第4回（平成26年度第1回）CRCフォーラム（平成26年6月16日（月）開催）
「TDUにおけるサイバーセキュリティ教育と研究発表会」

# CyS研究所での取り組みについて

佐々木　良一　教授
未来科学部情報メディア学科

TDU
東京電機大学

TDUにおけるサイバーセキュリティ
教育と研究発表会　　2014年6月16日

# ネットワークフォレンジック技術の研究開発
## ーLIFTシステムの開発ー

東京電機大学教授
サイバーセキュリティ研究所所長
佐々木良一

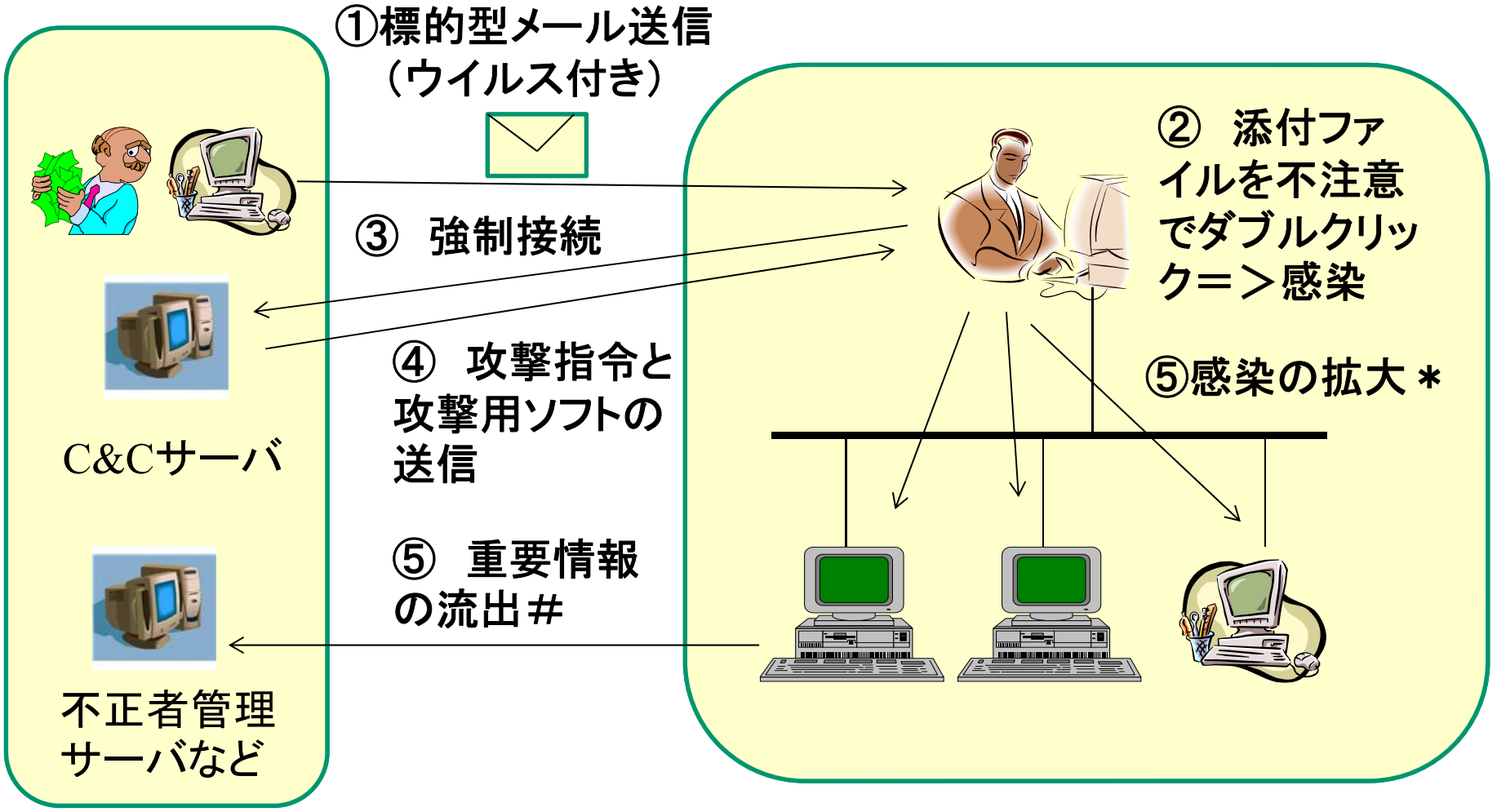
LIFT：Live Intelligent Network Forensic Technologies

# 目次



LIFT：Live Intelligent Network Forensic Technologies

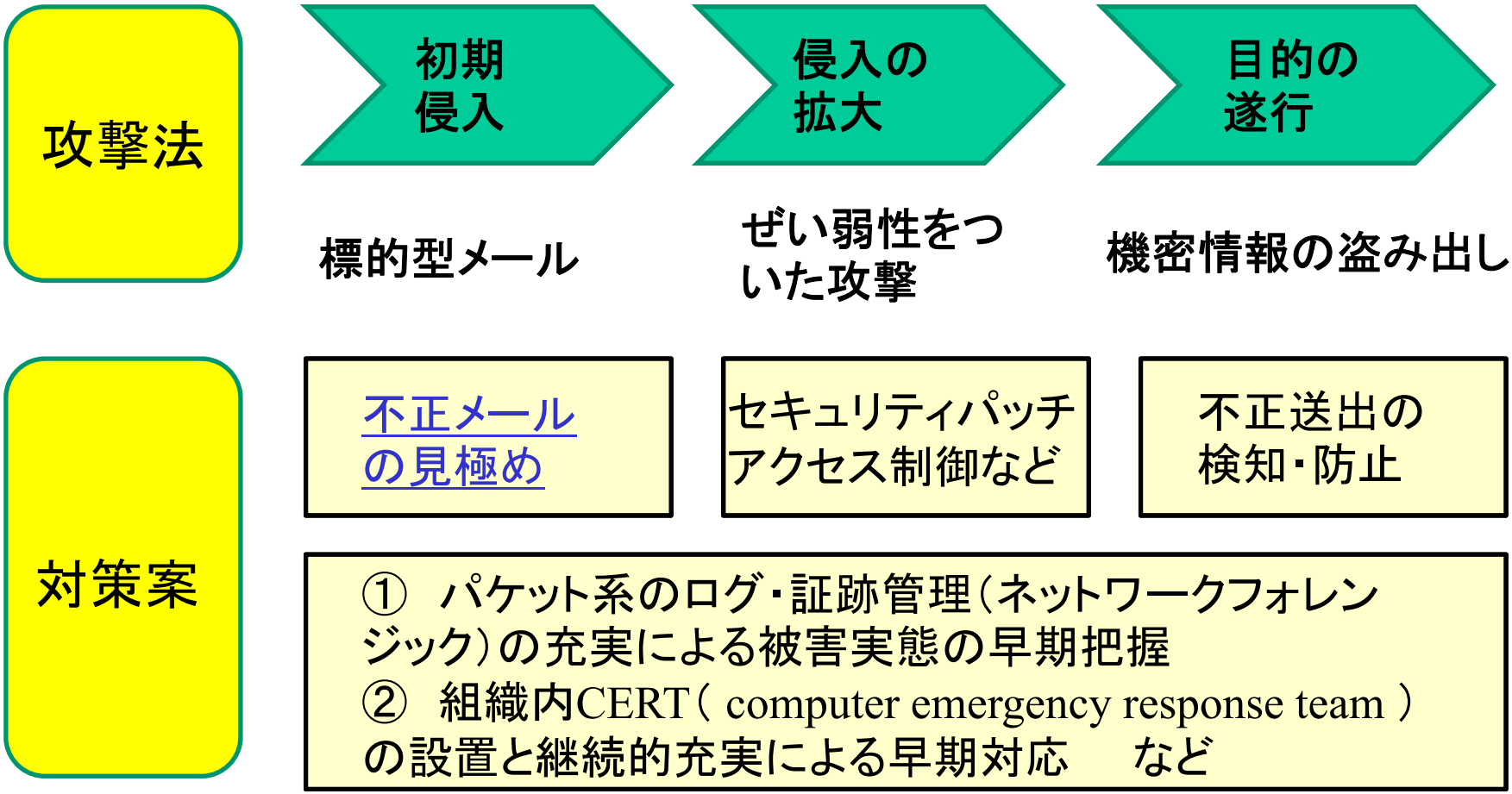
# 標的型メール攻撃の概要



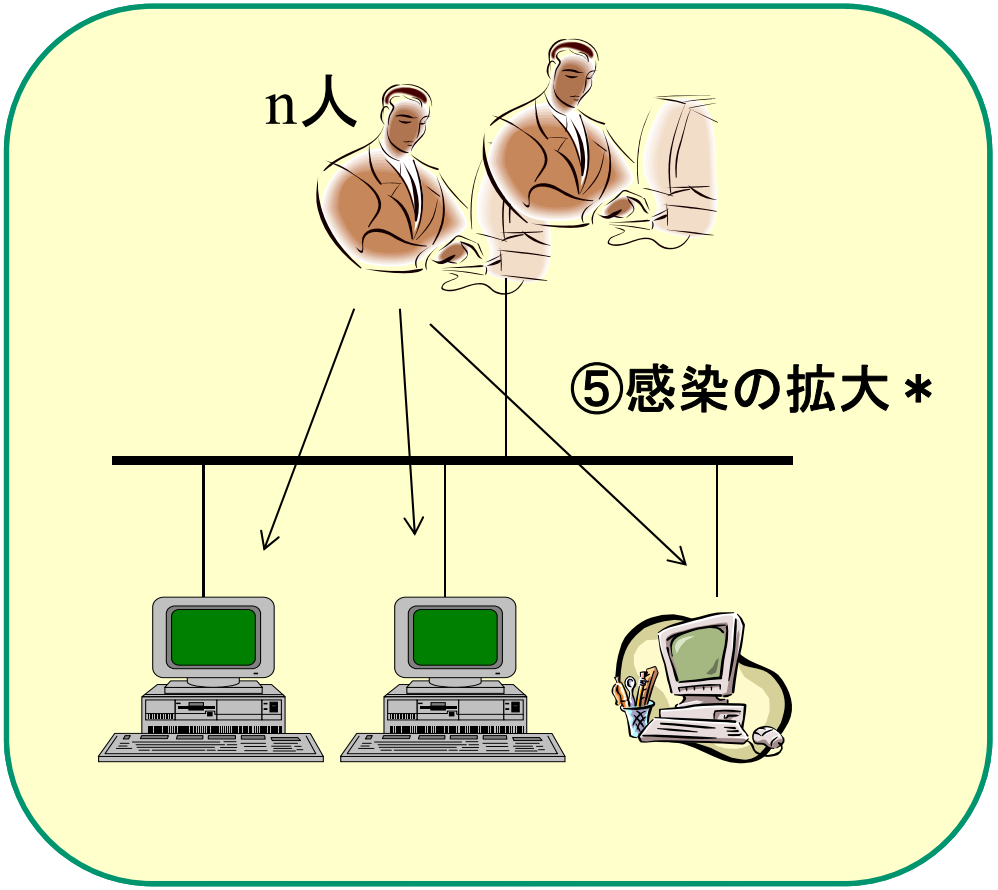

＊ソフトのぜい弱性やパスワード管理の不備を利用

＃HTTPのポスト機能等を利用

# 標的型攻撃と対策案



# だれか1人が標的型メールを開ける確率①




n人の組織で1人でも標的型メールを開けると被害はどんどん拡大する。
今、i人目の人が開ける確率をPiとすると組織の中で誰かひとりが開ける確率は次式で求められる。

$$PT = 1 - \prod_{i=1}^{n} (1-Pi)$$

# 政府における標的型メール対策

標的型不審メール攻撃訓練結果の概要（中間報告）

１．訓練期間：平成２３年１０月～１２月

２．訓練対象：内閣官房等１２の政府機関約６万名

３．訓練内容：①訓練対象者に対して事前教育の実施。
②訓練対象者に対して標的型不審メールを模擬したメールを２回送付。
③模擬メール中の添付ファイルを開封もしくは、ＵＲＬをクリックするなど不適切な扱いをした場合は、教育コンテンツに誘導。
④参加府省庁に個別の訓練結果を通知し、各府省庁内において適切な事後教育指導を実施。

開封率

第一回　添付メール　１０．１％　（１．１％－２３．８％）

第二回　リンクメール　３．１％　（０．４％－６．８％）

http://www.nisc.go.jp/active/general/pdf/hyoutekigata_120119.pdf

# だれか1人が標的型メールを開ける確率②

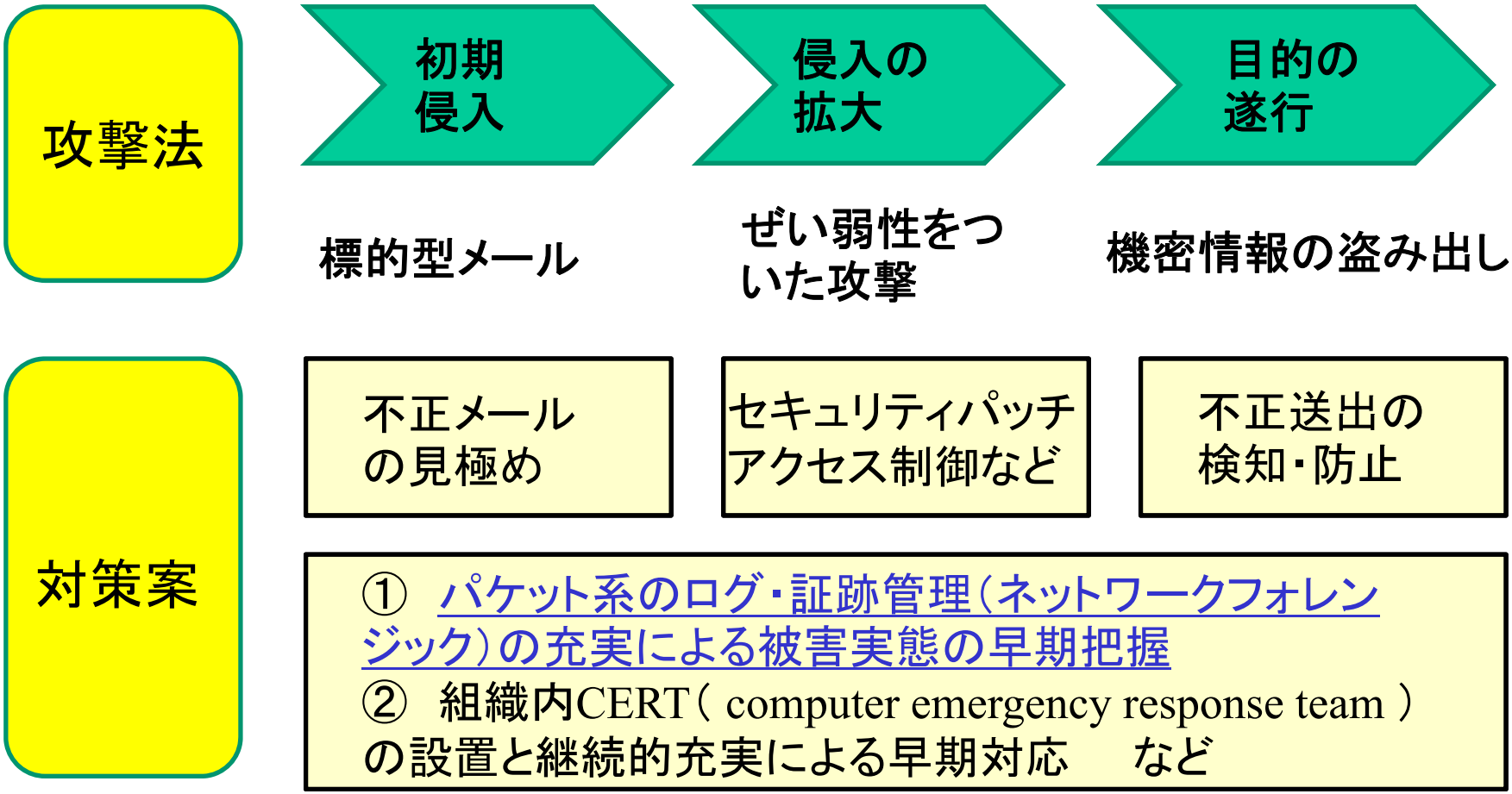
| n \ Pi | 1.0 | 0.1 | 0.03 | 0.01 |
|---|---|---|---|---|
| 10 | 1.0 | 0.65 | 0.26 | 0.096 |
| 50 | 1.0 | 1.0 | 0.87 | 0.39 |
| 100 | 1.0 | 1.0 | 0.95 | 0.63 |
| 500 | 1.0 | 1.0 | 1.0 | 0.99 |

n ： 組織に属する人の数　Pi ： i 番目の人が開ける確率

Piを少々小さくしてもnが大きいと効果がないことがわかる。

＝＞セグメントの分離なども考えるべき

# 標的型攻撃と対策案


攻撃法
初期
侵入
侵入の
拡大
目的の
遂行
標的型メール
ぜい弱性をつ
いた攻撃
機密情報の盗み出し
対策案
不正メール
の見極め
セキュリティパッチ
アクセス制御など
不正送出の
検知・防止
① パケット系のログ・証跡管理（ネットワークフォレン
ジック）の充実による被害実態の早期把握
② 組織内CERT（ computer emergency response team ）
の設置と継続的充実による早期対応 など

# 目次



LIFT：Live Intelligent Network Forensic Technologies

# ネットワークフォレンジック対策の必要性

- サーバ系のログは通常とられるようになってきたが、ネットワーク系のログはほとんど取られていない。

- ネットワークフォレンジック対策が必要

- ネットワーク系のログ管理のガイドラインの作成
（NISCとデジタル・フォレンジック研究会の協力）

# 標的型攻撃対策のための適切なログの管理（その１）

＜機器によらない全般的な対策＞

１．各ログ取得機器のシステム時刻を、タイムサーバを用いて同期する。
２．ログは1年間以上保存する。
３．複数のログ取得機器のログを、ログサーバを用いて一括取得する。
４．攻撃等の事象発生が確認された場合の対処手順を整備する。

内閣官房情報セキュリティセンター：
http://www.nisc.go.jp/active/general/pdf/logkanri_kanki_120705.pdf

# 標的型攻撃対策のための適切なログの管理（その2）

＜機器別の対策＞

1. ファイアウォール：「外⇒内で許可した通信」と「内⇒外で許可・不許可両方の通信」のログを取得する。

2. Web プロキシサーバ：接続を要求した端末を識別できるログを取得する。

3. 他のシステムや機器の権限を管理するサーバ（LDAP, Radius 等）：管理者権限による操作ログを取得する。

内閣官房情報セキュリティセンター：
http://www.nisc.go.jp/active/general/pdf/logkanri_kanki_120705.pdf

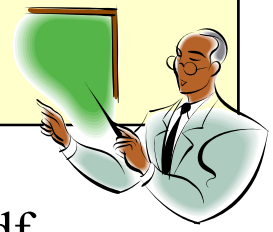
# 標的型攻撃対策のための適切なログの管理（その3）

＜機器別の対策＞

4. メールサーバ:「メールの送受信アドレス」及び「メッセージID」のログを取得する。

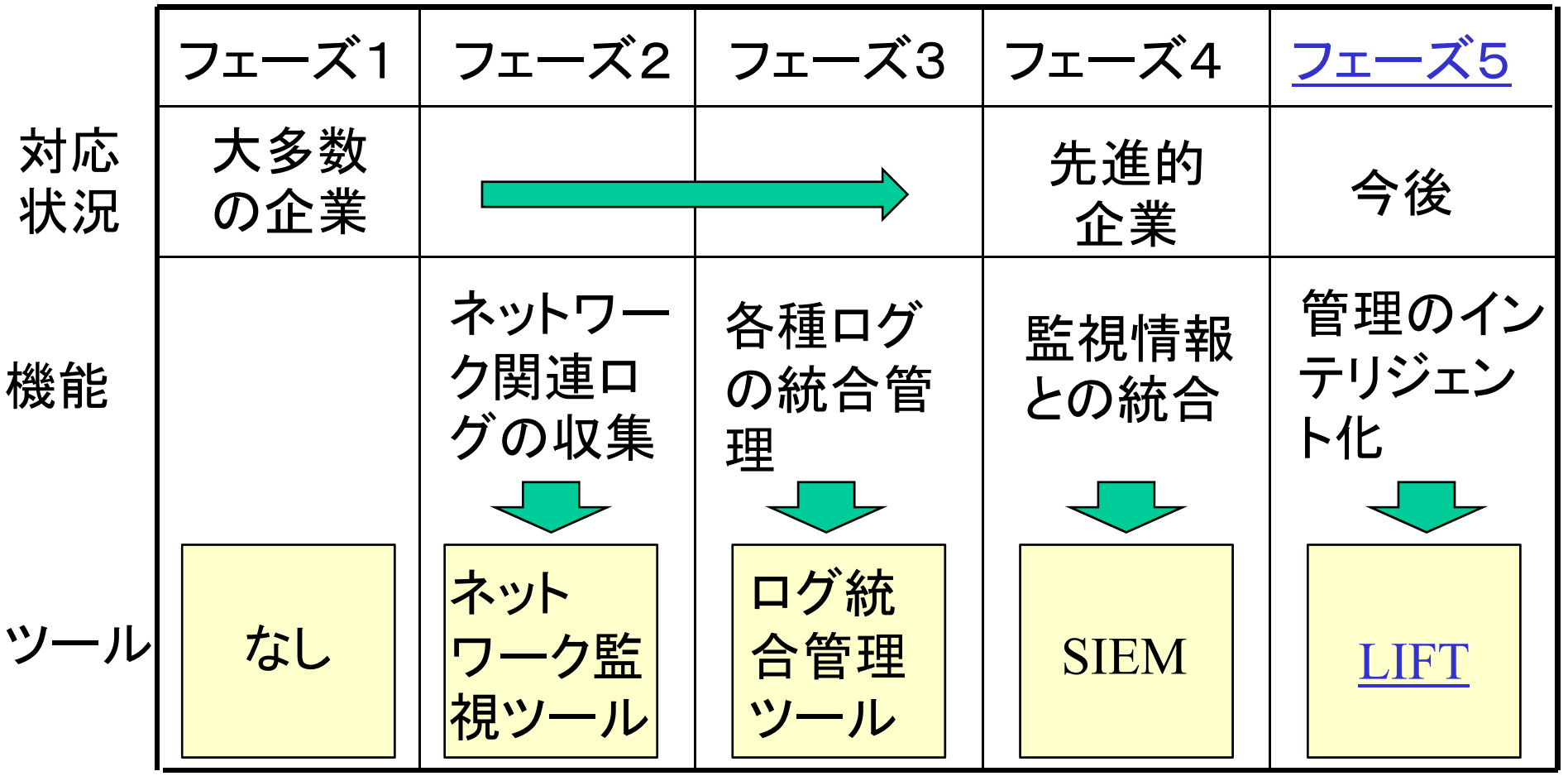
5. クライアントPC:マルウェア対策ソフトウェアの検知・スキャンログ・パターンファイルのアップデートログを取得する。

6. DB サーバ・ファイルサーバ:特別なログ設定は不要だが、確実にログを取得する。



内閣官房情報セキュリティセンター:
http://www.nisc.go.jp/active/general/pdf/logkanri_kanki_120705.pdf

# ネットワークフォレンジックの対応フェーズ




SIEM: Security Information and Event Management
LIFT: Live and Intelligent Network Forensic Technologies

# SIEMの問題点と対策案

1．対策の総合的判断が過剰に運用者の能力に依存
　（1）判断の自動化
　（2）対応に関する適切なガイド　} ⇒ AI技術の活用

2．判断に利用する情報が不十分
　（1）パケットを流した元のアプリケーションの探索方法の確立
　（2）不当に消されたデータの持つ情報の有効利用
　（3）ゾーンニングなどの能動的行動によって得られる情報の有効利用　他

3．対象に合致したシステムの構築運用支援が不十分
　（1）計画支援システムとのリンク
　（2）実証実験システムとのリンク

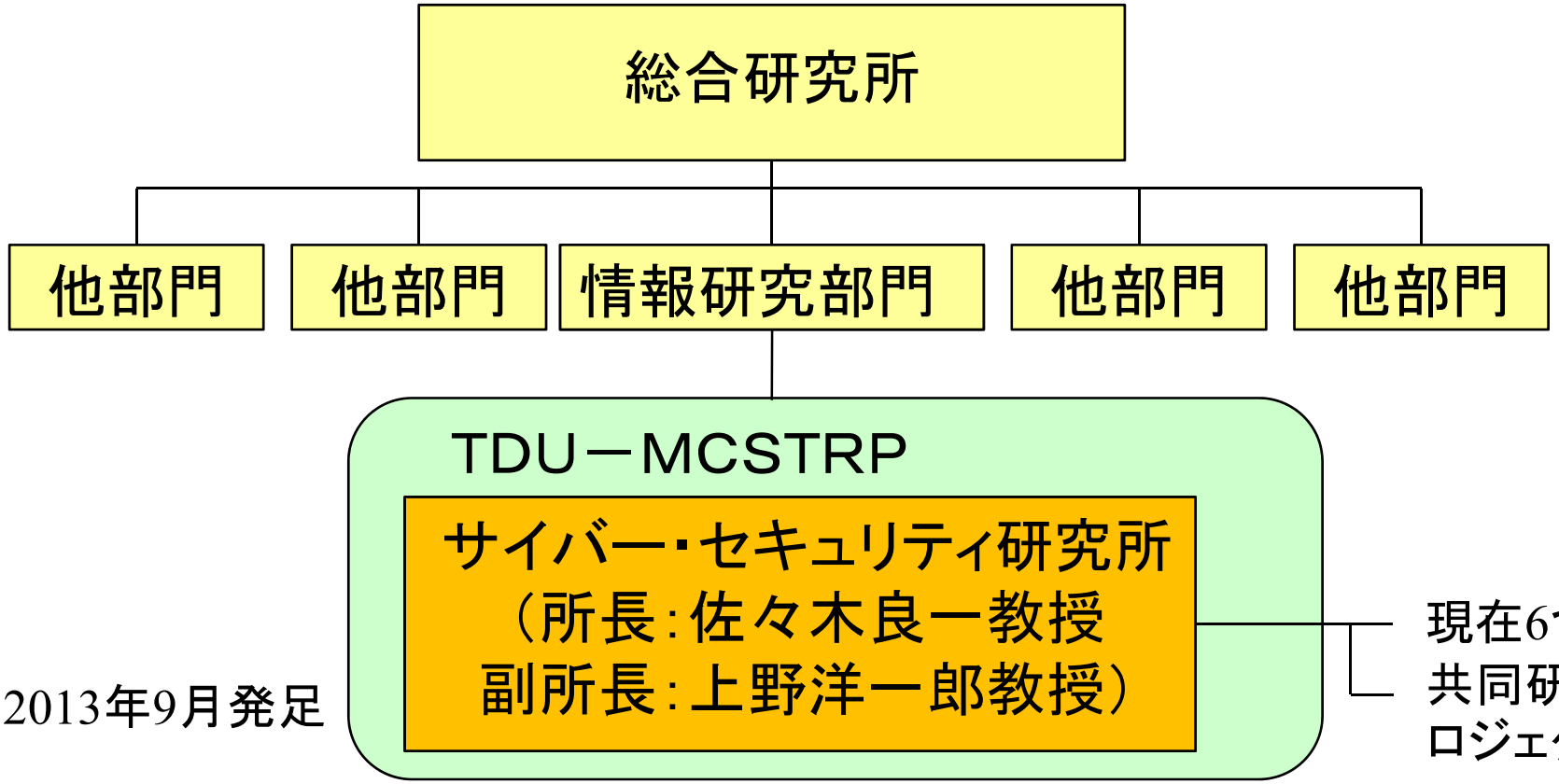
# 目次

1. 標的型メール攻撃と各種対策
2. ネットワーク・フォレンジックの現状
3. LIFTシステムの開発構想
4. LIFTシステムの開発状況
5. 今後の計画

LIFT：Live Intelligent Network Forensic Technologies

# サイバー・セキュリティ研究所

東京電機大学




2013年9月発足

TDU－MCSTRP：複合領域サイバー・セキュリティ技術研究開発プロジェクト

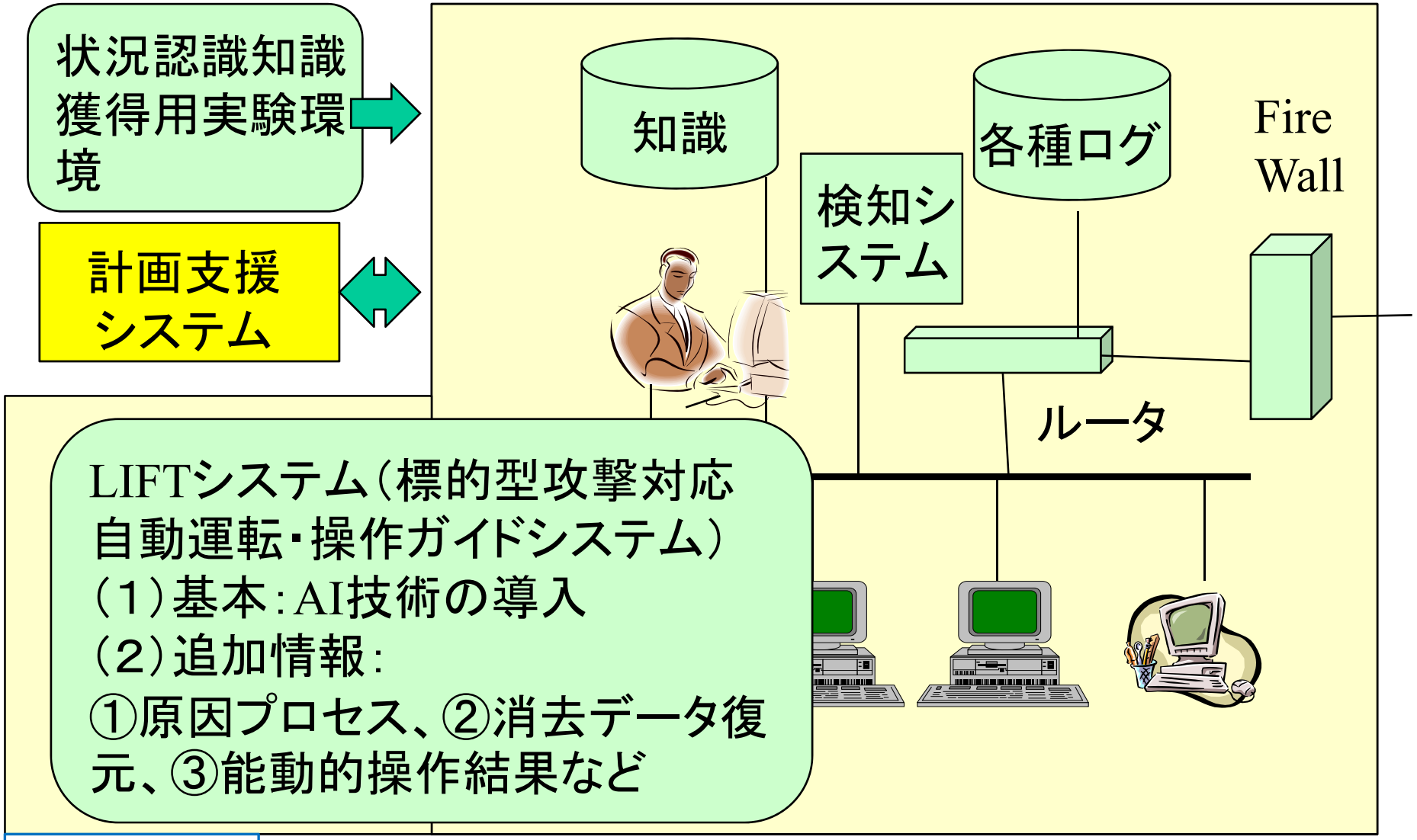
# 共同研究プロジェクト計画

1．メンバー
（1）佐々木良一（東京電機大学：リーダ）
（2）上原哲太郎（立命館大学教授、東京電機大学研究所客員教授）
（3）高倉弘喜（名古屋大学教授、東京電機大学研究所客員教授）
（4）松本隆（ネットエージェント、東京電機大学総合研究所研究員）
（5）佳山こうせつ（東京電機大学総合研究所研究員）
（6）杉本暁彦（日立、東京電機大学総合研究所研究員）
（7）名和利男（サイバーディフェンス研究所、東京電機大学総合研究所研究員）
（8）八槇博史准教授、柿崎淑郎助教　学生：比留間、橋本、三村（電機大）

2．研究項目
（1）LIFT（Live and Intelligent　Network　Forensic Technologies）技術の研究
（2）LIFTシステム　プロトプログラムの開発
（3）実システムへの適用・評価

3．スケジュール（案）
（1）スタート：2013年9月
（2）本格研究：2014年3月―2017年3月

# LIFTシステムの概要

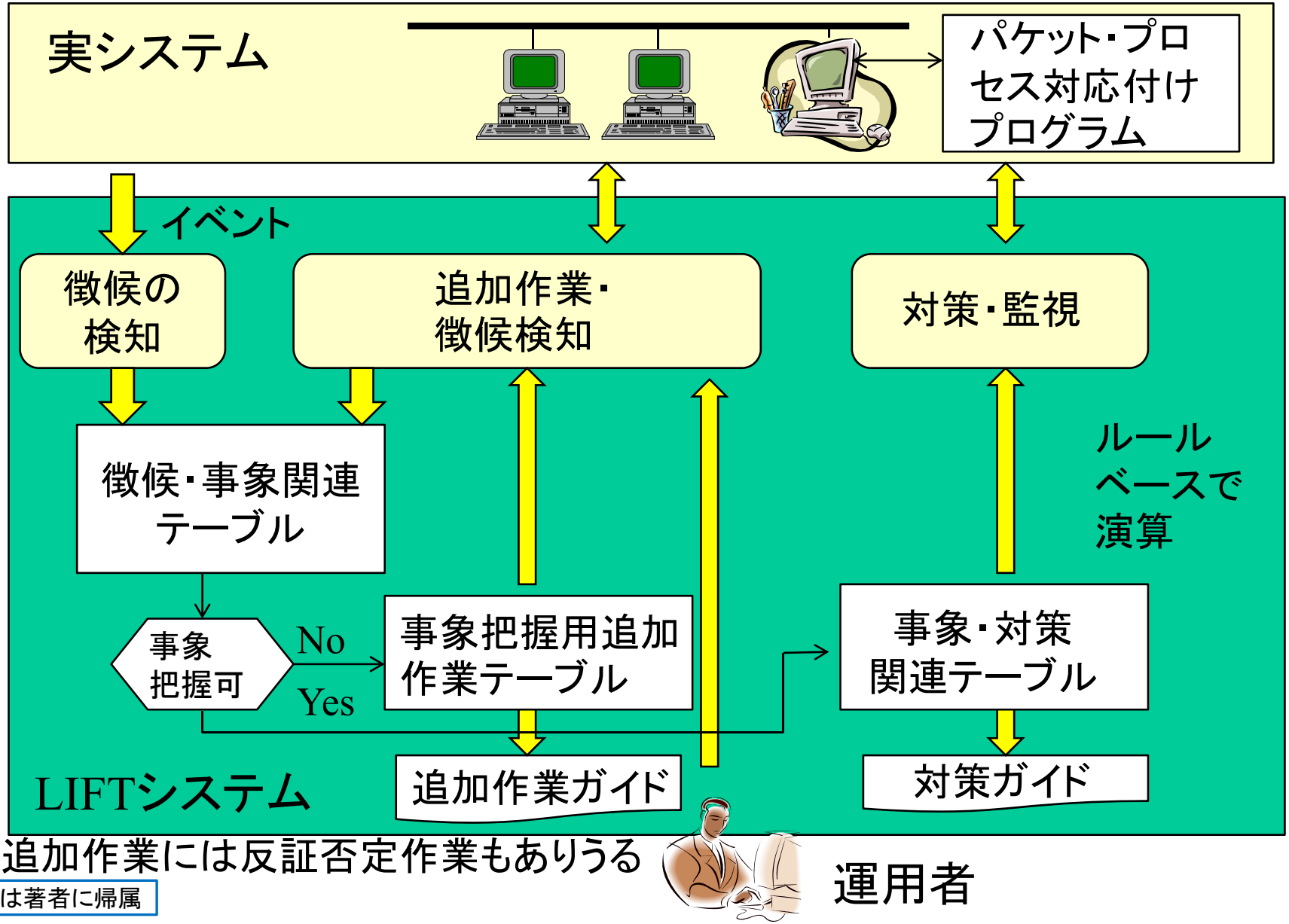


# 目次

1. 標的型メール攻撃と各種対策
2. ネットワーク・フォレンジックの現状
3. LIFTシステムの開発構想
4. LIFTシステムの開発状況
5. 今後の計画

LIFT：Live Intelligent Network Forensic Technologies

# LIFTシステムの運用イメージ

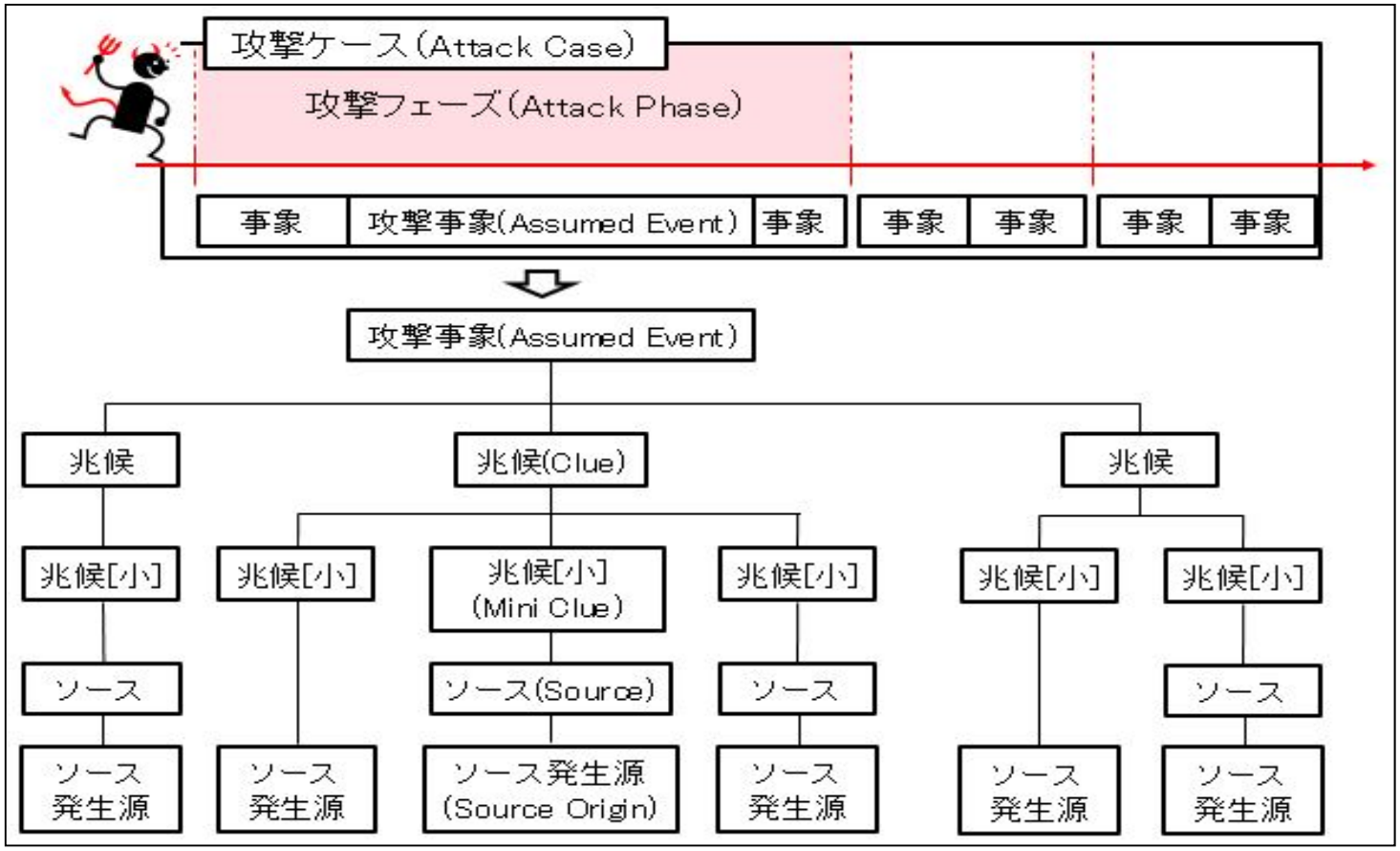


# 攻撃のフェーズ分け

| | |
|---|---|
| Ⅰ 侵入フェーズ | ：マルウェア添付メールを受信 |
| Ⅱ 基盤構築フェーズ | ：不正プログラムを起動<br>：C&Cサーバとの通信<br>：必要な機能のダウンロード<br>：端末の情報入手 |
| Ⅲ 内部侵入・調査フェーズ | ：内部ネットワークを探索<br>：他端末侵入<br>：踏み台PCの増殖<br>：管理端末/サーバへの侵入 |
| Ⅳ 目的遂行フェーズ | ：機密情報の送信<br>：端末の破壊 |

# 攻撃における各種名称の階層構造

＜例＞




標的型メール攻撃(ケース1)

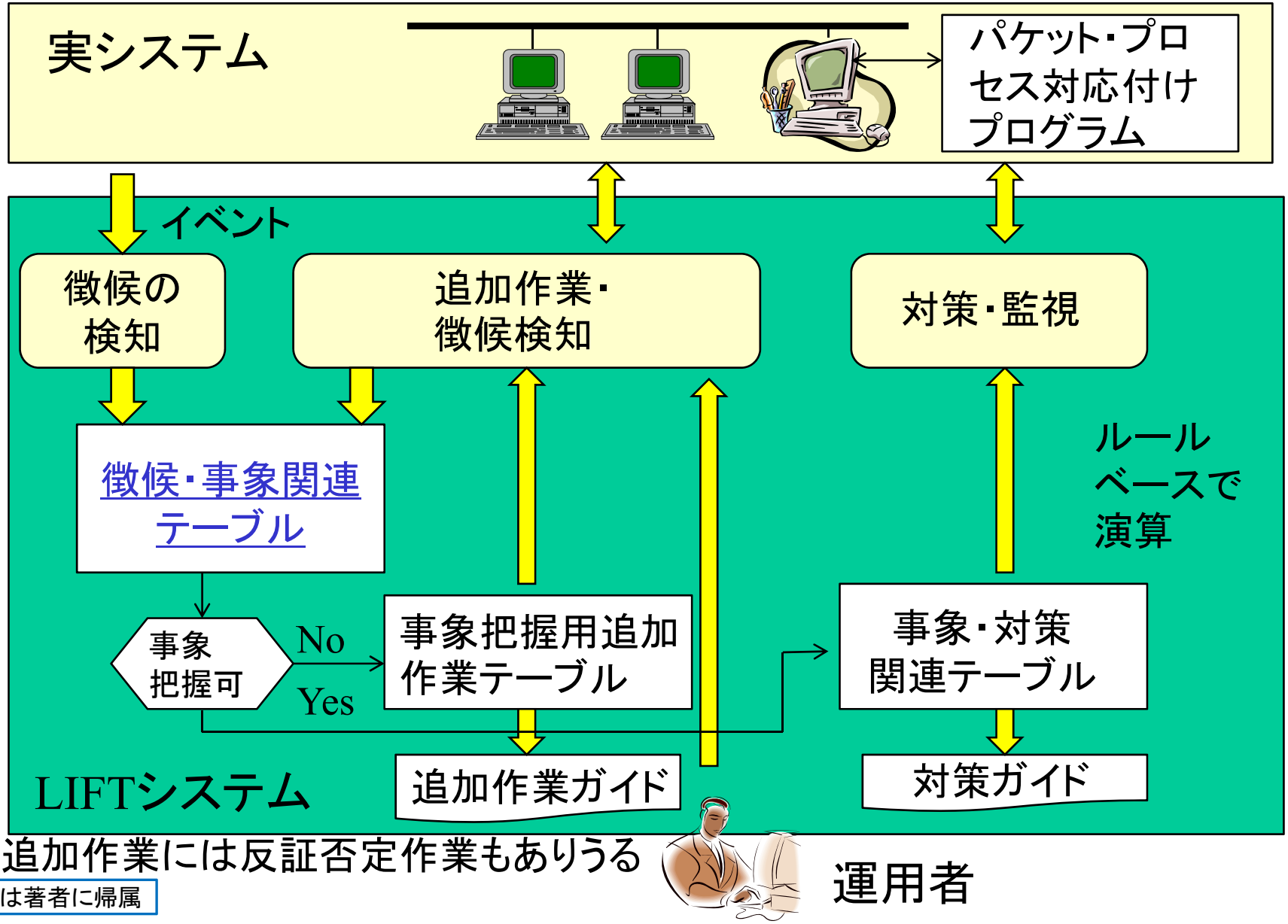
不正C&Cサーバへの接続

プロキシーを経由しない通信

PCの通信ログ

PCなど

# LIFTシステムの運用イメージ



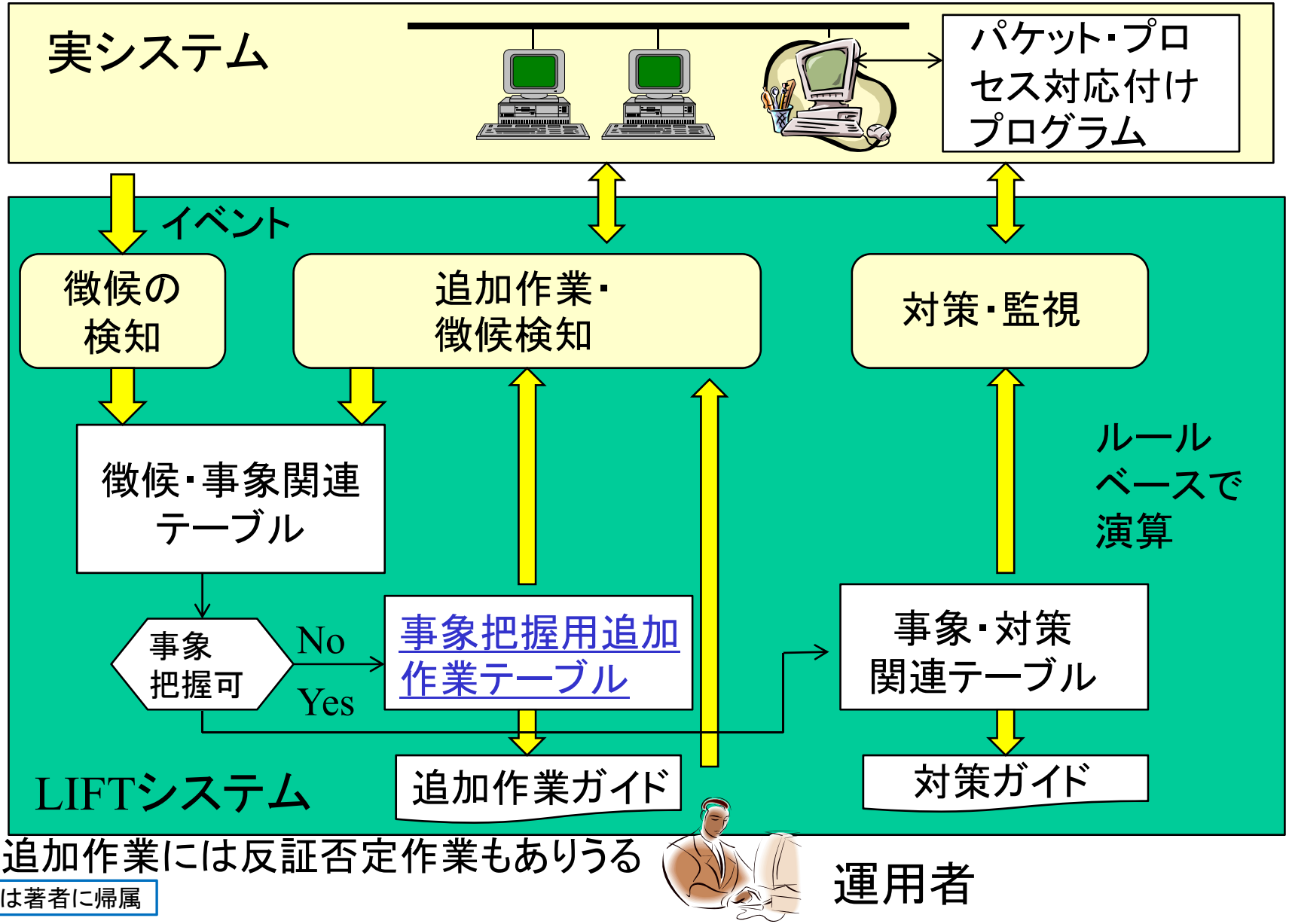
# 兆候・事象対応テーブル

| | | プロキシ | | | | | | |
|---|---|---|---|---|---|---|---|---|
| フェーズ | 事象＼兆候 | 不自然なプロセスの立ち上がり | プロキシを経由しない通信 | プロキシ認証試行に規則性 | 443以外のCONNECTメソッドを利用した通信 | 業務に不要なソフトのインストール | ファイル 共有試行 | サーバへの不自然な時間の認証 | 業務外通信 |
| 基盤構築フェーズ | 端末が不正プログラムを起動 | 2 | | | | | | | |
| | C&Cサーバへ接続 | | 5 | 2 | 2 | | | | 5 |
| | ユーザ端末のuser権限奪取 | | | | | 2 | 2 | 2 | 2 |
| | 感染端末のシステム情報窃取 | 2 | | | | 2 | 2 | 2 | 2 |

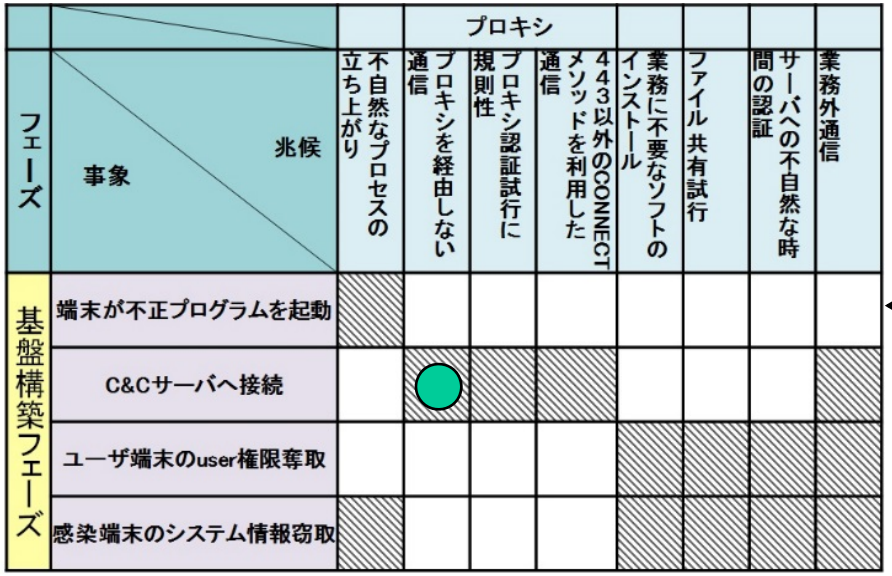
作成時：事象ごとに徴候をリストアップ
運用時：徴候群から事象を推定

数字は関連の強さ

# LIFTシステムの運用イメージ



# 事象把握用追加作業の例




徴候：プロキシーを経由しない通信

事象シーケンス

事象候補：不正C&Cサーバへの接続

②端末が不正プログラムを起動しているかどうかの確認

①確信度向上のための徴候チェック

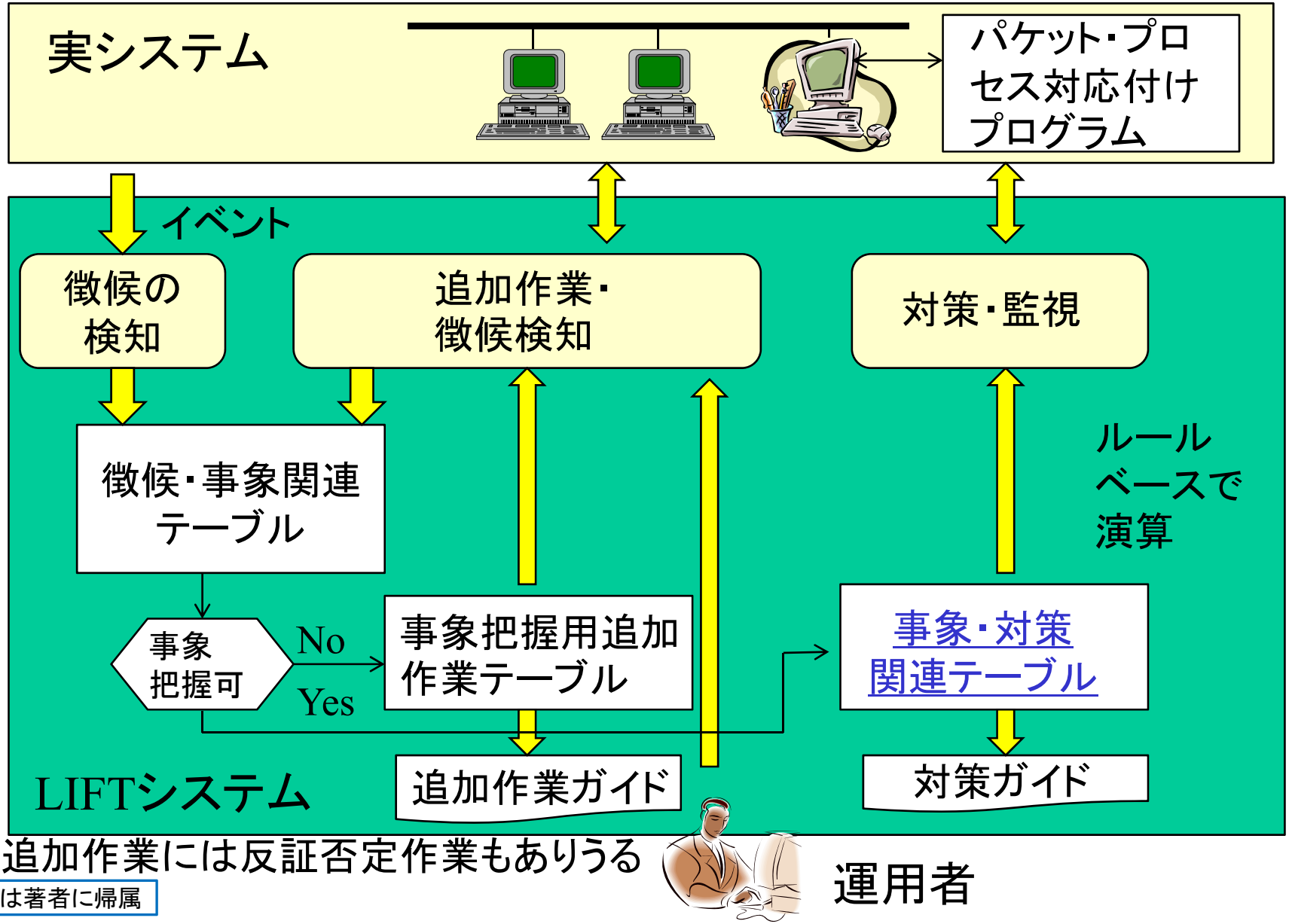
追加作業候補：候補PCの把握

候補PCのパケットと対応プロセスの関連付け

（パケット・プロセス対応付けプログラムの利用）

親プロセスが不正プログラムであることの発見

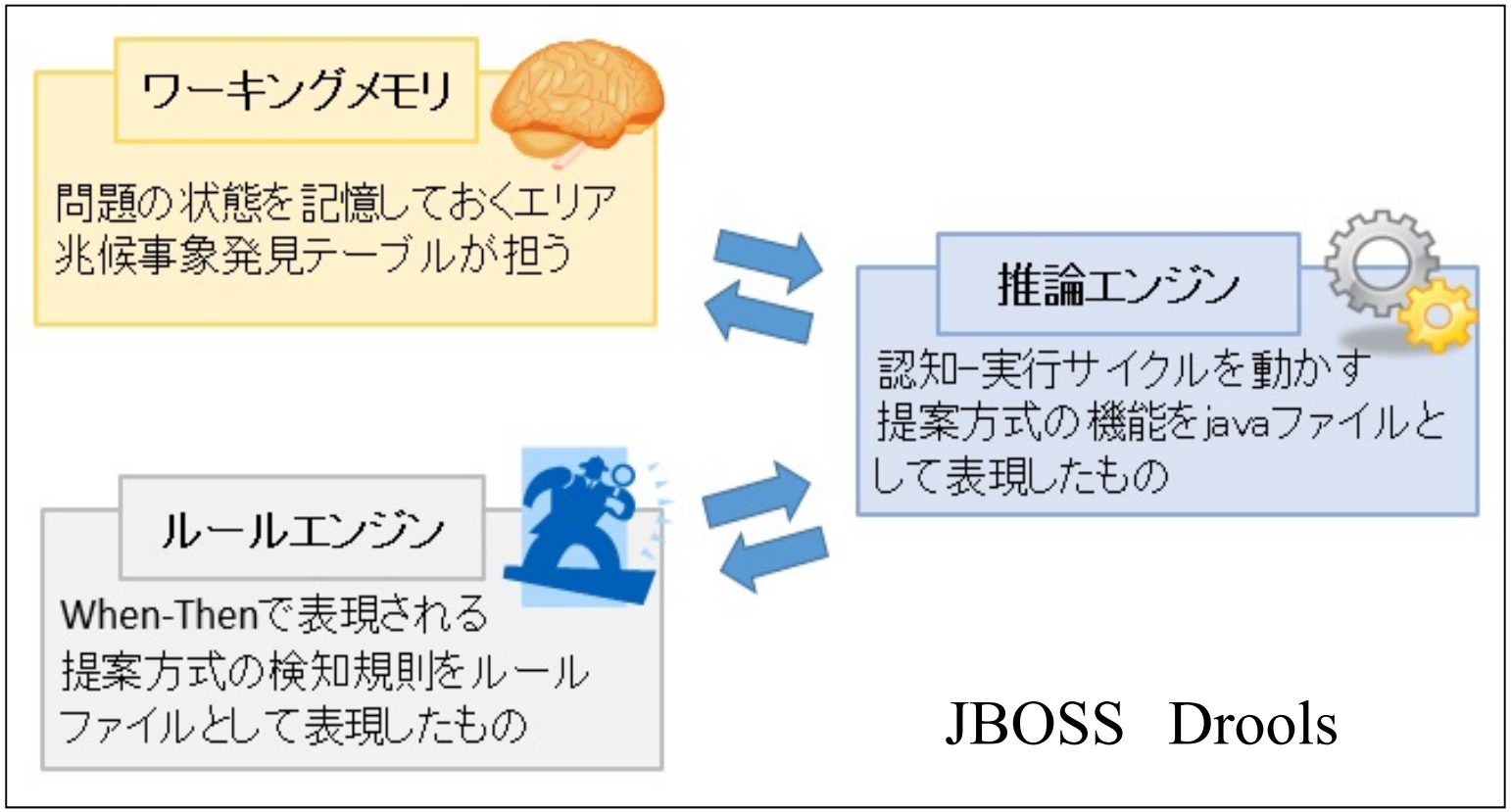
# LIFTシステムの運用イメージ



# 事象・対策関連テーブル

| フェーズ | 事象＼対応 | ルータで該当端末IPの遮断 | ルータで該当ポートの遮断 | ルータで該当通信ドメインの遮断 | 該当端末のインバウンド通信の遮断 | 該当端末のアウトバウンド通信の遮断 | 該当ネットワーク遮断 | 該当端末が所属するネットワークの隔離 | ネットワーク全体の遮断 | 該当端末の隔離 |
|---|---|---|---|---|---|---|---|---|---|---|
| 基盤構築フェーズ | 端末が不正プログラムを実行 | 有効でない | 有効でない | 有効でない | 有効でない | 有効でない | 有効でない | 有効でない | 有効でない | 有効 |
| | C&Cサーバへ接続 | 有効 | 有効 | 有効 | 有効 | 有効 | 有効 | 有効 | 有効 | 有効 |
| | ユーザ端末のuser権限奪取 | 有効でない | 有効でない | 有効でない | 有効 | 有効 | 有効 | 有効でない | 有効でない | 有効 |
| | 感染端末のシステム情報窃取 | 有効でない | 有効でない | 有効でない | 有効でない | 有効でない | 有効でない | 有効でない | 有効でない | 有効 |

# ルールベースプログラムの構成



# ルールベースプログラムの例

JBOSS　Droolsを用いた簡単なプログラムを作成

```
rule "Support"↓
        agenda-group "Aggregation"↓
    when↓
      sign : Sign ()      event : Event ()↓
      support : Support (signId == sign.getID() && eventId == event.getID() &&
effective != true)↓
      Detected (signId == sign.getID())↓
      sourceSupport : SourceSupport(signId == sign.getID() && collectFlag == tru
e)↓
    then↓
      support.setEffective (true);↓
      update (support);↓
      System.out.println ("※Rule Support fired.");↓
      System.out.println ("Assumed Event" + event.getID() + "(" + event.getDescr
iption () + ") is supported by detected Clue " + sign.getID() + "(" ↓
      + sign.getDescription() +  ") with score(確信度) = " + support.getScore();
↓
end↓
```

# 開発プログラム実行画面

```
1 Clue:308(プロキシを経由しない通信)がDetectされました↓
2 Assumed Event:305(C&Cサーバへの通信)の可能性があります↓
3 ↓
4 ※Rule NotSupport fired.↓
5 Mini Clue: 308(プロキシを経由しない通信) のソース 201(Router_log)は取られていま
6 せん↓
7 Mini Clue: 308 のソース 201の取得をONにします↓
8 ↓
9 ※Rule Support fired.↓
10 Assumed Event305(C&Cサーバへの通信) is supported by detected Clue 308(プロキシを
11 経由しない通信) with score(確信度) = 5↓
12 Certainty for Assumed Event305  is 5↓
13 ↓
14 ※Rule Calculate Certainty fired.↓
15 =====↓
16 Most likely event is Assumed Event(C&Cサーバへの通信) with certainty(確信度) = 5↓
17 =====↓
```

基本的適用可能性を確認

# 目次

1. 標的型メール攻撃と各種対策
2. ネットワーク・フォレンジックの現状
3. LIFTシステムの開発構想
4. LIFTシステムの開発状況
5. 今後の計画

LIFT：Live Intelligent Network Forensic Technologies

# 主な成果

＜技術的成果＞

① LIFTシステムの基本構想や概略仕様の提示

②自動運転での攻撃検知と攻撃事象推定機能，および対策方法の算出実行機能の提案

③簡単な例題に対し，ルールベースシステムであるJBOSS Droolsを用いプログラム開発を行い，基本的有効性を確認

＜産学協同的成果＞

パケット・プロセス対応付けプログラムと製品への組み込み決定

＜官学連携的成果＞

LIFTシステム基本技術研究が科研費に採択（2014年度より）

# 今後の対応

①実験を通じた各種テーブルの充実

②応急対応時の管理者へのわかりやすいガイド機能が行えるインターフェースの開発

③揮発情報を保全するメモリフォレンジック機能の導入の検討

④より広い対象に適用できるようにするためのオントロジーや知的システムの導入