Webtillämpningar

Webtillämpningar

Webben brukade vara så mycket enklare:

Mest statiska HTML-sidor, mest en fråga om öppen information, indata genom enkla formulär

Idag är det mycket annorlunda:

- Dynamiska websidor
- Webtillämpningar
- Accesskontroll

HTTP: Grunden för websidor

Hyper-Text Transfer protocol (HTTP) är ett protokoll i tillämpningslagret som levererar data till websidor

Uppfanns i CERN 1989

Klienten skickar POST- och GET-anrop

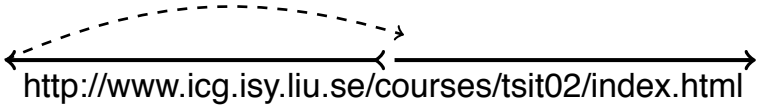
Websidan tolkar anropet, extraherar parametrar från av användaren

Servern svarar med status och sidan i HyperText Markup Language (HTML)

Uniform Resource Locator/Identifierare/Namn

URL-delen (adressen) skickas genom DNS-uppslagning

URN (filnamnet) används för att identifiera filen (eller dataelementet) som önskas av klienten



POST och GET

Det finns två typer av begäran, POST och GET

GET lägger begäran synligt i URL-raden (mindre säkert, loggas och är lätt att manipulera)

POST lägger begäran i requestanropet, syns inte i apropsraden men kan avlyssnas och manipuleras

För bättre säkerhet bör SSL (https) användas så din begäran inte kan avlyssnas

Klientens GET-anrop

Enkelt exempel: Be om en HTML-sida:

Du skriver: http://www.example.com/index.html

Detta delas upp till:

GET /index.html HTTP/1.1
Host: www.example.com

Serverns svar

```
GET /index.html HTTP/1.1
Host: www.example.com
HTTP/1.1200OK
Date:Mon,9Nov201622:38:34GMT
Content-Type:text/html;charset=UTF-8
Content-Encoding:UTF-8
Content-Length:138
Last-Modified:Wed,08Jan201323:11:55GMT
Server:Apache/1.3.3.7(Unix)(Red-Hat/Linux)
Accept-Ranges:bytes
Connection:close

<html>
<head>
    <title>An Example Page</title>
</head>
<body>
    Hello World, this is a very simple HTML document.
</body>
</html>
```

Klientens GET-anrop med PHP

PHP-exempel (från W3schools): Be om data från ett PHP-script

```
<?php
echo "Study " . $_GET['subject'] . " at " . $_GET['web'];
?>
```

hämtar data och presenterar det som:

Study PHP at W3schools.com

men detta gör att anropet syns:

`https://tryphp.w3schools.com/showphp.php?filename=demo_global_get`

POST-anrop

Exempel: Mitt *quiz*-system.

Frågenumret måste hela tiden skickas vidare till nästa steg.
För detta använder jag POST:

```
echo '<form action = IngisQuizForm.php method="post">';
echo '<input type="hidden" name="questionNumber" value="' .
                                    $questionNumber . '">';
echo '<input type="submit" value="Continue" class="button">';
echo "</form>";
```

och hämtar ut det på nästa sida:

```
$questionNumber = $_POST['questionNumber'];
```

Detta syns *inte* i webadressraden! Därmed är det svårare att avlyssna.

HTML, hypertext markup language

HTML är ett markup language (textbaserat filformat) som beskriver websidor

Element inkluderar formulär, ramar (frames), iframes, bilder, applets och scripts

Kombineras ofta med JavaScript och/eller PHP för att skapa dynamiska websidor

Aktiveras ofta med musklick på en knapp som aktiverar ett GET- eller POST-anrop

Andra händelser är mouseup, onmousehover mm

Cookies

Cookies, "kakor", är små mängder data som sätts av serversvar med headerfältet "Set-Cookie"

Innehåller nyckel+värde-par, domän, utgångsdatum, eventuell sökväg, samt flaggorna "secure" och "HTTP only"

"secure" påtvingar HTTPS-överföring

"HTTP only" förbjuder script-access från klienten

Sessions och cookies

HTTP saknar tillstånd! All information måste överföras från sida till sida eller sparas någonstans!

Sessions: Information (som identitet) sparas i servern (begränsad tid)

Cookies: Information (identitet) sparas i klienten

När en session skapas skickar servern ett ID till klienten som sparas som en cookie

Autenticering av sessions är ett separat problem som kan utföras i olika lager av nätverkssystemet.

Angrepp via webläsare

Databasangrepp

Cookiestöld

Scriptangrepp

Mer om detta nästa föreläsning!