A Methodical Approach for Detecting BloodHound

Hello!

I am Andy Robbins

Adversary Resilience Lead at SpecterOps

Co-creator of BloodHound

You can find me at @_wald0

Hello!

I am Jared Atkinson

Adversary Detection Lead at SpecterOps

You can find me at @jaredcatkinson

Agenda

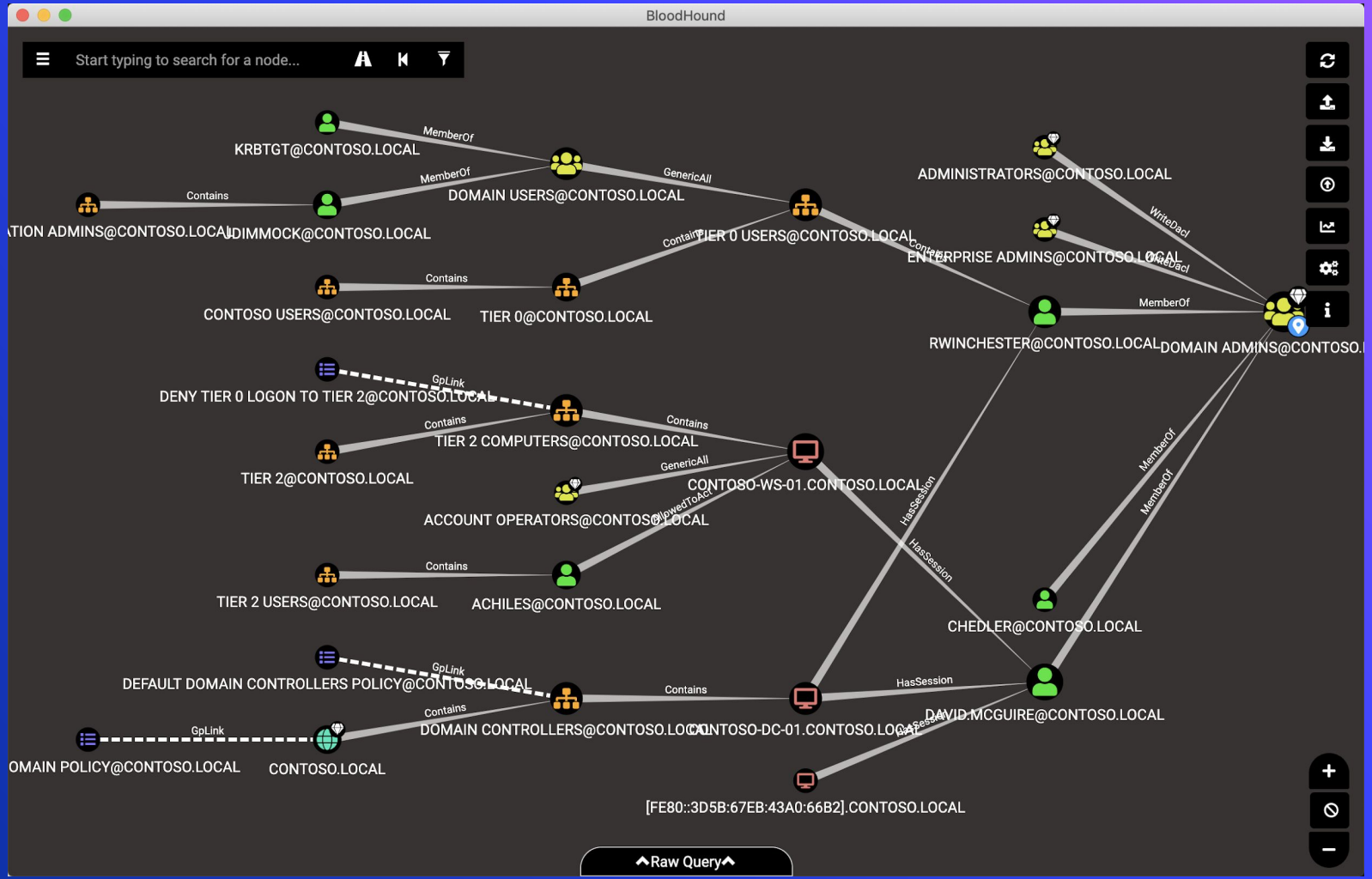
- What do we mean “Detect BloodHound”?
- How SharpHound works
- A closer look at Session collection
- Capability Abstraction

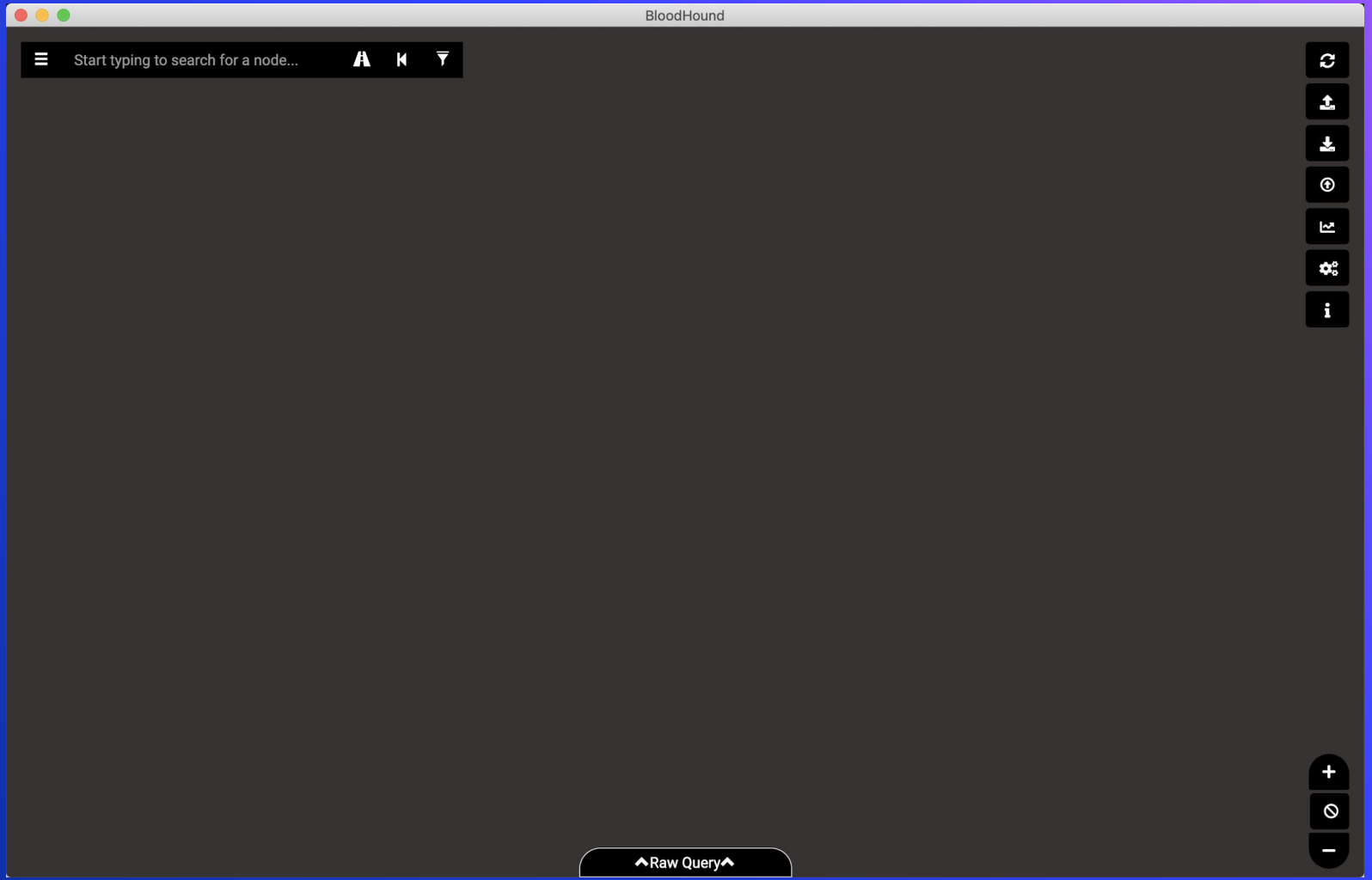
Link to this deck: https://bit.ly/2Wk9bAm

What do we mean by “Detect BloodHound”?

“Detect BloodHound”?

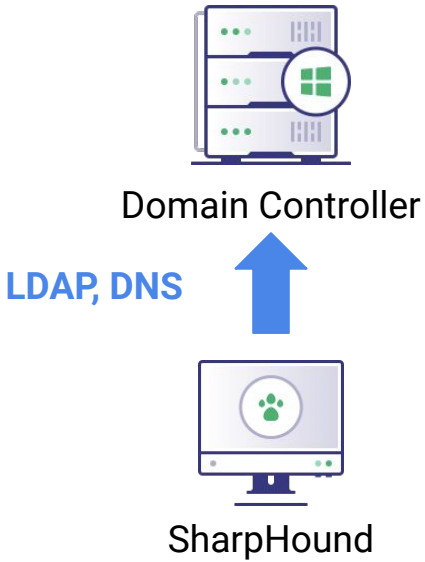
- BloodHound itself is simply an analysis tool
- BloodHound does not actually execute any attacks!
- Just like any analysis tool, BloodHound needs data
- BloodHound gets its data from **SharpHound**


BloodHound
Start typing to search for a node...
KRBTGT@CONTOSO.LOCAL
MemberOf
DOMAIN USERS@CONTOSO.LOCAL
GenericAll
ADMINISTRATORS@CONTOSO.LOCAL
WriteDacl
Contains
UDIMMOCK@CONTOSO.LOCAL
TIER 0 USERS@CONTOSO.LOCAL
ENTERPRISE ADMINS@CONTOSO.LOCAL
CONTOSO USERS@CONTOSO.LOCAL
TIER 0@CONTOSO.LOCAL
RWINCHESTER@CONTOSO.LOCAL
DOMAIN ADMINS@CONTOSO.
DENY TIER 0 LOGON TO TIER 2@CONTOSO.LOCAL
GpLink
TIER 2 COMPUTERS@CONTOSO.LOCAL
TIER 2@CONTOSO.LOCAL
CONTOSO-WS-01.CONTOSO.LOCAL
ACCOUNT OPERATORS@CONTOSO.LOCAL
AllowedToAct
HasSession
TIER 2 USERS@CONTOSO.LOCAL
ACHILES@CONTOSO.LOCAL
CHEDLER@CONTOSO.LOCAL
DEFAULT DOMAIN CONTROLLERS POLICY@CONTOSO.LOCAL
DOMAIN CONTROLLERS@CONTOSO.LOCAL
CONTOSO-DC-01.CONTOSO.LOCAL
DAVID.MCGUIRE@CONTOSO.LOCAL
CONTOSO.LOCAL
[FE80::3D5B:67EB:43A0:66B2].CONTOSO.LOCAL
Raw Query

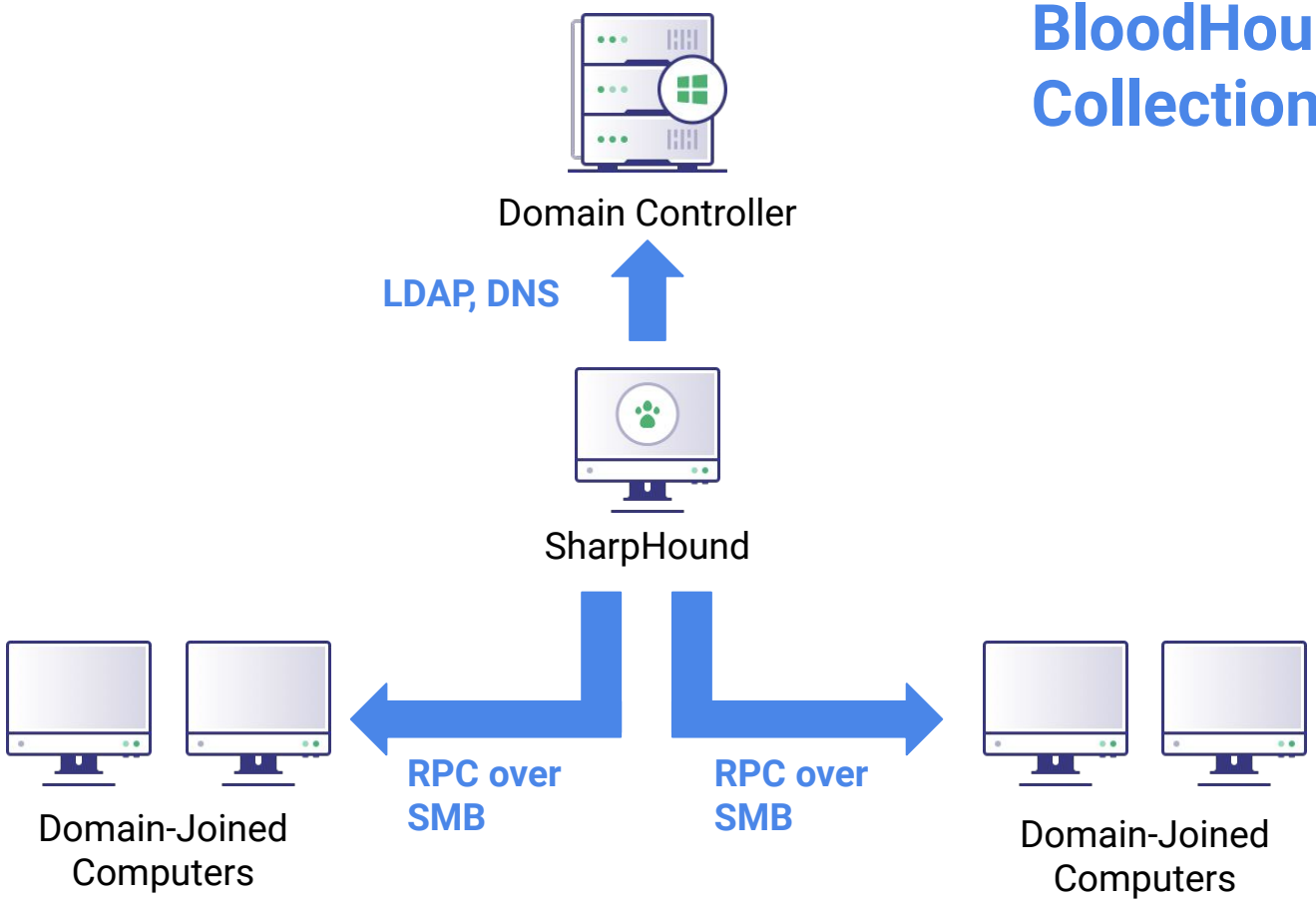

BloodHound
Start typing to search for a node...
Raw Query

How SharpHound Works

BloodHound Data Collection Architecture

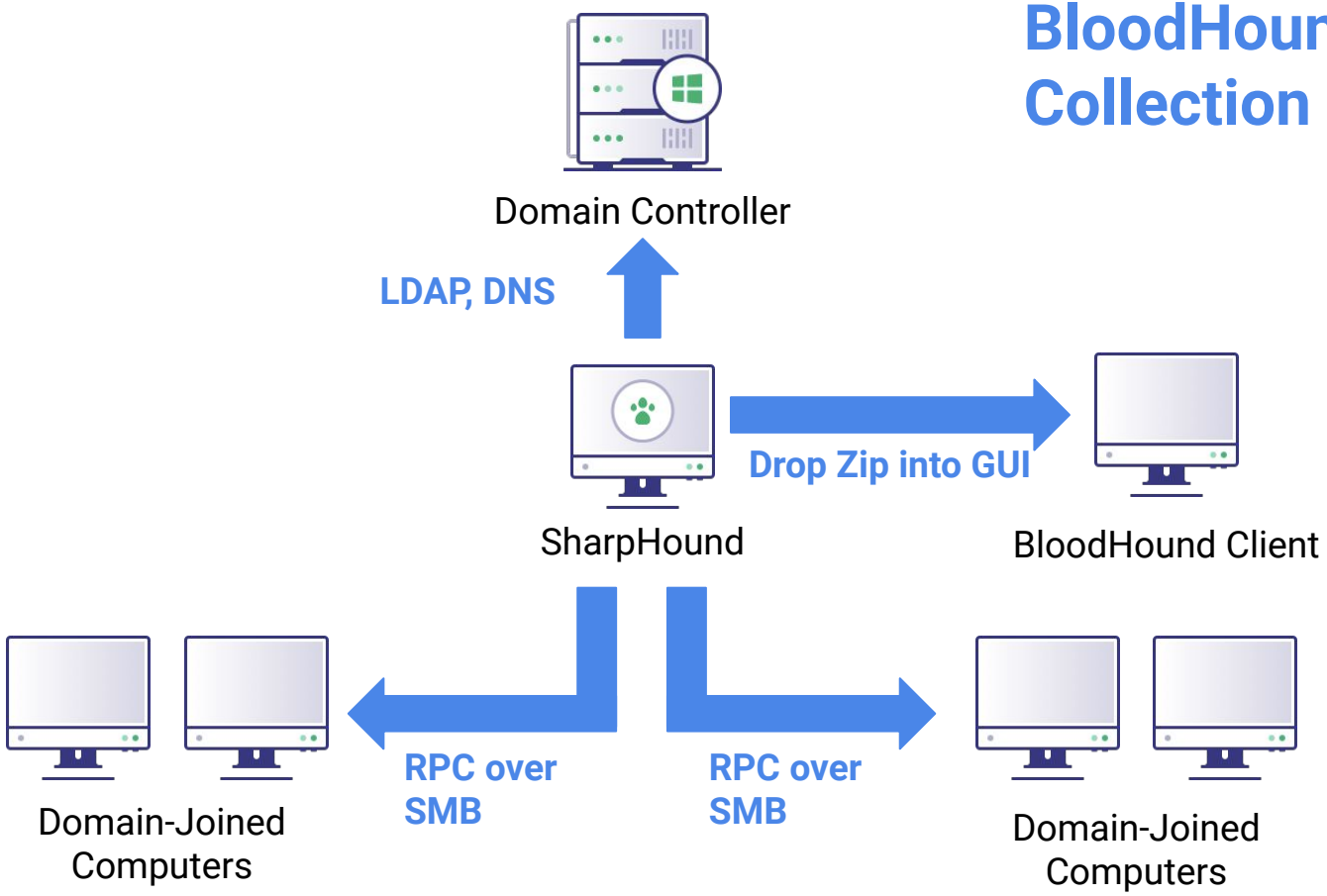


BloodHound Data Collection Architecture



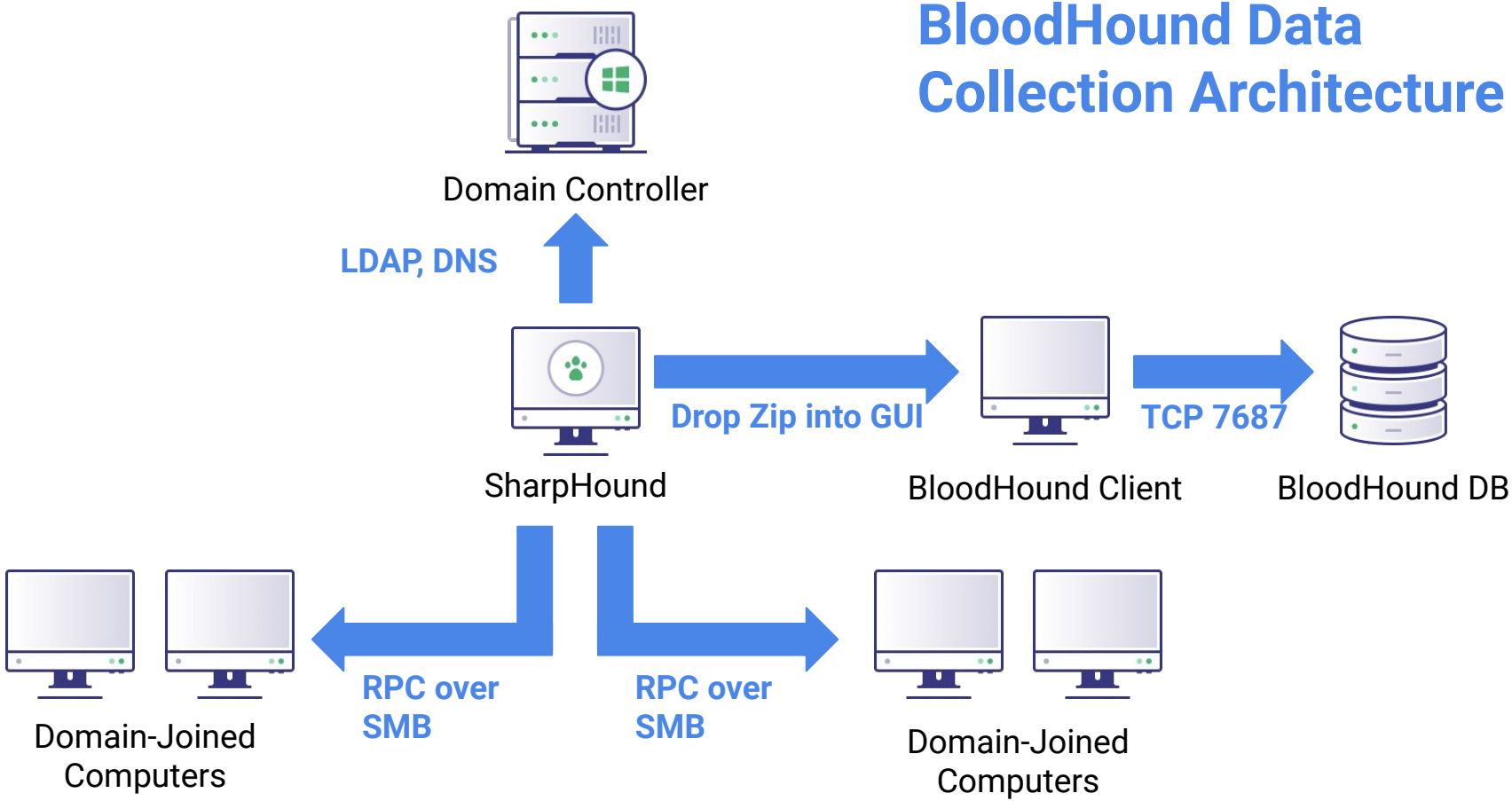
11

BloodHound Data Collection Architecture



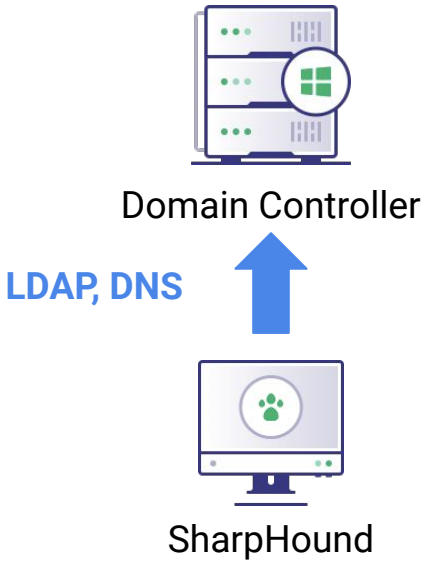
12

BloodHound Data Collection Architecture

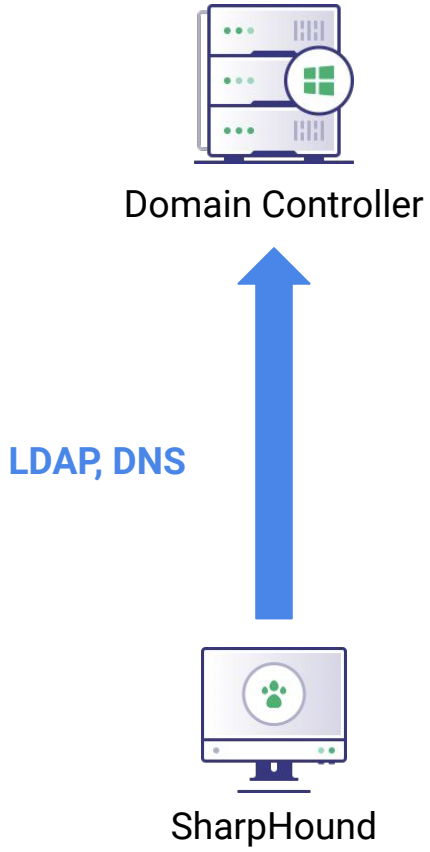


13

BloodHound Data Collection Architecture



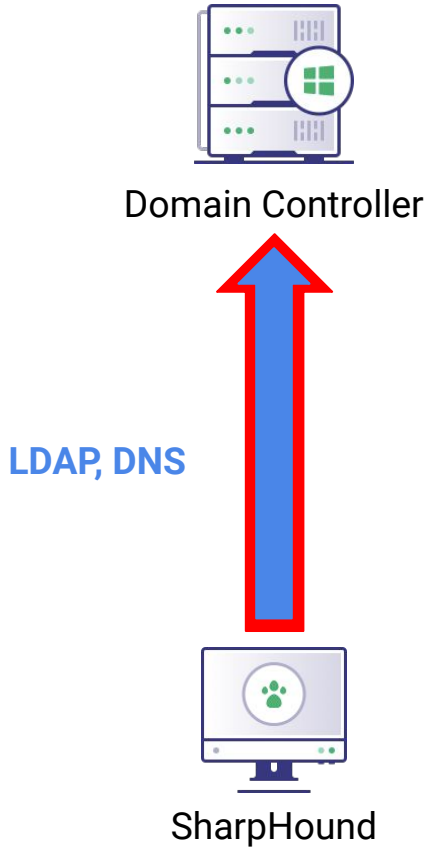
14



SharpHound's LDAP Collection:

- Collects from one domain controller per run

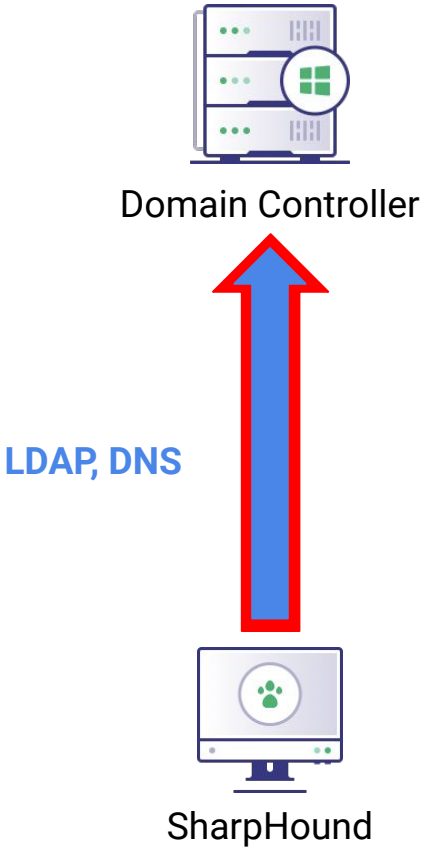
15



SharpHound's LDAP Collection:

- Collects from one domain controller per run
- Uses **Signed and Sealed LDAP** by default - TCP port 389

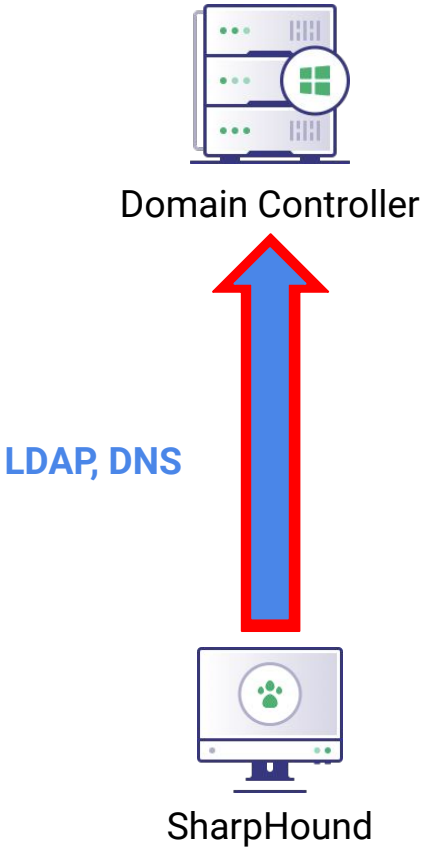
16



SharpHound's LDAP Collection:

- Collects from one domain controller per run
- Uses Signed and Sealed LDAP by default - TCP port 389
- Uses LDAP methods defined in the System.DirectoryServices.Protocols .Net namespace to interact with LDAP on the Domain Controller

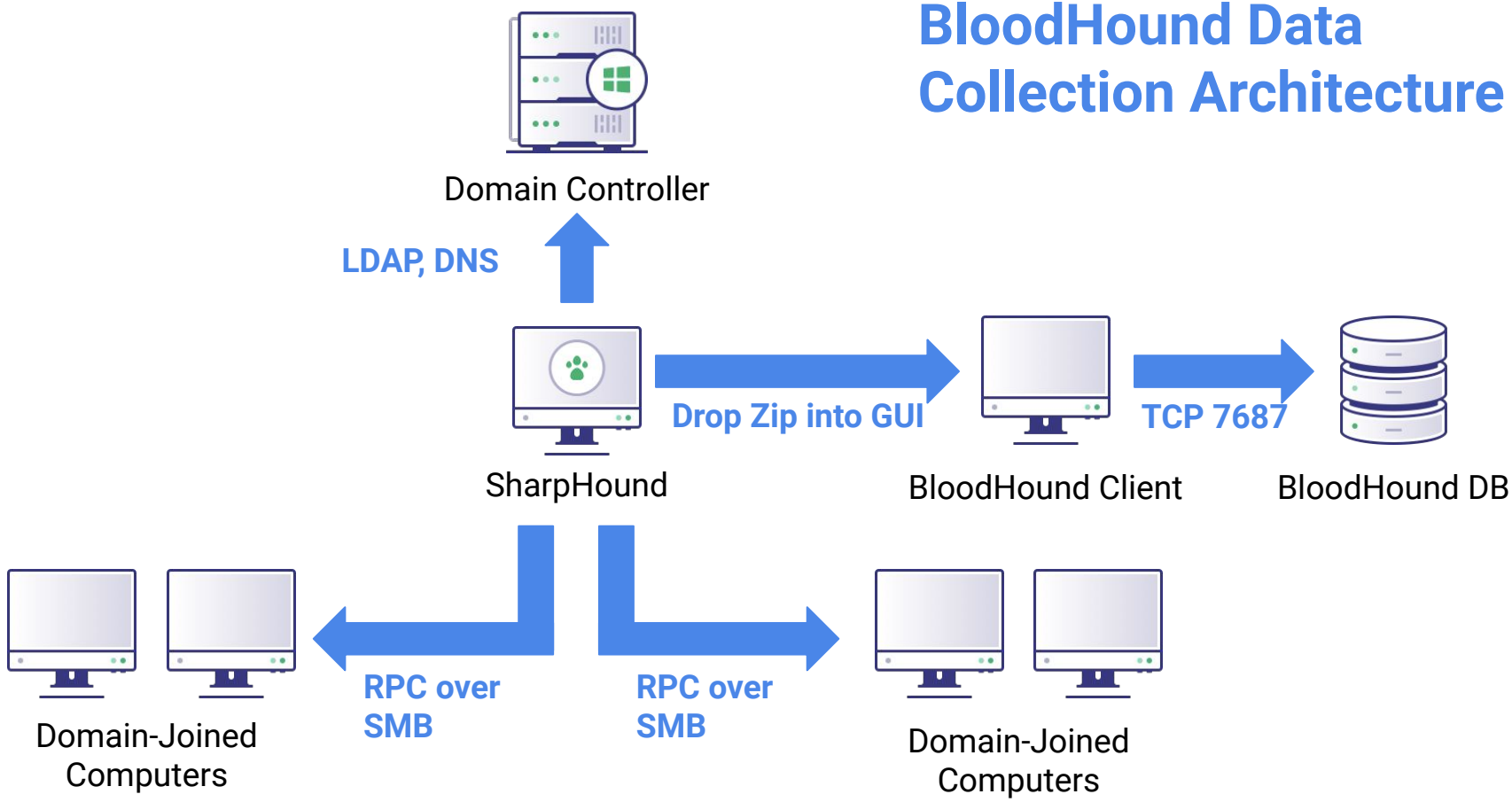
17



SharpHound's LDAP Collection:

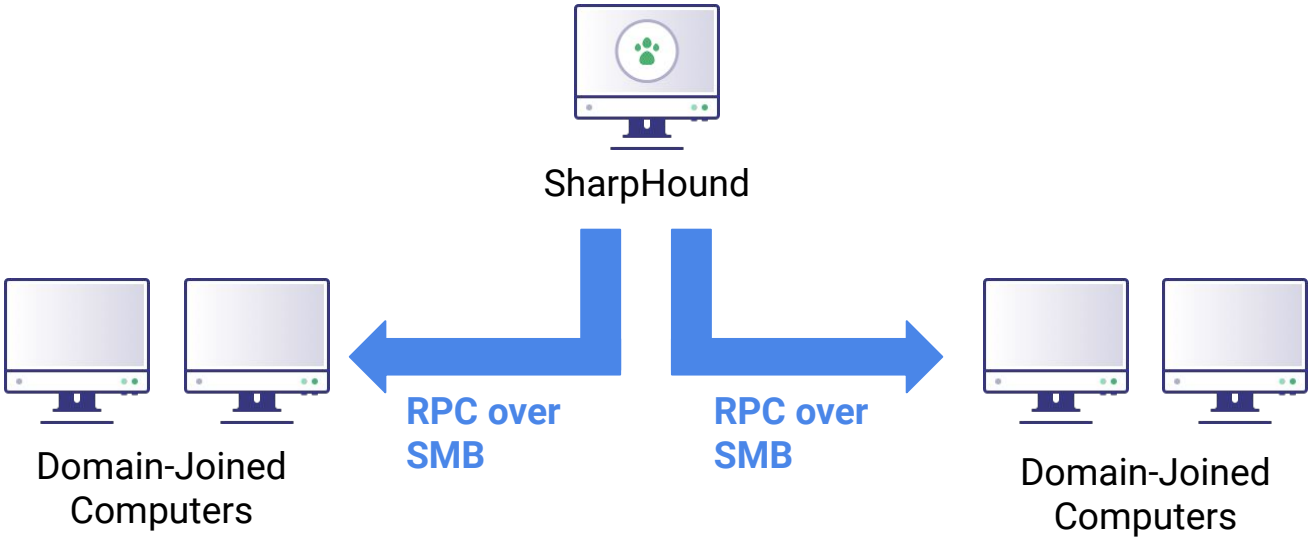
- Collects from one domain controller per run
- Uses Signed and Sealed LDAP by default - TCP port 389
- Uses LDAP methods defined in the System.DirectoryServices.Protocols .Net namespace to interact with LDAP on the Domain Controller
- Performs read operations against AD objects of the following classes:
  - domain
  - organizationalUnit
  - user
  - computer
  - group
  - groupPolicyContainer
  - ms-DS-Group-Managed-Service-Account

BloodHound Data Collection Architecture

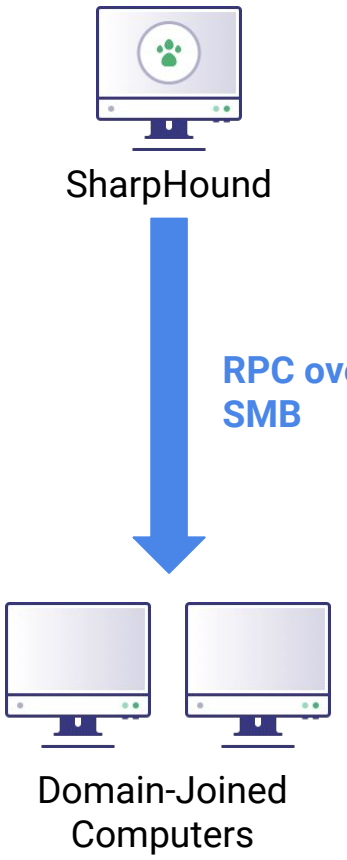


19

BloodHound Data Collection Architecture



20



SharpHound's endpoint collection:

- Gets a list of computers to scan from LDAP, the local SharpHound cache, or an input file
- Attempts to do a TCP connect on each computer on port 445. Scans the system if 445 is open.
- Uses WIN32API functions to collect data:
  - NetSessionEnum - User sessions
  - NetWkstaUserEnum - User sessions (privileged)
  - NetLocalGroupGetMembers - Local group members
- Those API functions are executed via RPC over SMB

21

A closer look at Session collection

Session collection

- Tells us what users are logged in to which computers
- Become an admin on that computer…
- …and impersonate users logged onto it
- The unprivileged method uses NetSessionEnum

Filter by title

Lmshare.h
CONNECTION_INFO_0 structure
CONNECTION_INFO_1 structure
FILE_INFO_2 structure
FILE_INFO_3 structure
NetConnectionEnum function
NetFileClose function
NetFileEnum function
NetFileGetInfo function
NetSessionDel function
NetSessionEnum function
NetSessionGetInfo function
NetShareAdd function
NetShareCheck function
NetShareDel function
NetShareDelEx function
NetShareEnum function
NetShareGetInfo function
NetShareSetInfo function
SESSION_INFO_0 structure
SESSION_INFO_1 structure

NetSessionEnum function

12/05/2018 • 5 minutes to read

Provides information about sessions established on a server.

Syntax

C++ Copy

```
NET_API_STATUS NET_API_FUNCTION NetSessionEnum(
  LMSTR   servername,
  LMSTR   UncClientName,
  LMSTR   username,
  DWORD   level,
  LPBYTE  *bufptr,
  DWORD   prefmaxlen,
  LPDWORD entriesread,
  LPDWORD totalentries,
  LPDWORD resume_handle
);
```

Parameters

`servername`

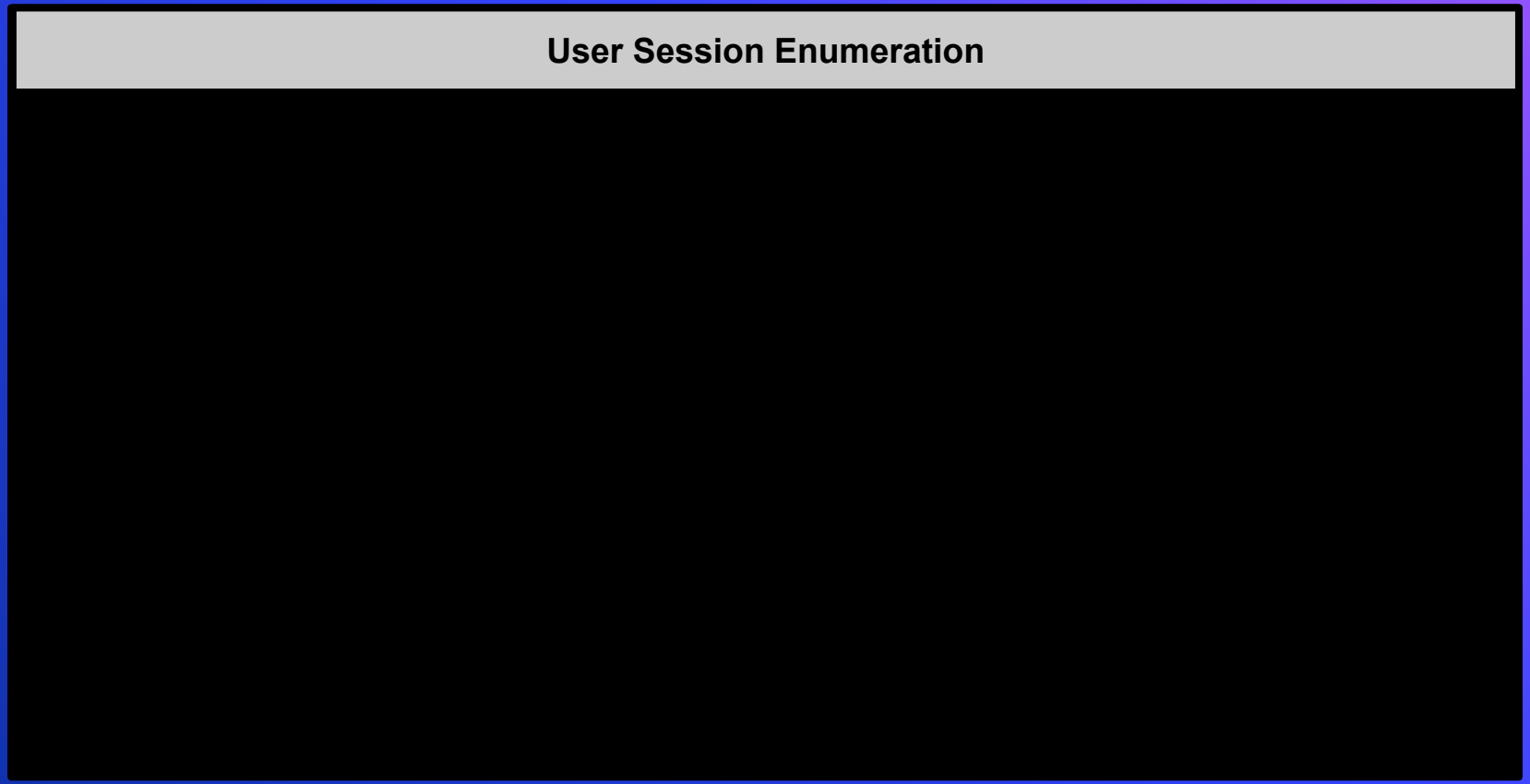
Pointer to a string that specifies the DNS or NetBIOS name of the remote server on which the function is to execute. If this parameter is **NULL**, the local computer is used.

Capability Abstraction

Abstraction Questions

1. Technique: What Technique are we interested in?
2. **Tools:** What Tool(s) do we know perform this technique?
3. **Functions:** What API Function enables the technique?
4. Files: Does this technique require the creation or access of any files?
5. Registry: Does this technique require configuration from the registry?
6. **Network:** Does this technique require network activity to occur?

User Session Enumeration



27

Tools

What Tool(s) do we know perform this technique?

- Sharphound (https://github.com/BloodHoundAD/SharpHound)
- Get-NetSession (https://github.com/PowerShellMafia/PowerSploit/blob/c7985c9bc31e92bb6243c177d7d1d7e68b6f1816/Recon/PowerView.ps1#L8173)
- NetSess (http://www.joeware.net/freetools/tools/netsess/)
- Bloodhound.py (https://github.com/fox-it/BloodHound.py)

User Session Enumeration

Tools	Sharphound

29

Florian Roth
@cyb3rops

Sigma rule to detect Bloodhound

github.com/Neo23x0/sigma/...

win_hack_bloodhound.yml ● | apt_wocao.yml | win_susp_rc4_kerberos.yml | johns-rule.yml | lnx_susp_failed_lc

```yaml
title: Bloodhound Hack Tool
id: f376c8a7-a2d0-4ddc-aa0c-16c17236d962
description: Detects command line parameters used by Bloodhound hack tool
author: Florian Roth
references:
    - https://github.com/BloodHoundAD/BloodHound
date: 2019/12/20
tags:
    - attack.discovery
    - attack.t1087
logsource:
    category: process_creation
    product: windows
detection:
    selection:
        CommandLine|contains: ' -CollectionMethod All '
    condition: selection
falsepositives:
    - Other programs that use these command line option and accepts an 'All' parameter
level: high
```

6:38 AM · Dec 20, 2019 · TweetDeck

153 Retweets **401** Likes

Florian Roth
@cyb3rops

Sigma rule to detect Bloodhound

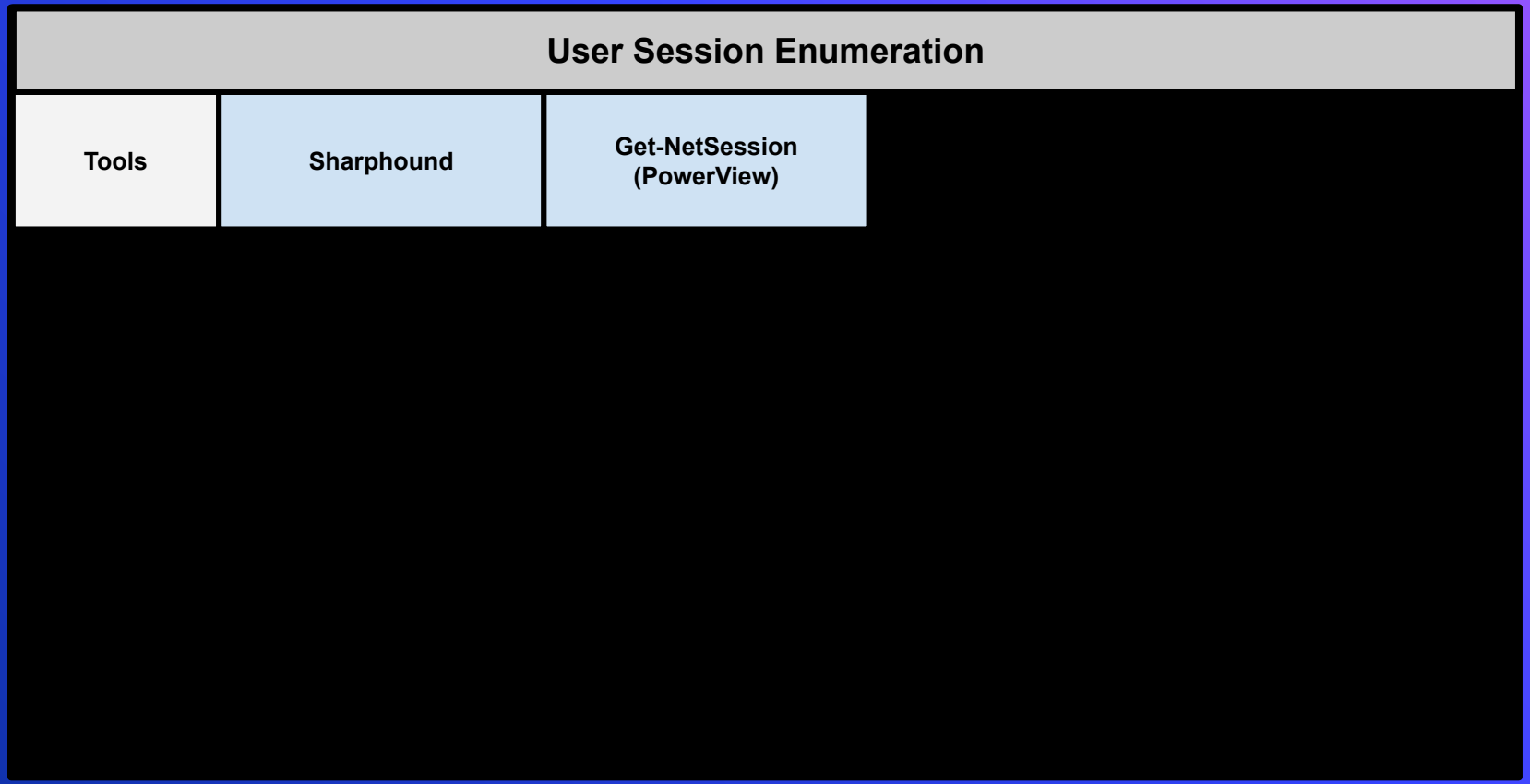
github.com/Neo23x0/sigma/...

win_hack_bloodhound.yml ● | apt_wocao.yml | win_susp_rc4_kerberos.yml | johns-rule.yml | lnx_susp_failed_lc

```yaml
title: Bloodhound Hack Tool
id: f376c8a7-a2d0-4ddc-aa0c-16c17236d962
description: Detects command line parameters used by Bloodhound hack tool
author: Florian Roth
references:
    - https://github.com/BloodHoundAD/BloodHound
date: 2019/12/20
tags:
    - attack.discovery
    - attack.t1087
logsource:
    category: process_creation
    product: windows
detection:
    selection:
        CommandLine|contains: ' -CollectionMethod All '
    condition: selection
falsepositives:
    - Other programs that use these command line option and accepts an 'All' parameter
level: high
```

6:38 AM · Dec 20, 2019 · TweetDeck

153 Retweets **401** Likes


User Session Enumeration
Tools
Sharphound
Get-NetSession
(PowerView)

32

Functions

- Are there managed (ex. .NET) APIs that we should consider?
- What unmanaged function(s) is/are used by the tool(s) that we previously identified?
- What DLL(s) export these functions?
- Are there alternative functions that can be called?
- Are there any native, undocumented, or underlying functions that are relied upon?

Sharphound - NetSessionEnum

```
[DllImport("NetAPI32.dll", SetLastError = true)]
private static extern int NetSessionEnum(
    [MarshalAs(UnmanagedType.LPWStr)] string ServerName,
    [MarshalAs(UnmanagedType.LPWStr)] string UncClientName,
    [MarshalAs(UnmanagedType.LPWStr)] string UserName,
    int Level,
    out IntPtr bufptr,
    int prefmaxlen,
    out int entriesread,
    out int totalentries,
    ref IntPtr resume_handle);

[StructLayout(LayoutKind.Sequential)]
public struct SESSION_INFO_10
{
    [MarshalAs(UnmanagedType.LPWStr)]
    public string sesi10_cname;
    [MarshalAs(UnmanagedType.LPWStr)]
    public string sesi10_username;
    public uint sesi10_time;
    public uint sesi10_idle_time;
}
```

Get-NetSession - NetSessionEnum

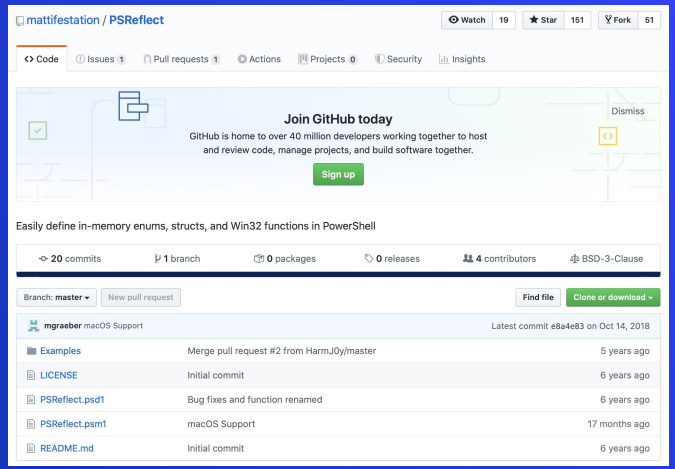
```
$Mod = New-InMemoryModule -ModuleName Win32

# all of the Win32 API functions we need
$FunctionDefinitions = @(
    (func netapi32 NetShareEnum ([Int]) @([String], [Int], [IntPtr].MakeByRefType(), [Int], [Int32].MakeByRefType(), [Int32].M
    (func netapi32 NetWkstaUserEnum ([Int]) @([String], [Int], [IntPtr].MakeByRefType(), [Int], [Int32].MakeByRefType(), [Int3
    (func netapi32 NetSessionEnum ([Int]) @([String], [String], [String], [Int], [IntPtr].MakeByRefType(), [Int], [Int32].Make
    (func netapi32 NetLocalGroupGetMembers ([Int]) @([String], [String], [Int], [IntPtr].MakeByRefType(), [Int], [Int32].MakeB
    (func netapi32 DsGetSiteName ([Int]) @([String], [IntPtr].MakeByRefType())),
    (func netapi32 DsEnumerateDomainTrusts ([Int]) @([String], [UInt32], [IntPtr].MakeByRefType(), [IntPtr].MakeByRefType())),
    (func netapi32 NetApiBufferFree ([Int]) @([IntPtr])),
    (func advapi32 ConvertSidToStringSid ([Int]) @([IntPtr], [String].MakeByRefType()) -SetLastError),
    (func advapi32 OpenSCManagerW ([IntPtr]) @([String], [String], [Int]) -SetLastError),
    (func advapi32 CloseServiceHandle ([Int]) @([IntPtr])),
    (func wtsapi32 WTSOpenServerEx ([IntPtr]) @([String])),
    (func wtsapi32 WTSEnumerateSessionsEx ([Int]) @([IntPtr], [Int32].MakeByRefType(), [Int], [IntPtr].MakeByRefType(), [Int32
    (func wtsapi32 WTSQuerySessionInformation ([Int]) @([IntPtr], [Int], [Int], [IntPtr].MakeByRefType(), [Int32].MakeByRefTyp
    (func wtsapi32 WTSFreeMemoryEx ([Int]) @([Int32], [IntPtr], [Int32])),
    (func wtsapi32 WTSFreeMemory ([Int]) @([IntPtr])),
    (func wtsapi32 WTSCloseServer ([Int]) @([IntPtr]))
)
```

Platform Invoke and PSReflect

Platform Invoke (P/Invoke)

01/18/2019 • 7 minutes to read •

P/Invoke is a technology that allows you to access structs, callbacks, and functions in unmanaged libraries from your managed code. Most of the P/Invoke API is contained in two namespaces: `System` and `System.Runtime.InteropServices`. Using these two namespaces give you the tools to describe how you want to communicate with the native component.



NetSessionEnum function

12/05/2018 • 5 minutes to read

Provides information about sessions established on a server.

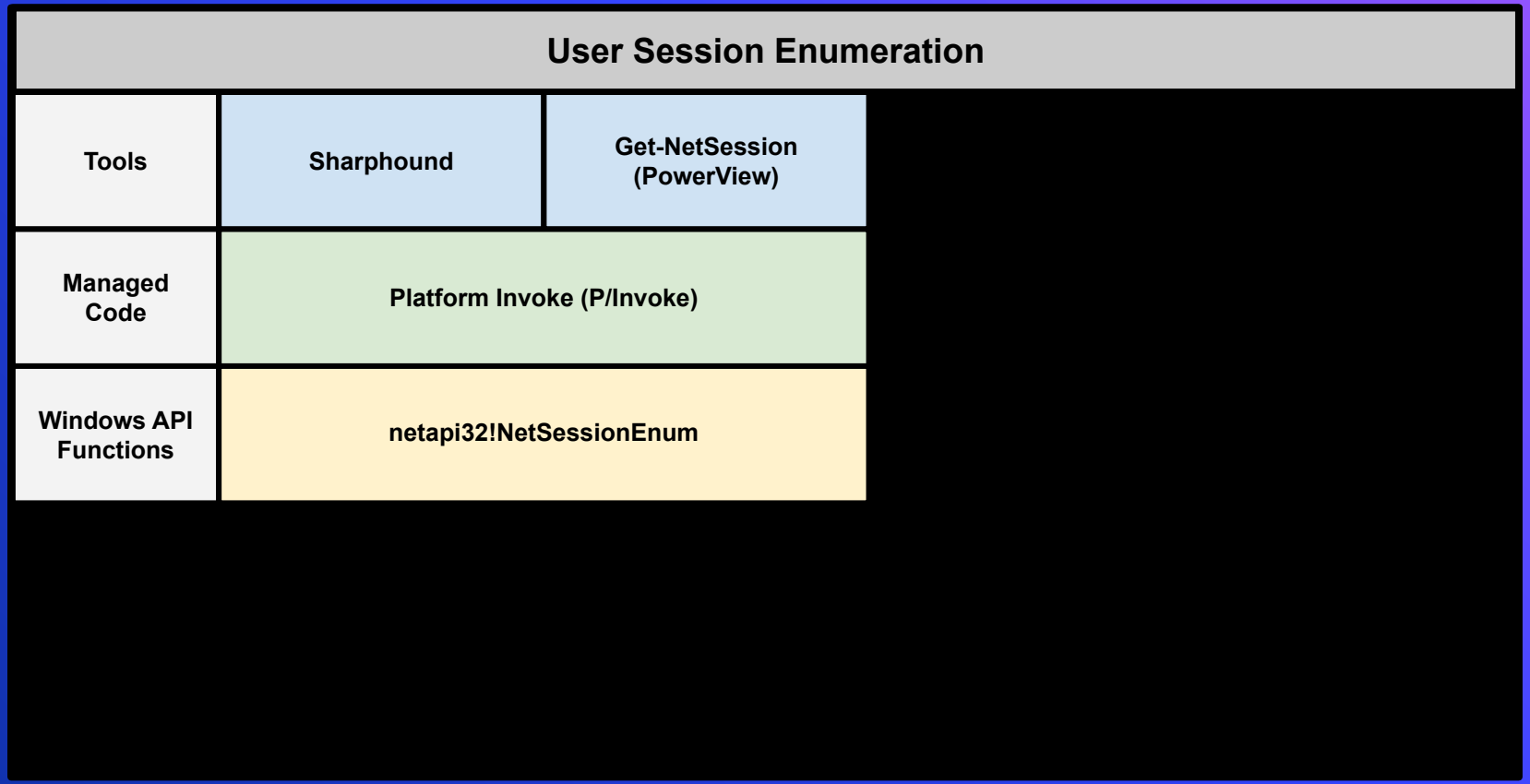
Syntax

C++ Copy

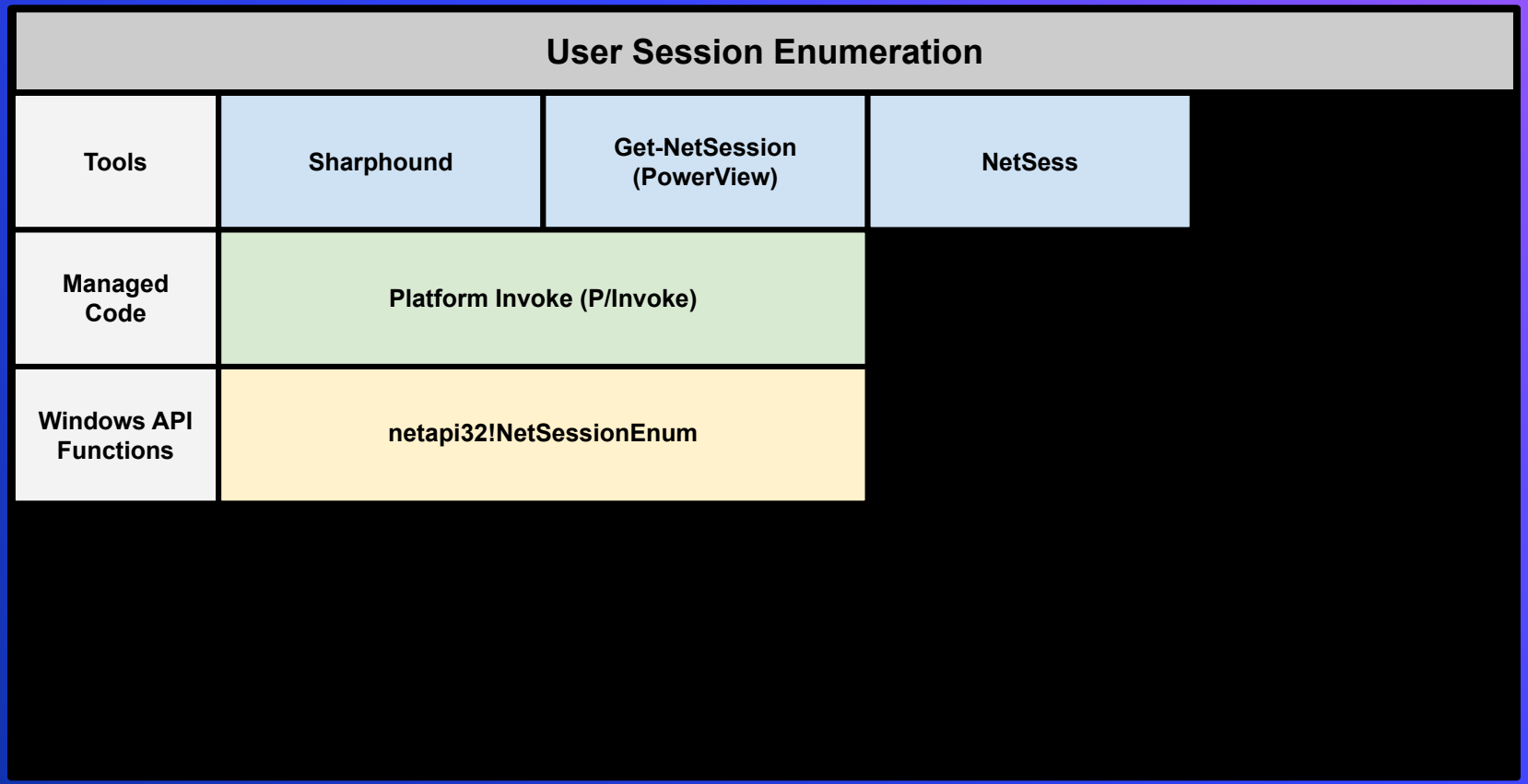
```
NET_API_STATUS NET_API_FUNCTION NetSessionEnum(
  LMSTR   servername,
  LMSTR   UncClientName,
  LMSTR   username,
  DWORD   level,
  LPBYTE  *bufptr,
  DWORD   prefmaxlen,
  LPDWORD entriesread,
  LPDWORD totalentries,
  LPDWORD resume_handle
);
```

Requirements

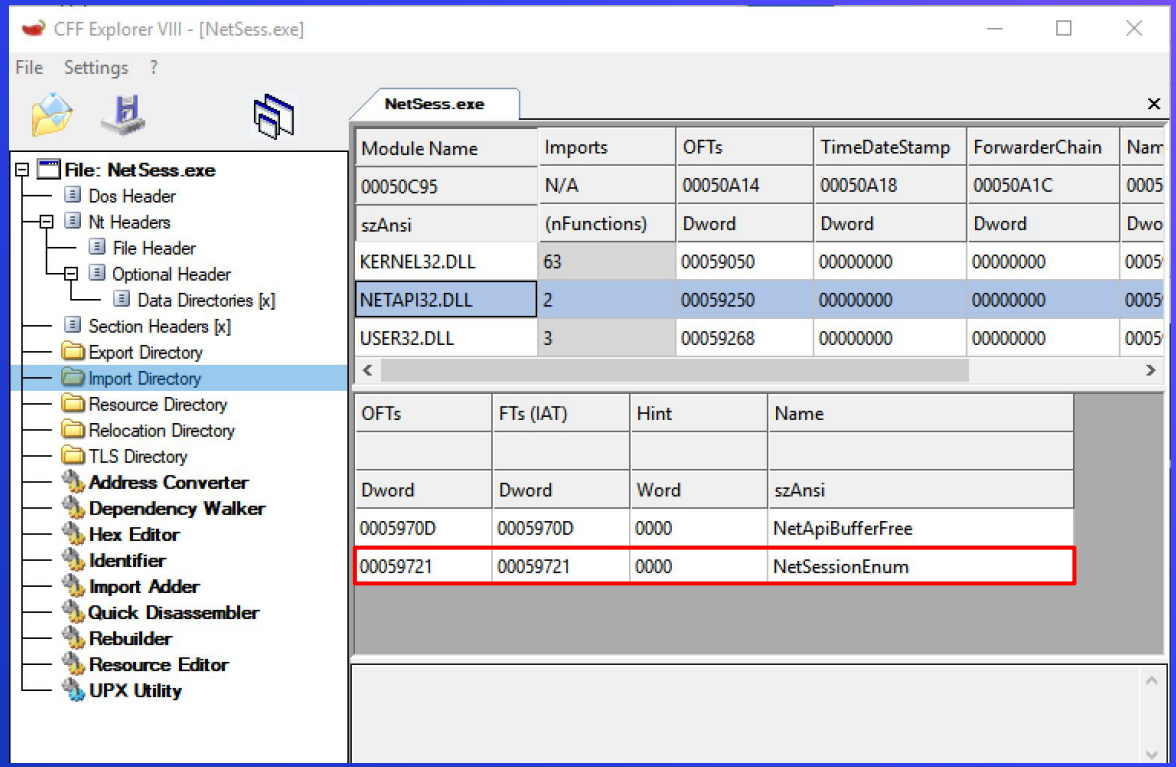
Minimum supported client	Windows XP [desktop apps only]
Minimum supported server	Windows Server 2003 [desktop apps only]
Target Platform	Windows
Header	lmshare.h (include Lm.h)
Library	Netapi32.lib
DLL	Netapi32.dll

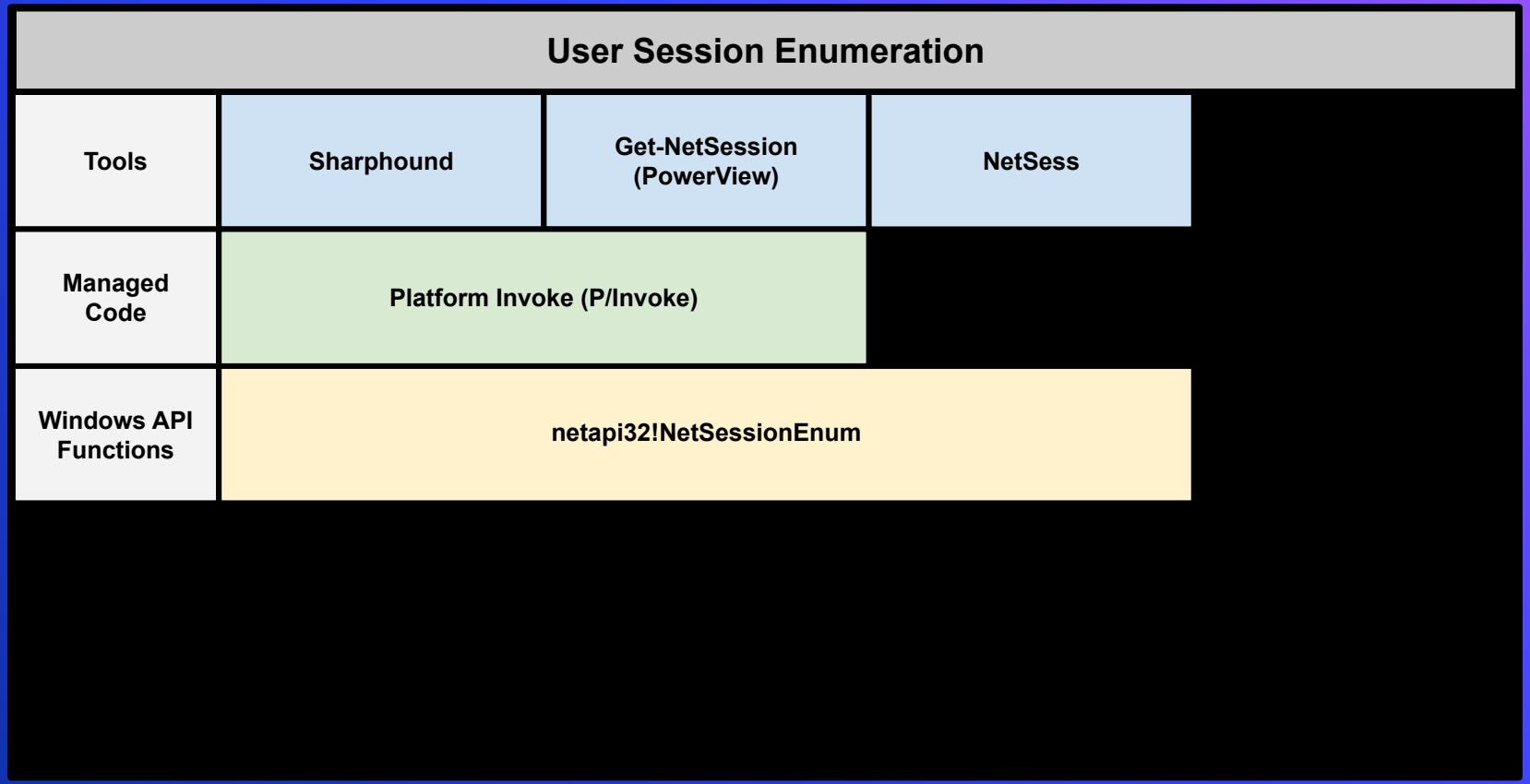

User Session Enumeration
Tools
Sharphound
Get-NetSession
(PowerView)
Managed
Code
Platform Invoke (P/Invoke)
Windows API
Functions
netapi32!NetSessionEnum

38

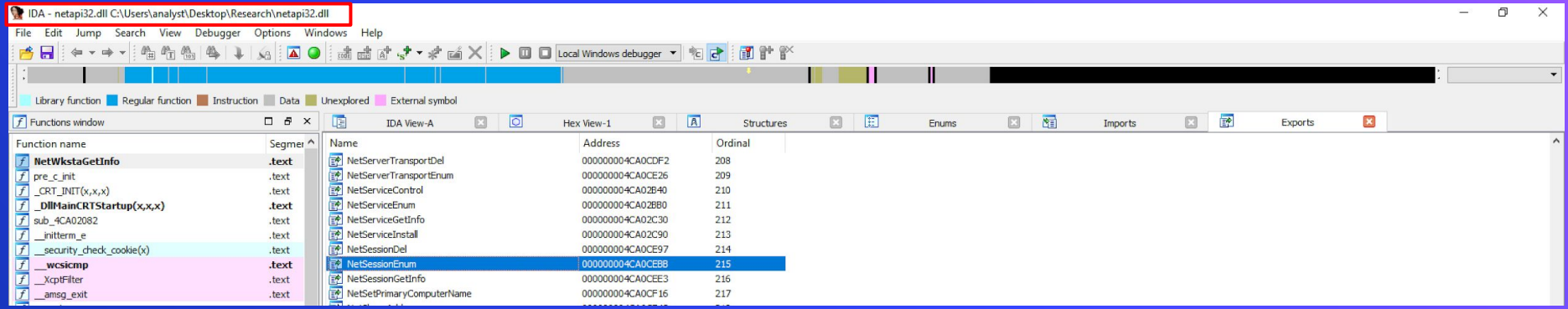

User Session Enumeration
Tools
Sharphound
Get-NetSession (PowerView)
NetSess
Managed Code
Platform Invoke (P/Invoke)
Windows API Functions
netapi32!NetSessionEnum

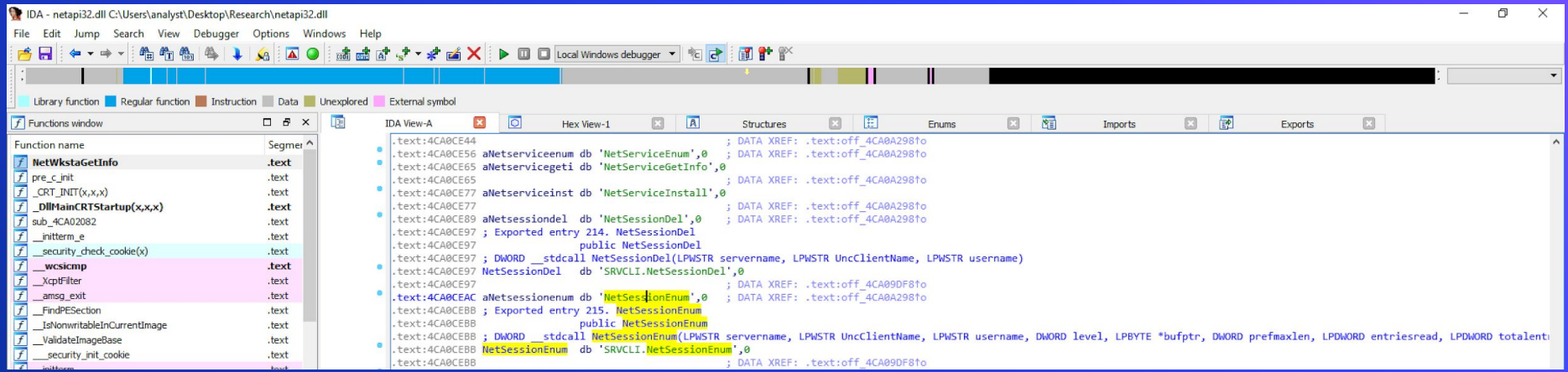
39

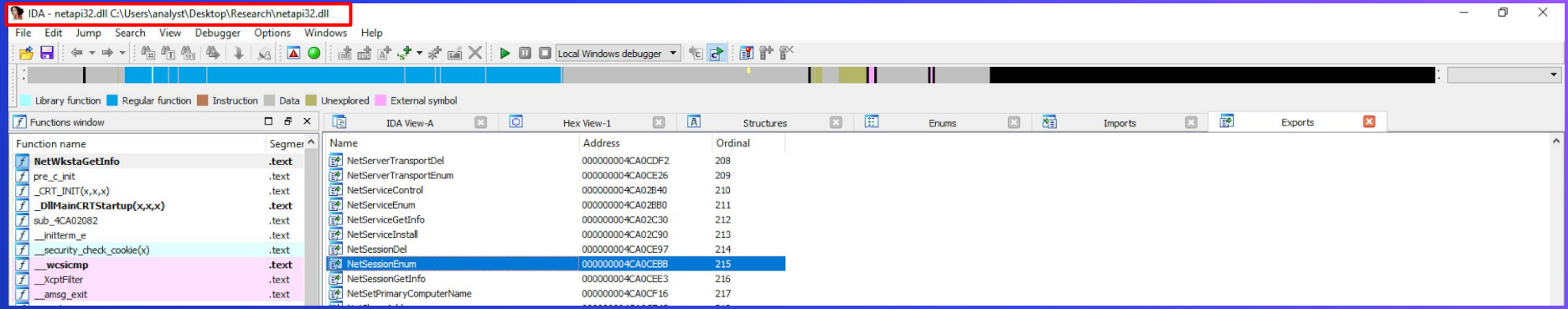

CFF Explorer VIII - [NetSess.exe]
File Settings ?
NetSess.exe
File: NetSess.exe
Dos Header
Nt Headers
File Header
Optional Header
Data Directories [x]
Section Headers [x]
Export Directory
Import Directory
Resource Directory
Relocation Directory
TLS Directory
Address Converter
Dependency Walker
Hex Editor
Identifier
Import Adder
Quick Disassembler
Rebuilder
Resource Editor
UPX Utility
Module Name | Imports | OFTs | TimeDateStamp | ForwarderChain
00050C95 | N/A | 00050A14 | 00050A18 | 00050A1C
szAnsi | (nFunctions) | Dword | Dword | Dword
KERNEL32.DLL | 63 | 00059050 | 00000000 | 00000000
NETAPI32.DLL | 2 | 00059250 | 00000000 | 00000000
USER32.DLL | 3 | 00059268 | 00000000 | 00000000
OFTs | FTs (IAT) | Hint | Name
Dword | Dword | Word | szAnsi
0005970D | 0005970D | 0000 | NetApiBufferFree
00059721 | 00059721 | 0000 | NetSessionEnum

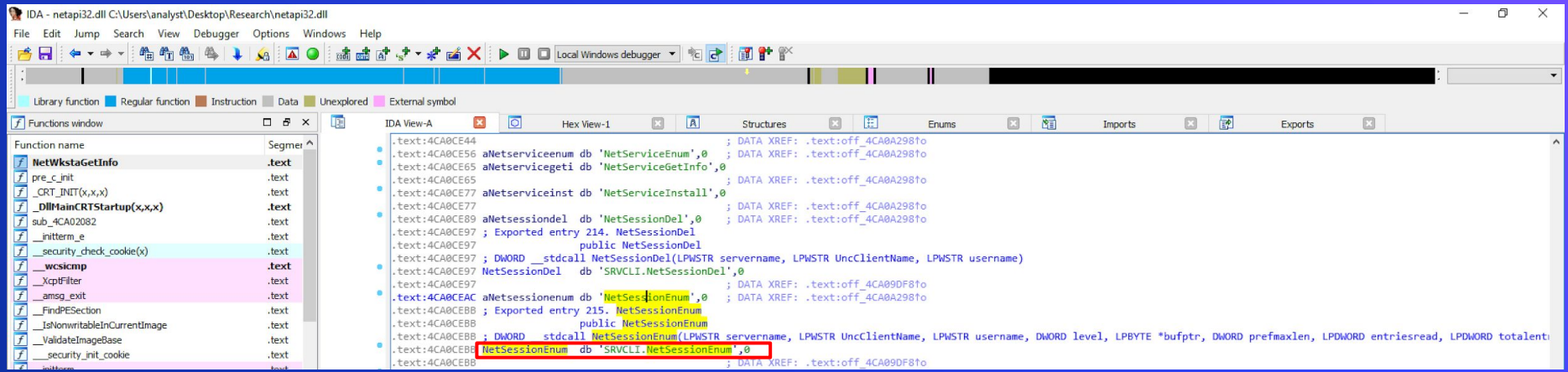

User Session Enumeration
Tools
Sharphound
Get-NetSession (PowerView)
NetSess
Managed Code
Platform Invoke (P/Invoke)
Windows API Functions
netapi32!NetSessionEnum

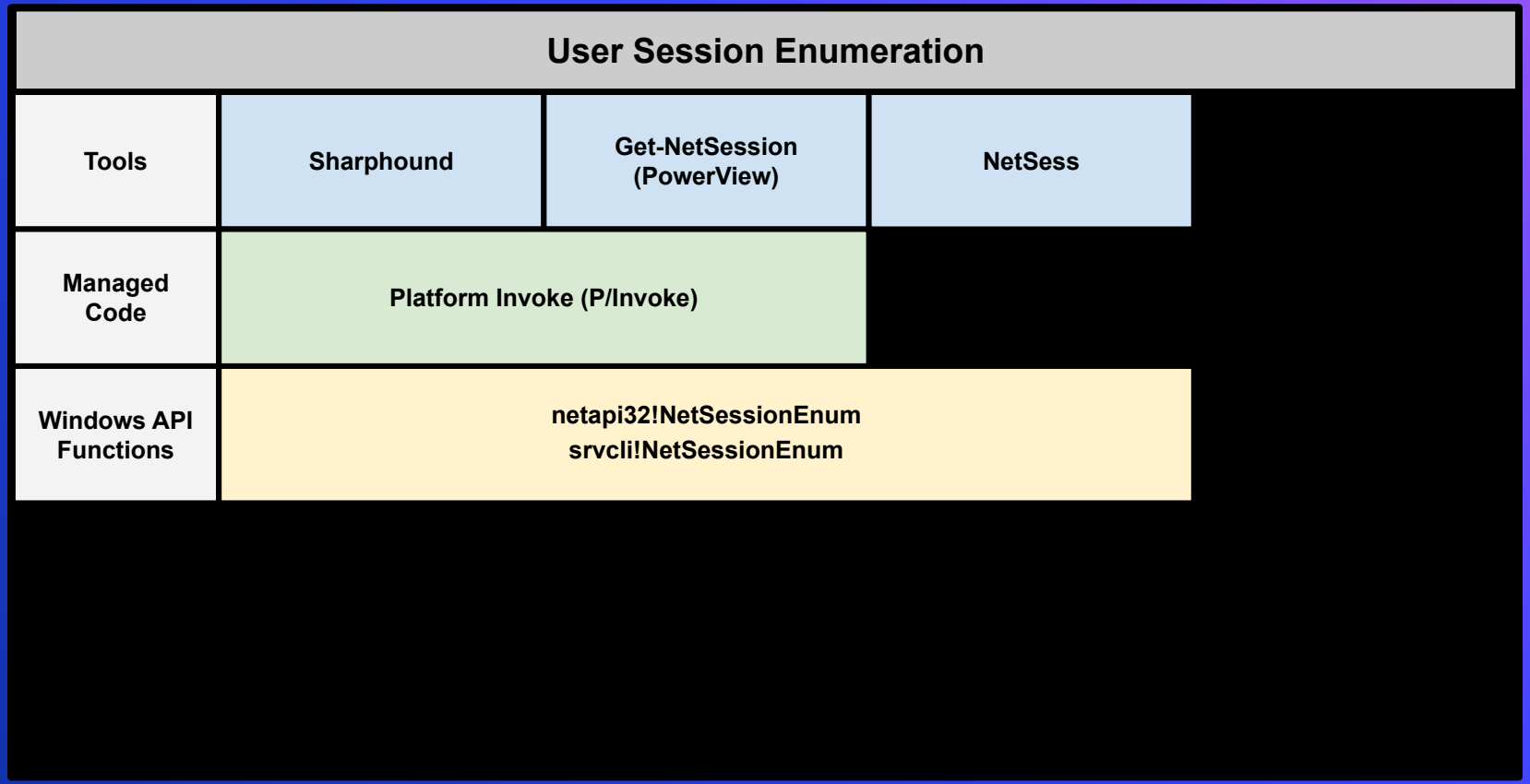
41


IDA - netapi32.dll C:\Users\analyst\Desktop\Research\netapi32.dll
File Edit Jump Search View Debugger Options Windows Help
Local Windows debugger
Library function Regular function Instruction Data Unexplored External symbol
Functions window
Function name
NetWkstaGetInfo
pre_c_init
_CRT_INIT(x,x,x)
_DllMainCRTStartup(x,x,x)
sub_4CA02082
__initterm_e
__security_check_cookie(x)
__wcsicmp
__XcptFilter
__amsg_exit
IDA View-A
Hex View-1
Structures
Enums
Imports
Exports
Name
Address
Ordinal
NetServerTransportDel 000000004CA0CDF2 208
NetServerTransportEnum 000000004CA0CE26 209
NetServiceControl 000000004CA02B40 210
NetServiceEnum 000000004CA02BB0 211
NetServiceGetInfo 000000004CA02C30 212
NetServiceInstall 000000004CA02C90 213
NetSessionDel 000000004CA0CE97 214
NetSessionEnum 000000004CA0CEBB 215
NetSessionGetInfo 000000004CA0CEE3 216
NetSetPrimaryComputerName 000000004CA0CF16 217


IDA - netapi32.dll C:\Users\analyst\Desktop\Research\netapi32.dll
File Edit Jump Search View Debugger Options Windows Help
Local Windows debugger
Library function Regular function Instruction Data Unexplored External symbol
Functions window
IDA View-A
Hex View-1
Structures
Enums
Imports
Exports
.text:4CA0CE44 ; DATA XREF: .text:off_4CA0A298↑o
.text:4CA0CE56 aNetserviceenum db 'NetServiceEnum',0 ; DATA XREF: .text:off_4CA0A298↑o
.text:4CA0CE65 aNetservicegeti db 'NetServiceGetInfo',0
.text:4CA0CE65 ; DATA XREF: .text:off_4CA0A298↑o
.text:4CA0CE77 aNetserviceinst db 'NetServiceInstall',0
.text:4CA0CE77 ; DATA XREF: .text:off_4CA0A298↑o
.text:4CA0CE89 aNetsessiondel db 'NetSessionDel',0 ; DATA XREF: .text:off_4CA0A298↑o
.text:4CA0CE97 ; Exported entry 214. NetSessionDel
.text:4CA0CE97 public NetSessionDel
.text:4CA0CE97 ; DWORD __stdcall NetSessionDel(LPWSTR servername, LPWSTR UncClientName, LPWSTR username)
.text:4CA0CE97 NetSessionDel db 'SRVCLI.NetSessionDel',0
.text:4CA0CE97 ; DATA XREF: .text:off_4CA09DF8↑o
.text:4CA0CEAC aNetsessionenum db 'NetSessionEnum',0 ; DATA XREF: .text:off_4CA0A298↑o
.text:4CA0CEBB ; Exported entry 215. NetSessionEnum
.text:4CA0CEBB public NetSessionEnum
.text:4CA0CEBB ; DWORD __stdcall NetSessionEnum(LPWSTR servername, LPWSTR UncClientName, LPWSTR username, DWORD level, LPBYTE *bufptr, DWORD prefmaxlen, LPDWORD entriesread, LPDWORD totalent
.text:4CA0CEBB NetSessionEnum db 'SRVCLI.NetSessionEnum',0
.text:4CA0CEBB ; DATA XREF: .text:off_4CA09DF8↑o


IDA - netapi32.dll C:\Users\analyst\Desktop\Research\netapi32.dll
File Edit Jump Search View Debugger Options Windows Help
Local Windows debugger
Library function Regular function Instruction Data Unexplored External symbol
Functions window
Function name
NetWkstaGetInfo
pre_c_init
_CRT_INIT(x,x,x)
_DllMainCRTStartup(x,x,x)
sub_4CA02082
__initterm_e
__security_check_cookie(x)
__wcsicmp
__XcptFilter
__amsg_exit
IDA View-A
Hex View-1
Structures
Enums
Imports
Exports
Name
Address
Ordinal
NetServerTransportDel 000000004CA0CDF2 208
NetServerTransportEnum 000000004CA0CE26 209
NetServiceControl 000000004CA02B40 210
NetServiceEnum 000000004CA02BB0 211
NetServiceGetInfo 000000004CA02C30 212
NetServiceInstall 000000004CA02C90 213
NetSessionDel 000000004CA0CE97 214
NetSessionEnum 000000004CA0CEBB 215
NetSessionGetInfo 000000004CA0CEE3 216
NetSetPrimaryComputerName 000000004CA0CF16 217


IDA - netapi32.dll C:\Users\analyst\Desktop\Research\netapi32.dll
File Edit Jump Search View Debugger Options Windows Help
Local Windows debugger
Library function Regular function Instruction Data Unexplored External symbol
Functions window
IDA View-A
Hex View-1
Structures
Enums
Imports
Exports
.text:4CA0CE44 ; DATA XREF: .text:off_4CA0A298↑o
.text:4CA0CE56 aNetserviceenum db 'NetServiceEnum',0 ; DATA XREF: .text:off_4CA0A298↑o
.text:4CA0CE65 aNetservicegeti db 'NetServiceGetInfo',0
.text:4CA0CE65 ; DATA XREF: .text:off_4CA0A298↑o
.text:4CA0CE77 aNetserviceinst db 'NetServiceInstall',0
.text:4CA0CE77 ; DATA XREF: .text:off_4CA0A298↑o
.text:4CA0CE89 aNetsessiondel db 'NetSessionDel',0 ; DATA XREF: .text:off_4CA0A298↑o
.text:4CA0CE97 ; Exported entry 214. NetSessionDel
.text:4CA0CE97 public NetSessionDel
.text:4CA0CE97 ; DWORD __stdcall NetSessionDel(LPWSTR servername, LPWSTR UncClientName, LPWSTR username)
.text:4CA0CE97 NetSessionDel db 'SRVCLI.NetSessionDel',0
.text:4CA0CE97 ; DATA XREF: .text:off_4CA09DF8↑o
.text:4CA0CEAC aNetsessionenum db 'NetSessionEnum',0 ; DATA XREF: .text:off_4CA0A298↑o
.text:4CA0CEBB ; Exported entry 215. NetSessionEnum
.text:4CA0CEBB public NetSessionEnum
.text:4CA0CEBB ; DWORD __stdcall NetSessionEnum(LPWSTR servername, LPWSTR UncClientName, LPWSTR username, DWORD level, LPBYTE *bufptr, DWORD prefmaxlen, LPDWORD entriesread, LPDWORD totalent
.text:4CA0CEBB NetSessionEnum db 'SRVCLI.NetSessionEnum',0
.text:4CA0CEBB ; DATA XREF: .text:off_4CA09DF8↑o

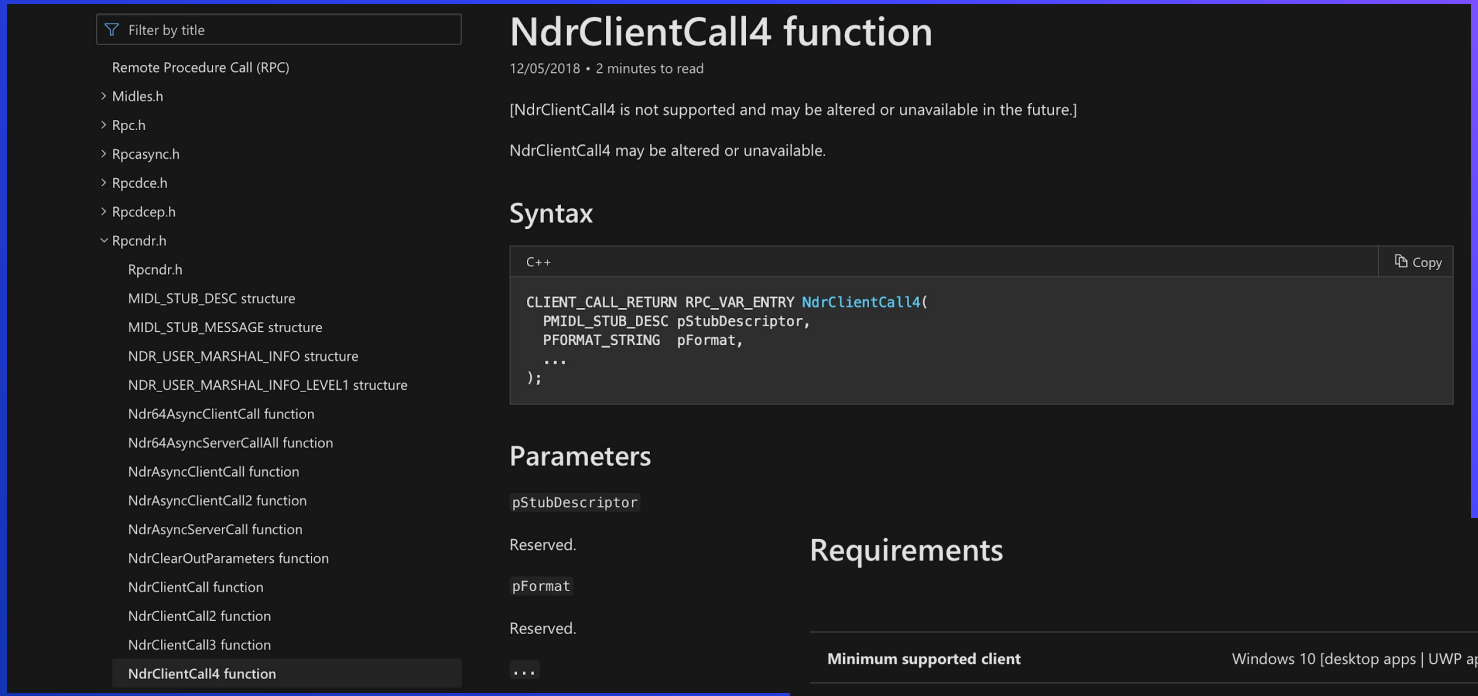

User Session Enumeration
Tools
Sharphound
Get-NetSession
(PowerView)
NetSess
Managed
Code
Platform Invoke (P/Invoke)
Windows API
Functions
netapi32!NetSessionEnum
srvcli!NetSessionEnum

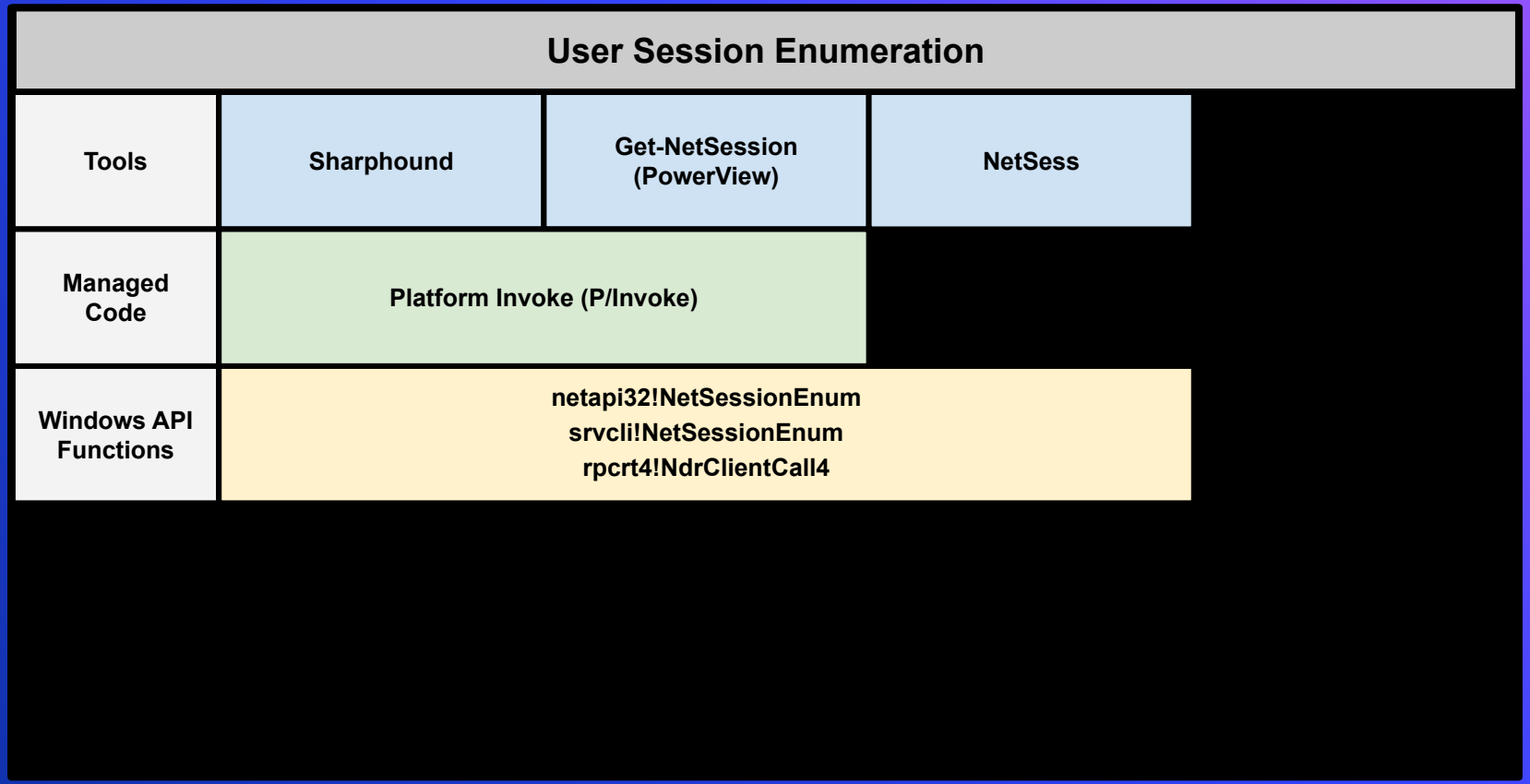
44

```
; Exported entry  41. NetSessionEnum


; Attributes: bp-based frame

; DWORD __stdcall NetSessionEnum(LPWSTR ServerName, LPWSTR UncClientName, LPWSTR username, DWORD level, LPBYTE *bufptr, DWORD prefmaxlen, LPDWORD entriesread, LPDWORD totalentries, LPDWORD resume_handle)
public NetSessionEnum
NetSessionEnum proc near

InfoStruct.Level= dword ptr -38h
InfoStruct.SessionInfo.EntriesRead= dword ptr -34h
InfoStruct.SessionInfo.Buffer= dword ptr -30h
var_2C= dword ptr -2Ch
var_28= dword ptr -28h
var_24= dword ptr -24h
var_20= dword ptr -20h
var_1C= dword ptr -1Ch
ms_exc= CPPEH_RECORD ptr -18h
ServerName= dword ptr  8
UncClientName= dword ptr  0Ch
username= dword ptr  10h
level= dword ptr  14h
bufptr= dword ptr  18h
prefmaxlen= dword ptr  1Ch
entriesread= dword ptr  20h
totalentries= dword ptr  24h
resume_handle= dword ptr  28h

push    28h
push    offset stru_1000E088
call    __SEH_prolog4
xor     esi, esi
mov     [ebp+var_2C], esi
mov     [ebp+InfoStruct.SessionInfo.Buffer], esi
lea     eax, [ebp+InfoStruct.SessionInfo.Buffer]
mov     [ebp+InfoStruct.SessionInfo.EntriesRead], eax
mov     eax, [ebp+level]
mov     [ebp+InfoStruct.Level], eax
mov     [ebp+var_20], 1
mov     [ebp+var_1C], esi
```

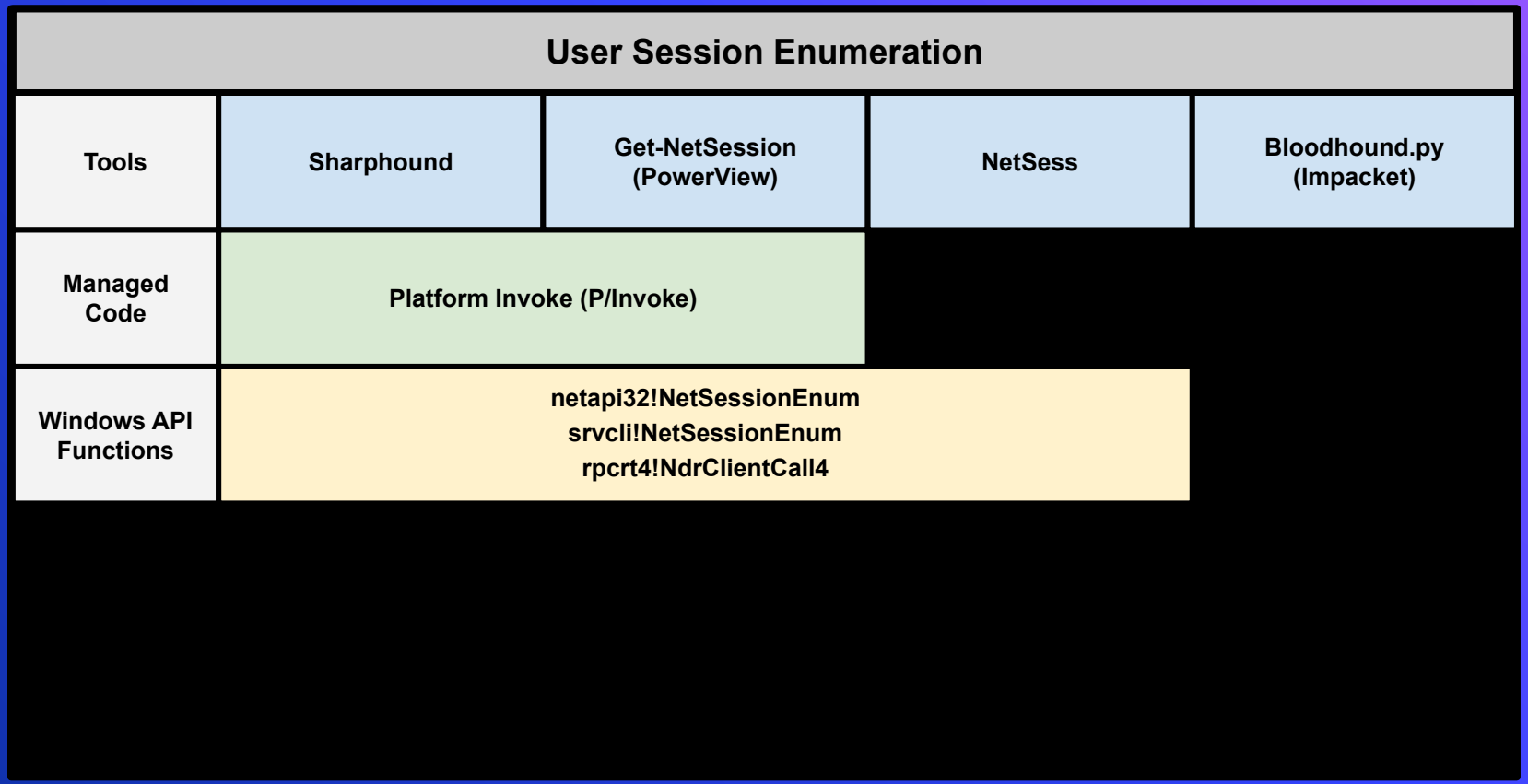
```
loc_100082EA:
mov     [ebp+ms_exc.registration.TryLevel], esi
push    [ebp+resume_handle]
push    [ebp+totalentries]
push    [ebp+prefmaxlen]
lea     eax, [ebp+InfoStruct.Level]
push    eax
push    [ebp+username]
push    [ebp+UncClientName]
push    [ebp+ServerName]
push    offset word_10001712
push    offset off_10001000
call    ds:__imp__NdrClientCall4
add     esp, 24h
mov     edx, eax
mov     [ebp+var_28], edx
mov     ecx, [ebp+var_2C]
mov     eax, [ebp+bufptr]
test    ecx, ecx
jz      short loc_10008331
```

NdrClientCall4


Filter by title
Remote Procedure Call (RPC)
Midles.h
Rpc.h
Rpcasync.h
Rpcdce.h
Rpcdcep.h
Rpcndr.h
Rpcndr.h
MIDL_STUB_DESC structure
MIDL_STUB_MESSAGE structure
NDR_USER_MARSHAL_INFO structure
NDR_USER_MARSHAL_INFO_LEVEL1 structure
Ndr64AsyncClientCall function
Ndr64AsyncServerCallAll function
NdrAsyncClientCall function
NdrAsyncClientCall2 function
NdrAsyncServerCall function
NdrClearOutParameters function
NdrClientCall function
NdrClientCall2 function
NdrClientCall3 function
NdrClientCall4 function
NdrClientCall4 function
12/05/2018 • 2 minutes to read
[NdrClientCall4 is not supported and may be altered or unavailable in the future.]
NdrClientCall4 may be altered or unavailable.
Syntax
C++
Copy
CLIENT_CALL_RETURN RPC_VAR_ENTRY NdrClientCall4(
PMIDL_STUB_DESC pStubDescriptor,
PFORMAT_STRING pFormat,
...
);
Parameters
pStubDescriptor
Reserved.
pFormat
Reserved.
...
Requirements
Minimum supported client
Windows 10 [desktop apps | UWP apps]
Minimum supported server
Windows Server 2016 [desktop apps | UWP apps]
Target Platform
Windows
Header
rpcndr.h (include Rpc.h)
Library
RpcRT4.lib
DLL
RpcRT4.dll

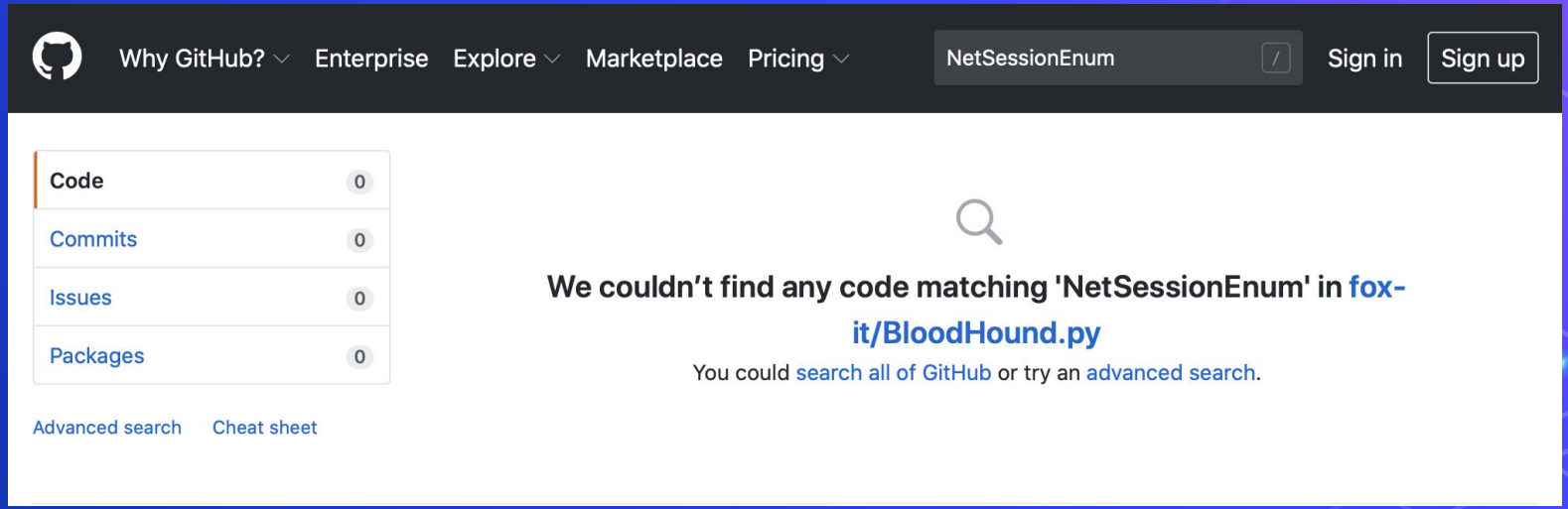

User Session Enumeration
Tools
Sharphound
Get-NetSession (PowerView)
NetSess
Managed Code
Platform Invoke (P/Invoke)
Windows API Functions
netapi32!NetSessionEnum
srvcli!NetSessionEnum
rpcrt4!NdrClientCall4

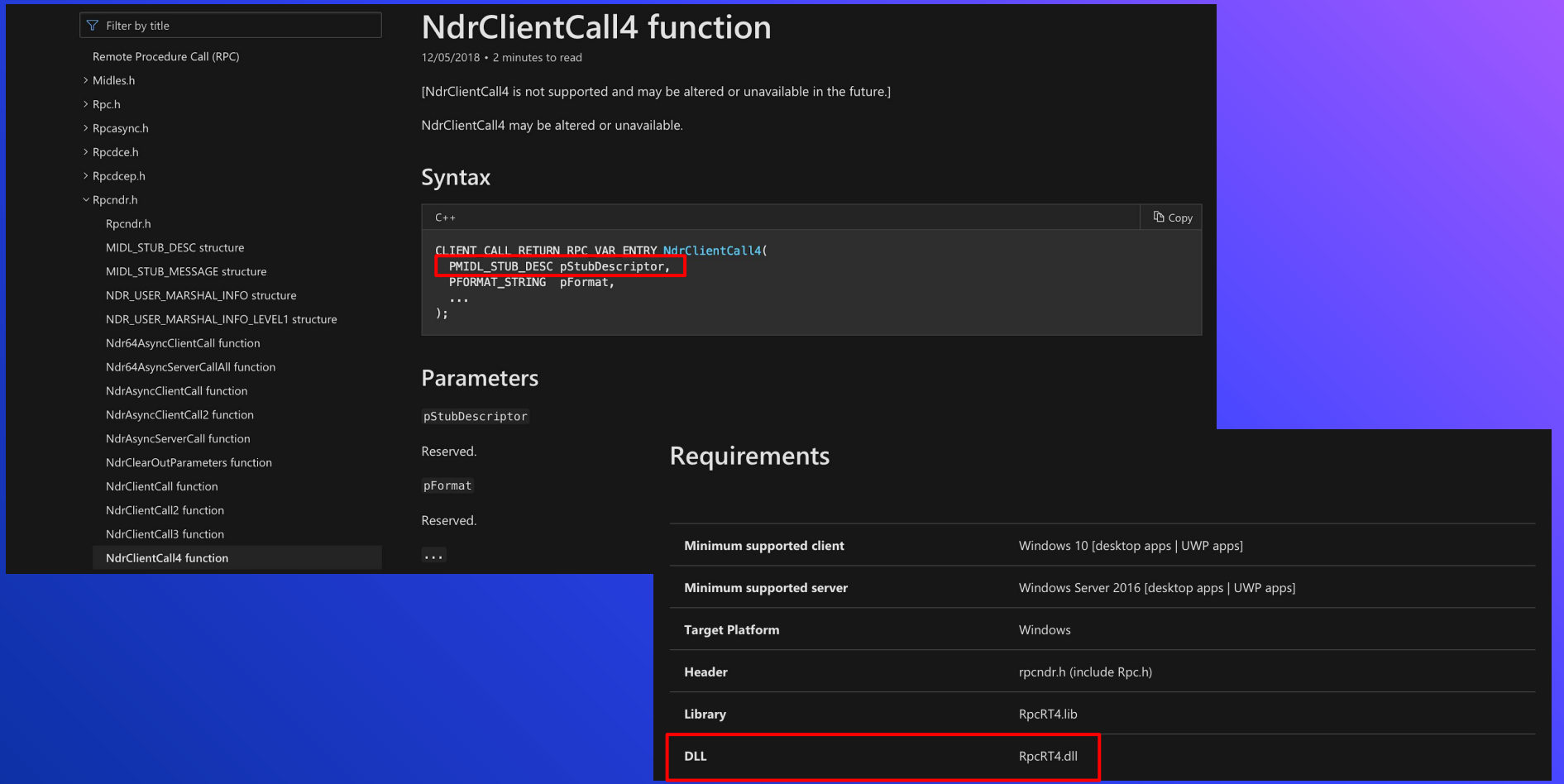
47


User Session Enumeration
Tools
Sharphound
Get-NetSession
(PowerView)
NetSess
Bloodhound.py
(Impacket)
Managed
Code
Platform Invoke (P/Invoke)
Windows API
Functions
netapi32!NetSessionEnum
srvcli!NetSessionEnum
rpcrt4!NdrClientCall4

48

No NetSessionEnum?




Filter by title
Remote Procedure Call (RPC)
Midles.h
Rpc.h
Rpcasync.h
Rpcdce.h
Rpcdcep.h
Rpcndr.h
Rpcndr.h
MIDL_STUB_DESC structure
MIDL_STUB_MESSAGE structure
NDR_USER_MARSHAL_INFO structure
NDR_USER_MARSHAL_INFO_LEVEL1 structure
Ndr64AsyncClientCall function
Ndr64AsyncServerCallAll function
NdrAsyncClientCall function
NdrAsyncClientCall2 function
NdrAsyncServerCall function
NdrClearOutParameters function
NdrClientCall function
NdrClientCall2 function
NdrClientCall3 function
NdrClientCall4 function
NdrClientCall4 function
12/05/2018 • 2 minutes to read
[NdrClientCall4 is not supported and may be altered or unavailable in the future.]
NdrClientCall4 may be altered or unavailable.
Syntax
C++
Copy
CLIENT_CALL_RETURN RPC_VAR_ENTRY NdrClientCall4(
PMIDL_STUB_DESC pStubDescriptor,
PFORMAT_STRING pFormat,
...
);
Parameters
pStubDescriptor
Reserved.
pFormat
Reserved.
...
Requirements
Minimum supported client
Windows 10 [desktop apps | UWP apps]
Minimum supported server
Windows Server 2016 [desktop apps | UWP apps]
Target Platform
Windows
Header
rpcndr.h (include Rpc.h)
Library
RpcRT4.lib
DLL
RpcRT4.dll

MIDL_STUB_DESC structure

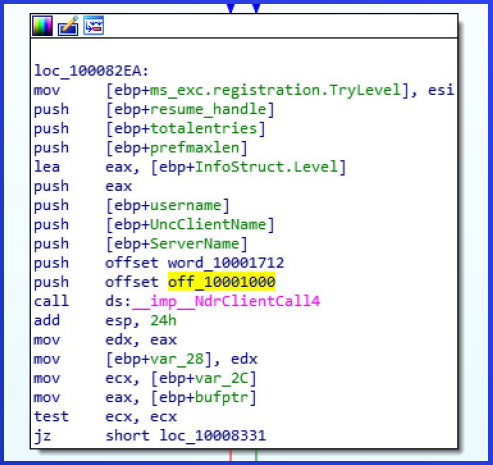
12/05/2018 • 2 minutes to read

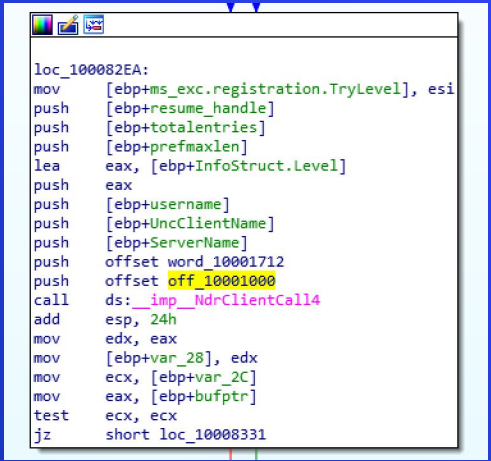
The **MIDL_STUB_DESC** structure is a MIDL-generated structure that contains information about the interface stub regarding RPC calls between the client and server.

Syntax

C++ Copy

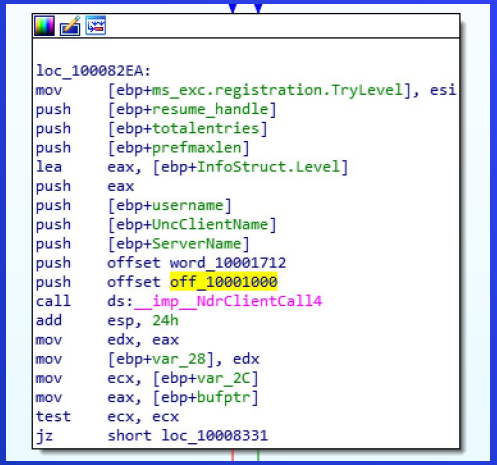
```
typedef struct _MIDL_STUB_DESC {
  void                                *RpcInterfaceInformation;
  void * )(size_t)                                 *(pfnAllocate;
  void()(void *)                                 * pfnFree;
  union {
    handle_t              *pAutoHandle;
    handle_t              *pPrimitiveHandle;
    PGENERIC_BINDING_INFO pGenericBindingInfo;
  } IMPLICIT_HANDLE_INFO;
  const NDR_RUNDOWN                   *apfnNdrRundownRoutines;
  const GENERIC_BINDING_ROUTINE_PAIR  *aGenericBindingRoutinePairs;
  const EXPR_EVAL                     *apfnExprEval;
  const XMIT_ROUTINE_QUINTUPLE        *aXmitQuintuple;
  const unsigned char                 *pFormatTypes;
  int                                 fCheckBounds;
  unsigned long                       Version;
  MALLOC_FREE_STRUCT                  *pMallocFreeStruct;
  long                                MIDLVersion;
  const COMM_FAULT_OFFSETS            *CommFaultOffsets;
  const USER_MARSHAL_ROUTINE_QUADRUPLE *aUserMarshalQuadruple;
  const NDR_NOTIFY_ROUTINE            *NotifyRoutineTable;
  ULONG_PTR                           mFlags;
  const NDR_CS_ROUTINES               *CsRoutineTables;
  void                                *ProxyServerInfo;
  const NDR_EXPR_DESC                 *pExprInfo;
} MIDL_STUB_DESC;
```


loc_100082EA:
mov [ebp+ms_exc.registration.TryLevel], esi
push [ebp+resume_handle]
push [ebp+totalentries]
push [ebp+prefmaxlen]
lea eax, [ebp+InfoStruct.Level]
push eax
push [ebp+username]
push [ebp+UncClientName]
push [ebp+ServerName]
push offset word_10001712
push offset off_10001000
call ds:__imp__NdrClientCall4
add esp, 24h
mov edx, eax
mov [ebp+var_28], edx
mov ecx, [ebp+var_2C]
mov eax, [ebp+bufptr]
test ecx, ecx
jz short loc_10008331



```
loc_100082EA:
mov     [ebp+ms_exc.registration.TryLevel], esi
push    [ebp+resume_handle]
push    [ebp+totalentries]
push    [ebp+prefmaxlen]
lea     eax, [ebp+InfoStruct.Level]
push    eax
push    [ebp+username]
push    [ebp+UncClientName]
push    [ebp+ServerName]
push    offset word_10001712
push    offset off_10001000
call    ds:__imp__NdrClientCall4
add     esp, 24h
mov     edx, eax
mov     [ebp+var_28], edx
mov     ecx, [ebp+var_2C]
mov     eax, [ebp+bufptr]
test    ecx, ecx
jz      short loc_10008331
```

```
.text:10001000 off_10001000     dd offset dword_100036A0
.text:10001000                                          ; DATA XREF: HEADER:10000124↑o
.text:10001000                                          ; HEADER:100001FC↑o ...
.text:10001004                  dd offset _MIDL_user_allocate@4 ; MIDL_user_allocate(x)
.text:10001008                  dd offset _MIDL_user_free@4 ; MIDL_user_free(x)
.text:1000100C                  dd offset _srvsvc_bhandle
.text:10001010                  dd 0
.text:10001014                  dd offset off_10001104
.text:10001018                  align 10h
.text:10001020                  dd offset word_100024EA
.text:10001024                  dd 1, 0A0000h, 0
.text:10001030                  dd 801026Eh, 3 dup(0)
.text:10001040                  dd 1, 3 dup(0)
```



```
loc_100082EA:
mov     [ebp+ms_exc.registration.TryLevel], esi
push    [ebp+resume_handle]
push    [ebp+totalentries]
push    [ebp+prefmaxlen]
lea     eax, [ebp+InfoStruct.Level]
push    eax
push    [ebp+username]
push    [ebp+UncClientName]
push    [ebp+ServerName]
push    offset word_10001712
push    offset off_10001000
call    ds:__imp__NdrClientCall4
add     esp, 24h
mov     edx, eax
mov     [ebp+var_28], edx
mov     ecx, [ebp+var_2C]
mov     eax, [ebp+bufptr]
test    ecx, ecx
jz      short loc_10008331
```

```
.text:10001000 off_10001000     dd offset dword_100036A0
.text:10001000                                          ; DATA XREF: HEADER:10000124↑o
.text:10001000                                          ; HEADER:100001FC↑o ...
.text:10001004                  dd offset _MIDL_user_allocate@4 ; MIDL_user_allocate(x)
.text:10001008                  dd offset _MIDL_user_free@4 ; MIDL_user_free(x)
.text:1000100C                  dd offset _srvsvc_bhandle
.text:10001010                  dd 0
.text:10001014                  dd offset off_10001104
.text:10001018                  align 10h
.text:10001020                  dd offset word_100024EA
.text:10001024                  dd 1, 0A0000h, 0
.text:10001030                  dd 801026Eh, 3 dup(0)
.text:10001040                  dd 1, 3 dup(0)
```

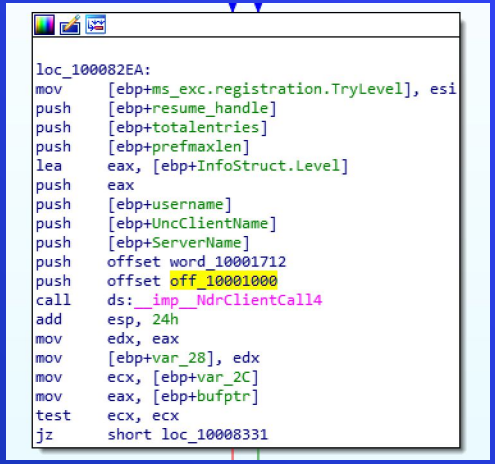
```
.text:100036A0 dword_100036A0  dd 44h, 4B324FC8h, 1D31670h, 475A7812h, 88E16EBFh, 3, 8A885D04h
.text:100036A0                                          ; DATA XREF: .text:off_10001000↑o
.text:100036A0                 dd 11C91CEBh, 8E89Fh, 6048102Bh, 2, 9 dup(0)
```

```
loc_100082EA:
mov     [ebp+ms_exc.registration.TryLevel], esi
push    [ebp+resume_handle]
push    [ebp+totalentries]
push    [ebp+prefmaxlen]
lea     eax, [ebp+InfoStruct.Level]
push    eax
push    [ebp+username]
push    [ebp+UncClientName]
push    [ebp+ServerName]
push    offset word_10001712
push    offset off_10001000
call    ds:__imp__NdrClientCall4
add     esp, 24h
mov     edx, eax
mov     [ebp+var_28], edx
mov     ecx, [ebp+var_2C]
mov     eax, [ebp+bufptr]
test    ecx, ecx
jz      short loc_10008331
```

```
.text:10001000 off_10001000     dd offset dword_100036A0
.text:10001000                                          ; DATA XREF: HEADER:10000124↑o
.text:10001000                                          ; HEADER:100001FC↑o ...
.text:10001004                  dd offset _MIDL_user_allocate@4 ; MIDL_user_allocate(x)
.text:10001008                  dd offset _MIDL_user_free@4 ; MIDL_user_free(x)
.text:1000100C                  dd offset _srvsvc_bhandle
.text:10001010                  dd 0
.text:10001014                  dd offset off_10001104
.text:10001018                  align 10h
.text:10001020                  dd offset word_100024EA
.text:10001024                  dd 1, 0A0000h, 0
.text:10001030                  dd 801026Eh, 3 dup(0)
.text:10001040                  dd 1, 3 dup(0)
```

```
.text:100036A0 dword_100036A0  dd 44h, 4B324FC8h, 1D31670h, 475A7812h, 88E16EBFh, 3, 8A885D04h
.text:100036A0                                          ; DATA XREF: .text:off_10001000↑o
.text:100036A0                 dd 11C91CEBh, 8E89Fh, 6048102Bh, 2, 9 dup(0)
```

```
100036A0  44 00 00 00 C8 4F 32 4B  70 16 D3 01 12 78 5A 47  D...ÈO2Kp.Ó..xZG
100036B0  BF 6E E1 88 03 00 00 00  04 5D 88 8A EB 1C C9 11  ¿ná^.....]^Šë.É.
100036C0  9F E8 08 00 2B 10 48 60  02 00 00 00 00 00 00 00  Ÿè..+.H`........
100036D0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
100036E0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
```



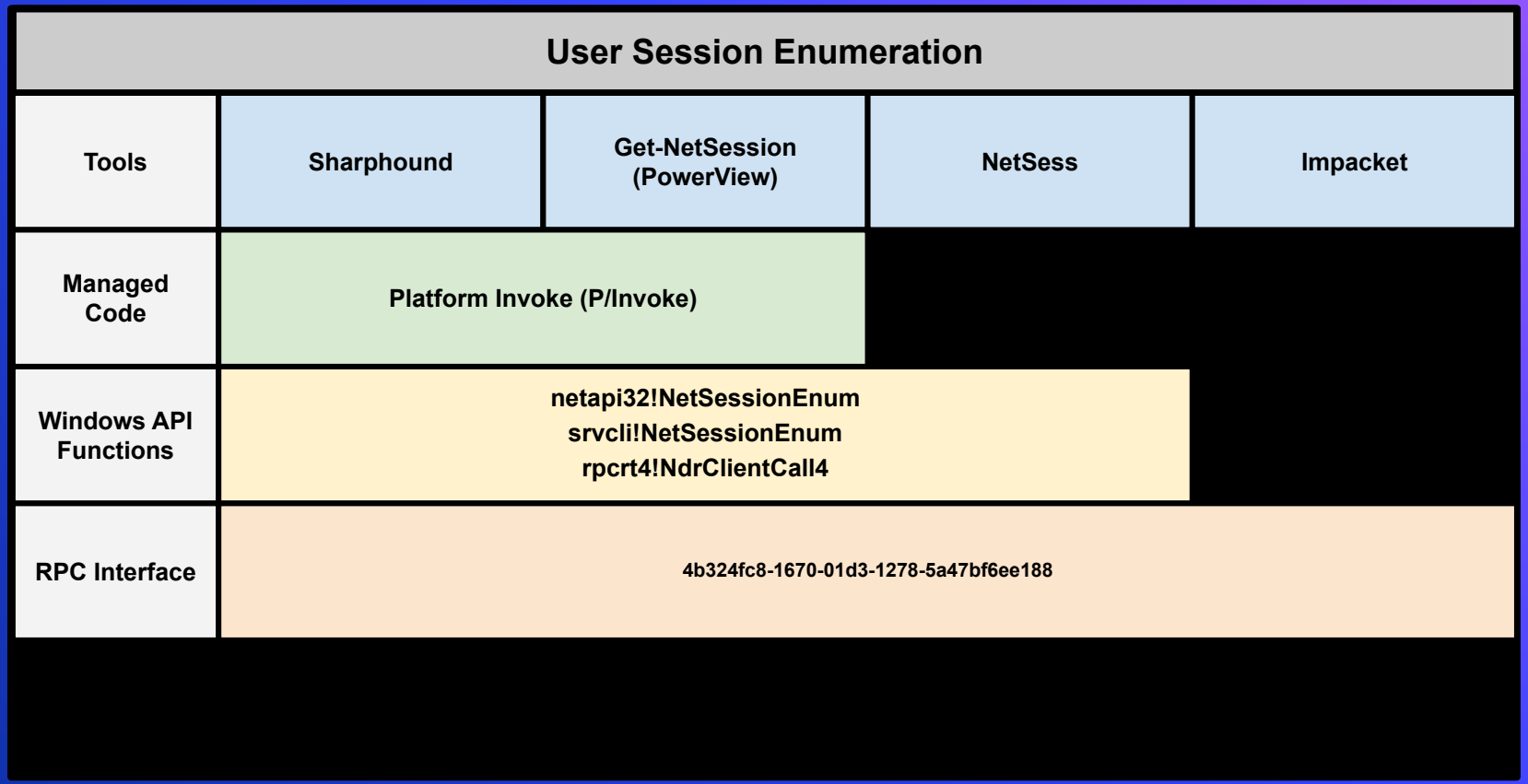
```
loc_100082EA:
mov     [ebp+ms_exc.registration.TryLevel], esi
push    [ebp+resume_handle]
push    [ebp+totalentries]
push    [ebp+prefmaxlen]
lea     eax, [ebp+InfoStruct.Level]
push    eax
push    [ebp+username]
push    [ebp+UncClientName]
push    [ebp+ServerName]
push    offset word_10001712
push    offset off_10001000
call    ds:__imp__NdrClientCall4
add     esp, 24h
mov     edx, eax
mov     [ebp+var_28], edx
mov     ecx, [ebp+var_2C]
mov     eax, [ebp+bufptr]
test    ecx, ecx
jz      short loc_10008331
```

```
.text:10001000 off_10001000     dd offset dword_100036A0
.text:10001000                                          ; DATA XREF: HEADER:10000124↑o
.text:10001000                                          ; HEADER:100001FC↑o ...
.text:10001004                  dd offset _MIDL_user_allocate@4 ; MIDL_user_allocate(x)
.text:10001008                  dd offset _MIDL_user_free@4 ; MIDL_user_free(x)
.text:1000100C                  dd offset _srvsvc_bhandle
.text:10001010                  dd 0
.text:10001014                  dd offset off_10001104
.text:10001018                  align 10h
.text:10001020                  dd offset word_100024EA
.text:10001024                  dd 1, 0A0000h, 0
.text:10001030                  dd 801026Eh, 3 dup(0)
.text:10001040                  dd 1, 3 dup(0)
```

```
.text:100036A0 dword_100036A0  dd 44h, 4B324FC8h, 1D31670h, 475A7812h, 88E16EBFh, 3, 8A885D04h
.text:100036A0                                          ; DATA XREF: .text:off_10001000↑o
.text:100036A0                 dd 11C91CEBh, 8E89Fh, 6048102Bh, 2, 9 dup(0)
```

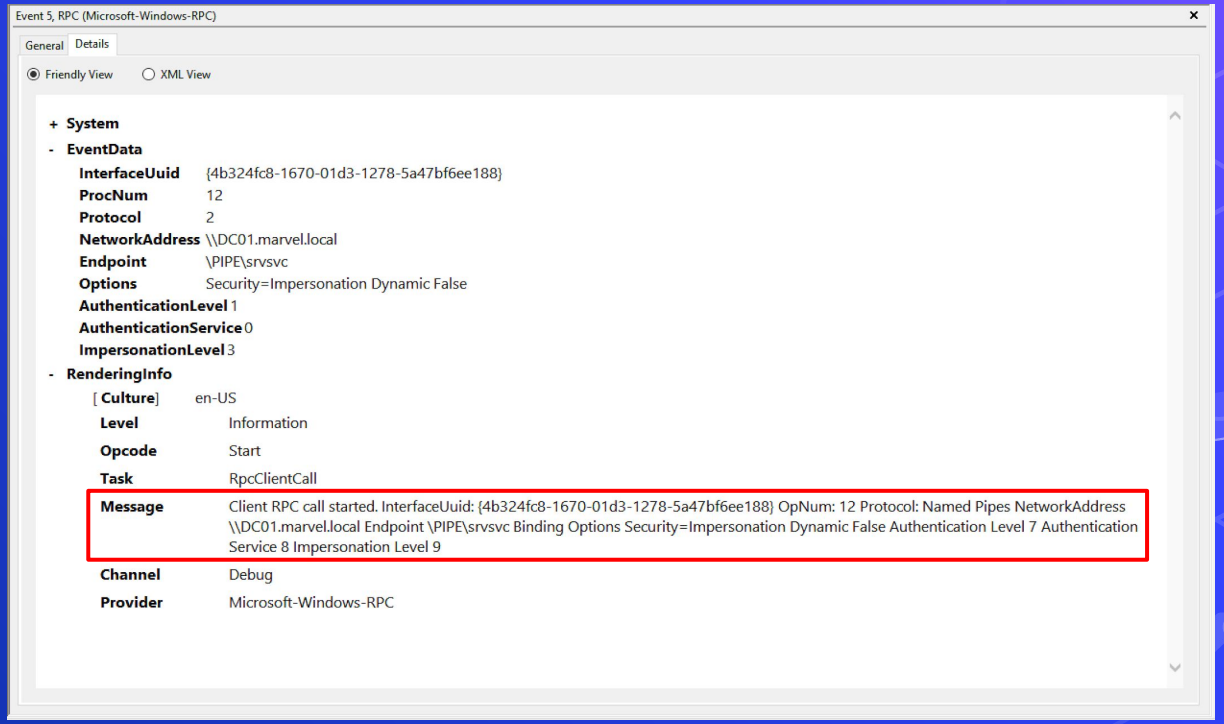
```
100036A0  44 00 00 00 C8 4F 32 4B  70 16 D3 01 12 78 5A 47  D...ÈO2Kp.Ó..xZG
100036B0  BF 6E E1 88 03 00 00 00  04 5D 88 8A EB 1C C9 11  ¿ná^.....]^Šë.É.
100036C0  9F E8 08 00 2B 10 48 60  02 00 00 00 00 00 00 00  Ÿè..+.H`........
100036D0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
100036E0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
```

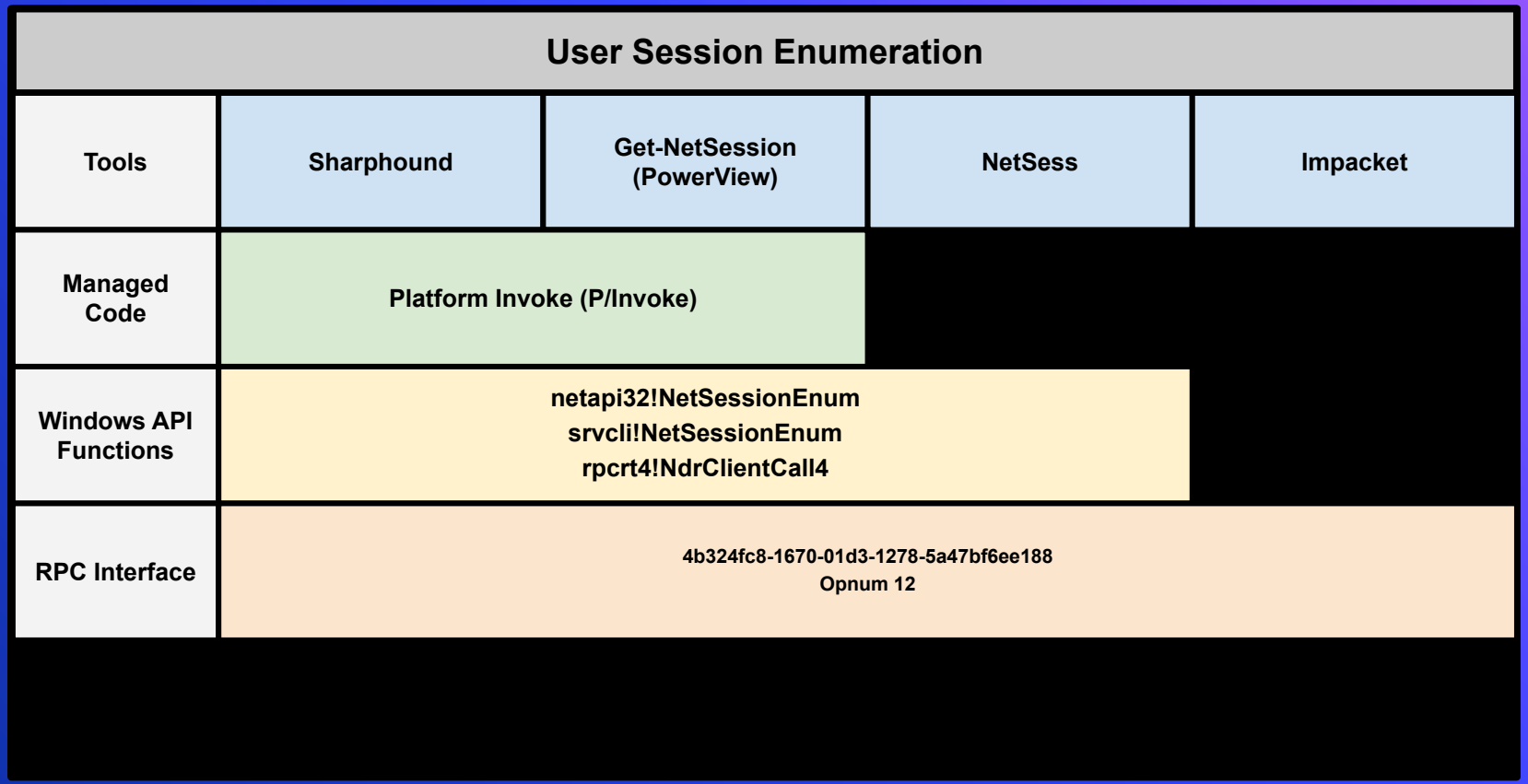
```
PS C:\Windows\system32> [Guid][Byte[]]@(0xC8,0x4F,0x32,0x4B,0x70,0x16,0xD3,0x01,0x12,0x78,0x5A,0x47,0xBF,0x6E,0xE,0x88)

Guid
----
4b324fc8-1670-01d3-1278-5a47bf6e0e88
```


User Session Enumeration
Tools
Sharphound
Get-NetSession (PowerView)
NetSess
Impacket
Managed Code
Platform Invoke (P/Invoke)
Windows API Functions
netapi32!NetSessionEnum
srvcli!NetSessionEnum
rpcrt4!NdrClientCall4
RPC Interface
4b324fc8-1670-01d3-1278-5a47bf6ee188

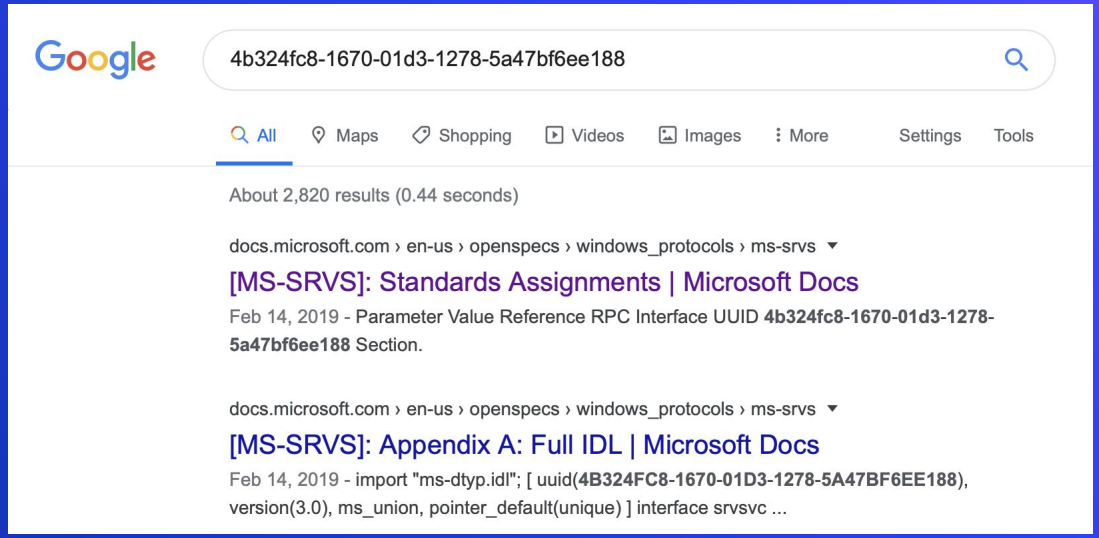
57

ETW - Microsoft-Windows-RPC Provider




User Session Enumeration
Tools
Sharphound
Get-NetSession
(PowerView)
NetSess
Impacket
Managed
Code
Platform Invoke (P/Invoke)
Windows API
Functions
netapi32!NetSessionEnum
srvcli!NetSessionEnum
rpcrt4!NdrClientCall4
RPC Interface
4b324fc8-1670-01d3-1278-5a47bf6ee188
Opnum 12

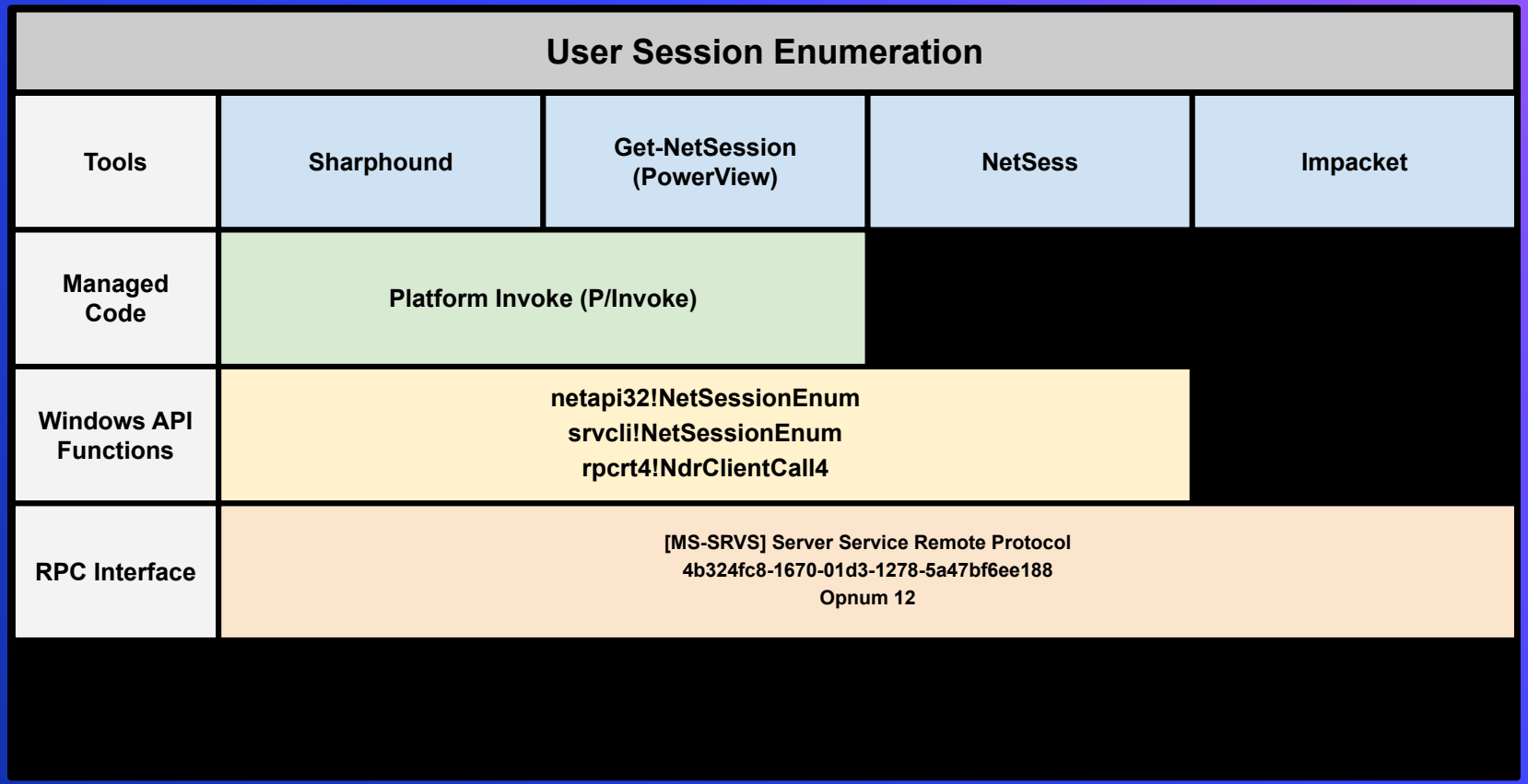
59



[MS-SRVS]: Server Service Remote Protocol

02/14/2019 • 4 minutes to read

Specifies the Server Service Remote Protocol, which remotely enables file and printer sharing and named pipe access to the server through the Server Message Block Protocol.


User Session Enumeration
Tools
Sharphound
Get-NetSession
(PowerView)
NetSess
Impacket
Managed
Code
Platform Invoke (P/Invoke)
Windows API
Functions
netapi32!NetSessionEnum
srvcli!NetSessionEnum
rpcrt4!NdrClientCall4
RPC Interface
[MS-SRVS] Server Service Remote Protocol
4b324fc8-1670-01d3-1278-5a47bf6ee188
Opnum 12

61

3.1.4.5 NetrSessionEnum (Opnum 12)

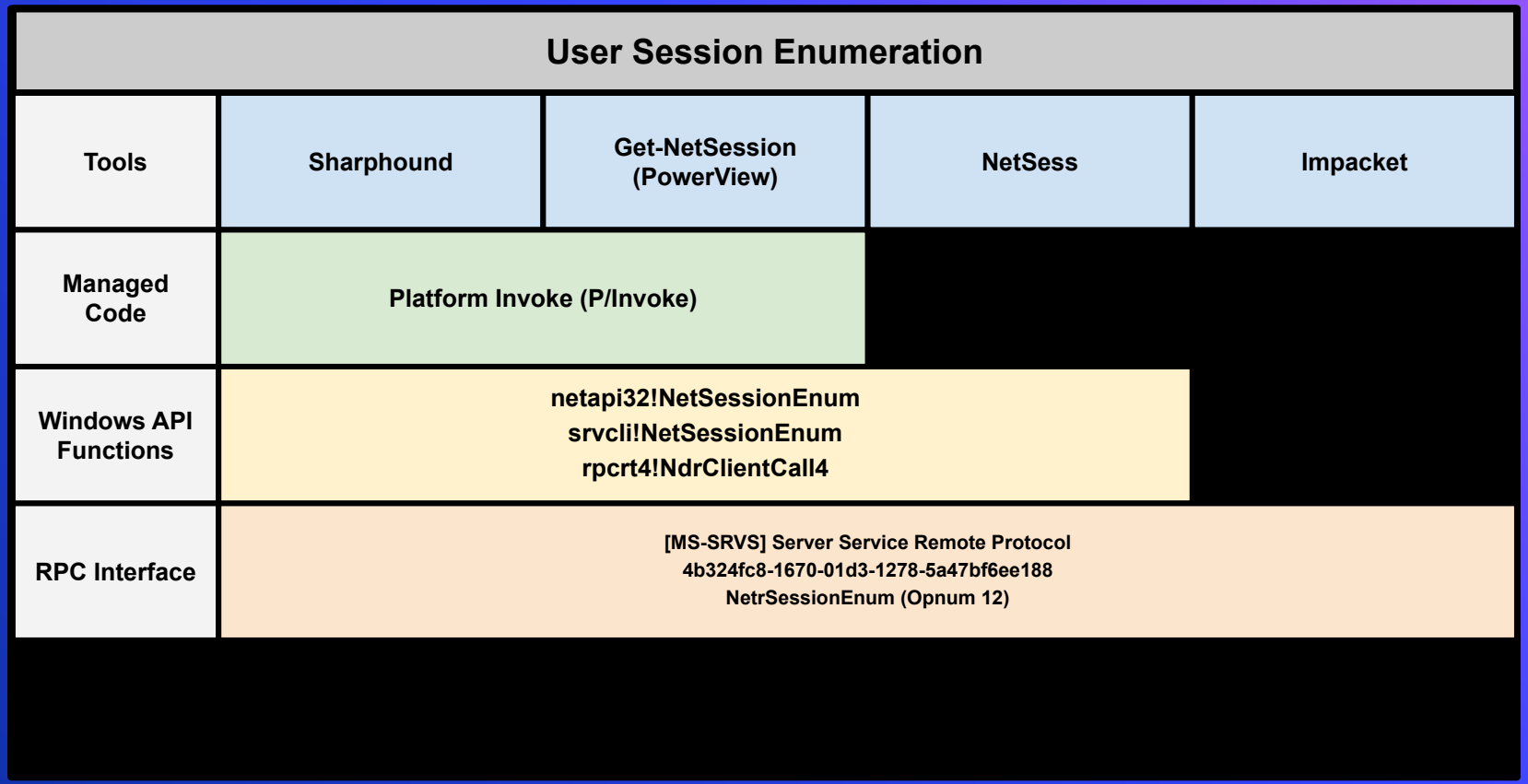
02/14/2019 • 7 minutes to read

The NetrSessionEnum method MUST return information about sessions that are established on a server or return an error code.

```
NET_API_STATUS NetrSessionEnum(
  [in, string, unique] SRVSVC_HANDLE ServerName,
  [in, string, unique] WCHAR* ClientName,
  [in, string, unique] WCHAR* UserName,
  [in, out] PSESSION_ENUM_STRUCT InfoStruct,
  [in] DWORD PreferedMaximumLength,
  [out] DWORD* TotalEntries,
  [in, out, unique] DWORD* ResumeHandle
);
```

Bloodhound.py
Impacket - NetrSessionEnum

```
322
323         if dce is None:
324             return
325
326         try:
327             resp = srvs.hNetrSessionEnum(dce, '\x00', NULL, 10)
328         except DCERPCException as e:
329             if 'rpc_s_access_denied' in str(e):
330                 logging.debug('Access denied while enumerating Sessions on %s, likely a patched OS', self.hostname)
331                 return []
332             else:
333                 raise
334         except Exception as e:
335             if str(e).find('Broken pipe') >= 0:
336                 return
337             else:
338                 raise
339
340         sessions = []
341
342         for session in resp['InfoStruct']['SessionInfo']['Level10']['Buffer']:
343             userName = session['sesi10_username'][:-1]
344             ip = session['sesi10_cname'][:-1]
345             # Strip \\ from IPs
```


User Session Enumeration
Tools
Sharphound
Get-NetSession (PowerView)
NetSess
Impacket
Managed Code
Platform Invoke (P/Invoke)
Windows API Functions
netapi32!NetSessionEnum
srvcli!NetSessionEnum
rpcrt4!NdrClientCall4
RPC Interface
[MS-SRVS] Server Service Remote Protocol
4b324fc8-1670-01d3-1278-5a47bf6ee188
NetrSessionEnum (Opnum 12)

64

Server Component for MS-SRVS

enigma0x3 / rpc.ps1

Last active 2 months ago

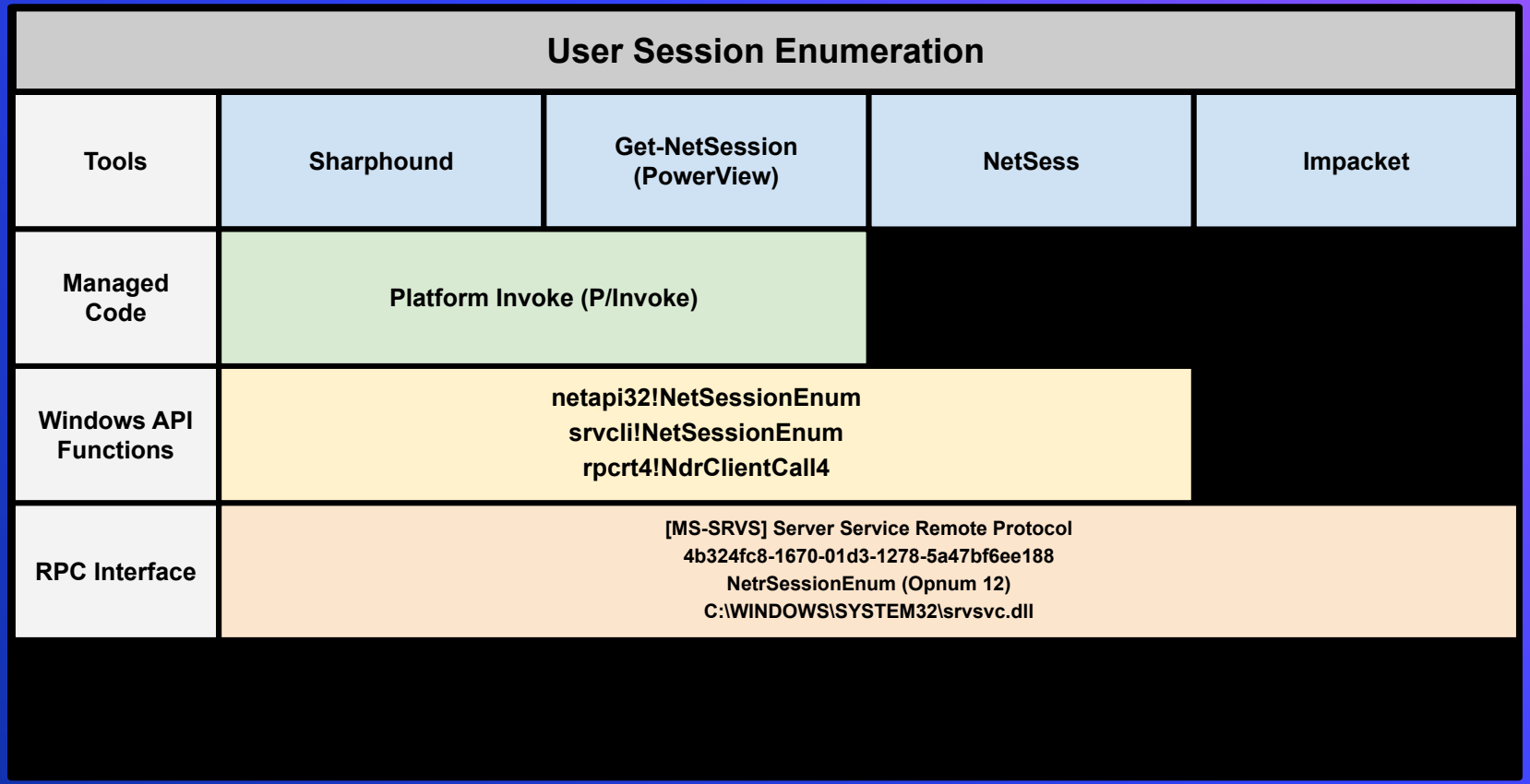
Star 0 Fork 0

Code Revisions 3 Embed <script src="https://gi Download ZIP

rpc.ps1 Raw

```
$rpc = ls C:\Windows\System32\*.exe, C:\Windows\System32\*.dll |Get-RpcServer -DbgHelpPath "C:\Program Files (x86)\Windows Kits

 foreach ($rpc1 in $rpc)
{
    $ourObject = New-Object -TypeName psobject
    $ourObject | Add-Member -MemberType NoteProperty -Name InterfaceID -Value $rpc1.InterfaceID
    $ourObject | Add-Member -MemberType NoteProperty -Name FileName -Value $rpc1.Name
    $ourObject | Add-Member -MemberType NoteProperty -Name IsRunning -Value $rpc1.IsServiceRunning
    $ourObject | Add-Member -MemberType NoteProperty -Name EndpointCount -Value $rpc1.EndpointCount
    $procs = $rpc1.Procedures.Name  | Out-String
    $ourObject | Add-Member -MemberType NoteProperty -Name Procedures -Value $procs
    $ourObject | fl | Out-file -Encoding ASCII rpc.txt -Append
}
```

```
3586 RPC  4b324fc8-1670-01d3-1278-5a47bf6ee188 (3.0) -- c:\windows\system32\srvsvc.dll
3587      0 -> NetrCharDevControl
          1 -> NetrCharDevControl
          2 -> NetrCharDevControl
          3 -> NetrCharDevControl
          4 -> NetrCharDevControl
          5 -> NetrCharDevControl
          6 -> NetrCharDevControl
          7 -> NetrCharDevControl
          8 -> NetrConnectionEnum
          9 -> NetrFileEnum
          10 -> NetrFileGetInfo
          11 -> NetrFileClose
          12 -> NetrSessionEnum
          13 -> NetrSessionDel
          14 -> NetrShareAdd
          15 -> NetrShareEnum
          16 -> NetrShareGetInfo
          17 -> NetrShareSetInfo
          18 -> NetrShareDel
          19 -> NetrShareDelSticky
3607      20 -> NetrShareCheck
3608      21 -> NetrServerGetInfo
3609      22 -> NetrServerSetInfo
3610      23 -> NetrServerDiskEnum
3611      24 -> NetrServerStatisticsGet
3612      25 -> NetrServerTransportAdd
3613      26 -> NetrServerTransportEnum
3614      27 -> NetrServerTransportDel
3615      28 -> NetrRemoteTOD
3616      29 -> I_NetrServerSetServiceBits
3617      30 -> NetprPathType
3618      31 -> NetprPathCanonicalize
3619      32 -> NetprPathCompare
3620      33 -> NetprNameValidate
3621      34 -> NetprNameCanonicalize
```


User Session Enumeration
Tools
Sharphound
Get-NetSession
(PowerView)
NetSess
Impacket
Managed
Code
Platform Invoke (P/Invoke)
Windows API
Functions
netapi32!NetSessionEnum
srvcli!NetSessionEnum
rpcrt4!NdrClientCall4
RPC Interface
[MS-SRVS] Server Service Remote Protocol
4b324fc8-1670-01d3-1278-5a47bf6ee188
NetrSessionEnum (Opnum 12)
C:\WINDOWS\SYSTEM32\srvsvc.dll

66

Network

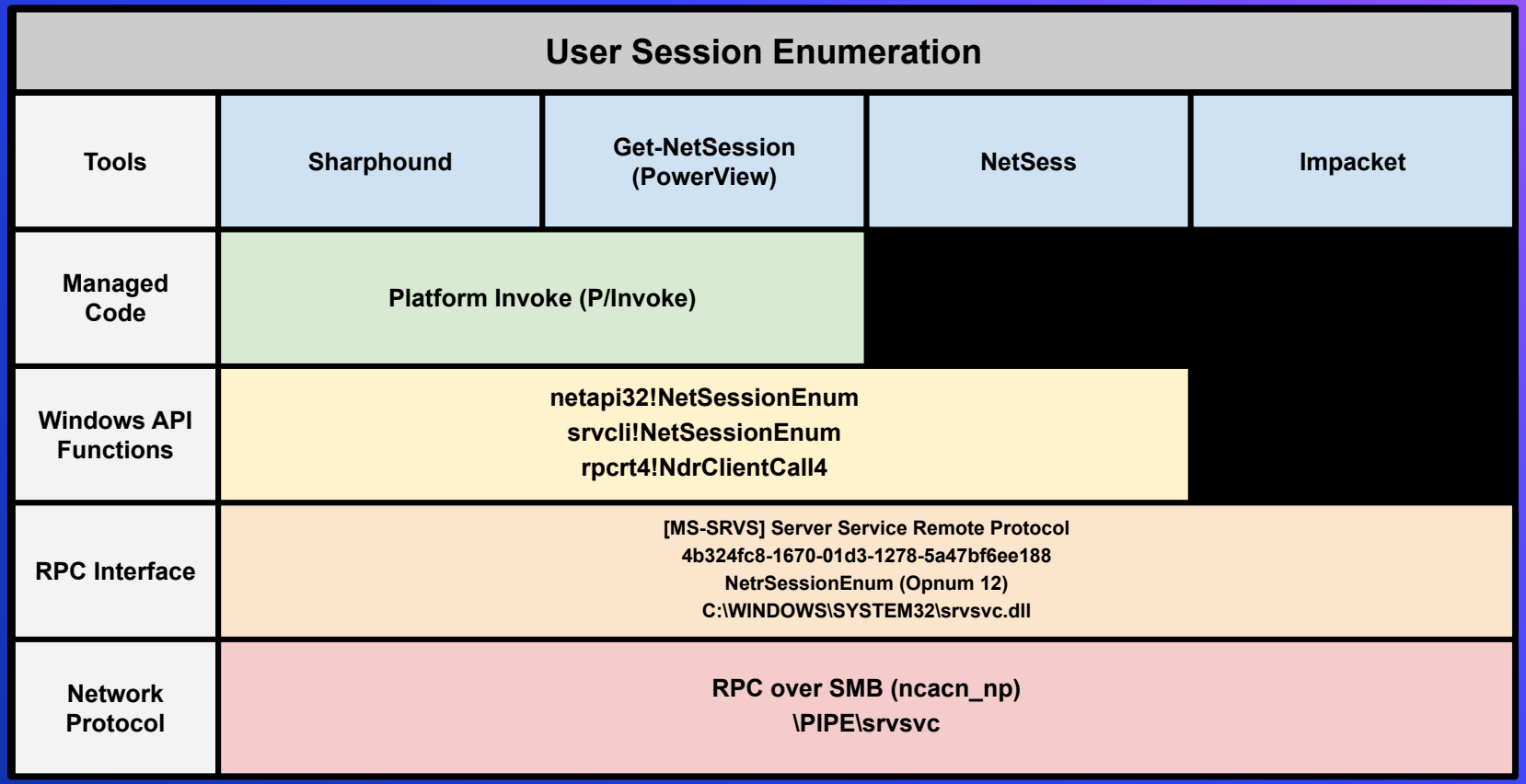
- Does this technique require network connectivity?
- What port, protocol, etc. is used?
- What specific details about the protocol can be used to differentiate this activity from other possibly benign activity?

Standards Assignment

1.9 Standards Assignments

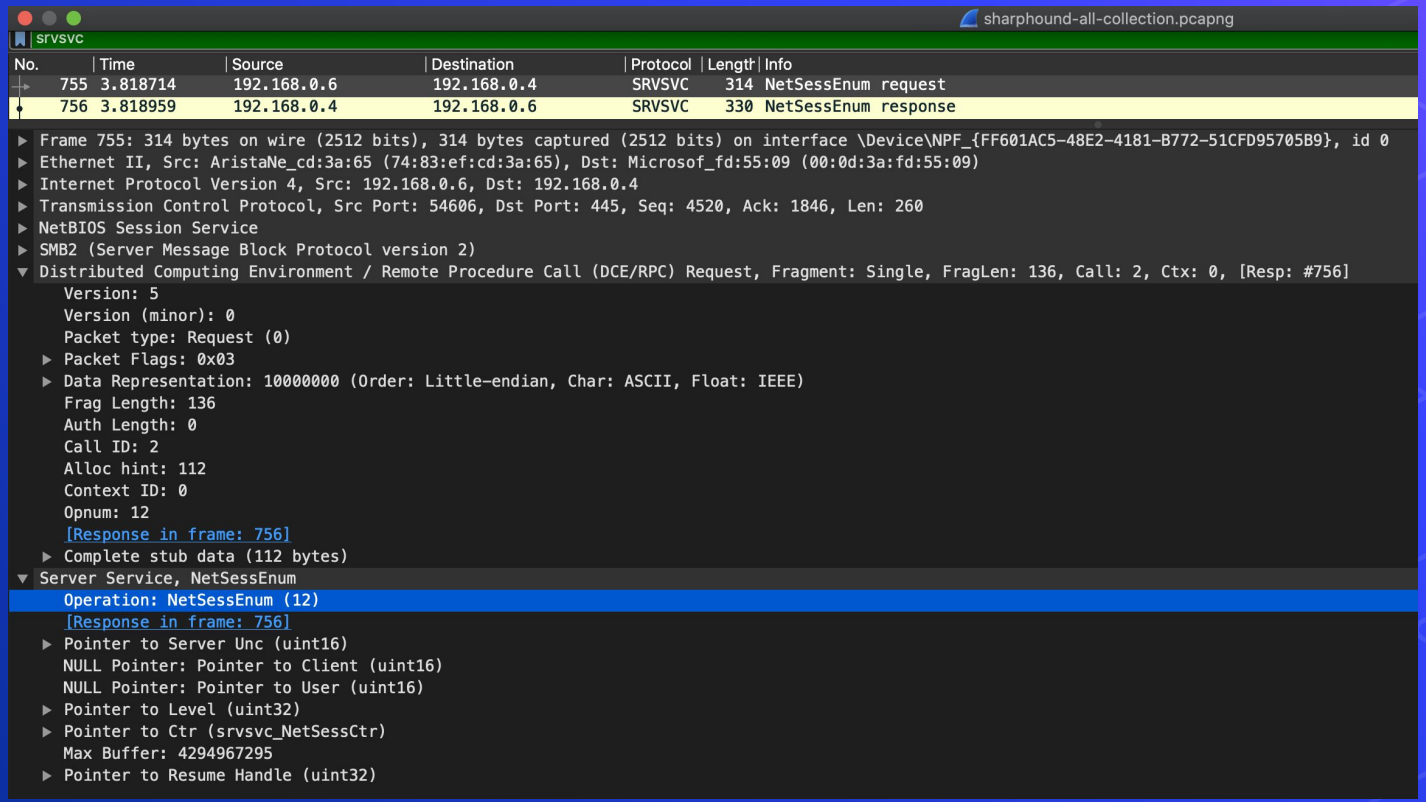
02/14/2019 • 2 minutes to read

Parameter	Value	Reference
RPC Interface UUID	4b324fc8-1670-01d3-1278-5a47bf6ee188	Section 2.1
Pipe Name	\PIPE\srvsvc	Section 2.1

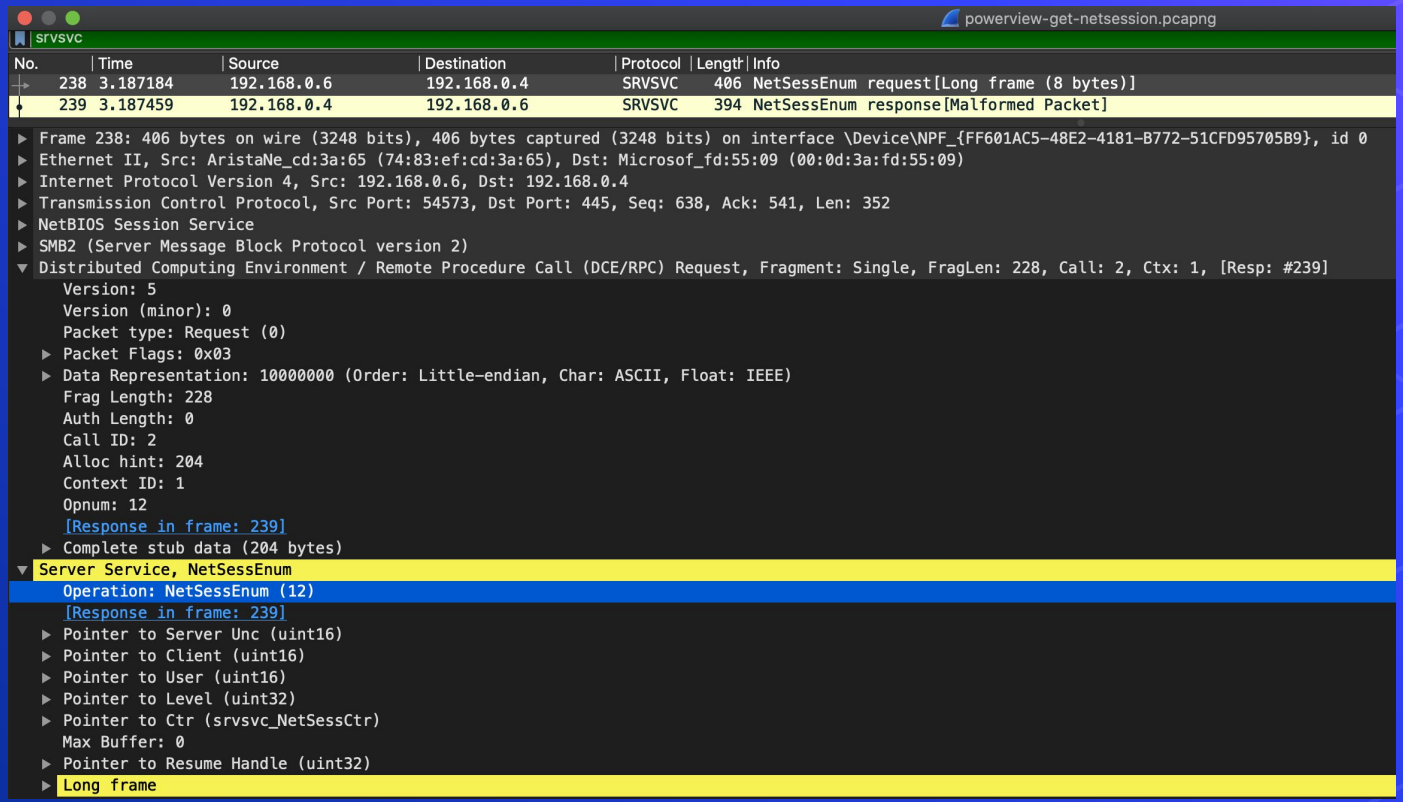

User Session Enumeration
Tools
Sharphound
Get-NetSession
(PowerView)
NetSess
Impacket
Managed
Code
Platform Invoke (P/Invoke)
Windows API
Functions
netapi32!NetSessionEnum
srvcli!NetSessionEnum
rpcrt4!NdrClientCall4
RPC Interface
[MS-SRVS] Server Service Remote Protocol
4b324fc8-1670-01d3-1278-5a47bf6ee188
NetrSessionEnum (Opnum 12)
C:\WINDOWS\SYSTEM32\srvsvc.dll
Network
Protocol
RPC over SMB (ncacn_np)
\PIPE\srvsvc

69

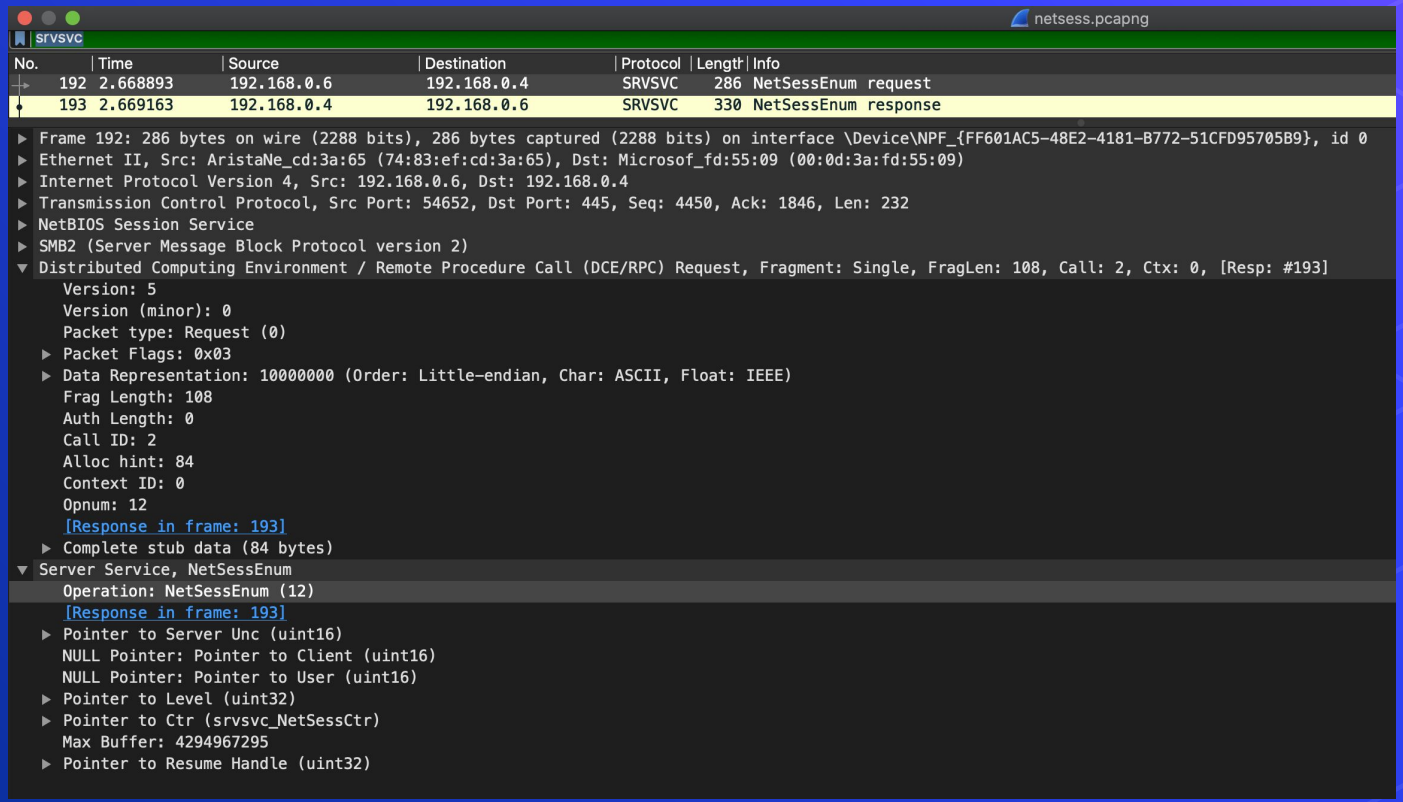
Network - Sharphound



Network - Get-NetSession



Network - NetSess



Network - Bloodhound.py

session-enum-bloodhound-py.pcapng

Apply a display filter ... <⌘/>

No.	Time	Source	Destination	Protocol	Length	Info
1040	7.612370	192.168.231.101	192.168.231.175	SMB2	190	Encrypted SMB3
1041	7.655544	192.168.231.175	192.168.231.101	TCP	54	50543 → 445 [ACK] Seq=1873 Ack=1767 Win=262656 Len=0
1042	7.663998	192.168.231.175	192.168.231.101	SMB2	223	Encrypted SMB3
1043	7.664282	192.168.231.101	192.168.231.175	SMB2	358	Encrypted SMB3
1044	7.672136	192.168.231.175	192.168.231.129	SMB2	306	Encrypted SMB3
1045	7.672633	192.168.231.129	192.168.231.175	SMB2	190	Encrypted SMB3
1046	7.718030	192.168.231.175	192.168.231.129	TCP	54	50540 → 445 [ACK] Seq=1883 Ack=1763 Win=262656 Len=0
1047	7.718108	192.168.231.175	192.168.231.101	TCP	54	50543 → 445 [ACK] Seq=2042 Ack=2071 Win=262144 Len=0
1048	7.733718	192.168.231.175	192.168.231.129	SMB2	223	Encrypted SMB3
1049	7.733992	192.168.231.129	192.168.231.175	SMB2	462	Encrypted SMB3
1050	7.750874	192.168.231.175	192.168.231.101	SMB2	178	Encrypted SMB3
1051	7.751245	192.168.231.101	192.168.231.175	SMB2	178	Encrypted SMB3
1052	7.780612	192.168.231.175	192.168.231.129	TCP	54	50540 → 445 [ACK] Seq=2052 Ack=2171 Win=262144 Len=0
1053	7.792150	192.168.231.175	192.168.231.101	SMB2	226	Encrypted SMB3
1054	7.792430	192.168.231.101	192.168.231.175	SMB2	190	Encrypted SMB3
1055	7.834043	192.168.231.175	192.168.231.129	SMB2	178	Encrypted SMB3
1056	7.834434	192.168.231.129	192.168.231.175	SMB2	178	Encrypted SMB3
1057	7.840764	192.168.231.175	192.168.231.101	SMB2	238	Encrypted SMB3
1058	7.841241	192.168.231.101	192.168.231.175	SMB2	262	Encrypted SMB3
1059	7.874290	192.168.231.175	192.168.231.129	TCP	54	50540 → 445 [ACK] Seq=2176 Ack=2295 Win=261888 Len=0
1060	7.885257	192.168.231.175	192.168.231.129	SMB2	226	Encrypted SMB3
1061	7.885570	192.168.231.129	192.168.231.175	SMB2	190	Encrypted SMB3
1062	7.889888	192.168.231.175	192.168.231.101	TCP	54	50543 → 445 [ACK] Seq=2522 Ack=2539 Win=261888 Len=0
1063	7.907450	192.168.231.175	192.168.231.101	SMB2	294	Encrypted SMB3

```
▶ Frame 1244: 258 bytes on wire (2064 bits), 258 bytes captured (2064 bits) on interface \Device\NPF_{1813C4D5-ACBC-46B2-974B-CD96B2484EB3}, id 0
▶ Ethernet II, Src: VMware_2b:2f:a8 (00:50:56:2b:2f:a8), Dst: VMware_a2:33:c1 (00:0c:29:a2:33:c1)
▶ Internet Protocol Version 4, Src: 192.168.231.129, Dst: 192.168.231.175
▼ Transmission Control Protocol, Src Port: 445, Dst Port: 50540, Seq: 10559, Ack: 11607, Len: 204
    Source Port: 445
    Destination Port: 50540
    [Stream index: 5]
    [TCP Segment Len: 204]
    Sequence number: 10559    (relative sequence number)
    Sequence number (raw): 1423873011
    [Next sequence number: 10763    (relative sequence number)]
    Acknowledgment number: 11607    (relative ack number)
    Acknowledgment number (raw): 1516581639
    0101 .... = Header Length: 20 bytes (5)
  ▶ Flags: 0x018 (PSH, ACK)
    Window size value: 2050
    [Calculated window size: 524800]
    [Window size scaling factor: 256]
```

Abstraction Take Aways

- The best place for detection is dependent on your monitoring posture
- Don’t rely on a single detection/analytic
- Always strive to understand as much about the attack as possible when developing detection logic
- Take the easy wins when possible, but keep the full scope in mind
- There likely isn’t one answer that covers everything!

Thanks!

Any questions?

You can find us at:

@_wald0

@jaredcatkinson

Link to this deck:
https://bit.ly/2Wk9bAm

Free templates for all your presentation needs

For PowerPoint and Google Slides

100% free for personal or commercial use

Ready to use, professional and customizable

Blow your audience away with attractive visuals