ネットワークセキュリティ演習

9回 Webセキュリティ2

演習レポートのURL

https://goo.gl/forms/eTBuYwNBlttnzjcb2

必要ソフトのインストール

sqlite3

```Bash
sudo apt install gcc
sudo apt install ruby-dev
sudo apt install libxml2
sudo apt install libxml2-dev
sudo apt install zlib1g-dev
sudo apt install nodejs
sudo apt install sqlite3
sudo apt install libsqlite3-dev
```

rubygems sqlite3

```
sudo gem install sqlite3
```

サーバ側でセッションIDを生成してクッキーに入れる方法

セッション開始ページ

```Bash
sudo nano /var/www/html/session.html
```

```Markup
<meta http-equiv="content-type" charset="utf-8">
<html>
 <head></head>
 <body>
 <h1>山崎サーバ</h1>
  <h2>セッション</h2>
 <form method="GET" action="/cgi-bin/session.cgi">
 <p><input type="submit" value="セッション開始"></p>
 </form>
 </body>
</html>
```

サーバ側でセッションIDを生成

```Bash
sudo nano /usr/lib/cgi-bin/session.cgi
```

安全な乱数でセッションIDを生成する

```Ruby
!/usr/bin/env ruby
# coding: utf-8
require 'cgi'
require 'securerandom'
cgi=CGI.new

puts "Content-Type: text/html;"
puts "Set-Cookie: sessionid=#{SecureRandom.hex(32)}; path=/"
puts "\n\n"
  print <<-EOM
  <meta http-equiv="content-type" charset="utf-8">
  <html><head>
    <script>
      function info(){alert(document.cookie);}
    </script>
  </head><body>
   <h1>山崎サーバ</h1>
  <h2>セッション確立</h2>
<form method='GET' action='/cgi-bin/service.cgi'>
  <input type='button' value="cookie" onclick="info();">
  <input type='submit' value="サービス">
</form>
</body></html>
EOM
```

サービス継続

```Bash
sudo nano /usr/lib/cgi-bin/service.cgi
```

```Ruby
#!/usr/bin/env ruby
# coding: utf-8
require 'cgi'
cgi=CGI.new
cookies=ENV['HTTP_COOKIE'].split(/;\s/).map{|x|x.split('=')}.to_h

puts "Content-Type: text/html;"
puts "Set-Cookie: sessionid=#{cookies["sessionid"]}; path=/"
puts "\n\n"
  print <<-EOM
  <meta http-equiv="content-type" charset="utf-8">
  <html><head>
    <script>
      function info(){alert(document.cookie);}
    </script>
  </head><body>
   <h1>山崎サーバ</h1>
  <h2>セション継続中</h2>
<p>sessionid=#{cookies["sessionid"]}</p>
<form method='GET' action='/cgi-bin/service.cgi'>
<input type='button' value="cookie" onclick="info();">
<input type='submit' value="サービス継続">
</form>
  </body></html>
EOM
```

リレーショナルデータベースとSQLの基本

cgiのディレクトリ　/usr/lib/cgi-bin/　にDBを作成する

webサーバからの書き込みの権限を与える

```Bash
cd  /usr/lib/cgi-bin/
sudo mkdir db
sudo chown www-data db
sudo chmod 777 db
```

DBのファイル名 user.db

```Bash
sqlite3 db/user.db
```

DBテーブル作成

```SQL
sqlite> create table user(user_id TEXT, password TEXT);
```

データ登録

```SQL
sqlite> insert into user values('terai','abcabc');
sqlite> insert into user values('takahashi','123123');
sqlite> insert into user values('yamasaki','efgefg');
```

sqlite3の終了

```SQL
sqlite> .q
```

DBファイルのパーミッションを変更

httpdのプロセス権限で user.db への更新を可能にする

```Bash
chmod 666 db/user.db
```

sqlite3 の再起動

```SQL
sqlite3 db/user.db
```

データの参照

```SQL
# 条件指定で1件の1カラムだけ

sqlite> select password from user where user_id='takahashi';
123123

### 条件指定で全カラム

sqlite> select * from user where user_id='takahashi';
takahashi|123123

### テーブル内全部

sqlite> select * from user;
terai|abcabc
takahashi|123123
yamasaki|efgefg
```

```
.q
```

ruby からsqlite3を利用

```Bash
irb
```

```Ruby
>> require 'sqlite3'
>> db=SQLite3::Database.new("./db/user.db")

>> db.execute("insert into user values('fujio','98765');")

>> db.execute("select * from user;")
=> [
 ["terai", "abcabc"],
 ["takahashi", "123123"],
 ["yamasaki", "efgefg"],
 ["fujio", "98765"]]
```

ユーザ登録

ユーザ登録ページ

```Bash
sudo nano  /var/www/html/signup.html
```

```Markup
<meta http-equiv="content-type" charset="utf-8">
<html>
 <head></head>
 <body>
 <h1>山崎サーバ</h1>
 <h2>ユーザ登録</h2>
 <form method="GET" action="/cgi-bin/signup.cgi">
 <p>ユーザID</p>
 <p><input type="text" name="user_id"></p>
 <p>パスワード</p>
 <p><input type="password" name="password"></p>
 <p><input type="submit" value="登録"></p>
 </form>
 </body>
</html>
```

ユーザ登録CGI

```Bash
sudo nano /usr/lib/cgi-bin/signup.cgi
```

192.168.1.XX のログイン画面に遷移

```ruby
#!/usr/bin/env ruby
# coding: utf-8
require 'cgi'
require 'sqlite3'
cgi=CGI.new
db=SQLite3::Database.new("./db/user.db")
sql="insert into user values('#{cgi['user_id']}','#{cgi['password']}');"
db.execute(sql)

print "Content-Type: text/html\n\n"
print <<-EOM
<meta http-equiv="content-type" charset="utf-8">
<html>
 <head></head>
  <body>
<h1>山崎サーバ</h1>
  <h2>ユーザ登録しました</h2>
  <p>'#{cgi['user_id']}'</p>
  <p>'#{cgi['password']}'</p>
  <h2><a href="http://192.168.0.17/login.html">
ログインページ</a></h2>
</body>
</html>
EOM

```

クッキーを使ったログイン処理

login ページ

```bash
sudo nano  /var/www/html/login.html
```

パスワードをあえてテキストとして表示している

```Markup
<meta http-equiv="content-type" charset="utf-8">
<html>
 <head>
 </head>
 <body>
   <h1>山崎サーバ</h1>
   <h2>ログイン</h2>
    <form method="GET" action="/cgi-bin/login.cgi">
      <h2>ユーザID</h2>
        <input type="text" name="user_id">
      <h2>パスワード</h2>
        <input type="text" name="password">
        <p><input type="submit" value="login"></p>
      </form>
 </body>
</html>
```

login CGI

ユーザIDとパスワードをチェック

パスワードが合致していればセッションIDを生成する

```Bash
sudo nano /usr/lib/cgi-bin/login.cgi
```

Ruby

```ruby
#!/usr/bin/env ruby
# coding: utf-8
require 'cgi'
require 'sqlite3'
require 'securerandom'
cgi=CGI.new
db=SQLite3::Database.new("./db/user.db")
sql="select * from user where user_id='#{cgi['user_id']}' and password='#{cgi['
user=db.execute(sql)
def valid_user(user)
  (user!=[] ? true : false)
end
if valid_user(user) then
  session_id=SecureRandom.hex(32)
  puts "Content-Type: text/html;"
    puts "Set-Cookie: sessionid=#{session_id}; path=/"
  puts "\n\n"
  print <<-EOM
    <meta http-equiv="content-type" charset="utf-8"><html><head>
<h1>山崎サーバ</h1><h2>ログイン成功</h2><p>#{session_id}</p>
<form method='GET' action='/cgi-bin/service.cgi'>
      <input type='submit' value="サービス利用">
      </form>
</body></html>
EOM
else
  puts "Content-Type: text/html; \n\n"
    print <<-EOM
<meta http-equiv="content-type" charset="utf-8">
<html><head>
<h1>山崎サーバ</h1><h2>ログイン失敗</h2>
<p>#{cgi['user_id']}</p>
</body></html>
EOM
end
```

クッキーを使ったログイン処理への攻撃

iframeでログインサイトをつくる

Bash

```bash
sudo nano /var/www/html/wana.html
```

Markup

```
<meta http-equiv="content-type" charset="utf-8">
<html>
  <head>
    <script>
      function info(){alert(document.cookie);}
    </script>
  </head>
  <body>
    <iframe id="attack"
      width="300"  //iframeの幅（ピクセル）
      height="400" //iframeの高さ（ピクセル）
      src="http://192.168.0.17/login.html">
    </iframe>
    <form method='GET' action='/cgi-bin/service.cgi'>
      <input type='button' value="cookie" onclick="info();">
    </form>
  </body>
</html>
```

SQLインジェクションの例

パスワードに次の文字列を入れてみる

```
' OR 't'='t
```