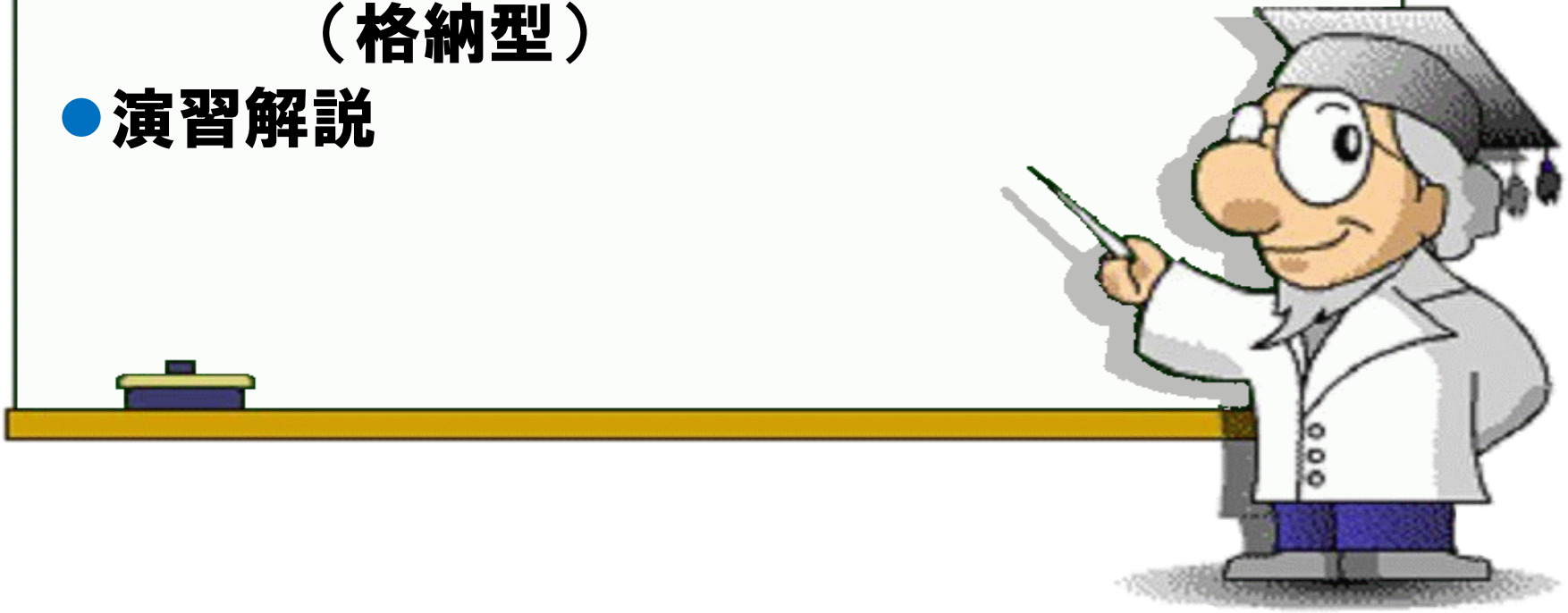
# AppGoatを利用した集合教育補助資料
## －クロスサイトスクリプティング編－

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

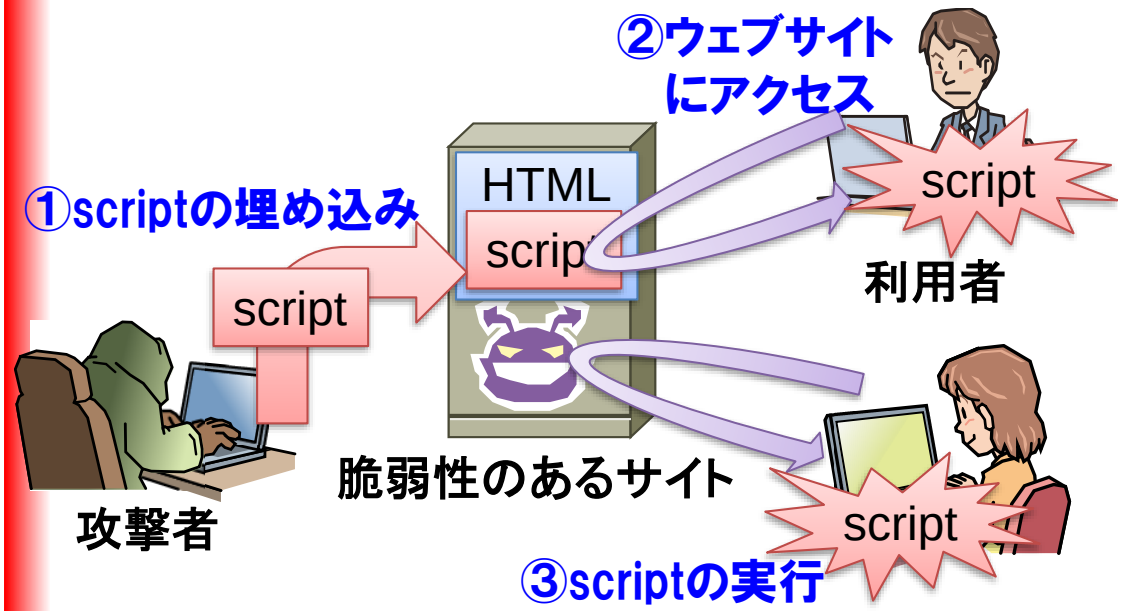
# 講義内容

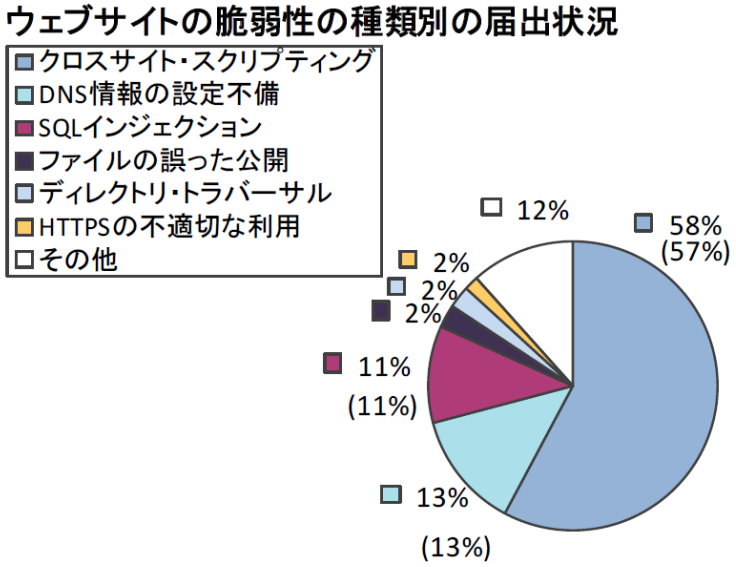
- 脆弱性の原理解説・基礎知識
- 脆弱性の発見方法
- 演習１：掲示板に埋め込まれるスクリプト
  （格納型）
- 演習解説



# クロスサイトスクリプティング（XSS）とは？

- XSS（Cross Site Scripting）とは、スクリプトをサイトに送り込み、スクリプトを含むHTMLを出力し、ブラウザ上で実行させる攻撃
- 「開発者」が最も作り込みやすい脆弱性




（10,408件の内訳、グラフの括弧内は前四半期までの数字）

出典：IPA・JPCERT/CC：ソフトウェア等の脆弱性関連情報に関する届出状況［2019年第3四半期（7月～9月）］

## HTMLとは？？

HTML (Hyper Text Markup Language) とは、タグ (<>) で囲まれた文字列を命令として取扱い、ブラウザで表示する言語。

**ウェブサーバから送付される HTMLソース (テキストファイル)**

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.
<html xmlns="http://www.w3.org/1999/xhtml" lang="ja" xml:lang="ja">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<title>IPA 独立行政法人 情報処理推進機構：情報セキュリティ</title>
<meta http-equiv="Content-Style-Type" content="text/css" />
<meta http-equiv="Content-Script-Type" content="text/javascript" />
<meta name="description" content="情報処理推進機構 (IPA) の「情報セキュリティ」ページで
<meta name="keywords" content="IPA,情報処理推進機構,情報セキュリティ,セキュリティセンタ
<meta http-equiv="X-UA-Compatible" content="IE=EmulateIE8; IE=EmulateIE9" />
<!-- base.css 旧css -->
<link rel="stylesheet" type="text/css" href="" media="screen,print" />
<!-- b-security.css 旧css -->
<link rel="stylesheet" type="text/css" href="" media="screen,print" />
<!-- default.css -->
<link rel="stylesheet" type="text/css" href="/files/default.css" media="all" />
<!-- script.js -->
```

エンコーディング

**ブラウザ表示 イメージ**

# HTMLの基礎知識とクロスサイトスクリプティング

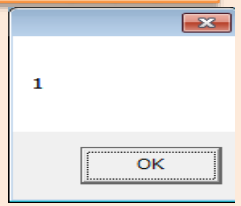
## ● スクリプトタグ(<script></script>)

<script>スクリプト</script>は、タグで囲まれたスクリプト(JavaScript)を実行する。

**HTMLソース**

```
<html>
<script>alert(1)</script>
…</html>
```

**ウェブブラウザ**




## ● エスタグ(<s></s>)

タグで囲まれた文字列に取り消し線を付ける。

**HTMLソース**

```
<html>
<s>IPA</s>
…</html>
```

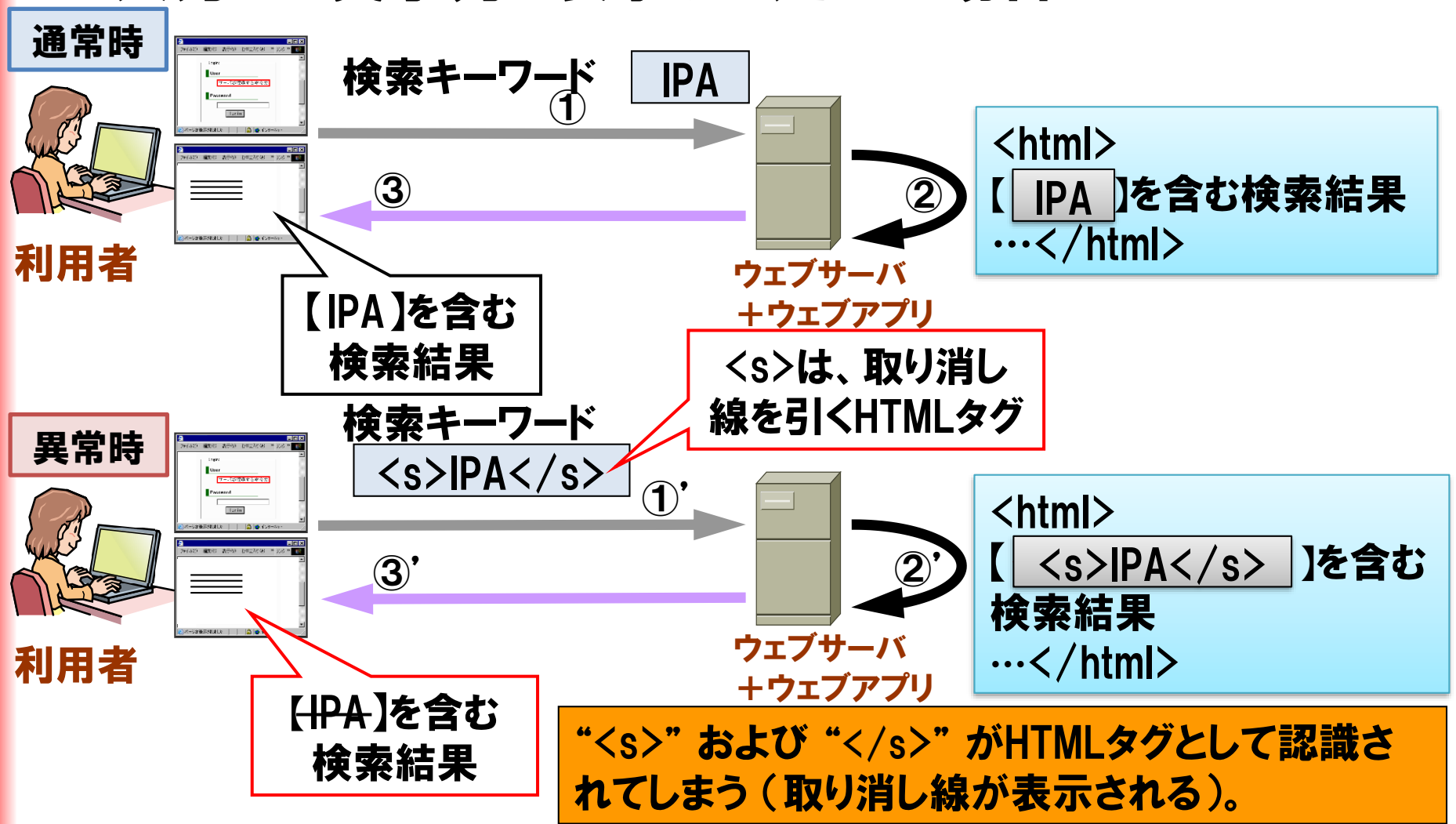
**ウェブブラウザ**

~~IPA~~

# クロスサイトスクリプティングとは
## ～入力値がHTMLタグとして認識されてしまう～

IPA

## ■入力した文字列を表示する処理の場合



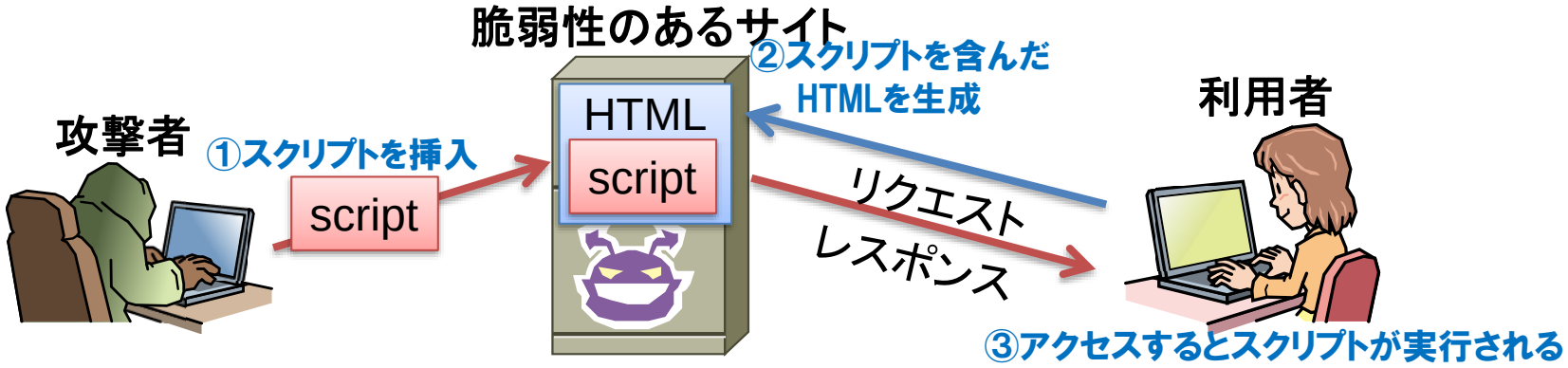

“<s>”および“</s>”がHTMLタグとして認識されてしまう（取り消し線が表示される）。

# XSS脆弱性のタイプ

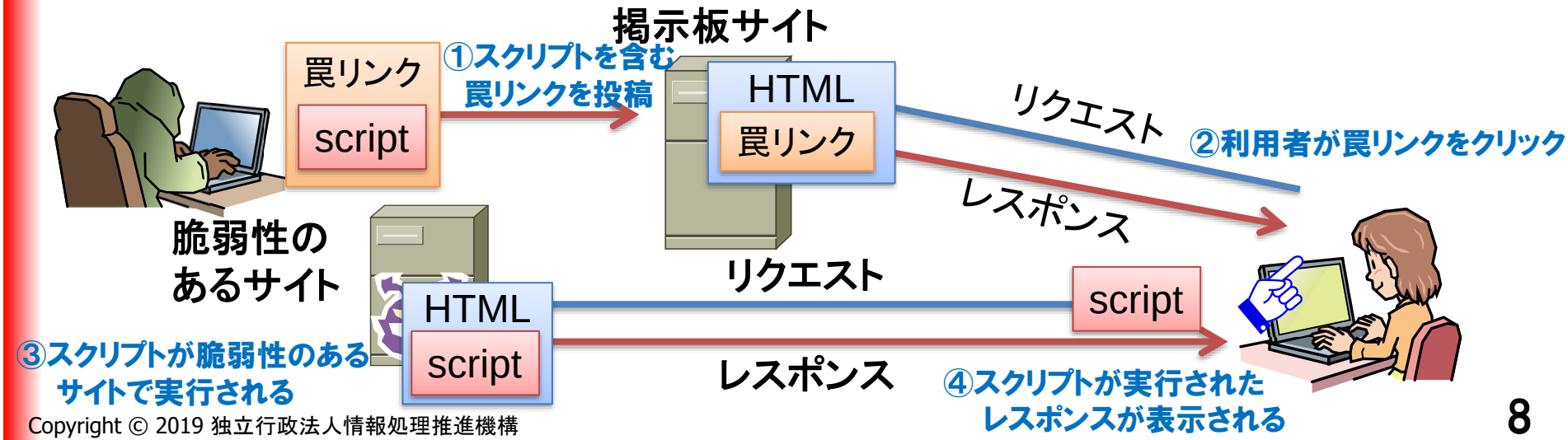
- **格納型（直接攻撃するタイプ）**
  - 攻撃には、脆弱性サイトのみを悪用する。
  - 脆弱性サイトにスクリプトを埋め込んだ後は、当該サイトにアクセスするたびにスクリプトが実行される。
  - 影響を受けるのは、脆弱性サイトにアクセスした人すべて
- **反射型（間接的に攻撃するタイプ）**
  - 攻撃には、罠サイトと脆弱性サイトの2サイトを悪用する。
  - 罠サイト経由で脆弱性サイトにアクセスしたときにスクリプトが実行される。
  - 影響を受けるのは、罠サイトのリンクからアクセスした人

# XSS脆弱性のタイプ

## ● 格納型（直接攻撃するタイプ）




## ● 反射型（間接的に攻撃するタイプ）



# 脆弱箇所の発見

IPA
AppGoat
～突いてみますか？脆弱性！～

## ポイント： HTMLタグが解釈される箇所を発見する

- 入力フォームに以下の値を入力します
  - ✓ ‘>”><hr>⇒脆弱な箇所は水平な罫線が引かれる

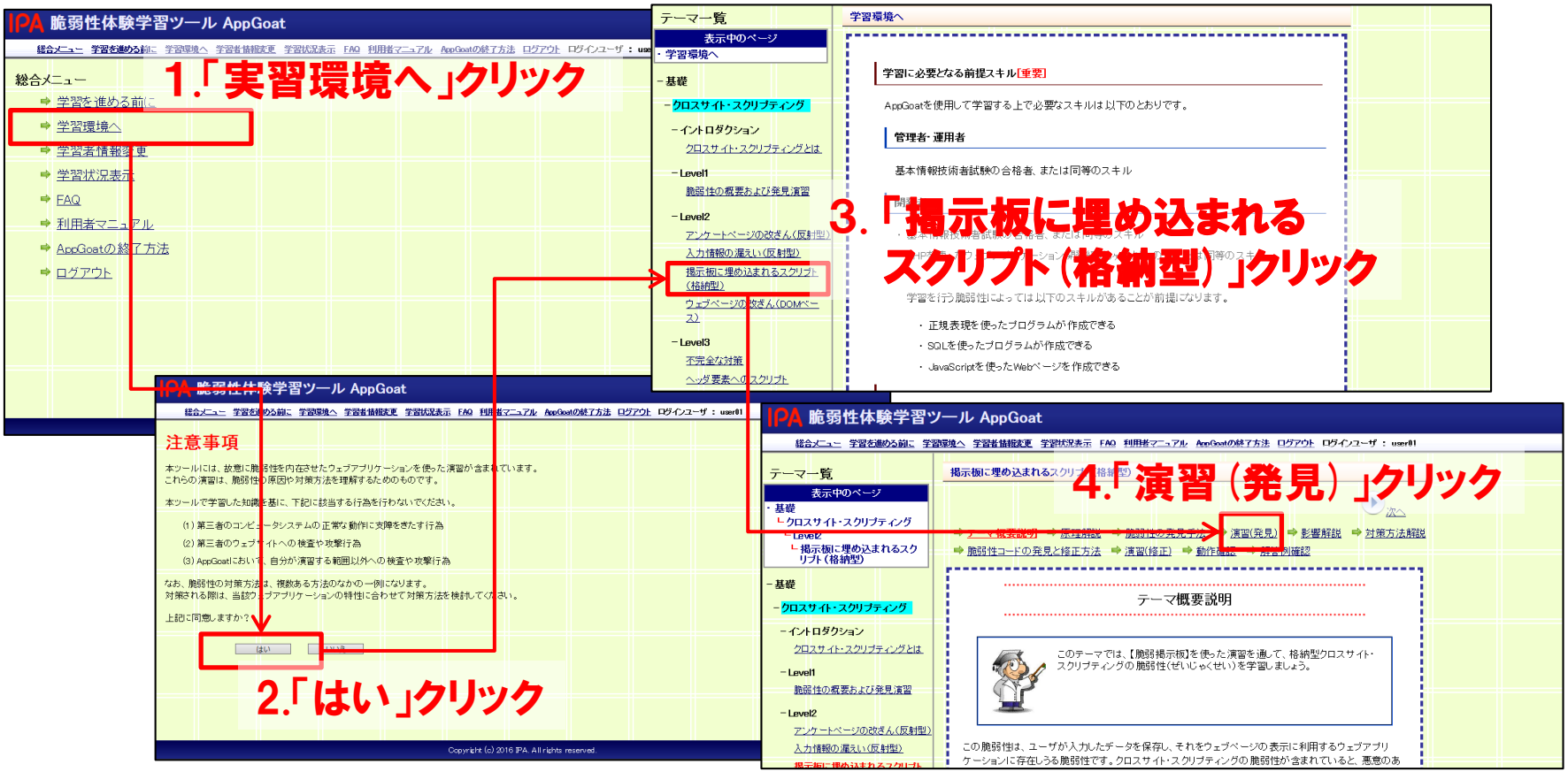

- 別の値を入力することでも発見できます
  - ✓ <script>alert (“alert test”) </script>

    ⇒脆弱な箇所はポップアップが表示される

# [演習] AppGoatの準備

①AppGoatを起動します

②以下の遷移で演習画面に移動します



# ［演習］AppGoatを用いた疑似攻撃体験

IPA

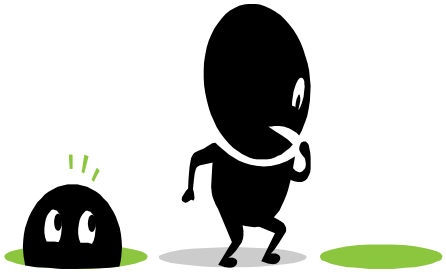
AppGoat
～突いてみますか？脆弱性！～

- **演習テーマ：**

  **「掲示板に埋め込まれるスクリプト（格納型）」**

- **ミッション：**

  **スクリプトを埋め込んでみましょう！**



Congratulations!!演習の目標を達成しました。

# [演習] 演習の進め方

IPA

AppGoat
～突いてみますか？脆弱性！～

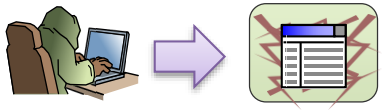
## Step1：脆弱性となる箇所を見つける

**手順1：掲示板の入力フィールドに様々な値を入力して、動作を確認する。**

**演習の手順**

HTMLタグを入力して、脆弱な箇所を発見してみましょう

## Step2：スクリプトを含むコメントを投稿する

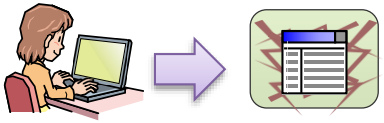


**手順2：スクリプトを掲示板サイトに投稿する。**

**演習の手順**

攻撃者の立場で、罠の設置に必要な手順を考えてみましょう

## Step3：掲示板サイトにアクセスする



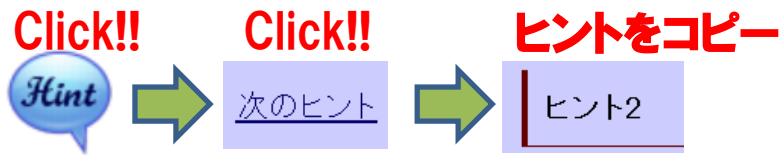
**手順3：掲示板サイトにアクセスして、スクリプトが実行されることを確認する。**

**演習の手順**

利用者の立場になり、罠サイトにアクセスした後の影響を考えてみましょう

# [演習] 演習の進め方

- 演習を円滑に進めるために
  - AppGoatのヒント機能を使い、攻撃用スクリプトをコピーして使用ください。




**ダイアログが表示されるスクリプト**

```
<script>alert ('Dialog by XSS') </script>
```

  - 下記の手順で、HTMLソースを確認できます。

**Internet Explorer**

掲示板上で右クリックし、「ソースを表示」でHTMLソースを表示する。

**Firefox**

掲示板上で右クリックし、「このフレーム」⇒「フレームのソースを表示」でHTMLソースを表示する。

# 演習

IPA

## 演習はじめてください。

※演習が終わったら次のページで解説を行います。

# [手順1]  入力フィールドに様々な値を入力する

IPA AppGoat ～突いてみますか？脆弱性！～

**演習の手順**

HTMLタグを入力して、脆弱な箇所を発見してみましょう

- 掲示板の複数の入力欄に「’ >”><hr>」を入れて、水平の罫線が引かれているところを探しましょう。

**ウェブブラウザ上の表示**

13 ’>”><hr>title : ’>”><hr>name 2013年07月05日 17:22 削除

’>”>

comment

- 「本文」を入力する箇所で、入力した値がそのまま使われていることが確認できます。

**HTMLソース**

```
<div class="comment_content"> ’>”><hr>comment</div>
```

# [演習] 演習の進め方

IPA

AppGoat ～突いてみますか？脆弱性！～

- Step1：脆弱性となる箇所を見つける 

  手順1：掲示板の入力フィールドに様々な値を入力して、動作を確認する。

**演習の手順**

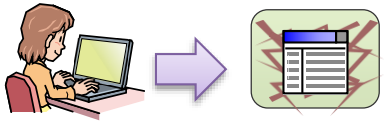
HTMLタグを入力して、脆弱な箇所を発見してみましょう

- Step2：スクリプトを含むコメントを投稿する 

  手順2：スクリプトを掲示板サイトに投稿する。

**演習の手順**

攻撃者の立場で、罠の設置に必要な手順を考えてみましょう

- Step3：掲示板サイトにアクセスする 

  手順3：掲示板サイトにアクセスして、スクリプトが実行されることを確認する。

**演習の手順**

利用者の立場になり、罠サイトにアクセスした後の影響を考えてみましょう

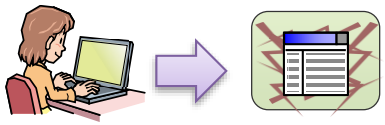
# ［手順2］ 
# スクリプトを掲示板サイトに投稿する

IPA AppGoat ～突いてみますか？脆弱性！～

1. **攻撃者の立場になり**、掲示板にスクリプトを埋め込みます。メッセージダイアログを表示させるスクリプトを使います。


2. 「投稿」ボタンを押下します。これでスクリプトの埋め込みが完了しました。

# ［手順3］スクリプトが実行されることを確認する



IPA

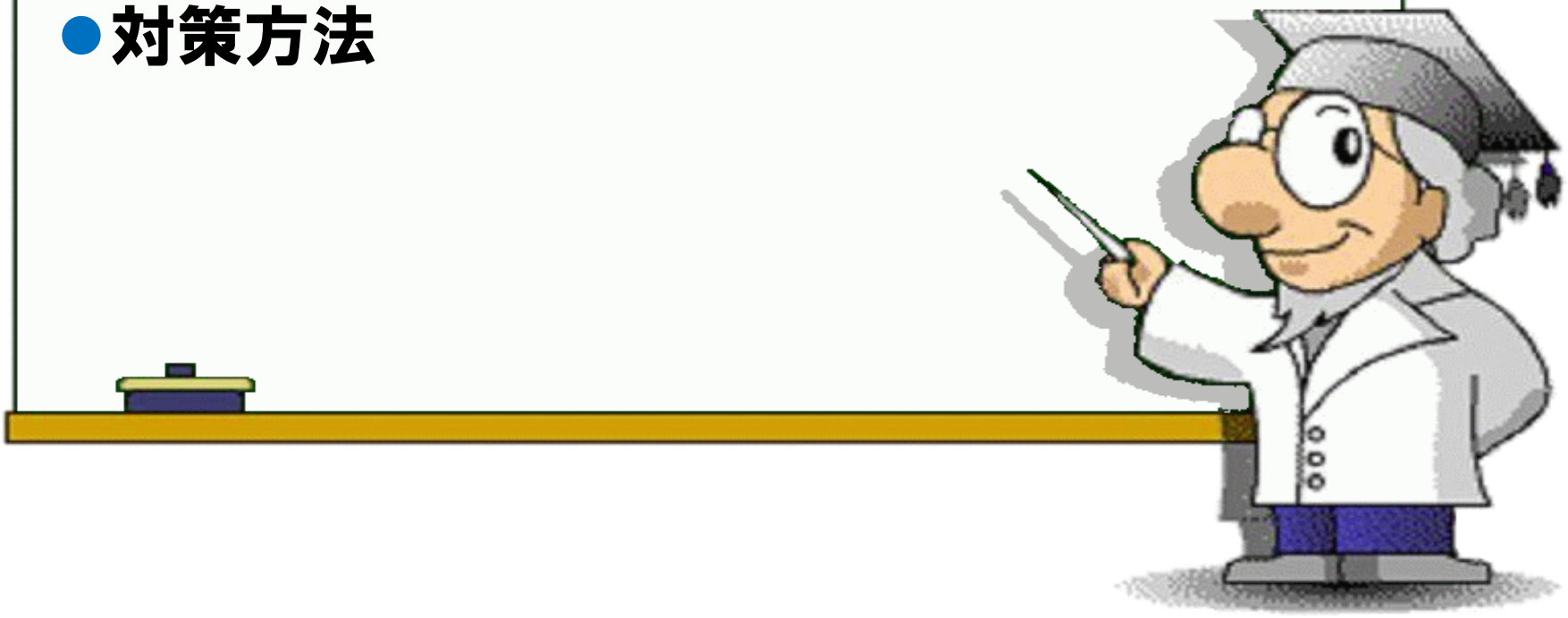
AppGoat
～突いてみますか？脆弱性！～

1. **利用者の立場になり**、掲示板サイトにアクセスします。

2. その結果、**利用者のウェブブラウザ上**でスクリプトが実行されます。

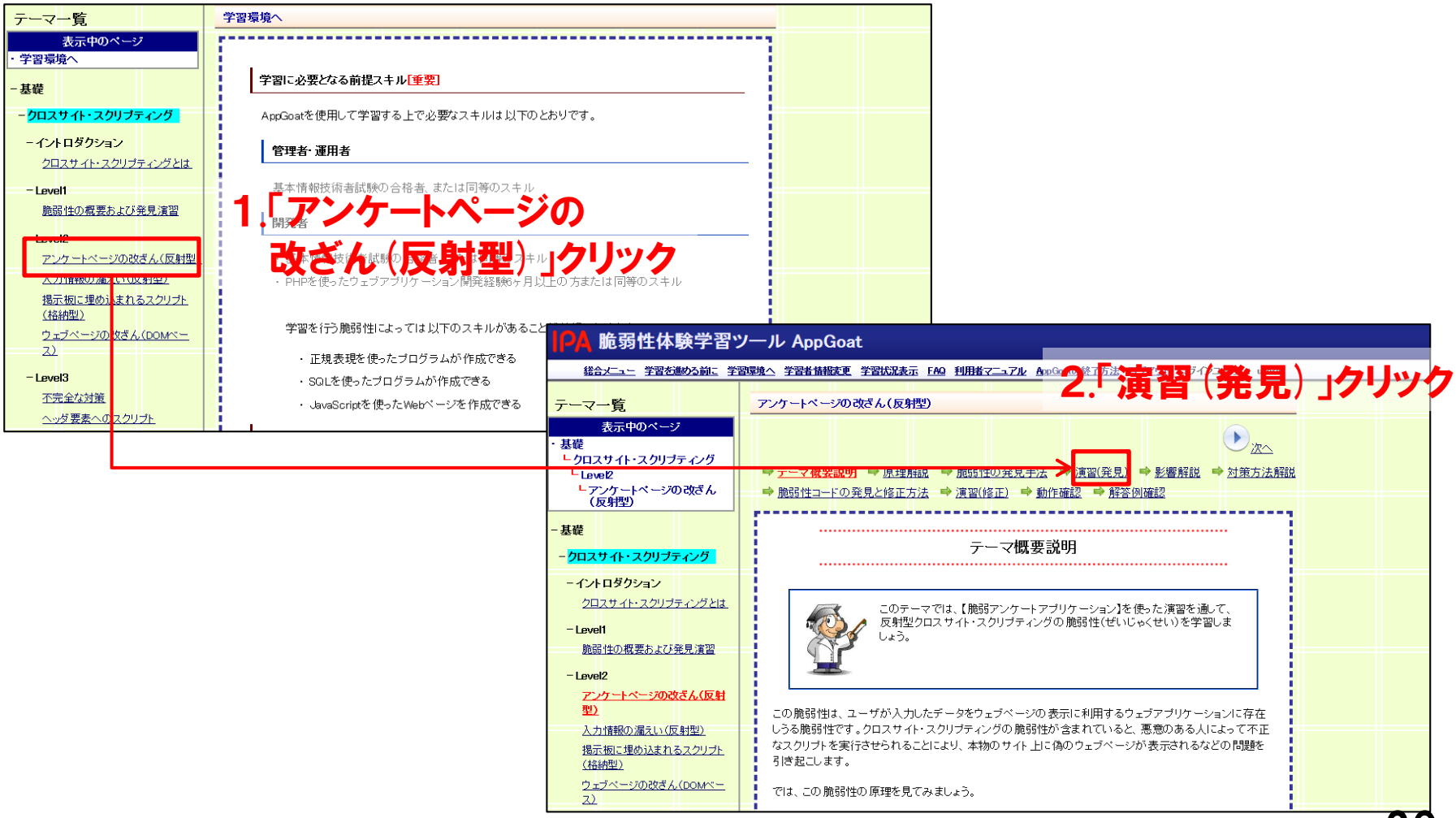
# 内容（演習2）

- 脆弱性の原理解説・基礎知識
- 演習2：アンケートページの改ざん（反射型）
- 演習解説
- 対策方法



# [演習] AppGoatの準備

## ①以下の遷移で演習画面に移動します



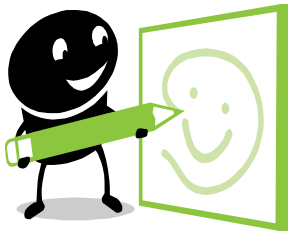
# ［演習］AppGoatを用いた疑似攻撃体験

IPA

AppGoat
～突いてみますか？脆弱性！～

- **演習テーマ：**

  **「アンケートページの改ざん（反射型）」**

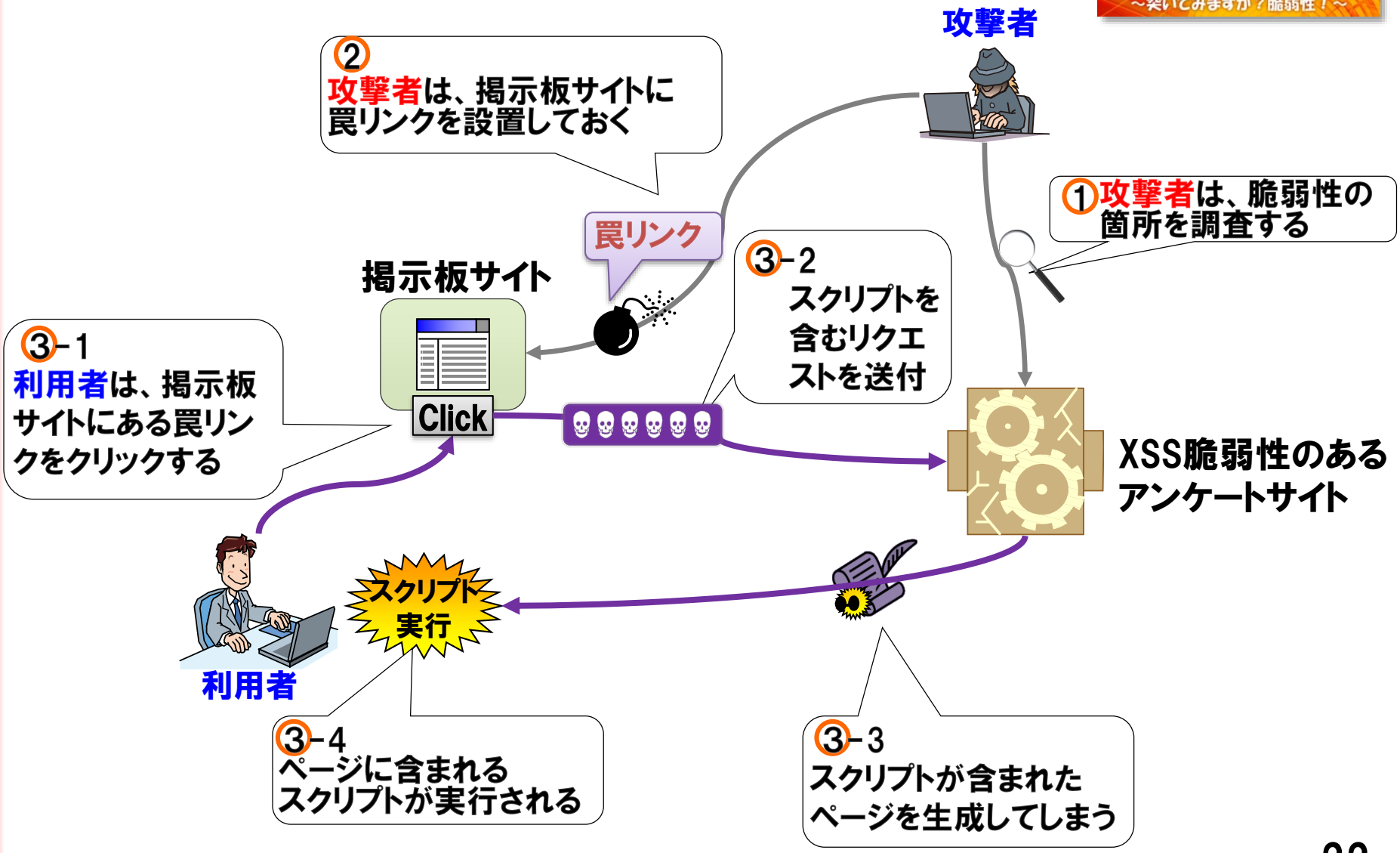


- **ミッション：**

  **アンケートページを改ざんしてみましょう**

Congratulations!!演習の目標を達成しました。

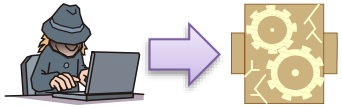
# [演習] 疑似攻撃のイメージ



# [演習] 演習の進め方

IPA

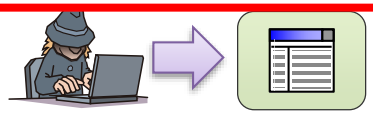
AppGoat

- Step1：脆弱性となる箇所を見つける 

  **手順1：**アンケートページの入力フィールドに様々な値を入力して、動作を確認する。

  **演習の手順**

  HTMLタグを入力して、脆弱な箇所を発見してみましょう

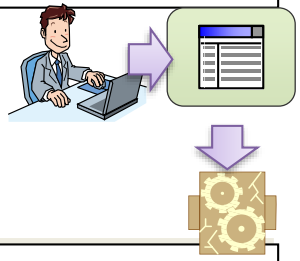
- Step2：罠リンクを設置する 

  **手順2：**「ヒント」などを参考に、スクリプトを実行させるURLを考える。

  **手順3：**罠リンクを設置するため、Step2のURLを掲示板サイトに投稿する。

  **演習の手順**

  攻撃者の立場で、罠リンクの作成や設置に必要な手順を考えてみましょう

- Step3：罠リンク経由でアンケートサイトにアクセスする 

  **手順4：**罠リンクをクリックして、スクリプトが実行されることを確認する。

  **演習の手順**

  利用者の立場になり、罠リンクをクリックした後の影響を考えてみましょう

# [演習] 演習の進め方

- 演習を円滑に進めるために
  - AppGoatのヒント機能を使い、攻撃用スクリプトをコピーして使用ください。


  - 下記の手順で、HTMLソースを確認できます。

**Internet Explorer**

アンケートページ上で右クリックし、「ソースを表示」でHTMLソースを表示する。

**Firefox**

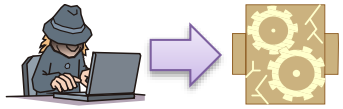
アンケートページ上で右クリックし、「このフレーム」⇒「フレームのソースを表示」でHTMLソースを表示する。

# 演習

IPA

## 演習はじめてください。

※演習が終わったら次のページで解説を行います。

# [Step1] 
# 入力フィールドに様々な値を入力する

IPA AppGoat ～突いてみますか？脆弱性！～

**演習の手順**

HTMLタグを入力して、脆弱な箇所を発見してみましょう

- アンケートサイトの複数の入力欄に「’ >”><hr>」を入れて、
  アンケート内容に関するエラーページを表示してみましょう。

**ウェブブラウザ上の表示**

名前に不正な文字列が含まれています。あなたの入力した名前は’>”>
です。

- アンケートページに送ったリクエスト（URL）のnameパラメータに
  脆弱性の存在する箇所と推測

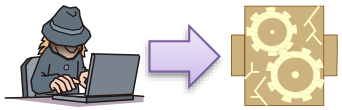
**URL**

http://IPアドレス/Users/ログインID/Web/Scenario1121/VulSoft/enquete.php?page=2&name=’ >”><hr>&sex=0&old=50&company=’ >”><hr>&xss=1&trouble=1&content=’ >”><hr>

# [演習] 演習の進め方

IPA

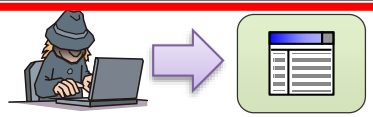
AppGoat ～突いてみますか？脆弱性！～

- Step1：脆弱性となる箇所を見つける 

  **手順1：**アンケートページの入力フィールドに様々な値を入力して、動作を確認する。

**演習の手順**

HTMLタグを入力して、脆弱な箇所を発見してみましょう

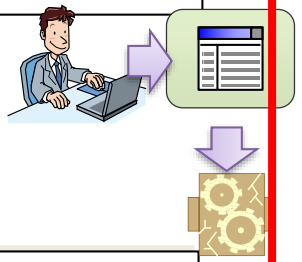
- Step2：罠リンクを設置する 

  **手順2：**「ヒント」などを参考に、スクリプトを実行させるURLを考える。

  **手順3：**罠リンクを設置するため、手順2のURLを掲示板サイトに投稿する。

**演習の手順**

攻撃者の立場で、罠リンクの作成や設置に必要な手順を考えてみましょう

- Step3：罠リンク経由でアンケートサイトにアクセスする 

  **手順4：**罠リンクをクリックして、スクリプトが実行されることを確認する。

**演習の手順**

利用者の立場になり、罠リンクをクリックした後の影響を考えてみましょう

# [Step2] スクリプトを実行させるURLを考える

IPA AppGoat ～突いてみますか？脆弱性！～

- アンケートサイトのXSS脆弱性を突いて、ヒントで表示されたスクリプトを実行させるURLを考えましょう。

1. アンケートに答えて「アンケート投稿」ボタンを押下し、アクセスするURLを確認します。

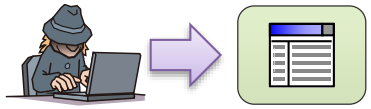
**アクセスするURL**

```
http://IPアドレス/Users/ログインID/Web/Scenario1121/VulSoft/enquete.php?page=2&name=test&sex=0&old=30&company=&xss=1&trouble=1&content=
```

2. 脆弱性となる箇所（nameパラメータ）に、スクリプトに相当する文字列を入れます。

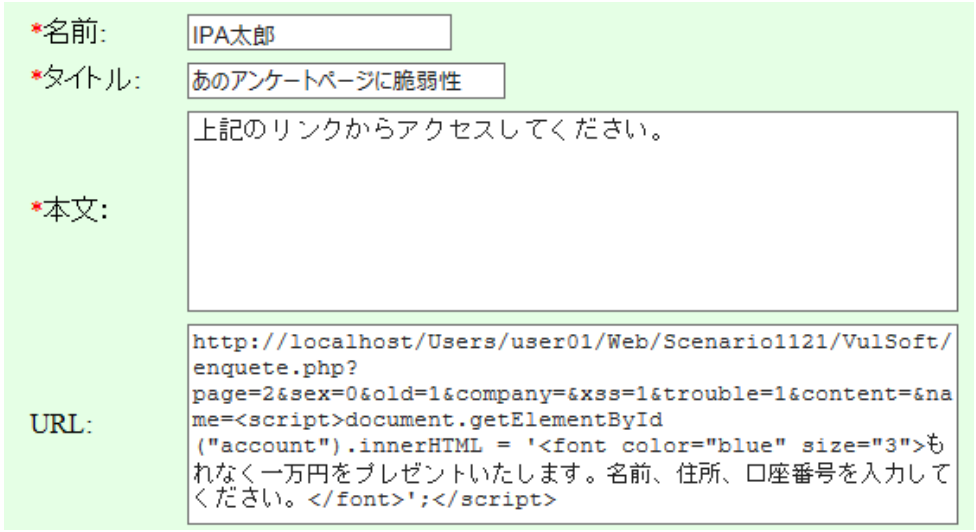
**スクリプトを実行させるURL**

```
http://IPアドレス/Users/ログインID/Web/Scenario1121/Vulsoft/enquete.php?page=2&sex=0&old=1&company=&xss=1&trouble=1&content=&name=<script>document.getElementById("account").innerHTML = '<font color="blue" size="3">もれなく一万円をプレゼントいたします。名前、住所、口座番号を入力してくださ い。</font>';</script>
```

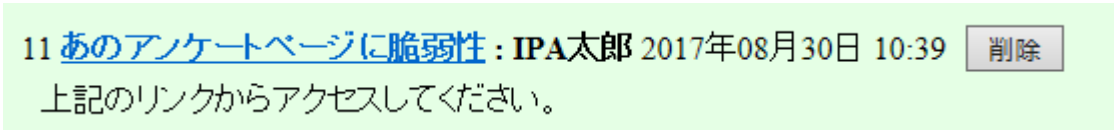
# [Step2] 
# 手順2のURLを掲示板サイトに投稿する

IPA
AppGoat ～突いてみますか？脆弱性！～

1. **攻撃者の立場になり**、掲示板に罠のリンクを作成します。罠のリンクには、先ほど作成したスクリプトを実行させるURLを入力します。



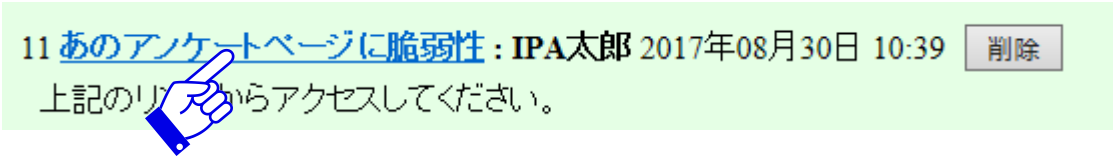

2. 「投稿」ボタンを押下します。これで罠リンクの設置が完了しました。



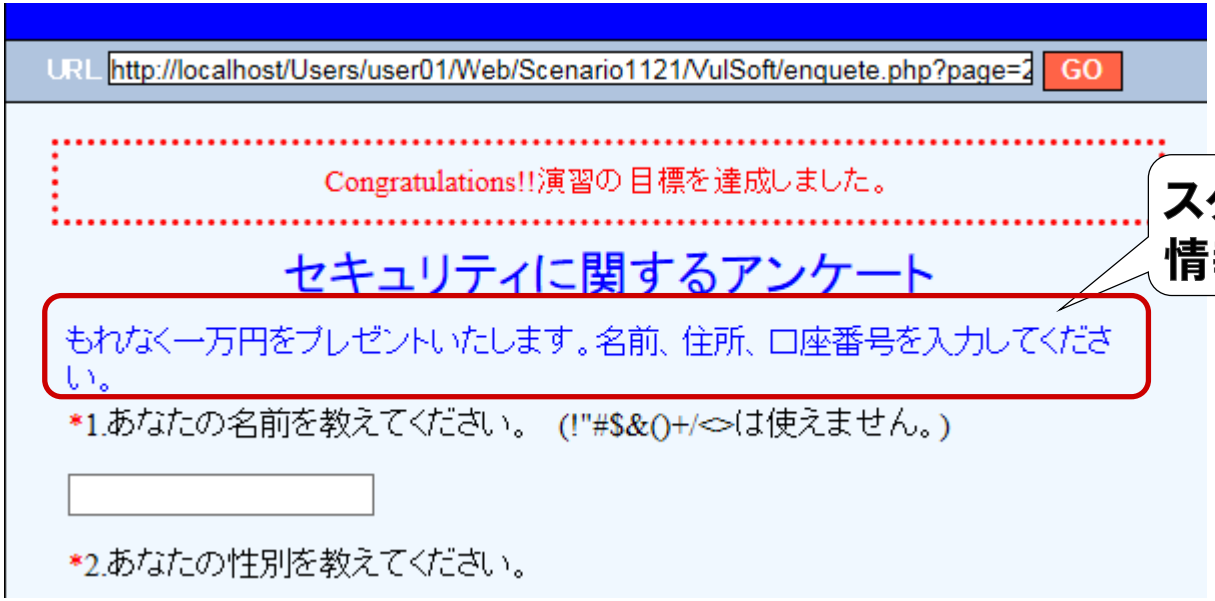
# [Step3] スクリプトが実行されることを確認する

IPA AppGoat ～突いてみますか？脆弱性！～

1. **利用者の立場になり**、罠のリンクをクリックし、アンケートページにアクセスします。

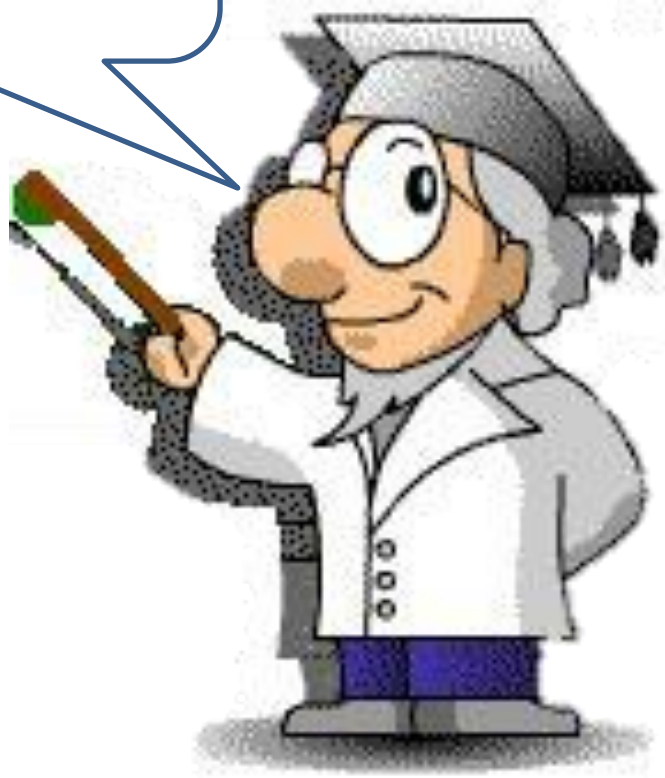



2. その結果、**利用者のウェブブラウザ上**でスクリプトが実行され、アンケートの内容が書きかえられることが確認できます。



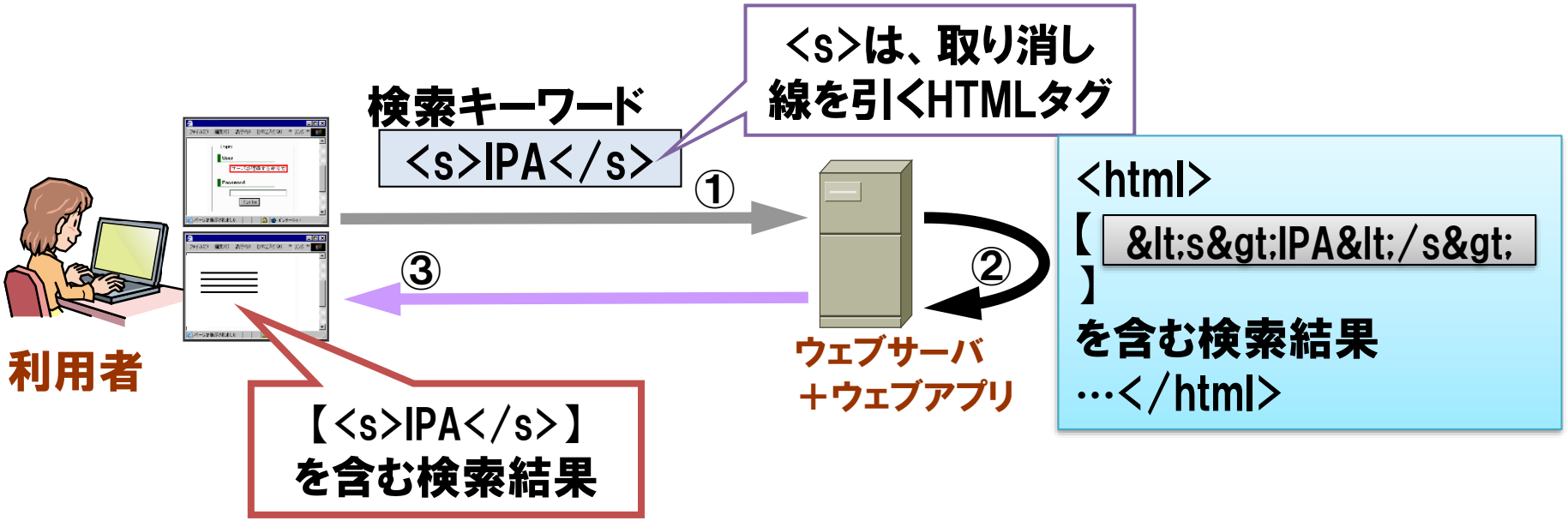
# 対策方法の解説

対策の解説



# 対策のポイント

- ユーザが入力した文字列をHTMLタグとして解釈しないように処理する



# 正しい対策の例

- HTMLにおける特別な意味を持つ「記号文字」を**エスケープ処理**する（文字参照に置換）

例：

| | | |
|---|---|---|
| < → < | " → " | & → & |
| > → > | ' → ' | |

入力値：<script>alert ("a") ;</script>
置換後：<script>alert ("a") ;</script>

- プログラム言語に用意されているエスケープ用の関数
  - ✓ Perl⇒escapeHTML ()
  - ✓ PHP⇒ htmlspecialchars ()

以上で、
クロスサイトスクリプティングの解説は終了です。