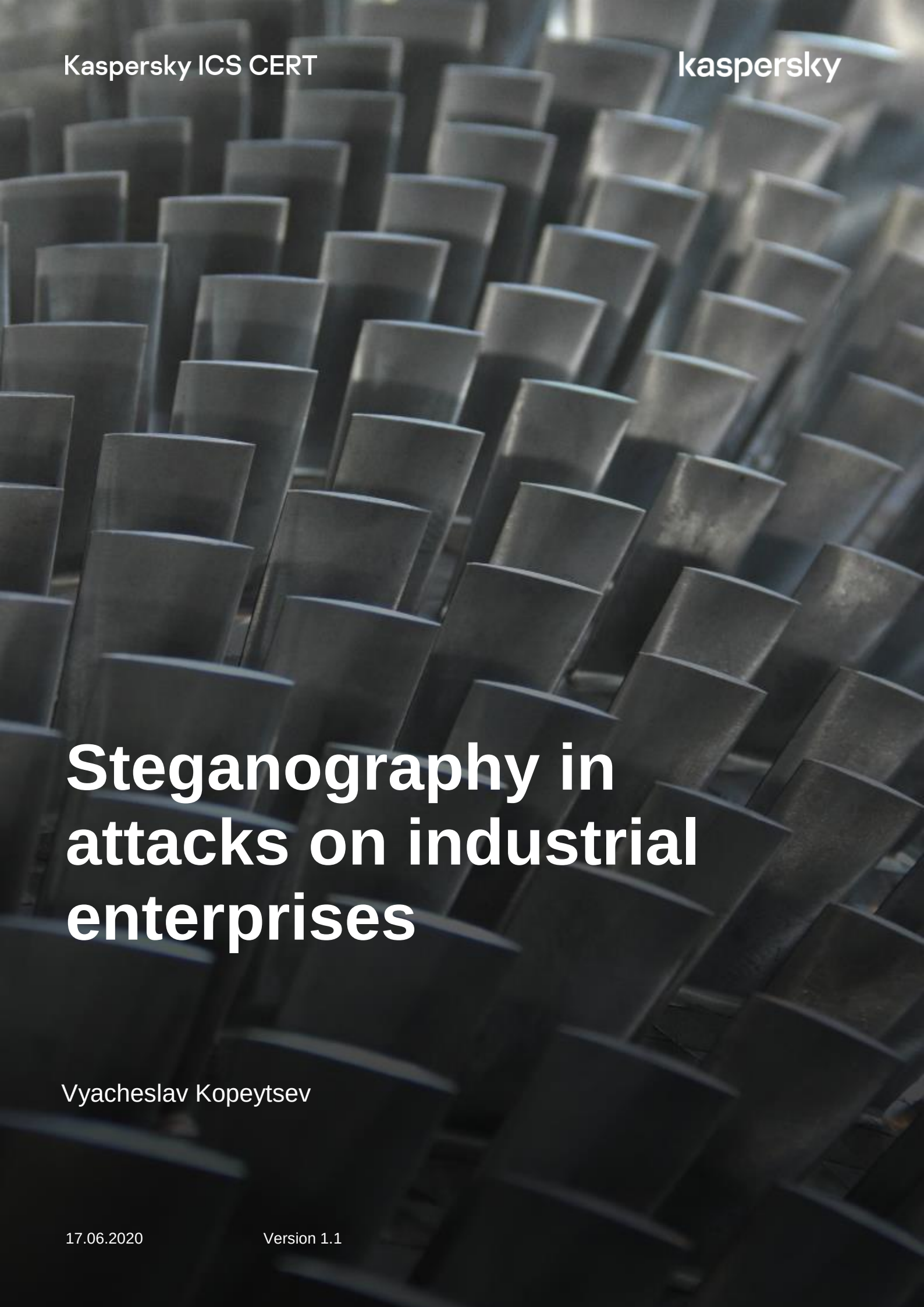


Kaspersky ICS CERT

kaspersky

Steganography in attacks on industrial enterprises

Vyacheslav Kopeytsev

17.06.2020 Version 1.1

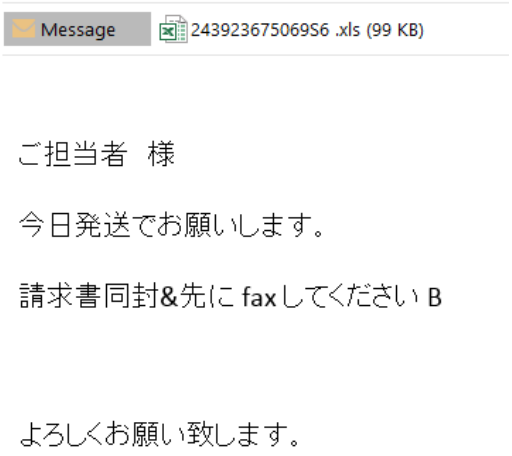
Kaspersky ICS CERT experts have identified a series of attacks on organizations located in different countries. As of early May 2020, there are known cases of attacks on systems in Japan, Italy, Germany and the UK. Up to 50% of the attackers' targets are organizations in various industrial sectors. Attack victims include suppliers of equipment and software for industrial enterprises. Attackers use malicious Microsoft Office documents, PowerShell scripts, as well as various techniques that make it difficult to detect and analyze malware.

Phishing emails, used as the initial attack vector, were customized using text in the language of each specific country. The malware used in this attack continued to run only if the operating system had a localization that matched the language used in the phishing email. For example, in the case of an attack on a company operating in Japan, the text of a phishing email and a Microsoft Office document containing a malicious macro were written in Japanese. Also, to successfully decrypt the malware module, the operating system had to have a Japanese localization as well.

As a result of an attack, banker Trojans from the Bebloh family (Shiotob, URLZone) and the Ursnif family (Gozi, ISFB) are installed on victim computers.

Technical analysis

The attackers send a phishing email to the victim. The email contains a request to open the attached document urgently.



Screenshot of a phishing email with a malicious attachment

The Excel document attached to the email contains a malicious macro script (verdict: Trojan.MSExcel.Agent.be). After opening the document, the user sees a message with the request to enable the document's active content. If the user agrees to do that, the malicious macro is executed.

The main task of the macro is to decrypt and execute a PowerShell script. The script is executed with the following parameters:

-ExecutionPolicy ByPass,
-WindowStyle Hidden,
-NoProfile,

i.e., the script is executed in spite of the configured policy, in a hidden window and without loading the user configuration.

```
Function GeneralCatalog()
GeneralCatalog = "pzq /p pzq /p CBjREFuryy -rC  OLCnfF -j 1 -abAvAgR -ABCEbsVY    "". ( $cFubZR[
End Function


Function lAstReport()
lAstReport = "ynpr(',','/')).ercynpr(';','+') ),[VB.pBzceRFfVBa.pbZCErfFVBAzBQr]::QRpbzcerff ) |
End Function
```

Fragment of the obfuscated macro that executes the PowerShell script

The PowerShell script (verdict: HEUR:Trojan.PowerShell.Generic) randomly selects one of the URL addresses listed in it. The URLs lead to public image hosting services, imgur.com and imgbox.com. The script downloads the image to which the URL points and starts the data extraction procedure.

Image downloaded by the malware

The data is hidden in the image using steganographic techniques and is extracted by the malware from pixels defined by the algorithm. Using steganography enables the attackers to evade some security tools, including network traffic scanners.

The data extracted from the image is consecutively encoded using the Base64 algorithm, encrypted with the AES algorithm and encoded using Base64 again. Curiously, the script has an error in its code, included on purpose, with the exception message used as the decryption key. Notably, the text in the exception message depends on the language pack installed in the operating system. Apparently, the attackers prepare the malicious script specifically for victims from a particular country.

The decrypted and decoded data makes up one more PowerShell script, which is executed.

```
(0..286)|.('%')
{
        foreach(${x} in(0..635))
        {
                ${p}=${G}.("GetPixel").Invoke(${x},${_});
                ${O}[${_}*636+${x}]=([math]::("Floor").Invoke((${p}."B"-band15)*16)-bor(${p}."g" -band 15))
        }
};

${EE}=[System.Text.Encoding]::"UTf8"."gETsTRING"(${O}[0..182171]);
${eRROr}.("Clear").Invoke();
${ErRorActIOnPReFeReNCE} ="SilentlyContinue";
&("null")|.("Out-Null");[string]${ness}=${eRRoR}[0]."EXCeption";
${VUS}=[Regex]::("Matches").Invoke(${NeSS},'(?<=: ).*?(?='')') -join'';
```

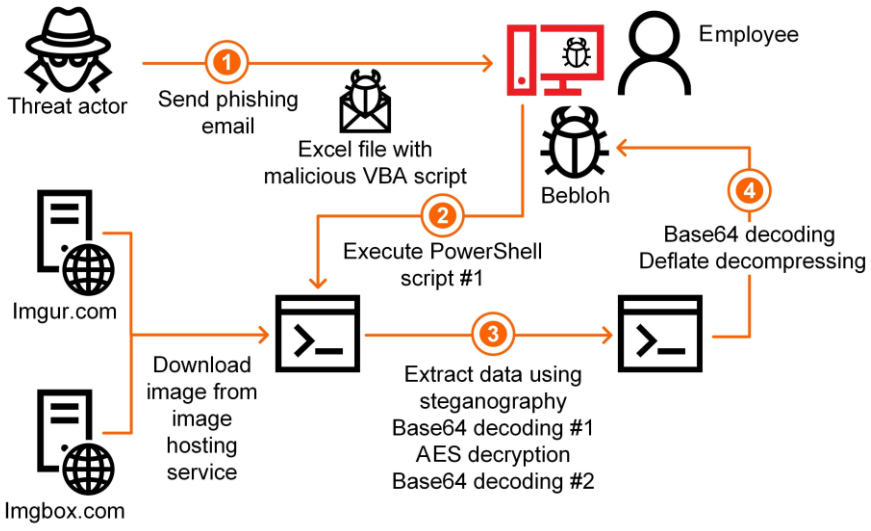
Data decryption code fragment

The second PowerShell script also decodes part of its contents using the Base64 algorithm, after which it unpacks the resulting data buffer using the Deflate algorithm. As a result, the malware gets one more PowerShell script – in this case, an obfuscated sample of malware from the Bebloh family (Shiotob, URLZone). In other cases, the attackers have used Trojans from the Ursnif family (Gozi, ISFB).

```
${A&LSQUO;TTR&LSQUO;IBUtEs} = 'AutoLayout, AnsiClass, Class, Public, SequentialLayout, Sealed, BeforeFieldInit'
${tY&LSQUO;p&LSQUO;ebU&LSQUO;ILDEr} = ${modU&LSQUO;lEbui&LSQUO;lDer}.`D&LSQUO;Effi&LSQUO;NEtyPE`('IMAGE_NT_HEADERS32', ${AT&LSQUO;T&L
${t&LSQUO;ypebu&LSQUO;ilder}.`dEFIN&LSQUO;eFi&LSQUO;elD`('Signature', [UInt32], 'Public') | &(`{0}{2}{1}` -f'Ou','ull','t-N')
${T&LSQUO;YpeB&LSQUO;UiLdEr}.`dEfin&LSQUO;E&LSQUO;FIeLD`('FileHeader', ${imag&LSQUO;e_FiL&LSQUO;e_hEad&LSQUO;Er}, 'Public') | &(`{0}
${tyPEB&LSQUO;Ui&LSQUO;ld&LSQUO;ER}.`De&LSQUO;Fi&LSQUO;N&LSQUO;eFIElD`('OptionalHeader', ${imaGe_OPTi&LSQUO;ON&LSQUO;AL&LSQUO;_&LSQU
${Im&LSQUO;AGe&LSQUO;_nt&LSQUO;_hEadEr&LSQUO;s32} = ${t&LSQUO;YPebu&LSQUO;iL&LSQUO;DEr}.`cR&LSQUO;EATeT&LSQUO;YpE`()
${w&LSQUO;I&LSQUO;N32Types} | &(`{1}{0}{2}` -f'-Me','Add','mber') -MemberType (`{3}{0}{1}{2}`-f 'r','opert','y','NoteP') -Name (`{4}
${a&LSQUO;TtrIBu&LSQUO;TeS} = 'AutoLayout, AnsiClass, Class, Public, SequentialLayout, Sealed, BeforeFieldInit'
${TyP&LSQUO;EBU&LSQUO;I&LSQUO;ldeR} = ${m&LSQUO;oDULEbuI&LSQUO;der}.`dEF&LSQUO;in&LSQUO;eTyPe`('IMAGE_DOS_HEADER', ${aTtrib&LSQUO;U
${tYPe&LSQUO;B&LSQUO;UilD&LSQUO;eR}.`D&LSQUO;EFinE&LSQUO;FieLd`('e_magic', [UInt16], 'Public') | &(`{0}{2}{1}` -f 'Ou','l','t-Nul')
${T&LSQUO;ypEbui&LSQUO;LDER}.`D&LSQUO;E&LSQUO;Fin&LSQUO;eFIeLd`('e_cblp', [UInt16], 'Public') | &(`{1}{0}`-f 'll','Out-Nu')
${TyPEb&LSQUO;UI&LSQUO;LD&LSQUO;Er}.`defIneFi&LSQUO;e&LSQUO;ld`('e_cp', [UInt16], 'Public') | &(`{0}{1}`-f 'Out-Nul','l')
${TY&LSQUO;pEb&LSQUO;UILD&LSQUO;eR}.`DE&LSQUO;F&LSQUO;InEFIElD`('e_crlc', [UInt16], 'Public') | &(`{1}{2}{0}` -f 'll','O','ut-Nu')
${TYPEB&LSQUO;Ui1&LSQUO;D&LSQUO;ER}.`d&LSQUO;eF&LSQUO;inE&LSQUO;FiEld`('e_cparhdr', [UInt16], 'Public') | &(`{2}{1}{0}` -f 'll','t-N
${TyPE&LSQUO;B&LSQUO;UI&LSQUO;lDer}.`d&LSQUO;E&LSQUO;FINEFiElD`('e_minalloc', [UInt16], 'Public') | &(`{0}{1}` -f 'Out','-Null')
```

Fragment of obfuscated script

The Bebloh/Shiotob family has been known since 2009. Trojans from that family steal FTP client and email account passwords and 'listen' to browsers' internet traffic to steal authentication data for various websites. The Ursnif family has broader functionality; its latest versions include updated data stealing modules, including those designed to steal cryptocurrency wallets.



Attack kill chain

Conclusions

Researchers observed previous variants of this attack starting at least from 2018. The new wave of attacks has caught the attention of researchers because industrial enterprises and companies that develop and supply products for industrial enterprises make up a high percentage of potential victims and because the attackers use several unconventional technical solutions.

First, the malicious module is encoded in an image using steganographic techniques and the image is hosted on legitimate web resources. This makes it virtually impossible to detect such malware using network traffic monitoring and control tools while it is being downloaded. From the standpoint of technical solutions, this activity is indistinguishable from sending ordinary requests to legitimate image hosting services.

A second curious feature of the malware is the use of the exception message as the decryption key for the malicious payload. This technique can help the malware evade detection in automatic analysis systems of the sandbox class and makes analyzing the functionality of the malware slightly more difficult for researchers if they do not know what language pack was used on the victim's computer.

In all the cases identified, the malware was blocked by Kaspersky solutions.

If you have encountered an attack of this kind, you can report it to us through a special form on our website.

Recommendations

- Train employees at enterprises in using email securely and, specifically, in identifying phishing messages.
- Restrict macros in Microsoft Office documents.
- Restrict PowerShell script execution (wherever possible).
- Pay special attention to events of launching PowerShell processes initiated by Microsoft Office applications.
- Restrict the ability of programs to gain SeDebugPrivilege privileges (wherever possible).
- Install antivirus software with support for centrally managing the security policy on all systems; keep the antivirus databases and program modules of security solutions up to date.
- Use accounts with domain administrator privileges only when necessary. After using such accounts, restart the system on which the authentication was performed.
- Implement a password policy with password strength and regular password change requirements.
- If it is suspected that some systems are infected, scan these systems with antivirus software and force a change of passwords for all accounts that have been used to log on to compromised systems.

Kaspersky Industrial Control Systems Cyber Emergency Response Team (Kaspersky ICS CERT) is a global project of Kaspersky aimed at coordinating the efforts of automation system vendors, industrial facility owners and operators, and IT security researchers to protect industrial enterprises from cyberattacks. Kaspersky ICS CERT devotes its efforts primarily to identifying potential and existing threats that target industrial automation systems and the industrial internet of things.

Kaspersky ICS CERT ics-cert@kaspersky.com