SQL Injection Attack

Anant Bhardwaj
anantb@cs.stanford.edu

Goal of the project

- You have already been given a rails project "netslip"
- It contains a loophole
- You need to
  - identify the loophole
  - exploit it to get some secret data
  - fix the loophole

Things that you need to know

- Using Sockets
- SQL Injection Attack
- Safeguards against SQL Injection Attacks

Network Communication

- What uniquely identifies a machine on a computer network?
- Is IP address alone enough for the communication between two endpoints?
- Why do we need ports?
- Did you ever think why do you get an error if you try to start another rails server if there is already one running?

Socket

- Definition:
    - is one endpoint of a two way communication-link between two programs running on a network.
    - is bound to a port number so that data can be sent to a particular application running on the endpoint.
    - In short: an endpoint is a combination of an IP address and a port number which an application uses for data transfer.

Understanding HTTP request and Response

- start rails-server

```
GET -U -s -e "http://localhost:3000/calc/add?
num1=5&num2=9"
```

HTTP GET request

```
anantb@anantb-lnx:~$ telnet localhost 3000
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
GET /calc/add?num1=7&num2=5 HTTP/1.0
```

HTTP POST request

```
anantb@anantb-lnx:~$ telnet localhost 3000
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
POST /calc/add HTTP/1.0
Accept: text/xml,application/xml,application/xhtml+xml,text/html*/*
Accept-Language: en-us
Accept-Charset: iso-8859-1,*,utf-8
Content-type: application/x-www-form-urlencoded
Content-length: 13

num1=7&num2=5
```

Understanding HTTP Response Header

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
X-Ua-Compatible: IE=Edge
Etag: "f0e83ab47de5964c16dbcf267f7df3d0"
Cache-Control: max-age=0, private, must-revalidate
X-Request-Id: 3d7ab084f09dd127bcd0bdbf87994aea
X-Runtime: 0.205363
Content-Length: 991
Server: WEBrick/1.3.1 (Ruby/1.9.2/2012-04-20)
Date: Fri, 01 Jun 2012 04:08:42 GMT
Connection: close
Set-Cookie: _session_id=83d5217dee59c3a4248f6352ffe0e677; path=/;
HttpOnly
```

Ruby Socket

- useful module: socket, CGI
- see socket.rb

SQL Injection Attack

- What is the vulnerability?
- How you can exploit it by injecting SQL code in your form data?
- Let's see an example.

Questions