10章 セキュリティ

10.1 セキュリティ管理業務

10.1.1 SUID/SGIDが有効になっているファイルの検索

- SUIDやSGIDビットが不要な実行ファイルに設定されているのはセキュリティリスク
    - どのユーザで実行しても、あるユーザの権限で実行されてしまう
- 検索方法は「find」で可能

```
# SUIDが有効になっているファイルを検索し、出力
find /usr -perm -4000 -ls

# SUIDを無効化する
chmod u-s /bin/touch
```

10.1.2 shadowファイルの特徴とユーザー/パスワードの有効期限

- パスワード情報は「/etc/shadow」ファイルに暗号化されて格納されている
- 権限は000で、rootしか参照できない
- 一般ユーザはpasswdでパスワード変更ができる
    - **passwdコマンドの実行ファイルにSUIDが有効になっているから**
- パスワードの有効期限が切れれば、ユーザはログインできなくなる

```
usermod [オプション] ユーザ名

    -e 年-月-日      : アカウントの有効期限を設定
    -L               : アカウントをロック(無効化)
    -U               : アカウントのロックを解除
```

```
chage [オプション] ユーザ名

    -l           : 有効期限情報を表示
    -m 日数      : パスワードを変更できる最短日数を設定
    -M 日数      : パスワードの有効期限（使用できる最大日数）を設定
    -I 日数      : パスワードの有効期限が切れてからアカウントを無効化するまでの日数を設定
    -W 日数      : パスワードの有効期限切れの警告が何日前から表示されるかを設定
    -E 年-月-日 : アカウントの有効期限を設定(usermod -eと同じ)

    オプションなしなら対話形式で設定可能
```

```
passwd オプション ユーザ名

    -x 日数      : パスワードの有効期限を設定(chage -Mと同じ)
    -l           : アカウントをロック(無効化) (usermod -Lと同じ)
    -u           : アカウントのロックを解除(usermod -U)
    -e           : パスワードを有効期限切れにする
```

- /etc/passwdのパスワード部分「x」を「!!」にすると未設定＝ログイン不可
- ログインシェルを以下にすると対話式ログイン不可
  - /bin/false
  - /sbin/nologin
  - touch /etc/nologin で一般ユーザはすべてログイン不可

10.1.3 ユーザ環境の切り替え

```
```bash
su [オプション] [ユーザ名]

    -        : カレントディレクトリや設定されている変数など、ユーザ環境を切り替える
```
```bash
sudo [オプション] コマンド

    -u ユーザ名       : 指定したユーザ権限でコマンドを実行(指定しなかった場合はrootユーザ権限でコマンドを実行)
```
```

- sudoコマンドを利用するには、/etc/sudoersファイルにsudoコマンドを実行できるユーザと実行できる操作を設定しておく必要がある
- sodoersファイルの編集は、**visudo**コマンド

```
ユーザ 許可するホスト(=対象ユーザ) 実行可能なコマンド

%wheel  ALL=(ALL)       ALL
```

10.1.4 ファイルやポートを開いているプロセスの確認

```
lsof [オプション] [ファイル]

    -i :ポート番号       :   指定したポートを開いているプロセスを表示
```

```
fuser [オプション] [ファイル]

    -v                  : 詳細な情報を表示
```

- unmountしたい時などに調べられる
- netstatやssでも可能
- 指定したホストが待機しているポートを確認

```
nmap [オプション] ホスト
```

10.1.5 使用するリソースの制限

```
ulimit オプション

    -a                      : 対象となるリソースと制限値の一覧を表示
    -c ブロックサイズ       : コアダンプファイルのサイズを制限
    -n ファイル数           : 同時に開くことができるファイル数を制限
```

10.1.6 ユーザのログイン情報の調査

- who ,w
    - /etc/run/utmpを参照

```
# 現在ログイン中のユーザと端末の情報を表示
who [オプション]

# 現在ログイン中のユーザと端末、実行しているプロセスの表示
w [オプション]
```

- last
    - /var/log/wtmpを参照

```
# ホスト上でのログイン履歴
last [オプション]
```

自動ログアウト

```
export TMOUT=60  #秒
```

10.2 ホストのセキュリティ設定

10.2.1 不要なネットワークサービスの停止

- init時代のサービスの止め方

```
# 自動起動の無効化
chkconfig httpd off

# HTTPDの停止
service httpd stop

# httpdの状態を確認
service httpd status
```

```
# 自動起動設定
chkconfig [サービス名] [on/off]

# サービスの操作
service サービス名 操作

    start
    stop
    status
```

10.2.2 スーパーユーザの利用

- スタンドアローンサーバー
- スーパーサーバー
    - inetd、xinetd
        - あるポートにユーザが接続してきたら対応するアプリを起動する
        - 起動時間がかかる
    - inetd の設定ファイルは「/etc/initd.conf」

      ```
      ftp  stream  tcp  nowait  root  /usr/libexec/ftpd  ftpd -l

      (1)サービス (2)ソケットタイプ (3)プロトコル (4)ウエイト (5)ユーザー名 (6)プログラム（デーモン）のパス (7)プログラム（デーモン）名と引数
      ```

    - xinetd の設定ファイルは「/etc/xinetd.conf」「/etc/xinetd.d/」
        - xinetdの設定ファイル内の項目
            - disable : 接続の有効化、無効化
            - only_from : 接続を許可するホスト/ネットワーク
            - no_access : 接続を拒否するホスト/ネットワーク

10.2.3 firewalldによるパケットフィルタリング

- Netfilter
    - firewalld
    - iptables
- firewalldを利用したNetfilterの制御

```
firewall-cmd --list-all
```

- firewall-cmdを利用して、firewallの制御が可能

```
# オプション

--get-default-zone                    : デフォルトゾーンの表示
--set-default-zone=ゾーン             : デフォルトゾーンの設定
--zone=ゾーン                         : 制御対象とするゾーンの指定
--add-service=サービス                : 指定したサービスを許可対象に追加
--remove-service=サービス             : 指定したサービスを許可対象から削除
--add-interface=インターフェイス      : インターフェイスをゾーンに追加
--remove-interface=インターフェイス   : インターフェイスをゾーンから削除
--list-all-zones                      : すべてのゾーンの情報を表示
--list-all                            : ゾーンの情報をすべて表示
--list-services                       : 許可対象となっているサービスを表示
--list-interfaces                     : ゾーンに含まれるインターフェイスを表示
--permanent                           : 永続的な設定にする(設定ファイルに書き込む)
```

```
    public   : パブリックエリア（デフォルト）
    dmz      : DMZネットワーク用
    trusted  : すべての接続を許可
    drop     : すべての接続を拒否
```

- Netfilterの制御

```
iptables オプション

    -L [チェイン]             ：ルールの表示
    -A [チェイン] ルール      : ルールを末尾に追加
    -D [チェイン] ルール      : ルールの削除
    -P [チェイン] ルール      : チェインに対して、ポリシー設定を行う
```

```
    ACCEPT   : パケットを許可
    DROP     : パケットを破棄
```

10.3 暗号化によるデータの保護

10.3.1 暗号化技術の概要

- 暗号化技術
  - 共通鍵暗号方式
  - 公開鍵暗号方式
    - 公開鍵
    - 秘密鍵

共通鍵暗号方式

- アルゴリズム
  - DES
  - AES(上位)
    - 無線LANなど

公開鍵暗号方式

- RSA

10.3.2 OpenSSH

- １。ホスト認証
- ２。ユーザ認証
  - 公開鍵
  - パスワード認証

```
ssh [オプション] [ユーザ@]ホスト [コマンド]

    -L           : ポート転送を行う
    -A           : ssh-agentによる鍵転送を行う
    -l ユーザ    : ユーザ名を指定
    -i           : 秘密鍵ファイルを指定
    -X           : Xクライアントの表示用にDISPLAY環境変数が自動で設定される
```

```
ssh-keygen [オプション]

    -t アルゴリズム      : 公開鍵ペアを生成する際に利用するアルゴリズムを指定
    -b ビット数          : 生成する鍵のビット数を指定
    -a ラウンド数        : ed25519で鍵を生成する際のKDFラウンド数を指定
    -C "コメント"        : 鍵データに付属するコメントを指定
```

鍵の種類	説明	ビット数
rsa	大きな数の素因数分解が困難さを利用	1024,2048(default),4096
dsa	1993年、離散対数問題の困難さを利用	1024
ecdsa	2009年、楕円曲線状の離散対数問題の困難さを利用	256,384,521
ed25519	楕円曲線Curve25519上の離散対数問題の困難さを利用	256

```
# SSHによりリモートホスト上に、もしくはリモートホスト上からファイルをコピー
scp [オプション] コピー元 コピー先

    -i  : 秘密鍵ファイルを指定

scp .ssh/*.pub 192.168.56.11:~
```

```
# sshコマンドを利用するとき、以下を参照
cat /etc/ssh/ssh_config

cat ~/.ssh/config
```

- ssh-agent
  - SSH用の認証エージェントを起動し、秘密鍵情報を保持
- ssh-add
  - ssh-agentに秘密鍵を追加

```
# 上記でパスフレーズの入力を省略できる
#「ssh-agent」はメモリ上に秘密鍵を保管しておく認証エージェント
ssh-agent bash
ssh-add .ssh/id_ecdsa

scp -i .ssh/id_ecdsa txt 192.168.56.11:~
```

10.3.4 SSHによるポート転送

```
ssh -L 2323:192.168.56.11:23 192.168.56.11
```

10.3.5 GnuPGによるデータの暗号化と復号

```
# GnuPGによる公開鍵ペアの管理、暗号化の実装
gpg [オプション]
```

```
    --gen-key                    : 鍵ペアの作成
    --list-keys                  : キーリングの内容を表示
    -a                           : インポート、エクスポートの際、ASCII情報で扱う
    -o ファイル名                : 出力先を指定
    --export メールアドレス      : 指定したアドレスの鍵をエクスポート
    -- import ファイル名         : 指定したファイルに含まれる鍵をインポート
    -- sign-key メールアドレス   : 鍵に署名を行う
    -e                           : 指定したファイルを暗号化
    -r メールアドレス            : 暗号化の際に使用する鍵のアドレスを指定
    -b                           : 署名ファイルを生成
    --verify                     : 署名ファイルを検証
```

```
# gpg-agent ・GnuGPで利用する秘密鍵を管理

gpg-agent [オプション]

    --daemon                     : デーモンモード（バックグラウンド）で動作
    --use-standard-socket        : ソケットファイルを標準の場所に配置
```

10.3.6 GnuPGによるデジタル署名

~~あとで

10.4 クラウドセキュリティの基礎

10.4.1 クラウドサービス

10.4.2 オンプレミス環境とパブリッククラウド環境

10.4.3 パブリッククラウド環境における認証構成

- 多要素認証
- NAT

10.4.4 ストレージの利用

- スナップショット

10.4.5 クラウドサービスにおける障害時の対応

- リージョン

v10.4.6 脆弱性検査について

- ペネトレーションテスト
- OW ASP ZAP というツール