MODULNUMMER: 133

Cedric Wieser

IFZ 826-003

1 INHALTSVERZEICHNIS

2 BLOCK 1

PHP → HYPERTEXT PREPROCESSOR

ANWENDUNG

Wordpress, E-Shop, Typo

PHP ist eine serverseitige Skriptsprache, mit der z.B. von der Webapp auf Datenbanken zugegriffen werden kann.

SICHERHEITSAUDITS

Massnahmen zur Risiko- und Schwachstellenanalyse.

SOCIAL ENGENEERING

Person, die jemande beeinflusst, ein Link anzuklicken oder Passwort einzugeben

DDOS ANGRIFFE (DISTRIBUTED DENIAL OF SERVICE)

Gleichzeitige Anfrage auf einen Server durch viele Rechner. → Wieviele Anfragen mach ein System verkrafen, bis es in die Knie geht?

THREAT MODELING

Modell der Bedrohung für ein Netz.

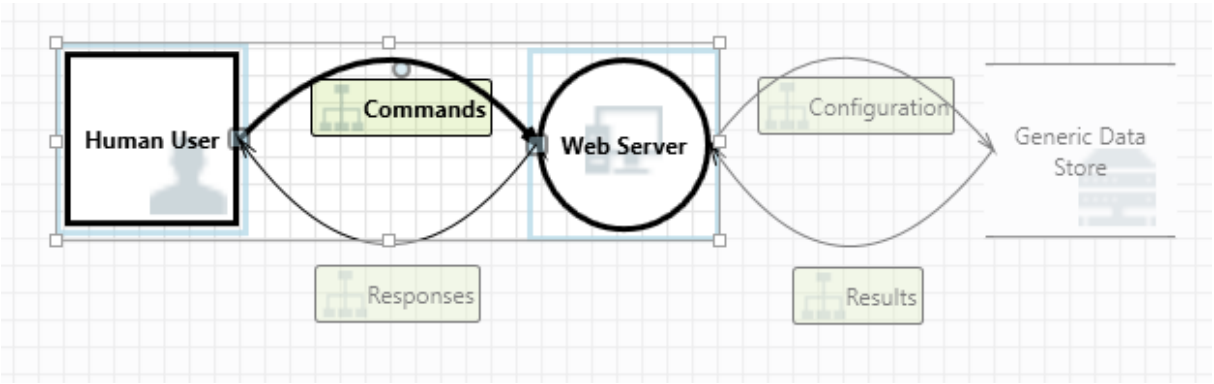

Abbildung 1: Microsoft Threat Modelling

Quelle: https://docs.microsoft.com/en-us/azure/security/develop/media/threat-modeling-tool-getting-started/interaction.png

C&C SERVER

Wenn mein Computer zu einem BotComputer gemacht wurde (wahrscheinlich durch von mir gedownloadete Schadsoftware), greift mein PC auf einen C&C Server zu, auf der der Hacker auch zugreift. Auf diesem wird sich untereinander unterhalten. Der Bad Guy übergibt Kommandos, wie z. B. greift diesen und diesen PC an.

ESCAPING

Formvalidierung: PLZ → nur Zahlen von 0-9, max. 4 (CH)

Vorname → nur Buchstaben von a - z

OWASP (OPEN WEB APPLICATION SECURITY PROJECT)

Das OWASP ist eine Organisation, die das Ziel hat, Anwendungen im www sicherer zu machen. Das wichtigste Dokument, was die Organisation zur Verfügung stellt, ist das OWASP Top 10. Darin sind die jährlich häufigsten Bedrohungen für Webanwendungen und Dienste veröffentlicht. Die 10 häufigsten Bedrohungen für das Jahr 2019 waren:

REFLECTED XSS (CROSS SITE SCRIPTING)

Ein Webseiten Benutzer wird umgeleitet, um z.B. sein Benutzername & Passwort einzugeben, wobei die Daten dann in dritte Hände gegeben werden. Beim reflected XSS wird solcher Schadcode einmalig (z.B. über falsche URL) eingeschleust. Beim neu laden der Seite (mit korrekter URL) ist der Schadcode weg.

"Das nicht-persistente (non-persistent) oder reflektierte (reflected) Cross-Site-Scripting ist ein Angriff, bei dem eine Benutzereingabe direkt vom Server wieder zurückgesendet wird. Enthält diese Eingabe Skriptcode, der vom Browser des Benutzers anschließend interpretiert wird, kann dort Schadcode ausgeführt werden.

Hierbei wird ausgenutzt, dass dynamisch generierte Webseiten ihren Inhalt oft an über URL (HTTP-GET-Methode) oder Formulare (HTTP-POST-Methode) übergebene Eingabewerte anpassen. Nicht-persistent heißt dieser Typ, da der Schadcode nur temporär bei der jeweiligen Generierung der Webseite eingeschleust, nicht aber gespeichert wird. Ruft man die Seite danach ohne die manipulierte URL oder das manipulierte Formular auf, ist der Schadcode nicht mehr enthalten."

Wikipedia

STORED XSS

Der Schadcode ist auf dem Server gespeichert und wird bei jedem Aufruf der Seite eingeschleust.

"Persistentes (persistent) oder beständiges (stored) Cross-Site-Scripting unterscheidet sich vom reflektierten XSS prinzipiell nur dadurch, dass der Schadcode auf dem Webserver gespeichert wird, wodurch er bei jeder Anfrage ausgeliefert wird. Dies ist bei jeder Webanwendung möglich, die Benutzereingaben serverseitig speichert und diese später wieder ausliefert, solange keine Prüfung der Benutzereingaben bzw. eine geeignete Kodierung der Ausgabe stattfindet."

Wikipedia

ENCODE 64, DECODE 64

Username und Passwort dürfen nie unverschlüsselt innerhalb einer Webseite übermittelt werden. Dafür kann z.B. Encode 64 / Decode 64 genutzt werden.

LAMP STACK

Linux, Apache, MySQL, PHP

3 TITEL BLOCK 2

ARRAYS

```php
<?php
$standorte = array(
  "Bern",
  "Zürich",
  "St.Gallen",
  "Shanghai",
  "Kobe"
);
$mitarbeiter = array(
  array("Bern",250),
  array("Zürich",7),
  array("St.Gallen",35),
  array("Shanghai",30),
  array("Kobe",50),
);
 ?>
```

```php
for ($row = 0; $row < 5; $row++) //Verschachtelte for Schleife
{
  echo "<p><b>Standort $row</b></p>";
  echo "<ul>";
  for ($col = 0; $col < 2; $col++)
  {
  echo "<li>".$mitarbeiter[$row][$col]."</li>";
  }
echo "</ul>";
}
```

EXTERNE PHP DATEI EINBINDEN

DATENKONSISTENZ

Entspricht die Usereingabe auch dem Datenbankeintrag?

SICHERHEIT

Validierung der Feldeingaben

PERFORMANCE

DDoS Testing

SOURCE CODE REVIEW

Ist die optimale Codequalität vorhanden? Manuell möglich, sehr aufwändig oder static Code Analysis oder dynamic Code Analysis. → zur Laufzeit

CIA MODEL

Model über die wichtigsten Eigenschaften einer App:

- Verfügbarkeit
- Integrität
- Vertraulichkeit
- zusätzlich: nicht abstreitbarkeit

Zwei Faktor Authorisierung

Das heisst, es wird sich über 2 Faktoren eingeloggt. 1. Benutzername und Passwort, 2. Token: Code per SMS, Fingerabdruck, QR Code, etc.

4 TITEL BLOCK 3

RELYING

Stelle nimmt Anfrage entgegen und leitet sie weiter → routing im Web.

(Passwörter und Benutzername werden an Dritte weitergeleitet.

API

Schnittstelle um mit Service zu kommunizieren

LOAD BALANCER

Load Balancer sind Server, die dazu genutzt werden, dass grosse Anfragen auf eine Seite verteilt werden können.

PHP KOPF

In einen PHP Kopf gehören:

- Projekt
- Datum
- Version
- Autor

COOKIE

In Cookies sind Daten über besuchte Webseiten gespeichert.

Mit ihnen werden Authentisierungen vorgenommen, Sessions gespeichert (Warenkörbe, etc) oder sie werden zum Tracking verwendet.

SUPERCOOKIES

Supercookies sind Cookies, die weiter Daten lesen, selbst wenn man sich von einem System ausgeloggt hat.

PHP FRAMEWORKS

- Laravel
- Symphonie
- CakePHP

5 TITEL BLOCK 4

TESTING

Whitebox Test

Nur der In- sowie Output wird getestet. Der Quellcode ist nicht bekannt.

Blackbox Test

Die Software wird getestet und der Quellcode liegt offen. Jede Verzeigung und Bedingung muss getestet werden.

Konstruktiver Test

Funktionieren die Funktionen? Die Funktionen, die die App bieten soll, werden getestet.

Destruktiver Test

Es wird alles darangesetzt, das System zum Absturz zu bringen

Graybox Test

Pen Test

Kommt ein Hacker über das Internet oder Social Engineering ins System?

Konzeptionelle Audits

Architektur und Dokumente werden geprüft

Web App Audits

Mobile App Audits

Phising Audits

Sensibilität der Mitarbeiter wird geprüft

Fuzzing

Negativ Testing → Anfragen, die das Programm nicht erwartet → Absturz

Datenvalidierung

Können nur Daten eingegeben werden, die nötig sind?

Bedienbarkeit

Kommt auch meine Oma mit der BankApp zurecht?

6 TITEL BLOCK 5

SESSION COUNTER ERSTELLEN

Um einen Session Counter zu erstellen, muss als erstes ein File erstellt werden, in das die Counts geschrieben werden können.
Code: Als erstes das File öffnen, dann überprüfen ob die Session-Variable ['views'] gesetzt ist. Falls nicht, die Session-Variable auf 1 setzen und in eine Variable speichern. Dann den Wert ins File schreiben.

Falls die Session-Variable gesetzt wurde, diese um eins erhöhen, und wieder ins File schreiben.

Die Varaible, in der die Anzahl Besuche gespeichert ist, im HTML ausgeben.

```php
<?php
//session Counter
session_start();
$handler = fopen('counter.txt', 'w');
if(!isset($_SESSION['views'])) {
  $data = $_SESSION['views'] = 1;
  fwrite($handler, $data);
}
else {
  $data = $_SESSION['views']++;
  fwrite($handler, $data);
  fread($handler, filesize('counter.txt'));
}
fclose($handler);

//Date
$date = "<p> Heute ist der " . date("d.m.Y");
 ?>
```

FINDEN UND ERSETZTEN

Um in einem String ein Zeichen oder eine Zeichenkette zu finden, kann mit der strpos() Methode gearbeitet werden. Diese gibt den Index zurück, an dem das Zeichen gefunden wurde.

Um einen Teil eines Strings zu ersetzen, nimmt man am besten die str_replace() Funktion.

```php
<?php
$str = "TOPOMEDICS:----- Messbar besser";
$find = "bar";
$ersetzt = "-";

$pos = strpos($str, $find);

if($pos == false) {
  echo "'" . $find . "' wurde nicht im String gefunden<br>";
}

else {
  echo "Im String: '" . $str . "', liegt '" . $find . "' an "
  . $pos . "ter Position<br>";
}

str_replace($ersetzt, "", $str)

echo "Im String: '" . $str . "', wurden alle '" . $ersetzt .
"' durch nichts ersetzt: " . str_replace($ersetzt, "", $str);
 ?>
```

7 TITEL BLOCK 6

File open und read

```php
<?php

$fn = fopen("beispiel.txt","r");

//echo $fn;
$fsleread = fread($fn, filesize("beispiel.txt"));
//echo $fn;
echo "fileread: " . $fsleread;

$result = fgets($fn);
echo "Result: " . $result;

$fs = fopen("beispiel.txt", "r");

  while(!feof($fs))  {
  $result = fgets($fs);
  echo "<br>$result<br>" ;
  }

  fclose($fs);

 ?>
```

```php
<?php //$target_path = $_SERVER['DOCUMENT_ROOT'] . "/m133/fileuploads/"
//. basename($_FILES['uploadedFile']['name']);

$dirpath =  $_SERVER['DOCUMENT-ROOT'];
echo $dirpaht;

$dirpaht = $dirpaht . 'm133/';
$imageFileType = strtolower(pathinfo($target_file,PATHINFO_EXTENSION));
// Check if image file is a actual image or fake image
if(isset($_POST["submit"])) {
    $check = getimagesize($_FILES["fileToUpload"]["tmp_name"]);
    if($check !== false) {
        echo "File is an image - " . $check["mime"] . ".";
        $uploadOk = 1;
    } else {
        echo "File is not an image.";
        $uploadOk = 0;
    }
}
?>
```

8 TITEL BLOCK 7

```php
<?php
session_start(); //Nicht vergessen
$name = $_POST['name'];
$name = preg_replace('/[^A-Za-z0-9\_\']/', '', $name); //1. Parameter:$
//Muster das gesucht werden soll, 2. Parameter: mit was ersetzt werden
//soll, 3. Parameter: String, der bearbeitet werden soll
//Regex Erklärung: ^negiert (was darf noch verwendet werden
//[Gross- und Kleinbuchstaben, Zahlen und Underlines])
$name = addslashes($name);

if(!isset($name) OR empty($name)) {
   $name = "Gast";
   echo "<SCRIPT type='text/javascript'> //not showing me this
   alert('Username eingeben');
   window.location.replace(\"formular.html\");
   </SCRIPT>";
}

  $_SESSION['username'] = $name;

  //Text ausgeben
  echo "Hallo $name <br />
  <a href=\"seite2.php\">Seite 2</a><br />
  <a href=\"logout.php\">Logout</a>";

//Session registrieren

?>
```

Konstante definieren

define('mwst', 0.08);

Zufallszahl generieren

$zufallswert = rand(1,10);

mulitdimensionales array

```php
$auto = array(
   array(
      "marke"=>"BMW",
      "typ"=>"140i",
      "farbe"=>"schwarz"),
   array(
      "marke"=>"Mercedes",
      "typ"=>"C43",
      "farbe"=>"grau")
);
```

```php
$mitarbeiter2 = [
  ["vorname"=>"Jacqueline", "nachname"=>"Motzer"],
  ["vorname"=>"Markus", "nachname"=>"Meier"]
]
```

9 TITEL BLOCK 8

```php
<?php
$target_path = "uploads/";
$target_path = $target_path.basename( $_FILES['fileToUpload']['name']);

if(move_uploaded_file($_FILES['fileToUpload']['tmp_name'], $target_path)) {
    echo "File uploaded successfully!";
} else{
    echo "Sorry, file not uploaded, please try again!";
}
?>
```

```php
<?php
$arr = array(1, 2, 3, 4);
foreach ($arr as &$value) {
    $value = $value * 2;
}
// $arr ist nun array(2, 4, 6, 8)
unset($value); // Entferne die Referenz auf das letzte Element
?>
```