新たな日常」の実現

1. 「新たな日常」構築の原動力となるデジタル化への集中投資・実装とその環境整備（デジタル・ニューディール）

(1) 次世代型行政サービスの強力な推進 ― デジタル・ガバメントの断行

③ 国・地方を通じたデジタル基盤の標準化の加速

国・地方を通じたデジタル基盤の統一・標準化を早急に推進するため、地方制度調査会の答申を踏まえ、法制上の措置を講じた上で、財源面を含め国が主導的な支援を行う。地方自治体の基幹系業務システムの統一・標準化について関係府省庁は内閣官房の下この1年間で集中的に取組を進める。年内に標準を設ける対象事務の特定と工程化を行う。

「成長戦略フォローアップ」(令和2年7月17日閣議決定)


6. 個別分野の取組

(2) 新たに講ずべき具体的施策

Ⅲ) スマート公共サービス

② 地方公共団体のデジタル化の推進

地方自治体の情報システムをより広域的なクラウドに移行するためには、各地方自治体が行っている情報システムのカスタマイズを無くすことが重要であり、国が主導して進めている標準化の取組を着実に進めるとともに、システムの機能要件等について法令に根拠を持つ標準を設けることとすべきであるとする地方制度調査会の答申を踏まえ、関係府省庁が連携して、セキュリティの基準を含め、情報システムの標準化について総合的な対応を検討し、早期に結論を得る。



セキュリティ要求工学？

- 多くのソフトウェア要求工学は
「業務の要求を把握し実装との間で調整する作業」
- コストの議論も活発ではなかった
- これからはセキュリティを要求項目に
- システム要件にセキュリティを入れる必要
「設計をセキュアに (Security by Design)」
- しかし課題も多い
- 要求に組み入れる動機付けが弱い



セキュリティ要求工学の比較

	SQUARE	CLASP	セキュリティ開発ライフサイクル
機関	SEI	Fortify	Microsoft
概要	要求エンジニアが主導して関係者との相互交流に基づきセキュリティ要求を定義	ソフトウェア開発サイクルを安全にするための一般的なプロセス要素を定義	設計段階でセキュリティ機能をモデル化
目標	セキュリティゴールを識別	ビジネス要求と関連付けて、リスクを緩和し不足や矛盾のないセキュリティ要求を抽出	顧客要望とセキュリティ標準へのコンプライアンスに基づくセキュリティ機能要求を識別
ミスユース脅威	生産物の定義に基づいてリスク分析し、セキュリティ要求を抽出・優先順位付け・レビュー	—	脅威モデルの設計でセキュリティ機能を抽出
資源	—	ネットワーク設計やデータ設計の段階で資源と信頼境界を識別	脅威モデルの設計で保護対象とする資源を識別

セキュリティ要求工学の要素

	留意点	備考
セキュリティ目標	上位レベルのセキュリティ・ゴールを識別する 1) 顧客が望む要求であること 2) 法制度、標準、ポリシーに適合すること	段階的な手順が開発者のために必要である
資産	1) 顧客、システム所有者、攻撃者の視点から資産の価値とリスクを判断する 2) 資産ごとに機密性、一貫性、可用性について視点ごとに優先度を判断する 3) リスクを発生確率と影響度で定義する	リスクの完全な定量化は困難
脅威分析	1) 最も重要な資産に注力する 2) 脅威カタログを事前に準備する 3) 最も重要な脅威を攻撃木を用いて識別する	脅威カタログの例: STRIDE（なりすまし、改変、否認、情報暴露、サービス拒否、特権昇格）
文書化	1) セキュリティ要求を「how」ではなく「what」で記述する 2) セキュリティ目標や資産をポリシーとして記述する 3) セキュリティ要求を1箇所にまとめる	セキュリティ要求の優先度と追跡性が必要である



参考：セキュリティ要求工学で読んでおくべき文献

- IEEE Software 2008 Jan/Feb.
Inger Anne Tondel, Martin Gilje Jaatun,
and Per Hakon Meland,
“Security Requirements for the Rest of Us: A Survey”
- 安全工学 54巻6号 (2015)
大久保隆夫 「セキュリティ要求工学」

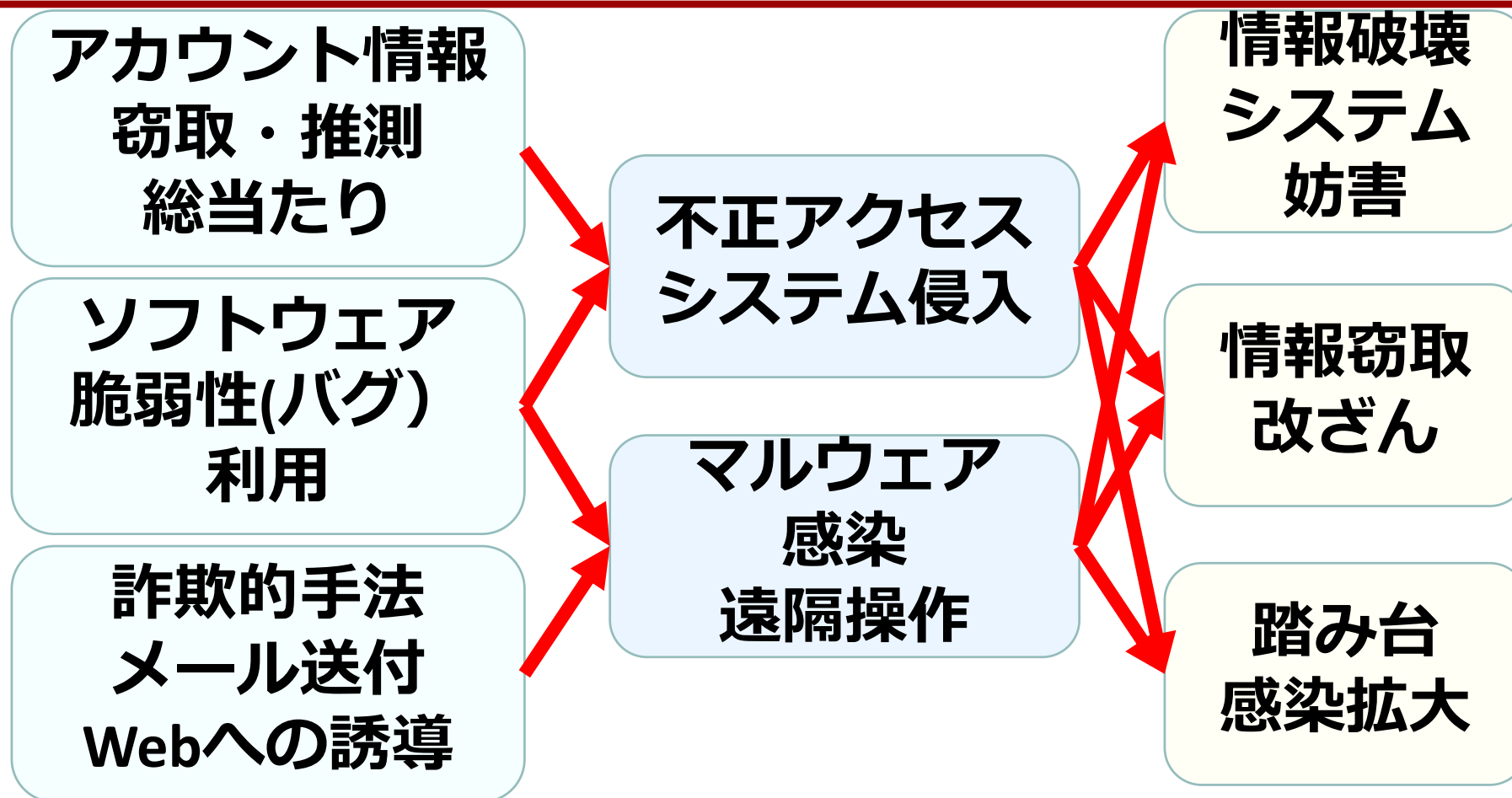



抽象的脅威より具体的リスク分析

- STRIDE手法(Microsoft)
 - Spoofing(なりすまし)
 - Tampering(改ざん)
 - Repudiation(否認)
 - Information Disclosure(情報漏えい)
 - Denial of Service(サービス拒否)
 - Elevation of Privilege(特権の昇格)
- これらの可能性と脅威の程度を評価



サイバー攻撃の構造





システム／ソフトウェアの 脆弱性とは何か？

- 脆弱性＝Vulnerability セキュリティホールともいう
- ソフトウェアの「バグ」、設定ミス、まれに仕様の誤り
- 外部からの操作でシステムに想定外の挙動をさせ、サービスに支障をきたす要因となるものの全て
- 影響の深刻度別に4つに大別できる
 - サービス不能攻撃（DoS）が可能になるもの
 - 特定のデータを入力するとシステムが停止する等
 - 権限昇格が可能になるもの
 - 認証後の一般ユーザに管理者権限を奪われる
 - 認証の回避が可能になる＝無認証ユーザの悪用を許すもの
 - データ漏洩等の原因になりやすい
 - 「丸見え」、XSS、SQL injectionなどは代表例
 - 「任意のプログラムの実行」が可能になるもの
 - OS command injection、バッファオーバーフロー等

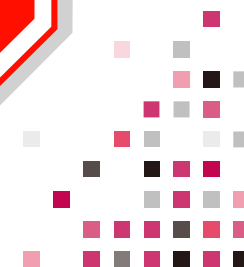


不正
アクセスの
原因



脆弱性によるサービス不能攻撃 (Denial of Service=DoS Attack)

- 外部からの特定のデータの入力によりプログラムが異常終了する/無限ループに入る等して機能が停止する
- 有名な例：Windowsへの”WinNuke”攻撃
- Windowsのファイル共有サービス(NetBIOS)に存在したバグを用いる
- 特定の packets を Port 139 に送るだけで OS そのものがバグで異常終了する (OOB Bug)
 - これ以降、「Port139はファイアウォールで遮断」が一般化
- 1997年7月認識され、攻撃プログラムが公開され広まる
数週間後にはMicrosoftは修正プログラムを公開
- しかしその後何年も多くのユーザは適用せず、影響が長く残る
- 同様の例は枚挙に暇なし

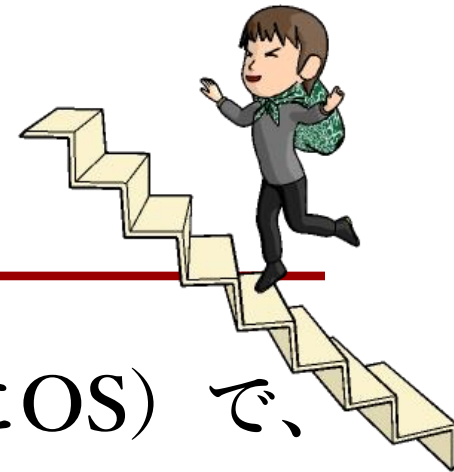




参考：脆弱性型以外のDoS

- 帯域消費型・サーバ過負荷型
 - 大量のアクセスを浴びせて、ネットワーク回線を消費させ尽くすか、サーバの処理能力を浪費する
 - 接続回線業者（ISP）との連携による回避
- SYN Flooding Attack
 - SYNフラグのみ立てたTCPセグメントをサーバに送りつける
 - 送信元IPアドレスは詐称の場合も

権限の昇格



- 一般ユーザと管理者権限が分離されたシステム（特にOS）で、
一般ユーザに管理者権限を奪われてしまうタイプの
バグ/設定ミス
- 特にOSで、
「管理者権限で動作し、一般ユーザから処理要求を受け付けて
行うプログラム」のバグが原因になりやすい
- たとえばUNIXの”setuid”なプログラムWindowsの”サービス”群など
- 権限の昇格に「任意のプログラムの実行」が加わると
大変な脅威



認証の回避が行われる（＝不正アクセス可能）

- 無認証のユーザに認証を回避されるタイプのバグ
／設定ミス
- 本来認証を必要とする操作を外部から認証なしで行われてしまう
- SQLインジェクションはじめWebアプリケーションで
比較的多いタイプの脆弱性
- 認証機構の設定ミス・設計ミスでもよく起きる

「任意のプログラムが実行可能」 Arbitrary Code Executoin(ACE)

- 「任意のプログラムを外部から送り込み、実行可能」
or 「システム内の任意のプログラムを起動可能」
- 後者は「インジェクション」あたりで可能な場合多し
- WindowsやLinuxのように内部がよく知られたシステムでは後者も前者と同様の脅威
 - Bashやcmd.exe=シェルプログラムを起動されたら？
- Webアプリケーションでは、結構、ある・・・
- SQLサーバの設定が悪い場合のインジェクション
(実質的にOSコマンドインジェクションが可能)
- バイナリ脆弱性は発見も利用もそれなりに高い技術が必要
 - だが、多くのウイルスが実際に利用している



「任意のプログラム」が実行されるまで

- 「コードの送り込み」と
「そのコードの起動」の2つのことが可能な脆弱性
- 単一の脆弱性で双方の条件がそろう場合大変な脅威
- バイナリではコードの送り込みには
「バッファオーバーフロー」を用いる
- スクリプト言語では方法は多彩



バッファオーバーフローとは



- 特にC言語/C++言語で記述されたプログラムに多く潜む脆弱性
- スタック領域またはヒープ領域にデータを代入するコードに不備があり、「入りきらない」量のデータを代入してしまう
- スタックオーバーフロー／ヒープオーバーフローと呼ぶ
- それを用いて任意のコードをそのプログラムに送り込む
- プログラム自身が乗っ取られる
- そのプログラムが動作している（OS上の）ユーザ権限が奪取される
- サーバ系のプログラムの多くが管理者権限で動いているので脅威が大きい

ネットに露出したプログラムにC/C++の利用はリスク大



今月のWindows Update(Trendmicro Security Blogより)

[CVE-2023-29336](#) – Win32k特権昇格の脆弱性

- これは今回のリリース時に悪用が確認された唯一の脆弱性です。Microsoft関連のこのタイプの脆弱性で攻撃の悪用事例がなかった月としては2022年5月まで遡る必要があります。今回の特権昇格の脆弱性も、通常、コード実行の脆弱性と組み合わせてマルウェアの拡散のために悪用されます。この脆弱性がセキュリティ企業から報告された点からも、こうした攻撃シナリオの懸念は妥当といえるでしょう。Microsoft社からの報告では、これらの悪用事例がどの程度の規模であるかの情報は提供されていません。

[CVE-2023-29325](#) – Windows OLE リモートコード実行の脆弱性

- 記述にはOLEとありますが、特に注意すべきコンポーネントはOutlookです。この脆弱性が悪用されると、攻撃者は細工したRTFメールを送信することで、影響を受けたシステム上でのリモートコード実行が可能となります。プレビューペインが攻撃経路となるため、作成されたメッセージを開封しなくても被害を受ける可能性があります。Outlookが可能性の高い攻撃経路である一方で、他のOfficeアプリケーションも影響を受けます。今回の脆弱性は、広く周知されたものの1つであり、Twitterでも議論が展開されています。Microsoft社からは、いくつかの回避策が提供されていますが、今回の更新を速やかにテストして適用することを推奨します。

[CVE-2023-24941](#) – Windowsネットワークファイルシステムリモートコード実行の脆弱性

- この脆弱性はCVSS 9.8に分類されており、悪用されると、未認証の攻撃者がリモートで特権を駆使し、影響を受けたシステム上で任意のコード実行が可能となります。この悪用では、ユーザ側の操作は必要ありません。この脆弱性に関して興味深い点は、NFSバージョン4.1に存在するものの、NFS 2.0、3.0のバージョンには存在しないことです。このため、以前のバージョンにダウングレードすることでこの脆弱性悪用のリスクを軽減できますが、Microsoft社は、2022年5月の[CVE-2022-26937](#)への修正パッチがインストールされていない限り、この軽減策を使用しないようにと警告しています。したがって、今回の更新を速やかにテストして適用することを推奨します。

[CVE-2023-24955](#) – Microsoft SharePoint Server リモートコード実行の脆弱性

- この脆弱性は、Pwn2OwnバンクーバーにおいてSTAR Labsチームによって実演され、ターゲットにしたサーバ上でコード実行を行うために使用された攻撃フローの一部でした。この脆弱性の悪用では、認証が必要となりますが、同イベントでは、認証回避との組み合わせにより実演されました。この悪用手法は、現実の攻撃シナリオでも起こり得ることだといえます。なお、SharePointへの対応では、今月リリースされる他の修正パッチもありますが、公表された悪用のリスクに完全に対処するには、さらに追加の修正パッチが必要になります。今後数か月で、Pwn2Ownで示されたこれらの脆弱性への修正対応がリリースされることを期待しましょう。

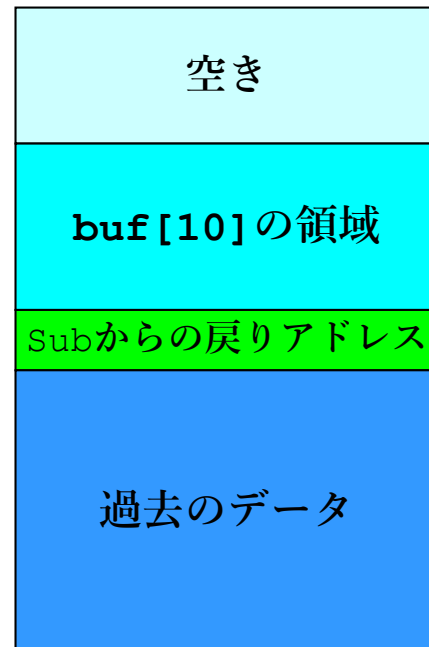
https://www.trendmicro.com/ja_jp/research/23/e/the-may-2023-security-update-review.html

スタックオーバーフロー

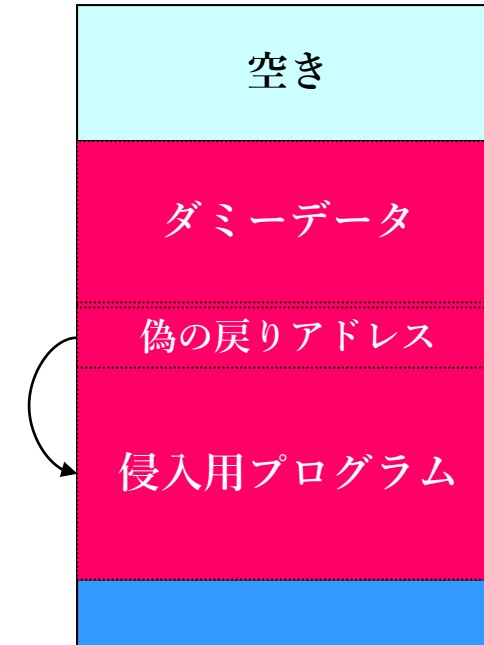
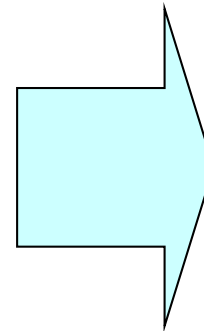
- C言語の文字列データは「NULLで終わる」以外のルールがないことを利用してバッファ領域を破壊

スタック

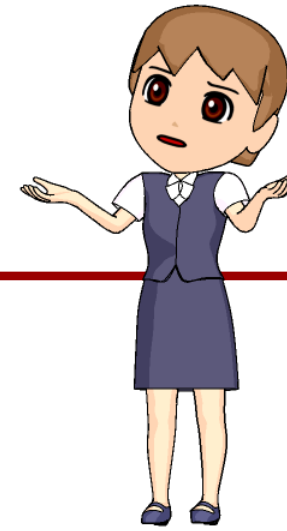
```
void sub()  
{  
    char buf[10];  
    ...  
    strcpy(buf, p);  
    ...  
    return;  
}
```



*pに10文字以上
突っ込むと・・・



こんなことも原因になる



- 整数オーバーフロー

- ```
int data, bufsize;
char buf[1024], *p;
// ... pには外部からのデータが入る
data = (int) *p; // データの先頭にサイズがある
bufsize = data * 4; // 1データあたり4バイト
if (bufsize <= 1024) { // ちゃんと境界チェック!
 strncpy(buf, p, bufsize);
} else {
 ERROR();
}
```

# バッファオーバーフローの対策

- プログラミングの工夫
  - バッファオーバーフローになりやすいライブラリや「常套句」を禁止
    - strcpyよりstrncpyなど しかし完璧は難しい
  - Cを捨てて「メモリ管理が楽な言語」に
    - 速度が必要ならgo, Rustを、そうでもなければC#,Javaを
    - 最近はJavaScriptでもかなり高速になった
    - 速度が不要ならPython等の利用も候補（生産性も高い）
  - OSやライブラリの工夫
- プログラム起動後に送り込まれた「データ」が「プログラム」として解釈可能なことに問題
  - ライブラリに工夫して、データがプログラム領域に使われそうになったら異常終了する（Stack破壊の検出など）
  - 速度低下、完全にうまく終了させるのが難しい
- OSに工夫して、データ領域はプログラムを置いても実行できない仕組みを入れる（WindowsのDEP、LinuxのExecShield、OpenBSDのW^Xなど）
  - CPUのメモリ管理機構に「実行禁止」機構が必要（現在はx86/x64系は備える）





# 脆弱性を体験するには

[トップ](#) | [個人学習向けツール概要](#) | [集合学習向けツール概要](#) | [FAQ](#) |



## 概要

脆弱性体験学習ツール「AppGoat」は、脆弱性の概要や対策方法等の脆弱性に関する基礎的な知識を実習形式で体系的に学べるツールです。利用者は、学習テーマ毎に用意された演習問題に対して、埋め込まれた脆弱性の発見、プログラミング上の問題点の把握、対策手法の学習を対話的に実施できます。

ウェブアプリケーションやサーバ・デスクトップアプリケーションの脆弱性対策に必要なスキルを習得したい開発者やウェブサイトの管理者におすすめです。

なお、本ツールは脆弱性対策の促進を目的とするものです。本ツールで学んだことを別の目的で悪用することはしないでください。

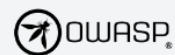
ダウンロードやダウンロードの詳細については下記を参照してください。

内容をご確認の上、同意いただける場合には「同意する」にチェックを入れてAppGoatを利用してください。

[「利用許諾条件合意書 \(PDF形式\)」](#)

☐ 利用許諾条件合意書に同意する

※AppGoatをご利用の際は、利用許諾条件合意書に同意いただく必要があります。



PROJECTS CHAPTERS EVENTS ABOUT

Search OWASP.org

## OWASP Juice Shop

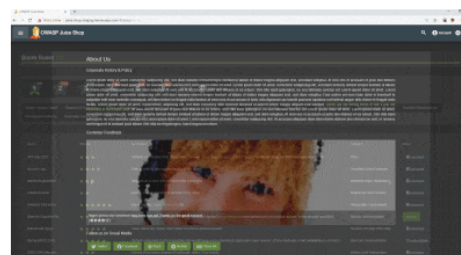
[Main](#) [Overview](#) [News](#) [Challenges](#) [Learning](#) [CTF](#) [Ecosystem](#) [Supporters](#)



owasp **flagship project** release **v14.5.1** GitHub ★ 8.1k [Follow](#)

openssf best practices **gold** Contributor Covenant **v2.0 adopted**

OWASP Juice Shop is probably the most modern and sophisticated insecure web application! It can be used in security trainings, awareness demos, CTFs and as a guinea pig for security tools! Juice Shop encompasses vulnerabilities from the entire [OWASP Top Ten](#) along with many other security flaws found in real-world applications!



## Description

Juice Shop is written in Node.js, Express and Angular. It was the first application written entirely in JavaScript listed in the [OWASP VWA Directory](#).





## 脆弱性管理の難しさ

---

- パッケージやクラウドサービスは脆弱性発見はベンダ・サービスの責任
- オンプレミスのシステムでは脆弱性管理そのものはユーザーが行う
- 完全に自製したソフトウェアでは脆弱性の発見も管理も自分で行う必要
- 特にWebアプリケーションに多い
- 連絡窓口が必要
- 困るのは「外注」したシステム…
- 脆弱性への対応は「保守」の範疇だが保守費は十分か？  
そもそも発見は??

# 楽天とPayPayがつまずいたセールスフォース製品の「設定不備」、被害は氷山の一角か

山端 宏実 日経クロステック／日経コンピュータ

2021.01.22

有料会員限定



全3212文字

PR

日本IBM、ビジネスパートナー14社を表彰。3つの共創型パートナーシップ推進  
富士通、東芝、OBC、DIS・・・日本IBMが協業を拡大、日本企業のDXに貢献  
ついに第3世代EPYC搭載サーバーが登場 その「進化」と「利用メリット」とは

15万社以上が使い、「世界No.1 CRM（顧客情報管理）」をうたうセールスフォース・ドットコム。同社が提供するクラウドサービスを使う企業で、本来アクセスできないはずの情報を第三者が閲覧できてしまう問題が明らかになった。この問題に気づいていない企業もあるとみられ、情報漏洩のリスクが高まっている。

「自宅の壁にいきなりマジックミラーを取り付けられたようなものだ」。セールスフォースのクラウドを使うネット企業のセキュリティ担当者こう憤る。

情報セキュリティに詳しい国際大学グローバル・コミュニケーション・センター（GLOCOM）の楠正憲客員研究員は「今はアンテナが高い企業で被害が判明している段階。金融以外の業種では被害をまだ十分に洗い出せていないのではないかと指摘する。セールスフォースのクラウドに何が起きたのか。

## PaaSの新機能追加が 思いがけない アクセスへの 隘路を作ってしまった例

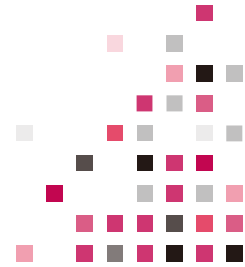
## 責任の 所在は？

楽天とPayPayがつまずいたセールスフォース製品の「設定不備」、被害は氷山の一角か | 日経クロステック (xTECH) (nikkei.com)



## とにかく発見されたら脆弱性対策

---

- セキュリティパッチ（セキュリティ修正プログラム）を導入する
  - しかし実際には大変
    - 多数ある計算機に漏れなくパッチできるか？
    - その作業に伴って、  
利用中のソフトウェアが動作しなくなったりしないか？
      - 事前の検証が必要な場合が多い  
無停止が求められるシステムではレプリカでの検証作業が必要
    - 停止することが難しいシステムの場合  
（メールや認証サーバなど）作業時間をどうする？
      - システムの多重化で回避など
  - メンテナンス計画の立案が重要
- 



# システムの運用計画にセキュリティの要素を入れる

---

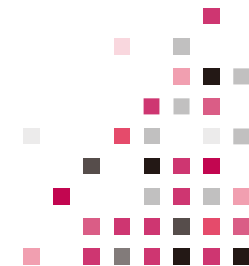
- 平時に必要な事
  - 平時正常に動作しているかを「能動的に」
    - 障害報告を受けてから動いていると発見が遅れる
  - 性能評価・ログの監査を行う
- 脆弱性情報が入ったときは…
  - パッチ適用計画を速やかに立てる
  - テスト環境の整備と運用
- 事故発生時は…
  - インシデントレスポンス



## 認証ログ監査に基づく不正アクセス発見

---

- 定期的に認証ログ（いつログイン・ログアウトしたか、ログインにどの機器（IPアドレス）が使われたか）を検査して異常が無いか調べる
  - 勤務時間外や通常利用しない時間での利用
  - 通常使用しないIPアドレス範囲からの利用
  - 同一IPアドレスからの多数のログイン・ログアウト
  - 多数のログイン失敗記録
- 侵入検知システム(IDS) で発見出来ることも
- 対応・当該アカウントに対する聞き取り調査や特定IPアドレスのファイアウォールでの遮断





## マルウェア対策ソフトウェアの代表：アンチウィルスソフトウェア

---

- コンピュータのファイル読み書きや通信を監視、ウィルス固有のパターンを発見したら中止させる
- ウィルスは新種がどんどん出てくるのでパターンを定義するファイルが重要
- よってサポートを受け続ける必要がある
- 最近是新種の登場にアンチウィルス会社は十分追従できていない  
= 検出は「運」に左右される
  - アンチウィルスが入っていても感染の危険はなくなる
  - 特にbotnetの検出は困難になりつつある
- クライアントで行うものとサーバで行うものがある
- 通常は併用する最近はフィッシングサイト検出機能、「ふるまい検知機能」なども





## 「パターン認識」の限界

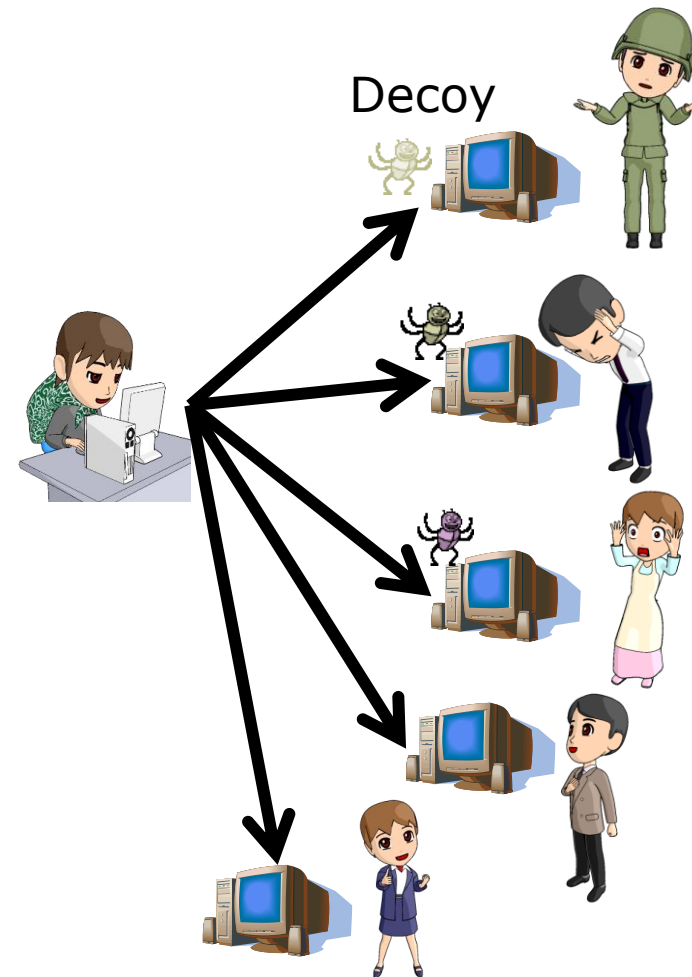
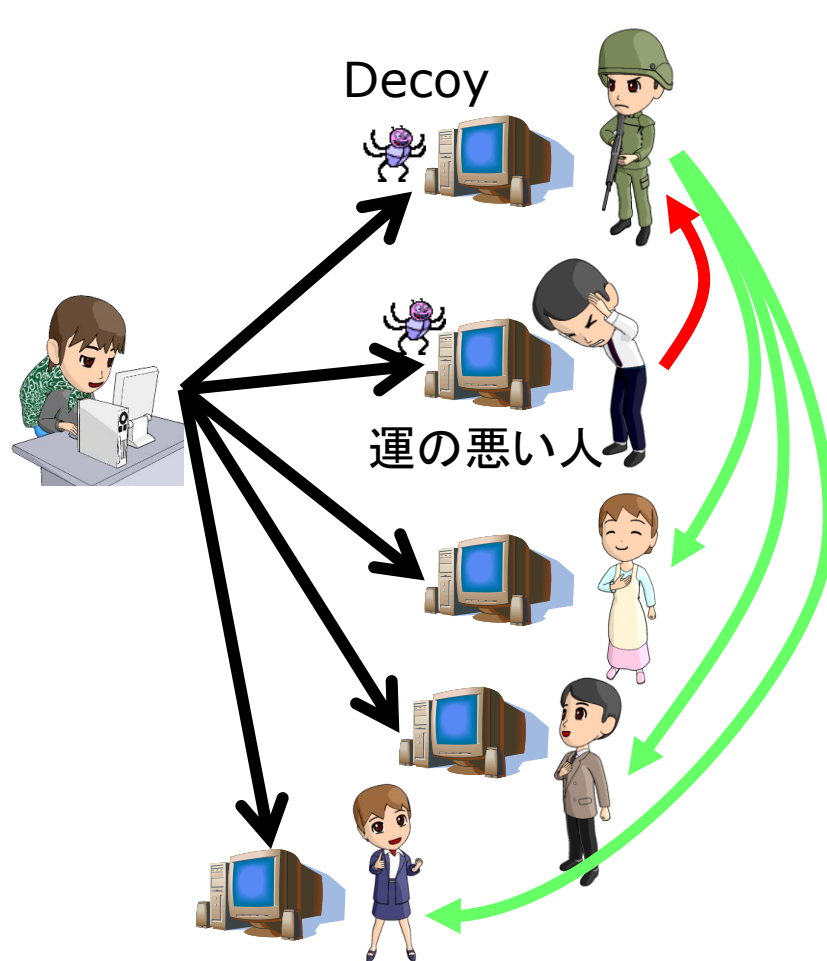
---

- ウィルス対策ソフトやスパイウェア対策ソフト、IDS（侵入検知システム）などの多くは基本的に「パターンファイル」を用いて侵入を検出
- パターンファイルは、いずれかの組織や「おとりシステム」への侵入結果から、そこで利用されたマルウェアそのものまたは通信の特徴を抽出して作成
- 脆弱性が明らかな場合には、脆弱性そのものを狙うコードの特徴を利用できる場合もある
- つまり「どこにも侵入したことがない」マルウェアは原理的に捕らえることが困難
- 特に既知の脆弱性を利用していない場合は絶望的

高度にカスタマイズされたマルウェアは検出が困難  
＝標的型攻撃の恐怖



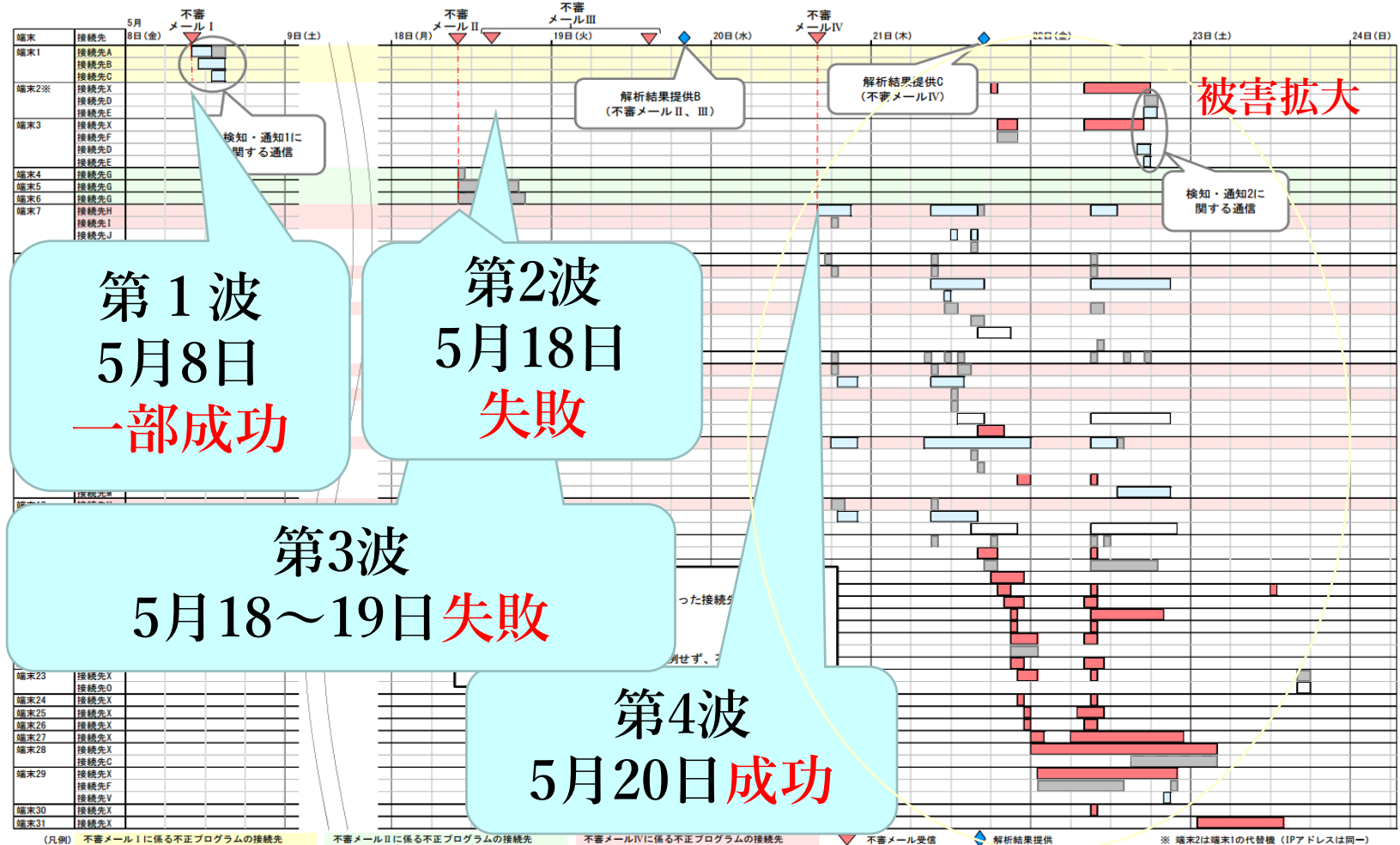
# マス攻撃と標的型攻撃





## 年金機構に対して行われた標的型波状攻撃

- 目的があれば達成までやる






## 「ふるまい検知」機能

---

- 最近の多くのマルウェア対策ソフトウェアが  
パターンファイルだけではなく  
マルウェア固有の「ふるまい」を検知する機能を搭載
- 通常使わないTCPポートへの接続  
スクリーンキャプチャやキーボードログなど  
「普通のアプリケーションが行わない」  
「危険な」動作の頻繁な実行などを検知
- 誤検知が課題





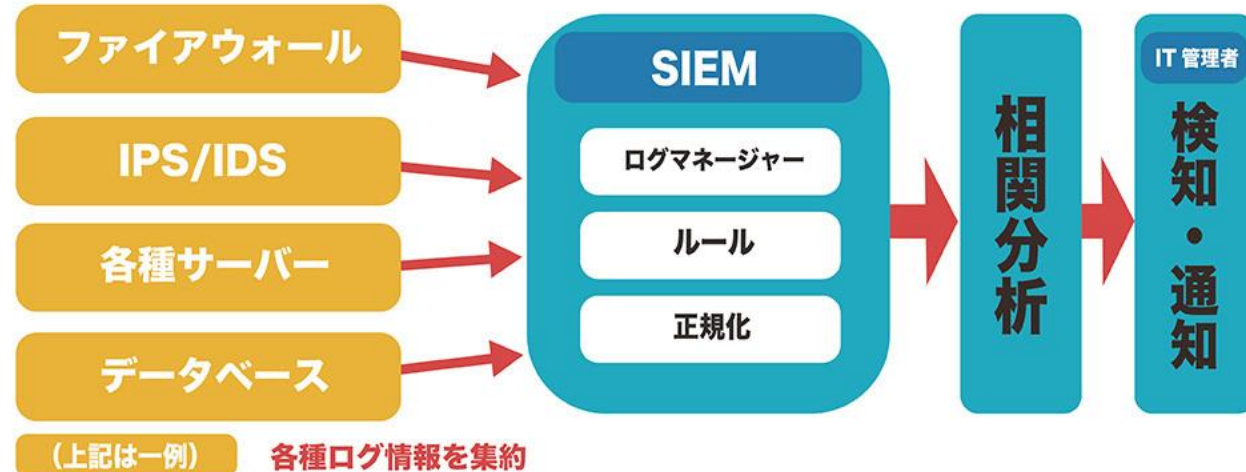
## 最近の流れ：EDR導入

---

- Endpoint Detection and Response
- 「ふるまい検知」をさらに進めて  
ふるまいをサーバに集約し分析
- 複数クライアントにおける相関分析
- 複数**組織**における相関分析
- Security Operation Center (SOC)による分析
- 分析はAIまたは「目視」で行う
- SOCによってはさらにクライアント内にアクセスして  
詳細な状況分析を行う場合も…！

# SIEM: Security Information and Event Management

- EDRはもちろん  
各種ネットワークログを集積し分析  
セキュリティ上のリスクを洗い出す

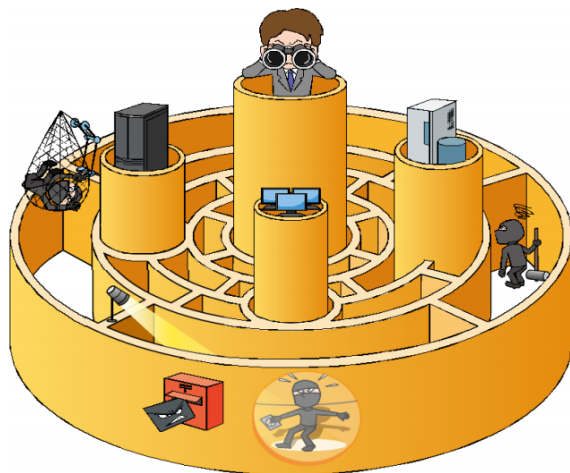


SIEMとは？ 主な機能や製品の選定ポイントについて | セキュリティコラム | 株式会社網屋 ([amiya.co.jp](http://amiya.co.jp))

# IPA「高度標的型攻撃」対策ガイド

## 「高度標的型攻撃」対策 に向けたシステム設計ガイド

～入口突破されても攻略されない内部対策を施す～



IPA 独立行政法人情報処理推進機構  
セキュリティセンター

2014年9月

このところ  
ほぼ毎年改訂されてきた  
システム設計ガイドの集大成

ポイントは  
「侵入されないこと」  
ではなく  
「侵入されたことを発見しやすい」  
「侵入されても被害が大きくなりにくい」  
こと


LAN内の細分化・監視強化



## インシデント・レスポンスとは


---

- インシデントを発見した際の対応
  - 原因の発見・究明
  - 情報システムの障害からの回復
  - 再発防止策の立案
  - (故意性があれば) 民事・刑事的な対応
    - (内部不正の場合) 内規での対応
    - 損害賠償や刑事事件化への対応
- 内部で処理 or 外部に依頼



## インシデントレスポンスは どうあるべきか

---

- 組織内体制の整備（CSIRT）が必要
  - シーサート（CSIRT: Computer Security Incident Response Team）とは、コンピュータセキュリティにかかるインシデントに対処するための組織の総称です。インシデント関連情報、脆弱性情報、攻撃予兆情報を常に収集、分析し、対応方針や手順の策定などの活動を行います。  
（日本シーサート協議会HPより）
  - 企業などでは**技術だけでは不十分**
  - 営利企業では対処は経営判断に直結
  - 法的対応には法務部門等とのリンクが不可欠
- 



# 日本コンピュータセキュリティ インシデント対応チーム協議会

Nippon CSIRT Association



## 日本シーサート協議会とは

シーサート (CSIRT: Computer Security Incident Response Team) にはさまざまな種類があり、目的、立場（組織内での位置づけ）、活動範囲、法的規制などの違いからそれぞれ独自で活動を行ってきました。

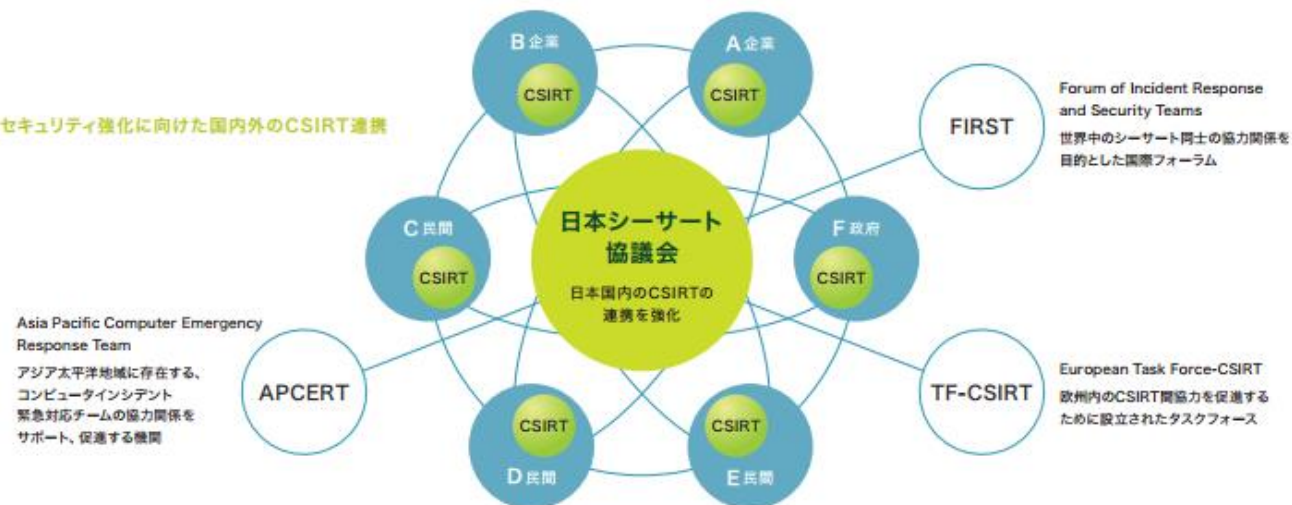
しかし、コンピュータセキュリティインシデントの攻撃がより巧妙かつ複雑になってきた現在、迅速な対応が、単独なシーサートでは困難な状況になっています。そこで、同じような状況や課題を持つシーサート同士が互いに協調し、共通の問題を解決する場として、日本コンピュータセキュリティインシデント対応チーム協議会（略称：日本シーサート協議会）が設立されました。CSIRT間の密接な連携、そして強い信頼関係に基づいた迅速かつ適切な対応、情報共有を実施する体制作りを目指します。

## シーサート (CSIRT) とは

シーサート (CSIRT: Computer Security Incident Response Team) とは、コンピュータセキュリティにかかるインシデントに対処するための組織の総称です。主な活動としては、インシデント関連情報・脆弱性情報・攻撃予兆情報の収集分析、対応方針や手順の策定などが挙げられます。

また、インシデント対応においてセンシティブな情報を扱う必要があることから、常日頃から他組織との信頼を醸成しておくことが求められます。そのため、国内はもとより海外の他組織と連携したり、情報交換を行うための窓口として機能することも、日本シーサート協議会の重要な役割です。

## セキュリティ強化に向けた国内外のCSIRT連携

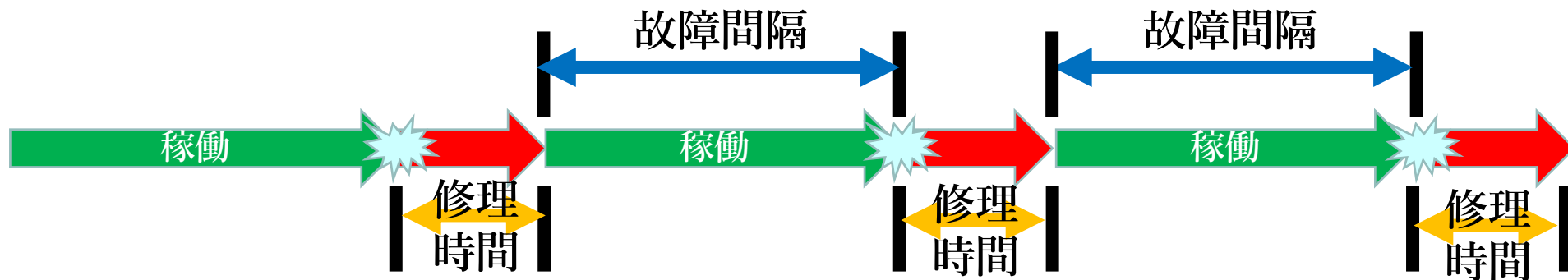




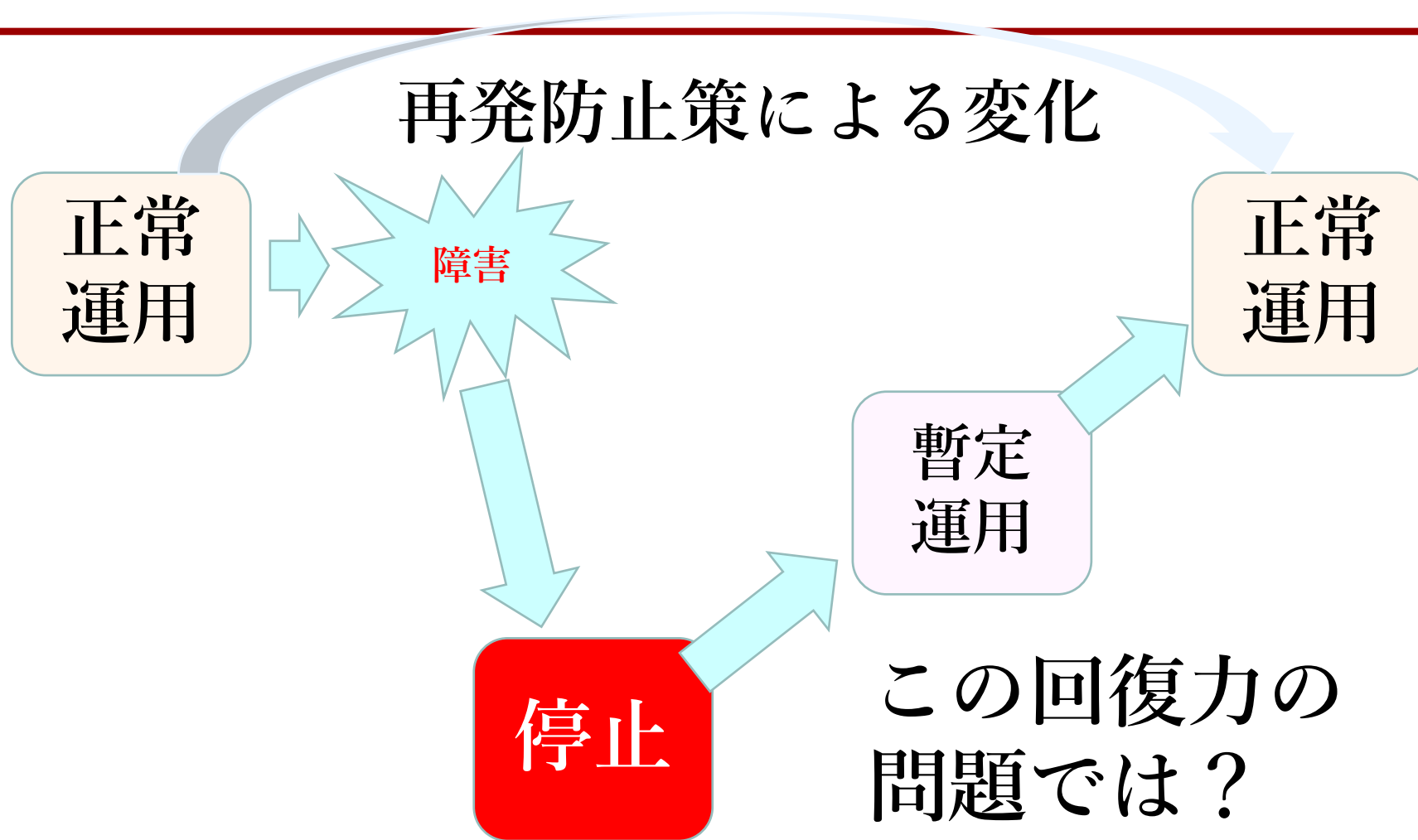
# システムの安定性をどう測るか

- 平均故障間隔 (MTBF)
  - 故障回復から故障までの間隔の平均時間
  - 修理できることが前提 交換部品は平均故障時間 (MTTF) で表現
- 平均修理時間 (MTTR)
  - 修理にかかる時間の平均時間
- 稼働率 =  $MTBF / (MTBF + MTTR)$

## 線形モデル



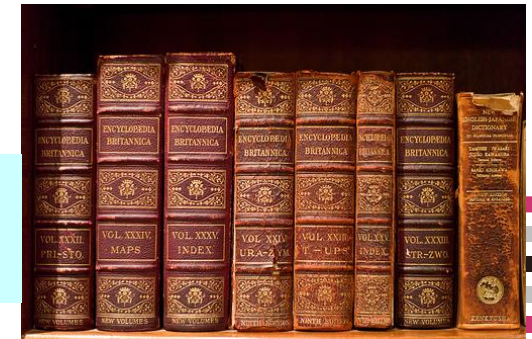
# 大規模情報システムの運用



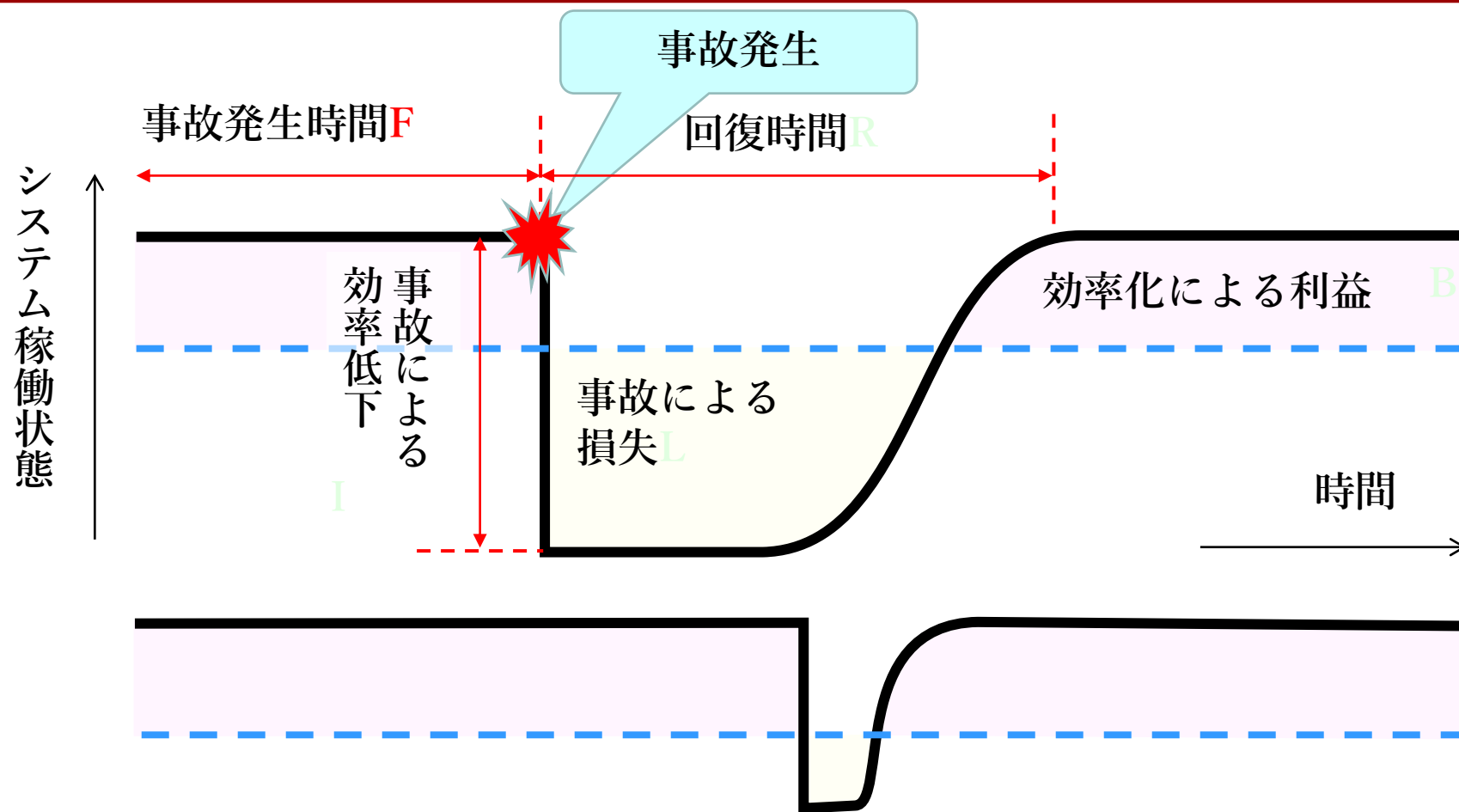
# レジリエンス (Resilience) from Merriam-Webster

1. the capability of a strained body to recover its size and shape after deformation caused especially by compressive stress  
応力がかかっても姿形が元に戻る能力
2. an ability to recover from or adjust easily to misfortune or change  
災害や変化から回復し適応する能力

モノ・人・組織・社会・システム…などが大きな外圧/環境変化に直面した際に本来の目的を維持し速やかに回復する能力



# 情報システムにおけるセキュリティとレジリエンス



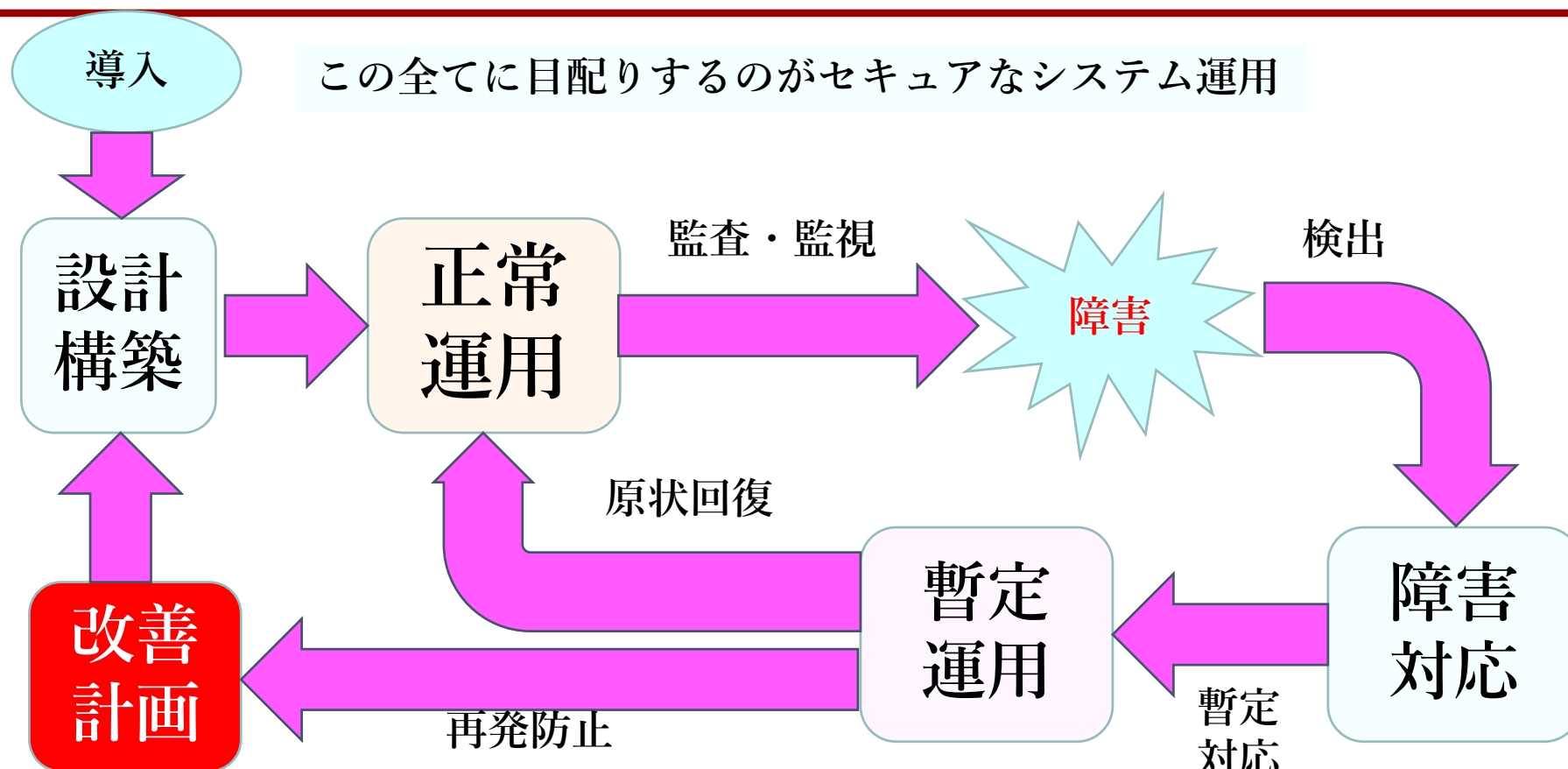
セキュリティとは  $F/(F+R)$  を抑えること レジリエンスは  $L$  を抑えること

A 15x15 grid of colored squares (red, black, grey) representing a sparse matrix. A red horizontal bar is at the bottom right.

- 



# 情報システムの レジリエンス・サイクル



# まずはガバナンスから

