security-assessment.com

White Paper

Title: 0x00 vs ASP file upload scripts

Prepared by: Brett Moore
Network Intrusion Specialist
Security-Assessment.com

Date: 13 July 2004

Traduction: Jérôme ATHIAS
Mise à jour : 24/08/2004

Vue d'ensemble

Les influences de « l'octet poison NULL » n'ont jamais été vraiment explorées dans ASP, mais comme d'autres langages, l'octet NULL peut poser des problèmes lorsque ASP transmet les données aux objets.

Beaucoup de systèmes d'upload écrits en ASP souffrent d'un problème courant du fait qu'un octet NULL peut être inséré dan le paramètre nom du fichier entraînant toute extension, après l'octet nul, à être ignorée lors de l'écriture du fichier.

Cela signifie que dans certains cas, il est possible d'outrepasser les vérifications d'extensions valides, même si l'une d'elle est ajoutée par l'application.

Ceci est très similaire aux attaques contre perl et PHP, la différence réside dans la manière où l'octet nul est envoyé à l'application.

Ce problème survient lorsque les données sont comparées et validées dans un script ASP mais passées au FileSystemObject sans vérification d'octets NULL.

Ce document va décrire comment les scripts d'upload ASP peuvent être impactés par l'attaque de l'octet poison NULL.

Portée

Les informations recueillies dans ce document sont basées sur une recherche réalisée en utilisant des systèmes d'upload qui intègrent posts multipart/form-data et l'objet de script FileSystemObject.

Tout au long de ce document, nous nous focaliserons sur la méthode CreateTextFile, qui est utilisée pour créer un fichier en écriture, mais il est possible que d'autres fonctions objets soient vulnérables au même genre de problème.

Un %00 ou NULL ne peut pas être envoyé à travers l'URL ou un post normal de formulaire car le serveur web interprète ceci comme la fin de chaîne, mais ne l'enregistre pas dans la variable de nom de fichier.

Lorsqu'un nom de fichier est envoyé en utilisant un post multipart/form-data, l'octet nul sera inclus dans la variable de nom de fichier, ceci affectant les appels au FileSystemObject.

Transfert montant de fichiers

Le transfert montant de fichiers est communément réalisé en utilisant un objet en entrée de type *file* et un type d'encodage de *multipart/form-data.*

Le type de contenu "application/x-www-form-urlencoded" est inopérant pour envoyer une grande quantité de données binaires ou de texte contenant des caractères non-ASCII. Le type de contenu "multipart/form-data" doit être employé pour soumettre des formulaires qui contiennent des fichiers, des données non-ASCII, et des données binaires.

Un message "multipart/form-data" contient un ensemble de parties, chacune représentant un contrôle réussi. Les parties sont envoyées à l'agent de traitement dans le même ordre que les contrôles correspondant apparaissent dans le flux du document.

```
<form method=post enctype="multipart/form-data" action=upload.asp>
        Votre photo:<BR><input type=file name=YourFile><BR><BR>
        <input type=submit name=submit value="Upload">
</form>
```

En validant le formulaire, les données seront envoyées au format multipart/form-data. Cela permet le transfert de tous les octets, y compris les nuls, dans les données transmises par le formulaire.

Recevant le post, la page ASP cible doit traiter et décoder les données transmises en un état utilisable.

Enregistrement de fichier

A un certain moment du processus d'upload, le fichier sera enregistré à un emplacement. Le code suivant est un exemple de code couramment utilisé pour faire cela.

```
Public Sub Save(filename)
        Dim objFSO, objFSOFile
        path=server.MapPath("/uploads/")
        Set objFSO = Server.CreateObject("Scripting.FileSystemObject")
        Set objFSOFile = objFSO.CreateTextFile(path + "\" + filename)
        objFSOFile.Write <file contents>
        objFSOFile.Close
End Sub
```

Lorsque le paramètre nom de fichier est passé à la fonction CreateTextFile(), il peut contenir des octets NULL. Cela peut impacté le nom du fichier créé puisque la fonction CreateTextFile lit uniquement jusqu'à l'octet NULL, puis créée le fichier.

```
Set objFSOFile = objFSO.CreateTextFile(path + "\" + filename)
```

Si le nom de fichier (filename) contient un octet NULL, tout ce qui suit cet octet sera ignoré.

Octet Null

L'octet NULL peut être inséré à la main par des modifications des données multipart transmises à l'aide d'un éditeur hexa ou en utilisant un proxy web.

Multipart Form Post

POST /upload.asp HTTP/1.0
Content-Type: multipart/form-data; boundary=----------
7d4cb161b009c
Host: localhost
Content-Length: 359
Pragma: no-cache
Cookie: ASPSESSIONIDSAADRCRS=LAKNNAKAGMIBJCOOLBIFEHIK

----------7d4cb161b009c
Content-Disposition: form-data; name="YourFile"; **filename="c:\nc.exe .bmp"**
Content-Type: text/plain

Proof Of Upload Test File
brett.moore@security-assessment.com
----------7d4cb161b009c
Content-Disposition: form-data; name="submit"

Upload
----------7d4cb161b009c

Le paramètre filename du post ci-dessus a été modifié comme suit;

N	C	.	E	X	E	**(null)**	.	B	M	P
4E	43	2E	45	58	45	**00**	2E	42	4D	50

Notez qu'un octet NULL (0x00) a été inséré entre le .exe et le .bmp.

Scripts de Test

Les deux scripts d'enregistrement de fichiers suivants sont des exemples de scripts d'upload où l'extension du fichier écrit peut être modifié arbitrairement.

Dans les deux cas tFile est le paramètre nom de fichier passé dans le post.

Exemple Un (Ajout de l'Extension de Fichier)

```
Public Sub Save(Path)
Dim objFSO, objFSOFile
Dim lngLoop

Set objFSO = Server.CreateObject("Scripting.FileSystemObject")
Set objFSOFile =
        objFSO.CreateTextFile(objFSO.BuildPath(Path, tFile + ".bmp"))

 ‘ Ecriture du fichier
 For lngLoop = 1 to LenB(m_Blob)
        objFSOFile.Write Chr(AscB(MidB(m_Blob, lngLoop, 1)))
 Next

 objFSOFile.Close
End Sub
```

Exemple Deux (Vérification de l'Extension de Fichier)

```
Public Sub Save(Path)
 Dim objFSO, objFSOFile
 Dim lngLoop

‘ Vérification de l’extension du fichier
if right(tFile,4) <>  “.bmp”  then exit sub

Set objFSO = Server.CreateObject("Scripting.FileSystemObject")
Set objFSOFile=
        objFSO.CreateTextFile(objFSO.BuildPath(Path, tFile))
‘ Write the file contents
For lngLoop = 1 to LenB(m_Blob)
  objFSOFile.Write Chr(AscB(MidB(m_Blob, lngLoop, 1)))
 Next

 objFSOFile.Close
End Sub
```

Résumé Final

Il a couramment été pensé que les applications internet écrites en ASP étaient immunisées aux problèmes associés aux octets NULL. Alors que dans la plupart des cas cela s'avère vrai, il peut être constaté ici que les applications faisant appel à des objets externes au langage de script ASP natif, peuvent être impactées par les octets NULL.

Il est probable que d'autres objets et zones puissent également être manipulés d'une certaine manière lorsque leurs données sont collectées à travers un post multipart/form-data.

Comme dans les autres endroits, une validation correcte des données entrées par l'utilisateur est primordiale pour la sécurité des applications internet. Il est donc important de vérifier l'entrée non seulement pour les chaînes courantes d'attaques utilisées pour traverser les répertoires, mais aussi pour les octets NULL avant d'utiliser l'entrée pour la création de fichiers.

Références

Perl CGI problems - rain.forest.puppy

http://www.phrack.org/show.php?p=55&a=7

Bugtraq Post Regarding PHP and null bytes

http://seclists.org/lists/bugtraq/2003/Jan/0159.html

OWASP HTML Version

http://www.cgisecurity.com/owasp/html/guide.html#id2846281

Forms in HTML documents

http://www.w3.org/TR/REC-html40/interact/forms.html#h-17.13.4

Security-Assessment.com

www.security-assessment.com

About Security-Assessment.com

Security-Assessment.com is an established team of Information Security consultants specialising in providing high quality Information Security Assurance services to clients throughout the UK, Europe and Australasia. We provide independent advice, in-depth knowledge and high level technical expertise to clients who range from small businesses to some of the worlds largest companies

Using proven security principles and practices combined with leading software and proprietary solutions we work with our clients to provide simple and appropriate assurance solutions to Information security challenges that are easy to understand and use for their clients.

Security-Assessment.com provides security solutions that enable developers, government and enterprises to add strong security to their businesses, devices, networks and applications. We lead the market in on-line security compliance applications with the SA-ISO Security Compliance Management system which enables companies to ensure that they are effective and in line with accepted best practice for Information Security Management.

Copyright Information