Born2beRoot

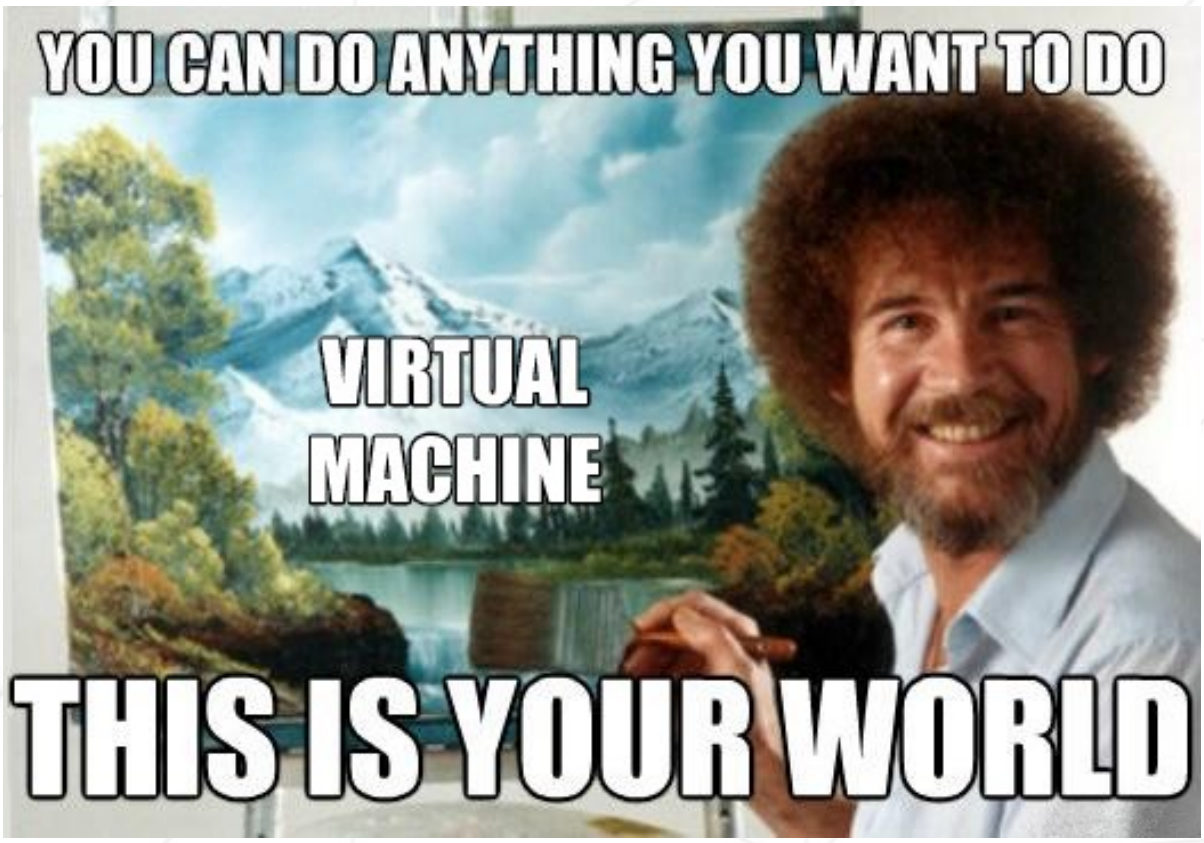
概要：このドキュメントは、システム管理に関する演習です。

バージョン：1

内容

第一章 前

文


YOU CAN DO ANYTHING YOU WANT TO DO
VIRTUAL
MACHINE
THIS IS YOUR WORLD

第二章 はじめに

このプロジェクトは、仮想化という素晴らしい世界を紹介することを目的としています。

VirtualBox（VirtualBoxが使えない場合はUTM）を使って、特定の指示のもと、最初のマシンを作成することになります。そして、このプロジェクトの最後には、厳密なルールを実行しながら、独自のOSをセットアップすることができるようになります。

第III章 一般的なガイドライン

- VirtualBox（VirtualBoxが使えない場合はUTM）の利用が必須となります。
  - リポジトリのルートにsignature.txtを回すだけです。その中に、あなたのマシンの仮想ディスクの署名を貼り付ける必要があります。詳しくは Submission and peer-evaluation に行ってください。

第4章 必須項目

このプロジェクトは、特定のルールに従って最初のサーバーを立ち上げてもらうというものです。

サーバーの立ち上げということで、必要最低限のサービスをインストールすることになります。このため、ここではグラフィカル・インターフェースは役に立ちません。したがって、X.orgやその他の同等のグラフィックサーバーをインストールすることは禁じられています。さもなければ、あなたの成績は0点となります。

OSはDebianの最新安定版（testing/unstableなし）、またはCentOSの最新安定版のいずれかを選択する必要があります。初めてシステム管理を行う場合は、Debianを強くお勧めします。

CentOSのセットアップはかなり複雑です。 したがって、KDumpを設定する必要はありません。ただし、SELinuxはスタートアップで起動している必要があり、その設定はプロジェクトのニーズに合わせて変更する必要があります。 DebianのAppArmorもスタートアップで起動しておく必要があります。

LVMを使用して、少なくとも2つの暗号化パーティションを作成する必要があります。以下は予想されるパーティション分割の例です。

```
wil@wil:~$ lsblk
NAME                  MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
sda                     8:0    0    8G  0 disk
├─sda1                  8:1    0  487M  0 part  /boot
├─sda2                  8:2    0    1K  0 part
└─sda5                  8:5    0  7.5G  0 part
  └─sda5_crypt        254:0    0  7.5G  0 crypt
    ├─wil--vg-root    254:1    0  2.8G  0 lvm   /
    ├─wil--vg-swap_1  254:2    0  976M  0 lvm   [SWAP]
    └─wil--vg-home    254:3    0  3.8G  0 lvm   /home
sr0                    11:0    1 1024M  0 rom
wil@wil:~$ _
```

ディフェンスでは、選んだOSについていくつか質問されます。例えば、aptitudeとaptの違いや、SELinuxやAppArmorが何なのかを知っておくとよいでしょう。要するに、自分が使っているものを理解することです。

SSH サービスはポート 4242 のみで実行されます。セキュリティ上の理由から、rootでSSHを使用して接続することはできないようにする必要があります。

SSHの使用は、新しいアカウントを設定することで防衛中にテストされます。そのため、その仕組みを理解しておく必要があります。

オペレーティングシステムにUFWファイアウォールを設定し、ポート4242のみを開放しておく必要があります。

仮想マシンを起動する際には、ファイアウォールが有効である必要があります。CentOSの場合、デフォルトのファイアウォールの代わりにUFWを使用する必要があります。これをインストールするには、おそらくDNFが必要です。

- 仮想マシンのホスト名は、42で終わるログイン名（例：wil42）である必要があります。評価中にこのホスト名を変更する必要があります。
- 強力なパスワードポリシーを導入する必要があります。
- sudoのインストールと設定は、厳密なルールに従って行う必要があります。
- ルートユーザーの他に、ユーザー名としてあなたのログインを持つユーザーが存在する必要があります。
- このユーザーはuser42とsudoグループに所属している必要があります。

防衛の際には、新しいユーザーを作成し、それをグループに割り当てる必要があります。

強力なパスワードポリシーを設定するには、以下の要件に準拠する必要があります。

- パスワードの有効期限は30日です。
- パスワードの変更までに許容される最短日数は、2日に設定されます。
- パスワードの有効期限が切れる7日前に警告メッセージを表示する必要があります。
- パスワードは10文字以上でなければなりません。また、大文字と数字を含む必要があります。また、同じ文字が3つ以上連続して含まれていてはいけません。

- パスワードにユーザー名を含んではならない。
- rootパスワードは、旧パスワードに含まれない7文字以上でなければなりません。
- もちろん、rootパスワードはこのポリシーに従わなければなりません。

設定ファイルの作成後、rootアカウントを含む、仮想マシン上に存在するすべてのアカウントのパスワードを変更する必要があります。

sudoグループに強力な設定を行うには、以下の要件を満たす必要があります。

- sudoを使用した認証は、パスワードが不正な場合、3回までしか試行できないようにする必要があります。
- sudo使用時にパスワード間違いによるエラーが発生した場合、任意のカスタムメッセージを表示する必要があります。
- sudoを使用する各アクションは、入力と出力の両方をアーカイブする必要があります。ログファイルは /var/log/sudo/ フォルダに保存されなければなりません。
- TTYモードは、セキュリティ上の理由から有効にしておく必要があります。
- セキュリティ上の理由からも、sudoが使用できるパスは制限する必要があります。例 /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin

最後に、monitoring.shという簡単なスクリプトを作成する必要があります。これは、bashで開発する必要があります。

サーバ起動時に、スクリプトは10分ごとにすべてのテリメインにいくつかの情報（以下にリストアップ）を表示します（壁を見てみてください）。バナーは任意である。エラーは表示されないようにしなければならない。

スクリプトは常に以下の情報を表示できるようにする必要があります。

- オペレーティングシステムのアーキテクチャとカーネルバージョン。
- 物理プロセッサの数。
- 仮想プロセッサーの数。
- サーバーで現在利用可能なRAMとその利用率をパーセントで表示します。
- サーバーで現在使用可能なメモリとその使用率をパーセントで表示します。
- プロセッサの現在の使用率をパーセンテージで表示します。
- 最後にリブートした日時。
- LVMがアクティブかどうか。
- アクティブな接続数。
- サーバーを使用しているユーザー数。
- サーバーのIPv4アドレスとそのMAC（Media Access Control）アドレスです。
- sudoプログラムで実行されたコマンドの数です。

答弁では、このスクリプトがどのように機能するかを説明することが求められます。また、改造せずに中断させる必要があります。 cronを覗いてみてください。

これは、スクリプトがどのように動作するかを想定した例である。

```
root@wil (tty1) からのブロードキャストメッセージです (Sun Apr 25 15:45:00 2021)。

        #アーキテクチャ  Linux wil 4.19.0-16-amd64 #1 SMP Debian 4.19.181-1 (2021-03-19) x86_64 GNU/Linux #CPU
        physical : 1
        #vCPU : 1
        #メモリ使用量  74/987MB (7.50%)
        #ディスク使用量  1009/2Gb (39%)
        #CPU負荷。       6.7%
        #最終起動時間 : 2021-04-25 14:45
        #LVMの使用: はい
        #接続TCP : 1 ESTABLISHED #ユーザ
        ーログです。1
        #ネットワーク  IP 10.0.2.15 (08:00:27:51:9b:a5) です。
        #スードー : 42 cmd
```

以下は、対象の要件の一部を確認するために使用できる2つのコマンドです。

CentOSの場合。

```
[root@wil ~]# head -n 2  /etc/os-release
NAME="CentOS Linux"
VERSION="8"
[root@wil ~]# sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:     actual (secure)
Max kernel policy version:      32
[root@wil ~]# ss -tunlp
Netid State  Recv-Q  Send-Q   Local Address:Port   Peer Address:Port
tcp   LISTEN 0       128           0.0.0.0:4242        0.0.0.0:*     users:(("sshd",pid=822,fd=5))
tcp   LISTEN 0       128              [::]:4242           [::]:*     users:(("sshd",pid=822,fd=7))
[root@wil ~]# ufw status
Status: active

To                         Action      From
--                         ------      ----
4242                       ALLOW       Anywhere
4242 (v6)                  ALLOW       Anywhere (v6)

[root@wil ~]# _
```

Debianの場合。

```
root@wil:~# head -n 2 /etc/os-release
PRETTY_NAME="Debian GNU/Linux 10 (buster)"
NAME="Debian GNU/Linux"
root@wil:/home/wil# /usr/sbin/aa-status
apparmor module is loaded.
root@wil:/home/wil# ss -tunlp
Netid State  Recv-Q Send-Q Local Address:Port  Peer Address:Port

tcp   LISTEN 0      128          0.0.0.0:4242       0.0.0.0:*     users:(("sshd",pid=523,fd=3))
tcp   LISTEN 0      128             [::]:4242          [::]:*     users:(("sshd",pid=523,fd=4))
root@wil:/home/wil# /usr/sbin/ufw status
Status: active

To                         Action      From
--                         ------      ----
4242                       ALLOW       Anywhere
4242 (v6)                  ALLOW       Anywhere (v6)
```

第五章 ボーナスパート

ボーナスリスト

- パーティションを正しく設定し、以下のような構造になるようにします。

```
# lsblk
NAME                    MAJ:MIN RM   SIZE RO TYPE  MOUNTPOINT
sda                       8:0    0  30.8G  0 disk
├─sda1                    8:1    0   500M  0 part  /boot
├─sda2                    8:2    0     1K  0 part
└─sda5                    8:5    0  30.3G  0 part
  └─sda5_crypt          254:0    0  30.3G  0 crypt
    ├─LVMGroup-root     254:1    0    10G  0 lvm   /
    ├─LVMGroup-swap     254:2    0   2.3G  0 lvm   [SWAP]
    ├─LVMGroup-home     254:3    0     5G  0 lvm   /home
    ├─LVMGroup-var      254:4    0     3G  0 lvm   /var
    ├─LVMGroup-srv      254:5    0     3G  0 lvm   /srv
    ├─LVMGroup-tmp      254:6    0     3G  0 lvm   /tmp
    └─LVMGroup-var--log 254:7    0     4G  0 lvm   /var/log
sr0                      11:0    1  1024M  0 rom
```

- Lighttpd、Mari- aDB、PHPの各サービスを利用して、機能的なWordPressサイトを構築する。
- あなたが便利だと思う好きなサービスを立ち上げてください（NGINX / Apache2 は除く！）。ディフェンスでは、あなたが選んだサービスを正当化する必要があります。

ボーナス・パートでは、追加サービスを設定することが可能です。　この場合、あなたのニーズに合わせてより多くのポートを開くことができます。もちろん、UFWのルールはそれに応じて適応させなければなりません。

ボーナスパーツは、必須パーツがPERFECTである場合にのみ査定されます。パーフェクトとは、必須パートが統合的に行われ、誤動作することなく動作することを意味します。必須条件をすべてクリアしていない場合、ボーナスパーツの評価は一切行われません。

第六章

提出と相互評価

Gitリポジトリのルートにsignature.txtを回すだけです。その中に、あなたのマシンの仮想ディスクの署名を貼り付けなければなりません。この署名を得るには、まずデフォルトのインストールフォルダを開く必要があります（あなたのVMが保存されているフォルダです）。

- Windows: %HOMEDRIVE%%HOMEPATH%VirtualBox VMs
- Linux: ~/VirtualBox VMs/ (英語)
- MacM1: ~/Library/Containers/com.utmapp.UTM/Data/Documents/
- MacOS: ~/VirtualBox VMs/ (英語)

次に、仮想マシンの「.vdi」ファイル（UTMの場合は「.qcow2」）から署名をsha1フォーマットで取得します。以下は、centos_serv.vdiファイルに対する4つのコマンド例です。

- Windows：certUtil -hashfile centos_serv.vdi sha1
- Linux：sha1sum centos_serv.vdi
- Mac M1の場合：shasum Centos.utm/Images/disk-0.qcow2
- MacOS：shasum centos_serv.vdi

どのような出力が得られるか、その一例をご紹介します。

- 6e657c4619944be17df3c31faa030c25e43e40af

初回評価後は、仮想マシンの署名が変更される場合がありますので、ご注意ください。この問題を解決するには、仮想マシンを複製するか、状態を保存を使用することができます。

もちろん、Gitリポジトリに仮想マシンを投入することは禁じ手です。ディフェンスでは、signature.txtの署名とあなたの仮想マシンの署名が比較されます。　もし両者が同一でない場合、あなたの成績は0点となります。