HTTP Basics
HTTP – HyperText Transfer Protocol

- HTTP – The protocol behind the web (WWW)
- Versions: 0.9, 1.0, 1.1
- RFCs: 1945, 2068, 2616
- By understanding how HTTP works, you’ll be able to:
  - Understand the interaction between web clients and web servers
  - Manually query web servers and receive low-level information that typical web browsers hide from the user
  - Understand how AppScan and ASE perform security tests
HTTP Basics

- Let’s take a look at the user’s request from the browser: http://www.site.com:80/
  - http:// - use the HTTP protocol
  - www.site.com - hostname
  - :80 – connect to the remote computer at port 80
  - / - anything after the hostname and optional port number is regarded as a document path.

- Let’s take a look at the actual message that the browser sends to the server:
HTTP Request and Response

Request:

GET / HTTP/1.1
Accept: application/vnd.ms-excel, application/msword, application/vnd.ms-powerpoint, image/gif, image/x-xbitmap, image/jpeg, image/pjpeg, */*
Accept-Language: he
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)
Host: www.site.com
Connection: Keep-Alive

Here is the server’s response:

HTTP/1.1 200 OK
Server: Microsoft-IIS/5.0
Date: Fri, 29 Aug 1997 02:14:00 GMT
Set-Cookie: SessionId=raqngbihvwpypof45rujbce45; path=/
Content-Type: text/html;
Content-Length: 39

<html>
  <h1>HELLO WORLD</h1>
</html>
HTTP Methods

- **HTTP Methods:**
  - GET - retrieve a document
  - HEAD - retrieve header information
  - POST - Send data to the server
  - PUT, DELETE - Uploads a representation of the specified resource; deletes a resource

- Note (i): There are more methods, but we won't talk about them now.

- Note (ii): It is possible to send data to a web application (such as CGIs and ASPs) using GET, the data is appended to the path (After the '?') and is called the QUERY.

- URL Encoding: data sent to web application should be encoded in a special format. Since it can be appended to the URL itself, it cannot contain special characters such as space, newlines, '&', '=', etc. The format is %HH where HH is a hexadecimal representation of the character needed.
HTTP Response Codes

- **Content-Type header:** text/html, text/plain, application/octet-stream, application/x-trash, application/x-www-urlencoded

- **Server response codes:**
  - 1XX – Informational. The client SHOULD continue with its request
  - 2XX – Successful. Client's request was successfully received, understood, and accepted
  - 3XX – Redirection. Further action needs to be taken by the user agent in order to fulfill the request
  - 4XX – Client Error
  - 5XX – Server Error. The server is aware that it has erred or is incapable of performing the request
HTML Forms Using the GET Method

- Let's take a look at an HTML form (Method=GET):

```html
<form method=GET action="/cgi-bin/script.asp">
<input type=text name="username" value=""/>
<input type=password name="password" value=""/>
<input type=hidden name="sessionId" value="12ouh349d9242uh"/>
<input type=submit name=submit value="click here"/>
</form>
```

- When the submit button is pressed, the browser will send the following request:

```
GET /cgi-bin/script.asp?username=myName&password=myPassword&sessionId=12ouh349d9242uh&submit=click+here HTTP/1.1
```
Now, let’s take a look at the same HTML form (Method=POST):

POST /cgi-bin/script.asp HTTP/1.1
Accept: application/vnd.ms-excel...
Accept-Language: en
Content-Type: application/x-www-form-urlencoded
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)
Host: www.site.com
Content-Length: 79
Connection: Keep-Alive
Cache-Control: no-cache

username=myName&password=myPassword&sessionId=12ouh349d9242uh&submit=click+here
Redirects

- Redirections (HTTP):
  - Redirection occurs when the server sends the following response:
    - HTTP/1.1 302 Found
    - Server: Microsoft-IIS/5.0
    - Date: Thu, 26 Oct 2006 16:26:39 GMT
    - Location: /path/to/file.asp