2 日で学ぶ情報セキュリティ

実習用補足資料

(Revision: 0.1)

目次

実習に使う PC について……1
前準備……2
課題１：ハッシュ暗号を使ってみる……7
課題２：ディジタル署名を利用してみる……8
課題３：インターネット上に公開される情報の確認……15
課題４：DNS の情報を確認してみる……16
課題５：ブラウザクラッシャーを体験してみる……18
課題６：whois データベースをつかってみる……19
課題７：ping スイープを実行する……20
課題８：対象機器のオープンポートを確認する……22
課題９：パスワード解析をやってみる……28
課題 10：XSS 攻撃を試してみる……32
課題 11：SQL インジェクション攻撃を試してみる……38
課題 12：CSRF 攻撃を試してみる……45
課題 13：CMS への攻撃を試してみる……49
課題 14：OpenVAS を使って、システムの脆弱性診断を実行してみる……52
課題 15：バックドアからの侵入を試してみる……59
課題 16：サーバのセキュリティ向上を行ってみる……61
課題 17：Snort を使ってシステムの侵入検知を実行してみる……66
課題 18：Windows のログを確認してみる……71

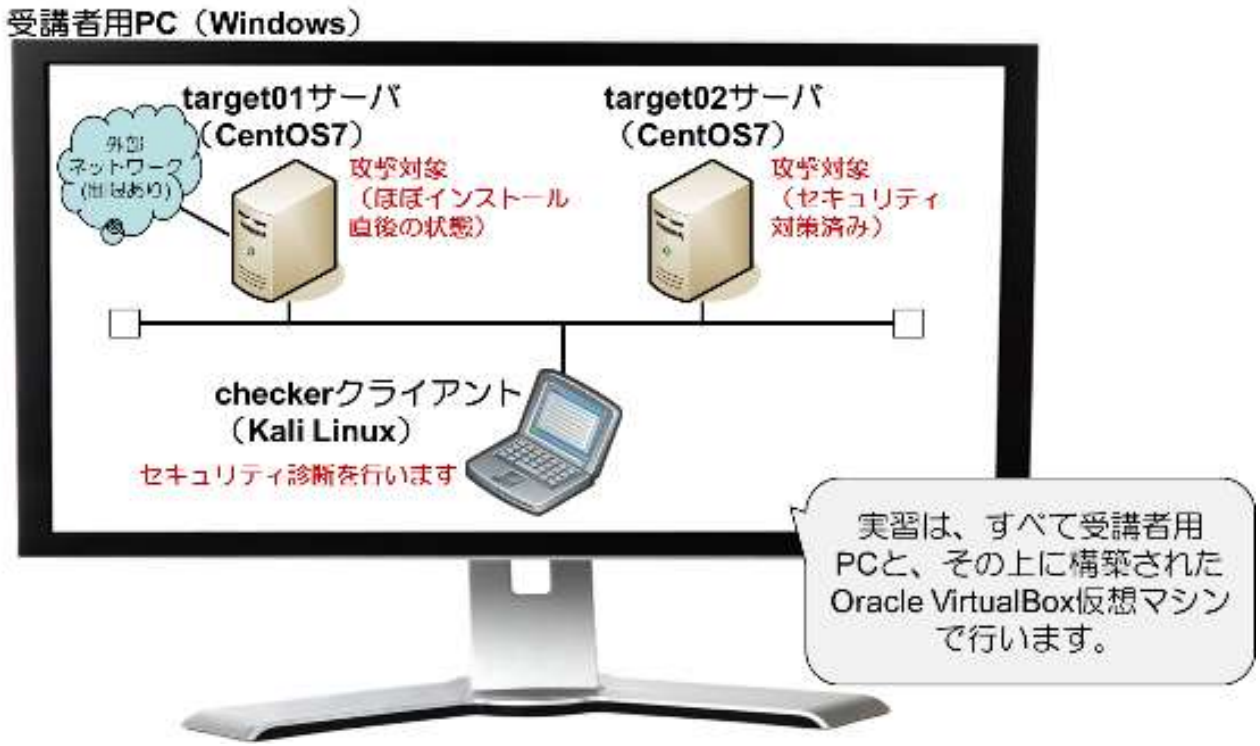
ステップアップ課題　解答……72

実習に使う PC について

- target01 サーバ
  OS: CentOS 7
  ホスト名：target01
  root パスワード：[redacted]

- target02 サーバ
  OS: CentOS 7 + 最新パッチ
  ホスト名：target02
  root パスワード：[redacted]

- checker クライアント
  OS: Kali Linux 2016.2 + 最新パッチ
  ホスト名：checker
  root パスワード：[redacted]



前準備

■Oracle VM VirtualBox の使い方

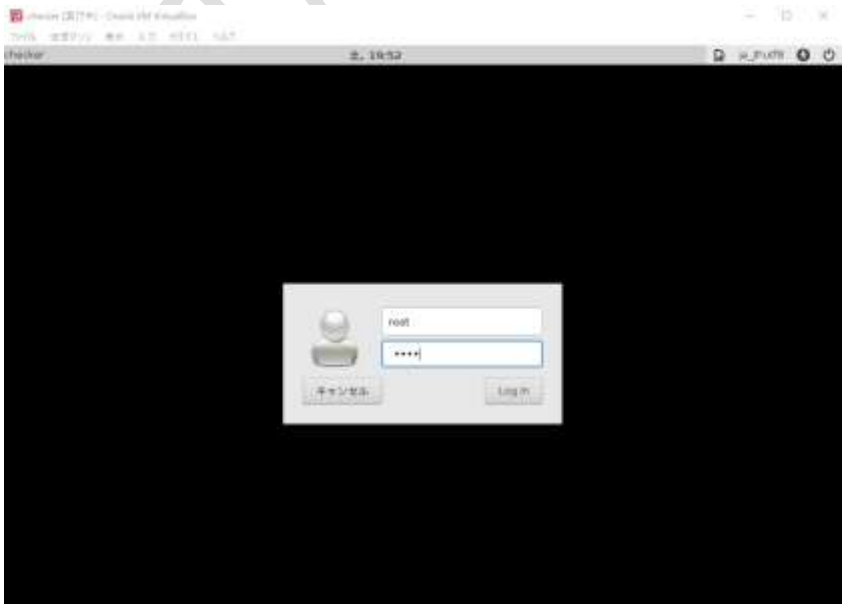
下記手順で、必要な仮想マシンを起動します。

1. [スタート] - [Oracle VM VirtualBox] - [Oracle VM VirtualBox]を選択します。
   下図のような画面が表示されます。

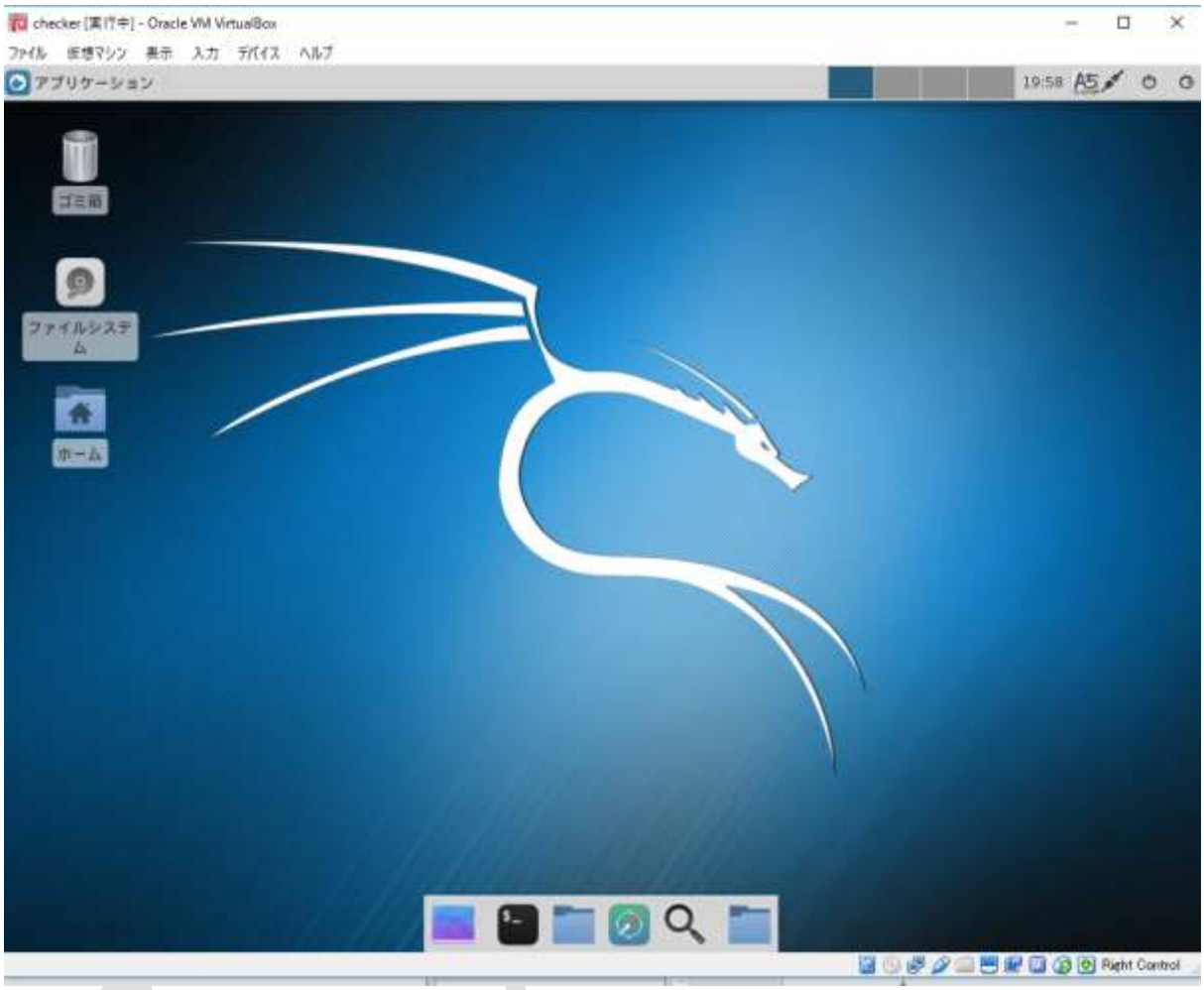
2. 起動したい仮想マシンを選択し、 

ボタンをクリックします。

3. Checkerクライアントの場合、下図のような画面が表示されるので、上欄にユーザ「**root**」、下欄にパスワード「[redacted]」を入力し、[Log in]ボタンをクリックします。

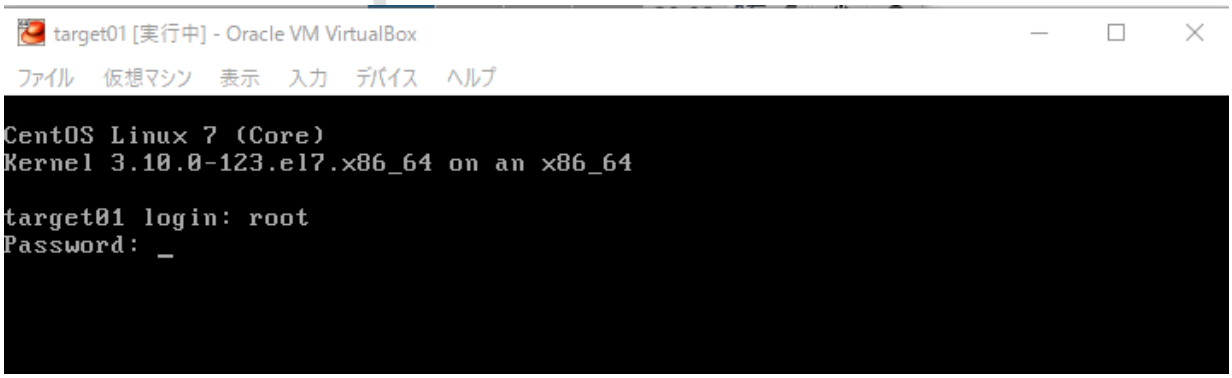


4. 正常にログインできれば、下図のような画面が表示されます。



実習では、主に「ターミナルエミュレータ」、「ブラウザ」、「ファイルマネージャ」を使います。
いずれも、画面下のアイコンをクリックすると起動します。

5. target01 サーバ、および target02 サーバの場合、下図のような画面が表示されますので、ユーザ「root」[Enter]、パスワード「[redacted]」（target02 サーバの場合は「[redacted]」）[Enter]を入力します。



課題 1：ハッシュ暗号を使ってみる

（テキスト：第２章 2.4 ハッシュ暗号とメッセージ認証コード 参照）

目的：

同一アルゴリズムのハッシュ関数を使って、同じファイルから作成されたハッシュ値は「同じになる」ことを確認します。

手順：

1. checker クライアントで実行：ファイルのハッシュ化

   「ターミナル」にて下記コマンドを実行し、結果を確認します。

```
# cd /root
# md5sum sample.txt
2de3b9bf1c5b133340a8ce0957faeeac  sample.txt <-例
```

2. target01 サーバで実行：ファイルのハッシュ化

   1.と同様に、下記コマンドを実行し、結果を確認します。

```
# cd /root
# md5sum sample.txt
2de3b9bf1c5b133340a8ce0957faeeac  sample.txt <-例
```

3. checker クライアントと target01 サーバとで、結果の値が同じであることを確認します。

4. 他のアルゴリズムの関数を使ってみて、md5sum で作成されたハッシュ値と違うことを確認します。

```
# cd /root
# sha512sum sample.txt
6d10fc5a00cb2d753ec72000bfc0691618bb3b5d13446bdd8ab168fac11a59a76db79849c570499963a55147a2
a203cf410e43b5b4ddf41e286704b6d0e4543a  sample.txt <-例
```

同一アルゴリズムのハッシュ関数（この実習の場合、md5sum）を使って、同じファイルから作成されたハッシュ値は「**同じになる**」ことが確認できます。

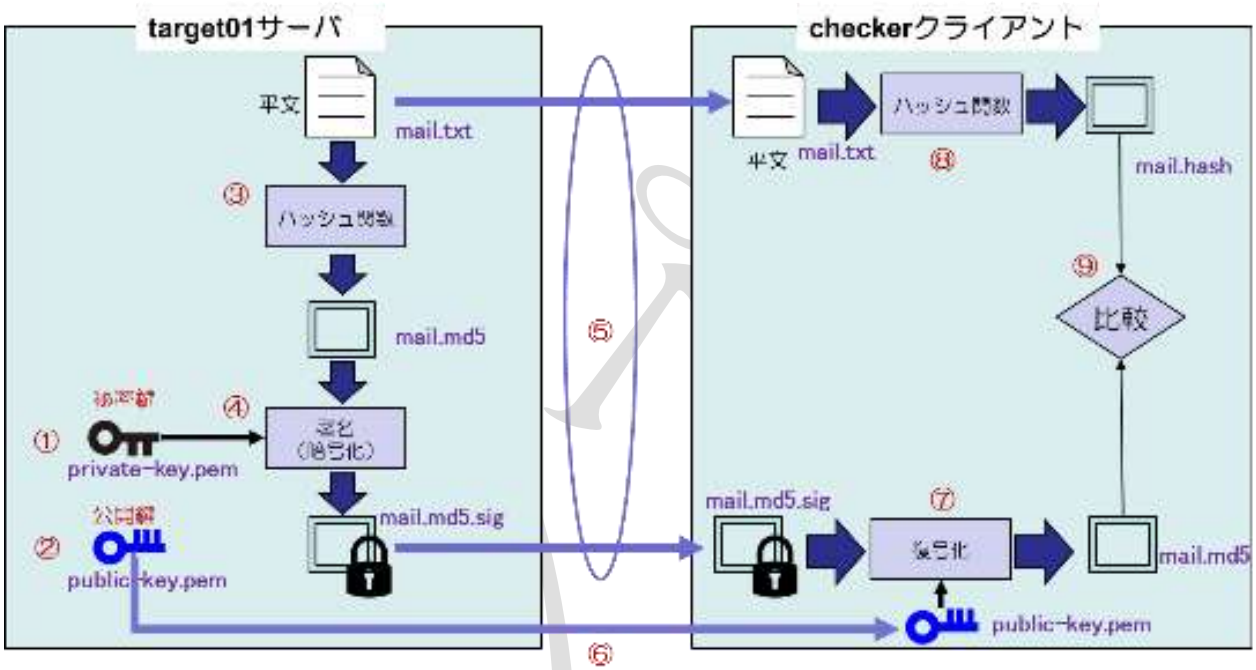
課題２：ディジタル署名を利用してみる

（テキスト：第２章 2.5 否認と否認防止 参照）

目的：

ディジタル署名を使うことによって、通信の途中で改ざんされた場合、検知できることを確認します。

手順：



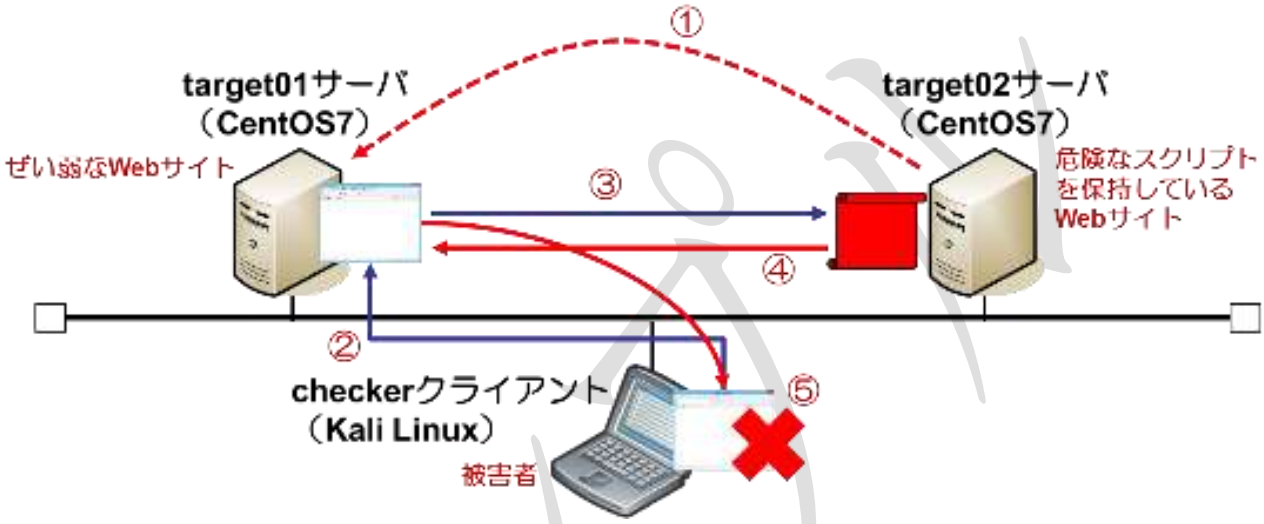
課題 10：XSS 攻撃を試してみる

（テキスト：第 5 章　5.5 Web アプリケーションへの攻撃　参照）

目的：

どのような経緯でクロスサイトスクリプティング(XSS)攻撃が行われるのかを疑似体験してみます。

攻撃パターン例：パワーポイントのスライド（課題 10：XSS 攻撃を試してみる-1）参照



1. 攻撃者はあらかじめぜい弱な Web サイトを見つけ、攻撃者が用意したサーバ(target02)内にある危険なスクリプトを、ぜい弱な Web サイト（target01）のページから呼び出せるように仕掛けを準備する。

2. 利用者（被害者）は、ぜい弱な Web サイト（target01）の仕掛けが施されたページを訪れ、仕掛けをクリックしてしまう。

3. target01 の Web ページは、攻撃者が用意した危険なスクリプトを呼び出しにいく。

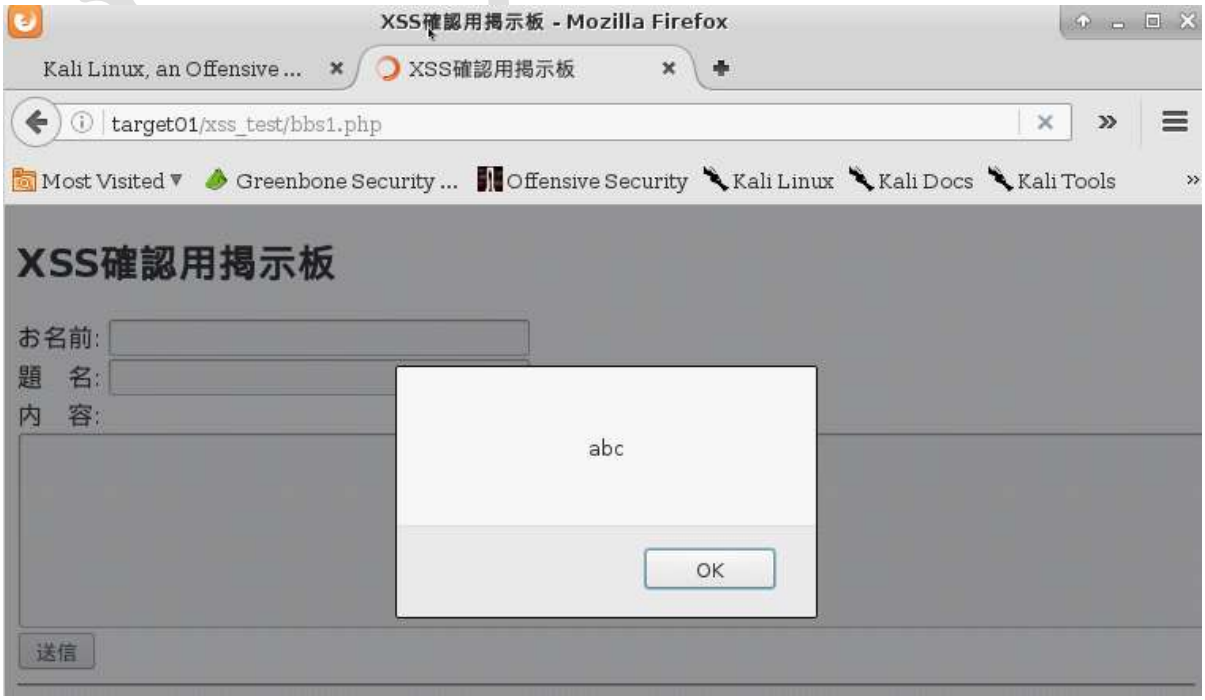
4. 攻撃者が用意した危険なスクリプトは、ぜい弱な Web サイトに呼び出されていく。

5. ぜい弱なサイト経由で、攻撃者が用意した危険なスクリプトが、checker クライアント上で実行され、被害にあう。

手順：

1. ぜい弱な Web アプリケーションを見つける。
   下記 URL へアクセスし、内容に「**<script>alert(“abc”);</script>**」と書き、[送信]ボタンを押してみる。
   お名前、題名は何でもよい。
   http://target01/xss_test/bbs1.php

2. 下記の状態になれば、ぜい弱な Web アプリケーションである可能性が高いことがわかる。



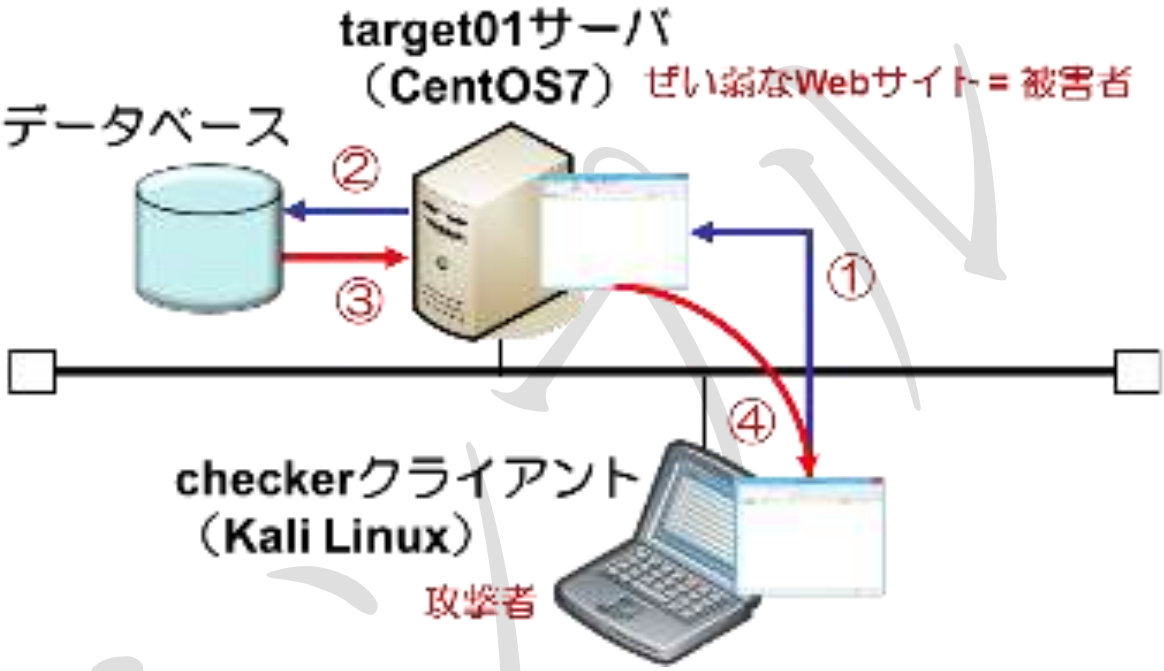
課題 11：SQL インジェクション攻撃を試してみる

（テキスト：第 5 章　5.5 Web アプリケーションへの攻撃　参照）

目的：

どのような方法で SQL インジェクション攻撃が行われるのかを疑似体験してみます。

攻撃パターン例：パワーポイントのスライド（課題 11：SQL インジェクション攻撃を試してみる-1）参照



1. 攻撃者（checker クライアント）は、ぜい弱な Web アプリケーションに対し、SQL コマンドを含んだ不正な入力を行います。
   ※ ぜい弱な Web アプリケーションは、データベースを利用していることが前提です。

2. ぜい弱なサーバ(target01)は、正常なデータベース操作と共に、攻撃者が入力した不正な SQL 文を実行します。
   ※ この場合、データベースがサーバ内にあるのか、他のサーバにあるのかは関係ありません。

3. データベースは正常な返答と共に、不正な SQL 文の結果を戻します。

4. 攻撃者のブラウザに、意図した内容が表示されます。