
MALPROXY
Leave your malware @home
Amit Waisel
Hila Cohen

About US

Amit Waisel

Offensive Cyber Security Expert

Technology lead, Security Research @ XM Cyber

Trusted Security Advisor

Favorite bit: 1

Private Pilot ✈️, Skipper ⛵ and cat lover 🐈

Hila Cohen

Security Researcher @ XM Cyber

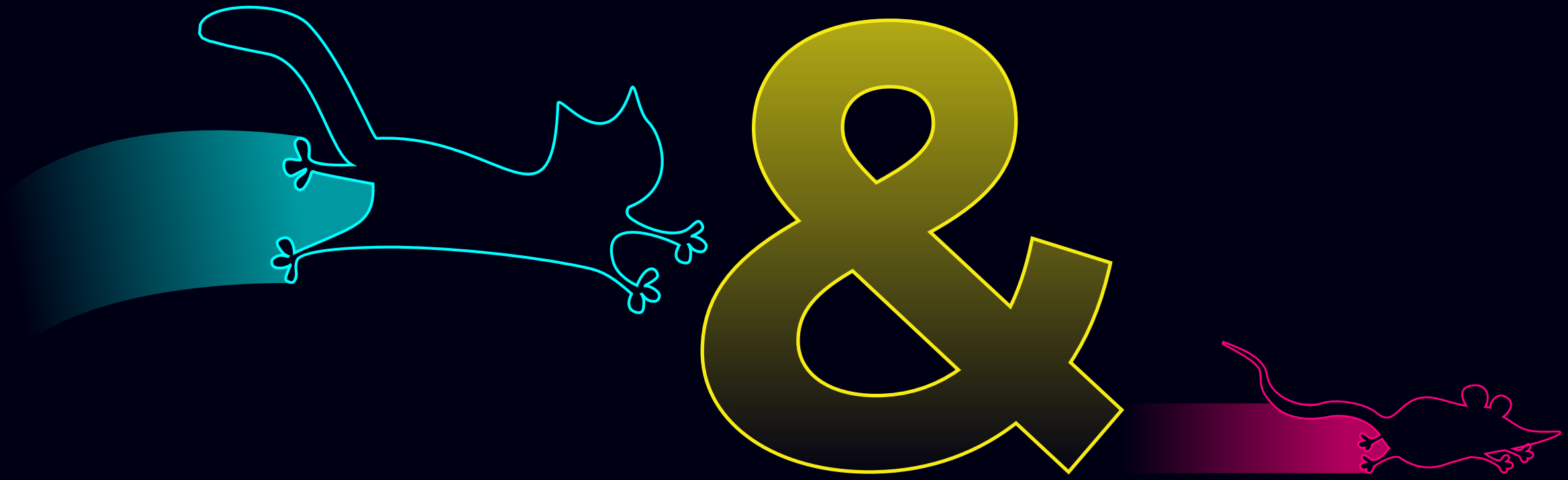
@hilaco10

Passionate about Windows Internals and Malware Analysis

Love to dance, travel the world 🌍 and capture moments with my camera 📷

TL;DR

- Endpoint protections introduction
- Malproxy - A new technique to bypass endpoint protections
- Demo
- Mitigations



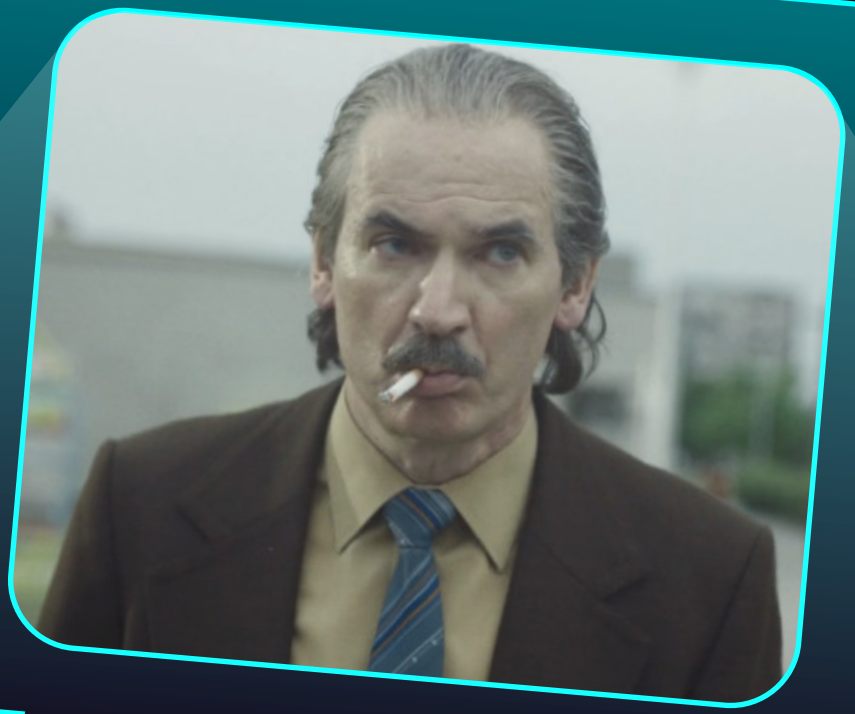
- Organizations heavily rely on endpoint protection solutions in their security stack
- Unfair cat-and-mouse game
- Security solutions evolved over time, so are the viruses

What do you
know about your

endpoint protection solutions?

not great,
not terrible

Anatoly Dyatlov

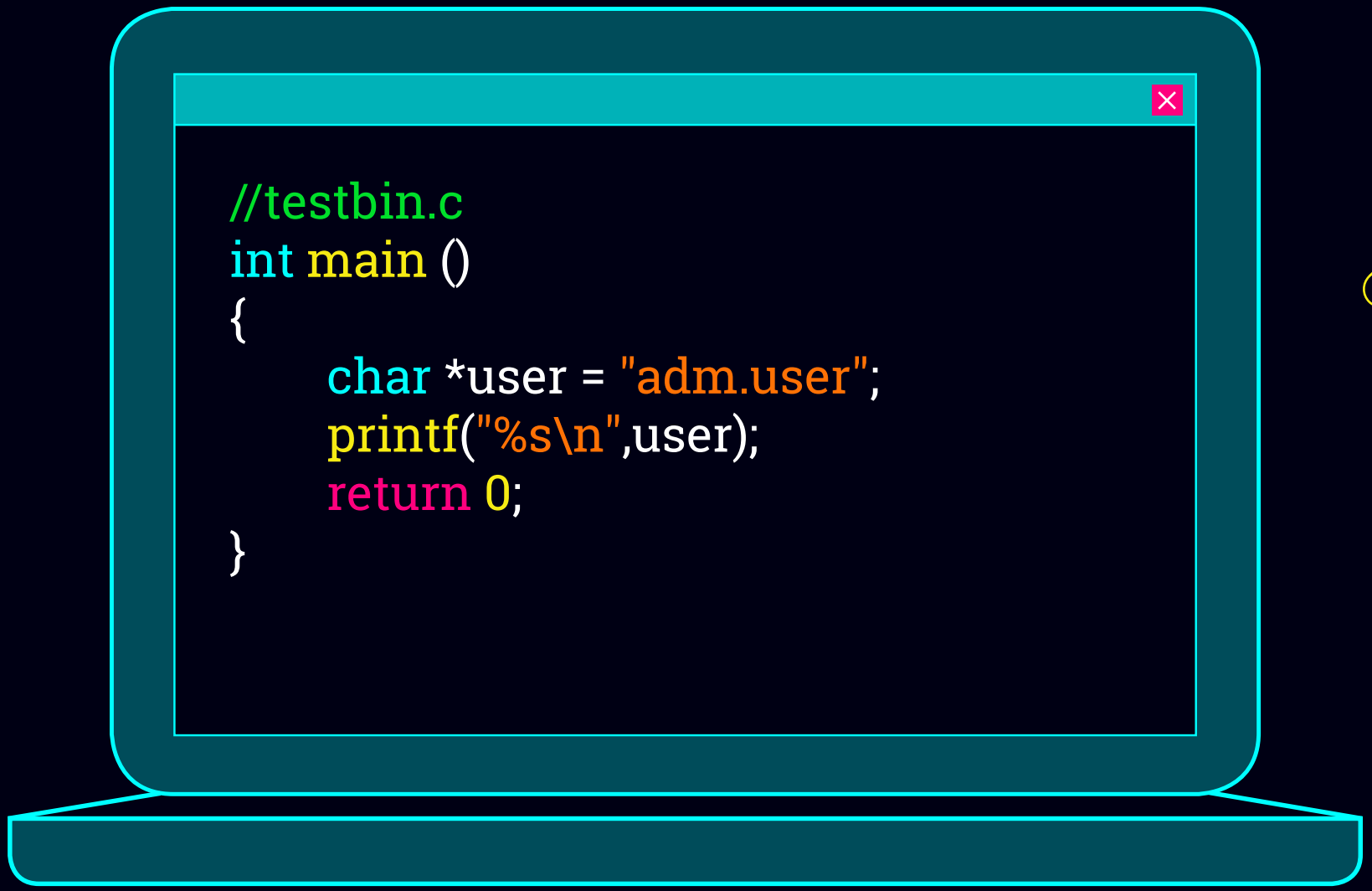


Endpoint Protection 101

malicious activity detection mechanisms

1. Static signatures
2. Heuristics
3. Behavioral signatures

1. Static signatures
2. Heuristics
3. Behavioral signatures

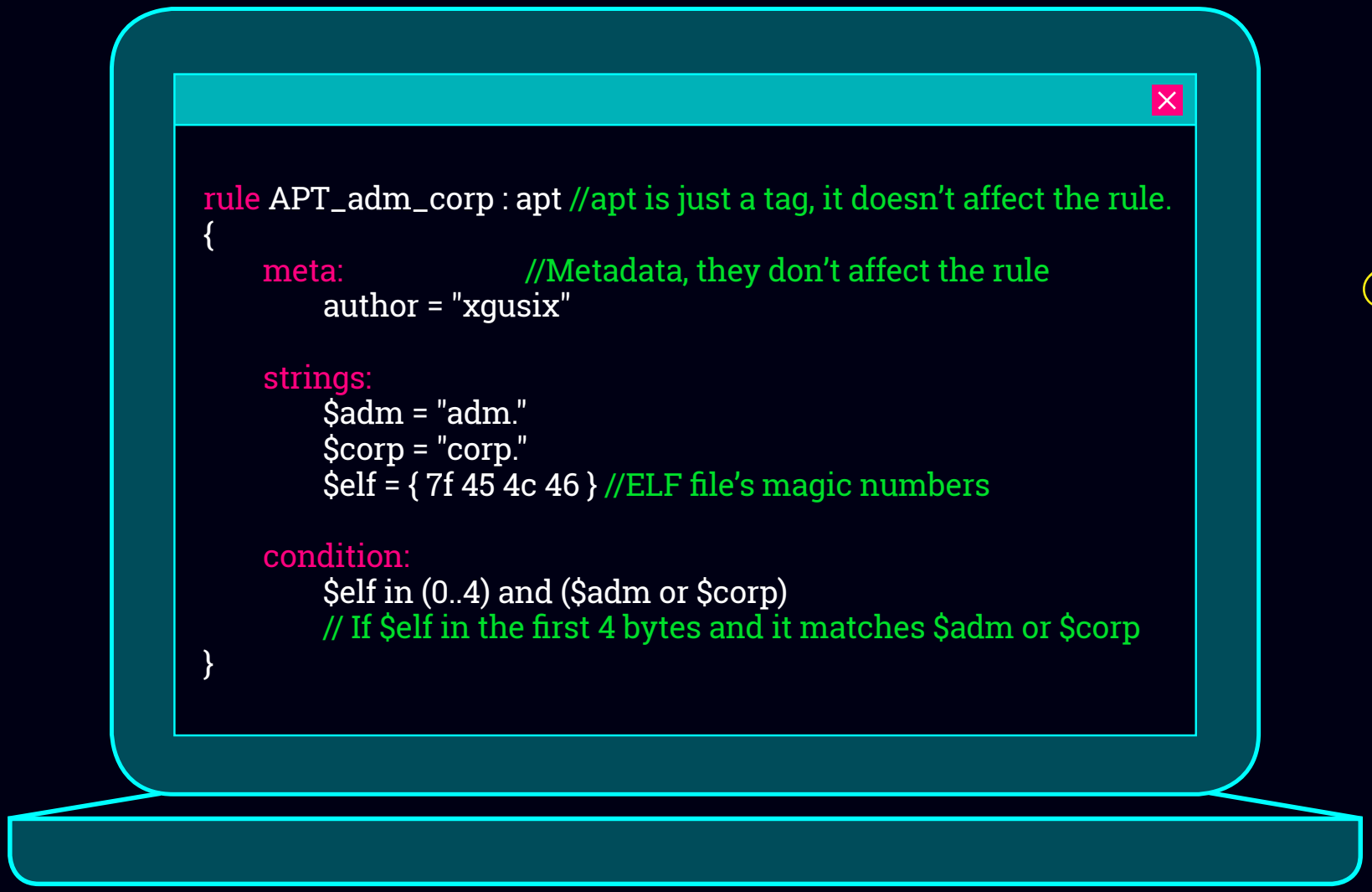


```c
//testbin.c
int main ()
{
        char *user = "adm.user";
        printf("%s\n",user);
        return 0;
}
```

1 Static signatures

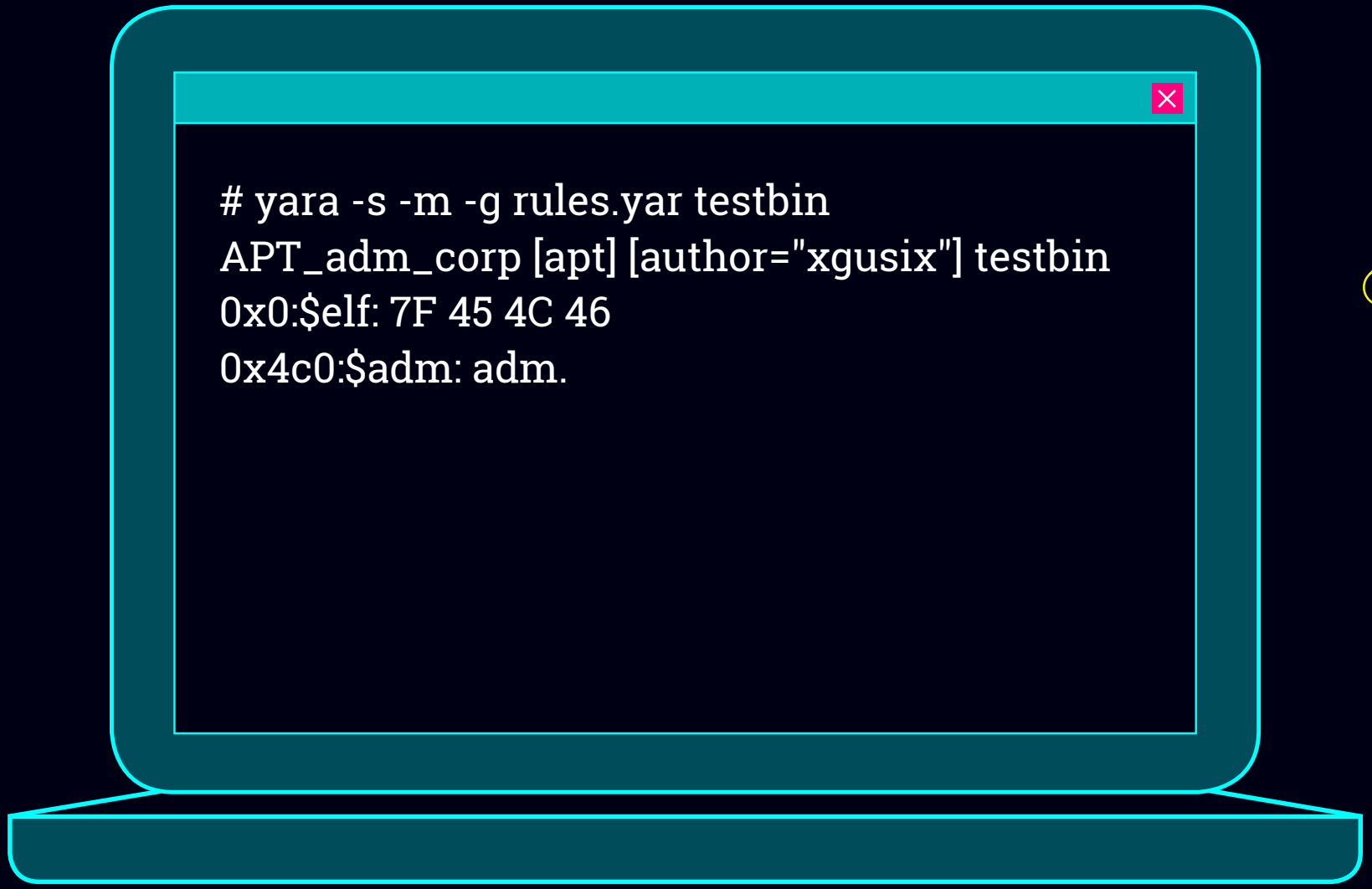
2 Heuristics

3 Behavioral signatures

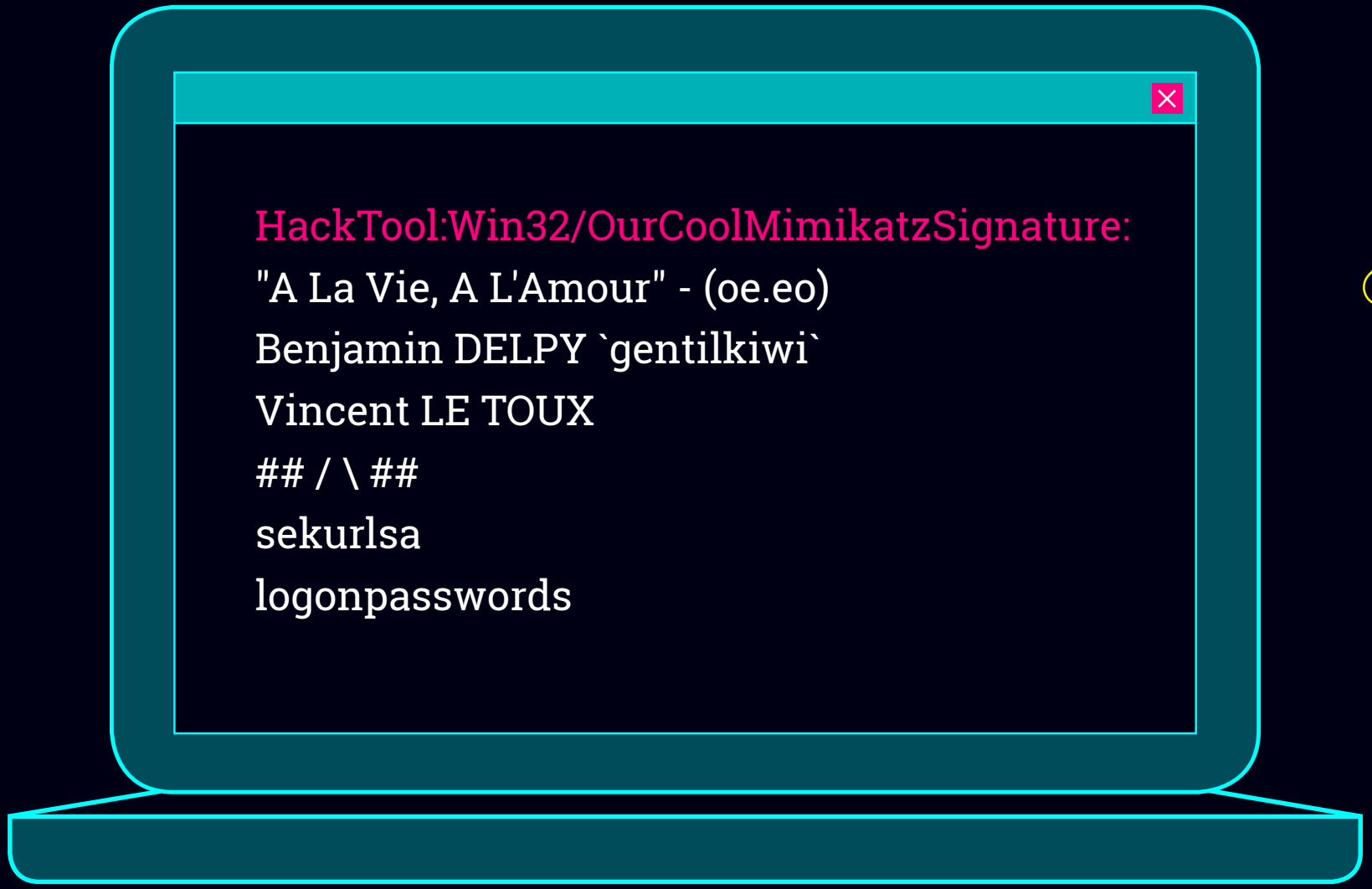


```
rule APT_adm_corp : apt //apt is just a tag, it doesn't affect the rule.
{
        meta:                   //Metadata, they don't affect the rule
                author = "xgusix"

        strings:
                $adm = "adm."
                $corp = "corp."
                $elf = { 7f 45 4c 46 } //ELF file's magic numbers

        condition:
                $elf in (0..4) and ($adm or $corp)
                // If $elf in the first 4 bytes and it matches $adm or $corp
}
```

1. Static signatures
2. Heuristics
3. Behavioral signatures



1. Static signatures
2. Heuristics
3. Behavioral signatures



1 Static signatures

2 Heuristics

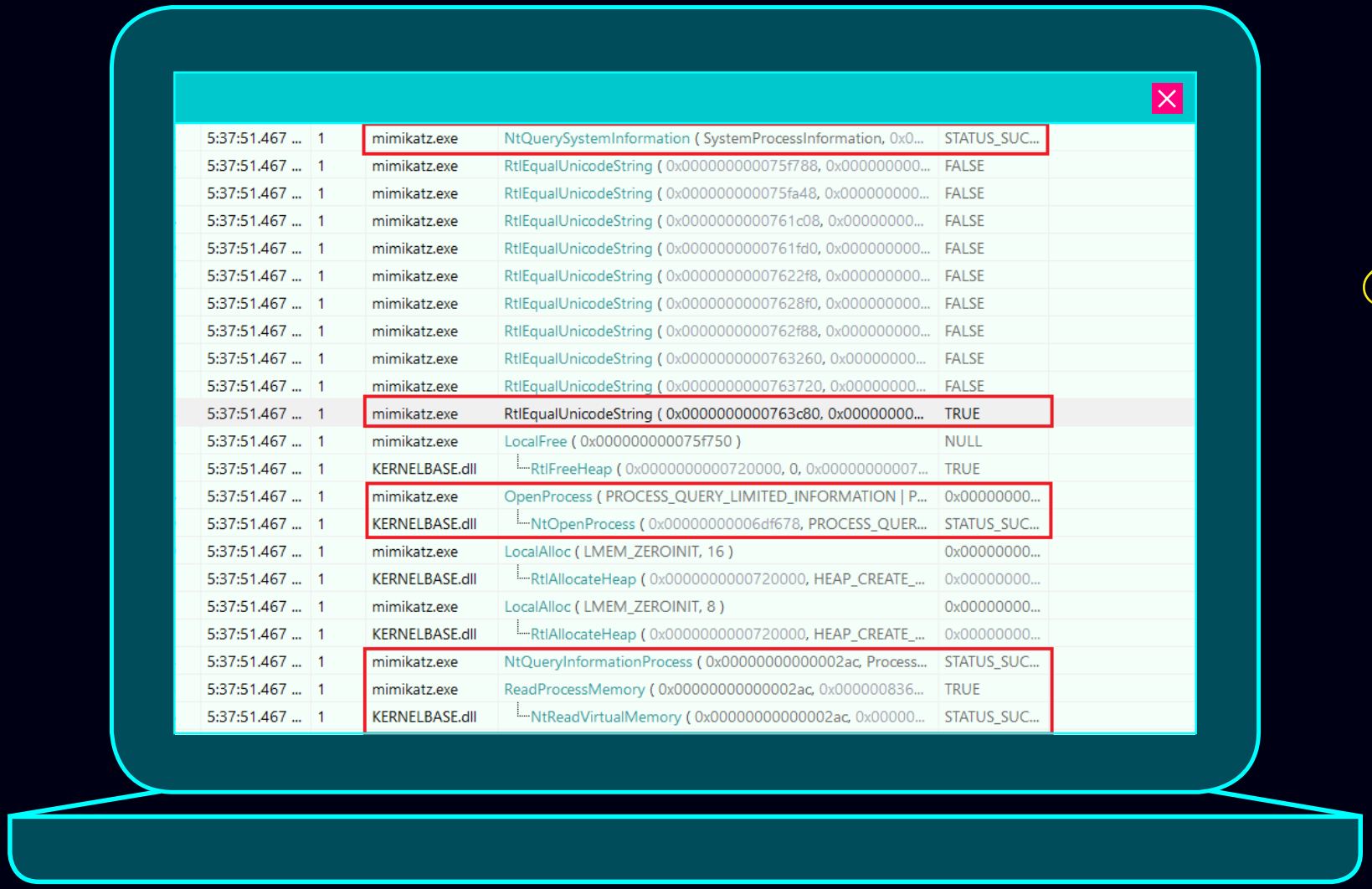
3 Behavioral signatures

Property	.text	.data	UPX2
Raw-address	0x00000400	0x00000400	0x00003400
Raw-size	0x0 bytes	0x3000 bytes	0x200 bytes
Virtual-address	0x00401000	0x00407000	0x0040A000
Virtual-size	0x6000 bytes	0x3000 bytes	0x1000 bytes
Executable	+	-	+
Writable	+	+	-

1 Static signatures

2 Heuristics

3 Behavioral signatures

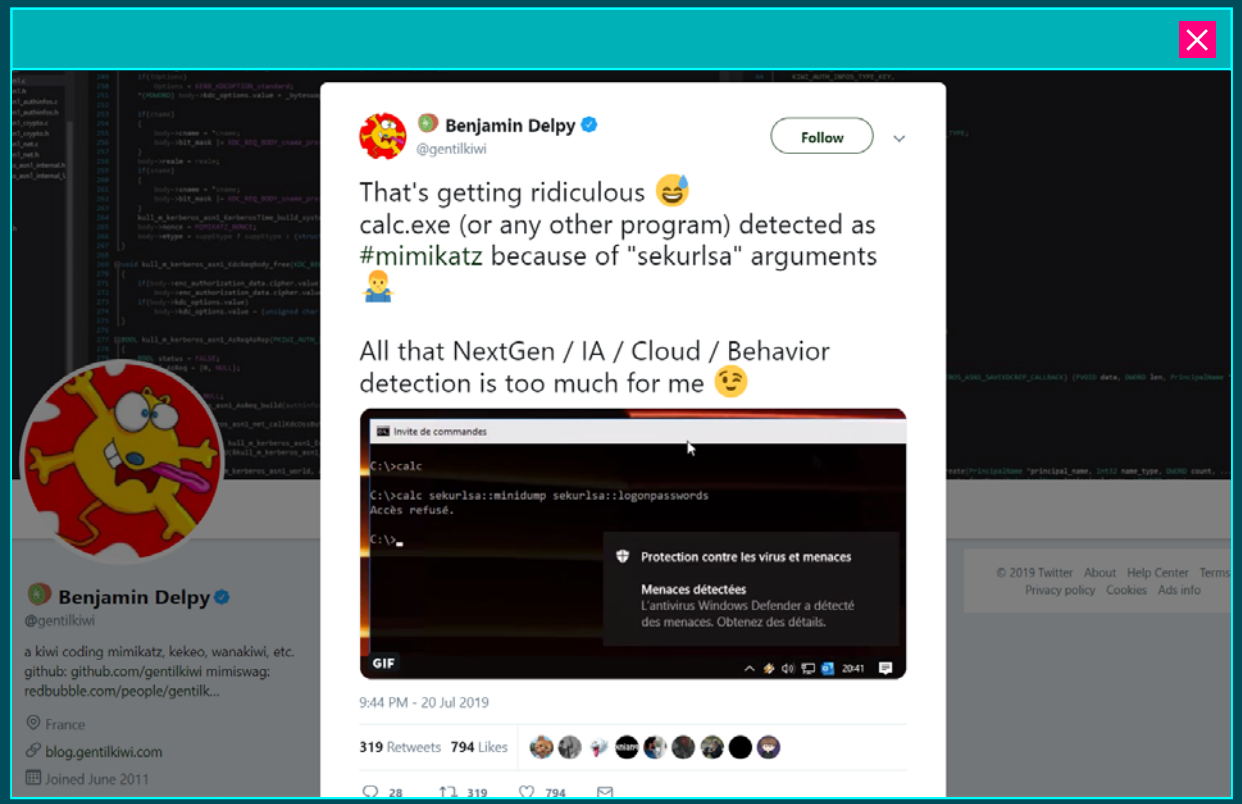


5:37:51.467 ...	1	mimikatz.exe	NtQuerySystemInformation (SystemProcessInformation, 0x0...	STATUS_SUC...
5:37:51.467 ...	1	mimikatz.exe	RtlEqualUnicodeString (0x0000000000075f788, 0x000000000...	FALSE
5:37:51.467 ...	1	mimikatz.exe	RtlEqualUnicodeString (0x0000000000075fa48, 0x000000000...	FALSE
5:37:51.467 ...	1	mimikatz.exe	RtlEqualUnicodeString (0x0000000000761c08, 0x000000000...	FALSE
5:37:51.467 ...	1	mimikatz.exe	RtlEqualUnicodeString (0x0000000000761fd0, 0x000000000...	FALSE
5:37:51.467 ...	1	mimikatz.exe	RtlEqualUnicodeString (0x00000000007622f8, 0x000000000...	FALSE
5:37:51.467 ...	1	mimikatz.exe	RtlEqualUnicodeString (0x00000000007628f0, 0x000000000...	FALSE
5:37:51.467 ...	1	mimikatz.exe	RtlEqualUnicodeString (0x0000000000762f88, 0x000000000...	FALSE
5:37:51.467 ...	1	mimikatz.exe	RtlEqualUnicodeString (0x0000000000763260, 0x000000000...	FALSE
5:37:51.467 ...	1	mimikatz.exe	RtlEqualUnicodeString (0x0000000000763720, 0x000000000...	FALSE
5:37:51.467 ...	1	mimikatz.exe	RtlEqualUnicodeString (0x0000000000763c80, 0x000000000...	TRUE
5:37:51.467 ...	1	mimikatz.exe	LocalFree (0x0000000000075f750)	NULL
5:37:51.467 ...	1	KERNELBASE.dll	RtlFreeHeap (0x0000000000720000, 0, 0x00000000007...	TRUE
5:37:51.467 ...	1	mimikatz.exe	OpenProcess (PROCESS_QUERY_LIMITED_INFORMATION | P...	0x00000000...
5:37:51.467 ...	1	KERNELBASE.dll	NtOpenProcess (0x00000000006df678, PROCESS_QUER...	STATUS_SUC...
5:37:51.467 ...	1	mimikatz.exe	LocalAlloc (LMEM_ZEROINIT, 16)	0x00000000...
5:37:51.467 ...	1	KERNELBASE.dll	RtlAllocateHeap (0x0000000000720000, HEAP_CREATE_...	0x00000000...
5:37:51.467 ...	1	mimikatz.exe	LocalAlloc (LMEM_ZEROINIT, 8)	0x00000000...
5:37:51.467 ...	1	KERNELBASE.dll	RtlAllocateHeap (0x0000000000720000, HEAP_CREATE_...	0x00000000...
5:37:51.467 ...	1	mimikatz.exe	NtQueryInformationProcess (0x00000000000002ac, Process...	STATUS_SUC...
5:37:51.467 ...	1	mimikatz.exe	ReadProcessMemory (0x00000000000002ac, 0x000000836...	TRUE
5:37:51.467 ...	1	KERNELBASE.dll	NtReadVirtualMemory (0x00000000000002ac, 0x00000...	STATUS_SUC...

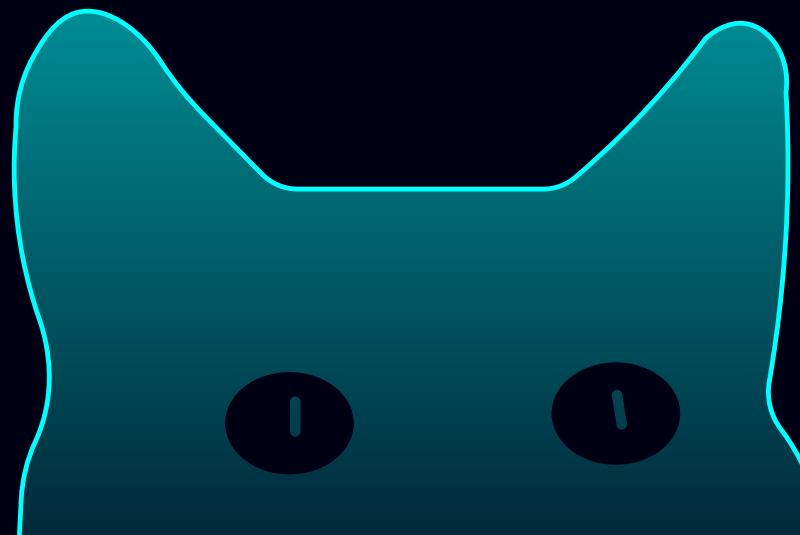
1 Static signatures

2 Heuristics

3 Behavioral signatures



Endpoint protection solutions bypass



MALPROXY

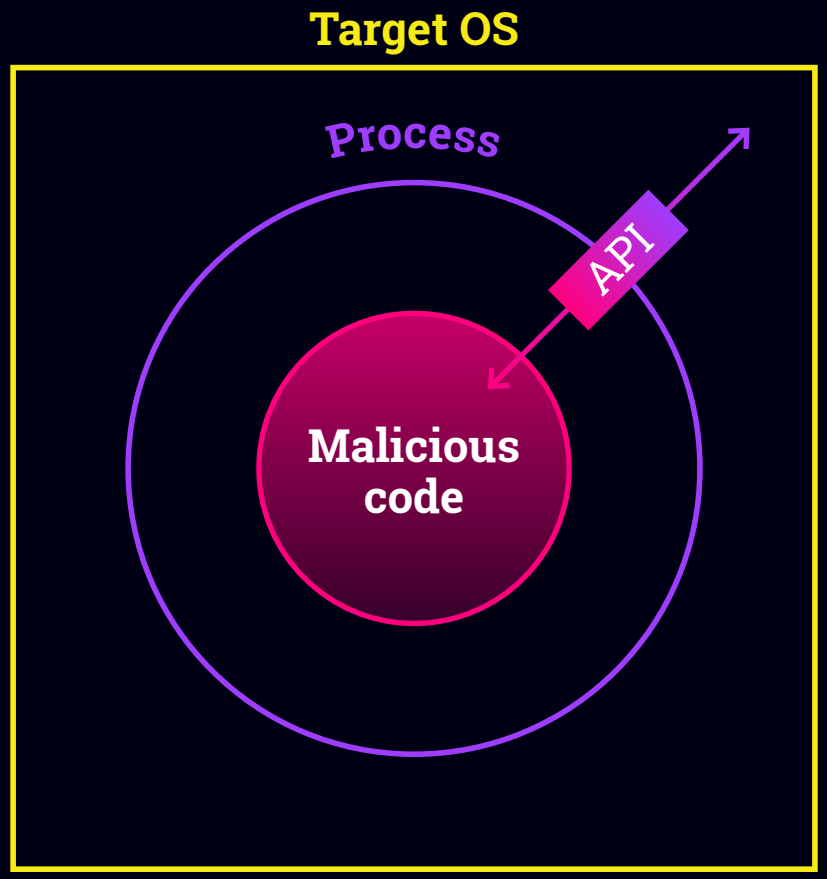
Endpoint protection solutions bypass

Looking for my code somewhere?

You will never get this!

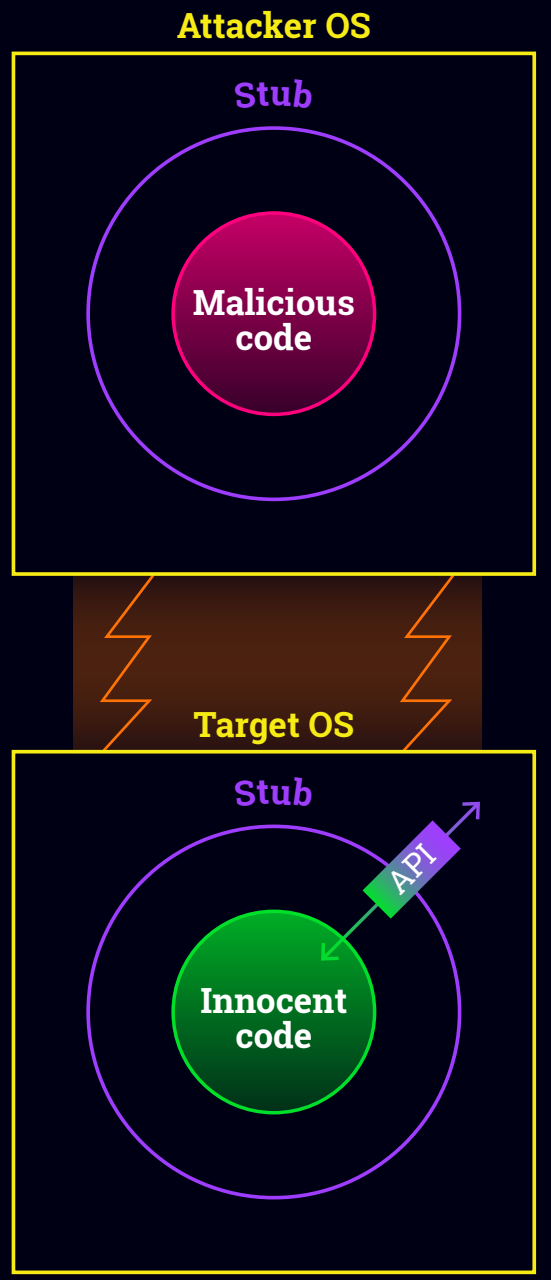
MALPROXY

- Malicious code interacts with the underlying OS using API function calls
- Those actions can be detected and blocked by any security solution



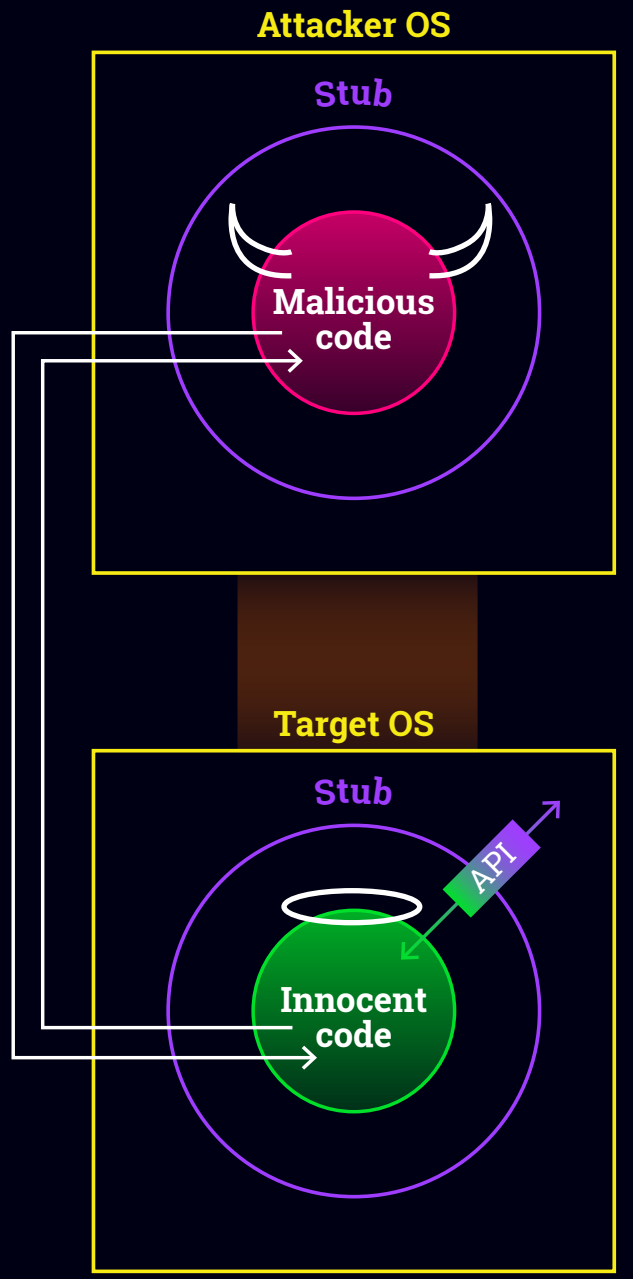
MALPROXY

- Proxy the malicious operations over the network
- Never deploying the actual malicious code on the target side
- Emulating needed API calls



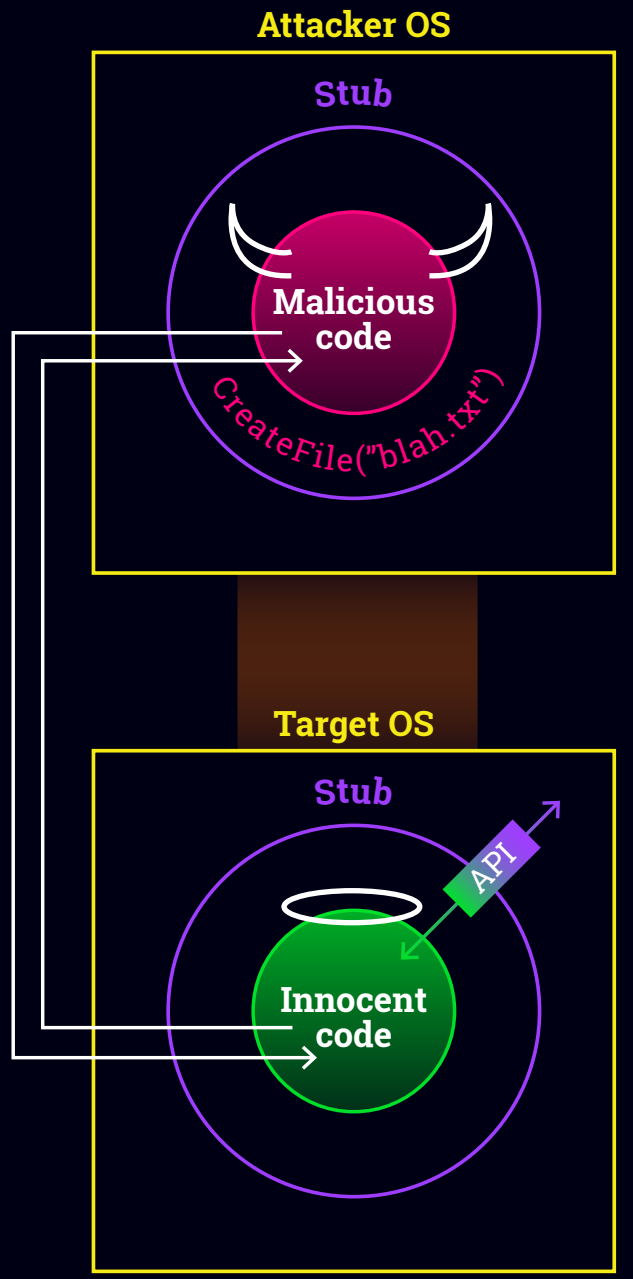
MALPROXY

- Target & attacker stubs
- Load the PE file and hook system API functions
- Execution flow – hook, serialize, send, execute, serialize, send, return. Repeat.



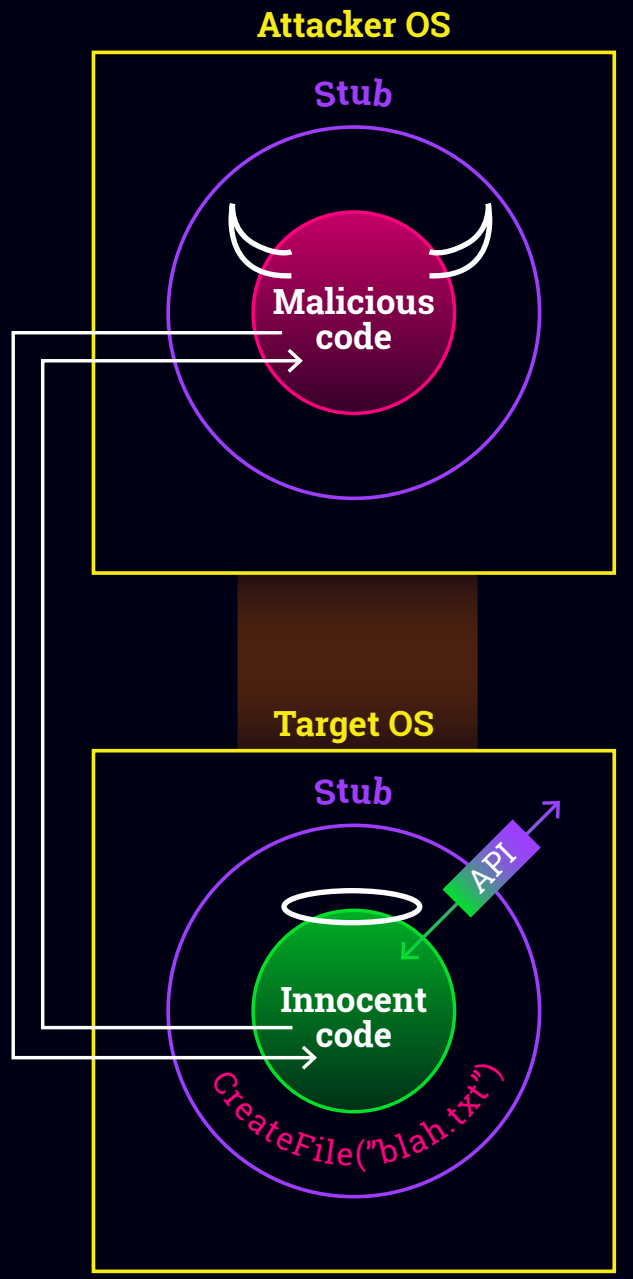
MALPROXY

- Target & attacker stubs
- Load the PE file and hook system API functions
- Execution flow – hook, serialize, send, execute, serialize, send, return. Repeat.



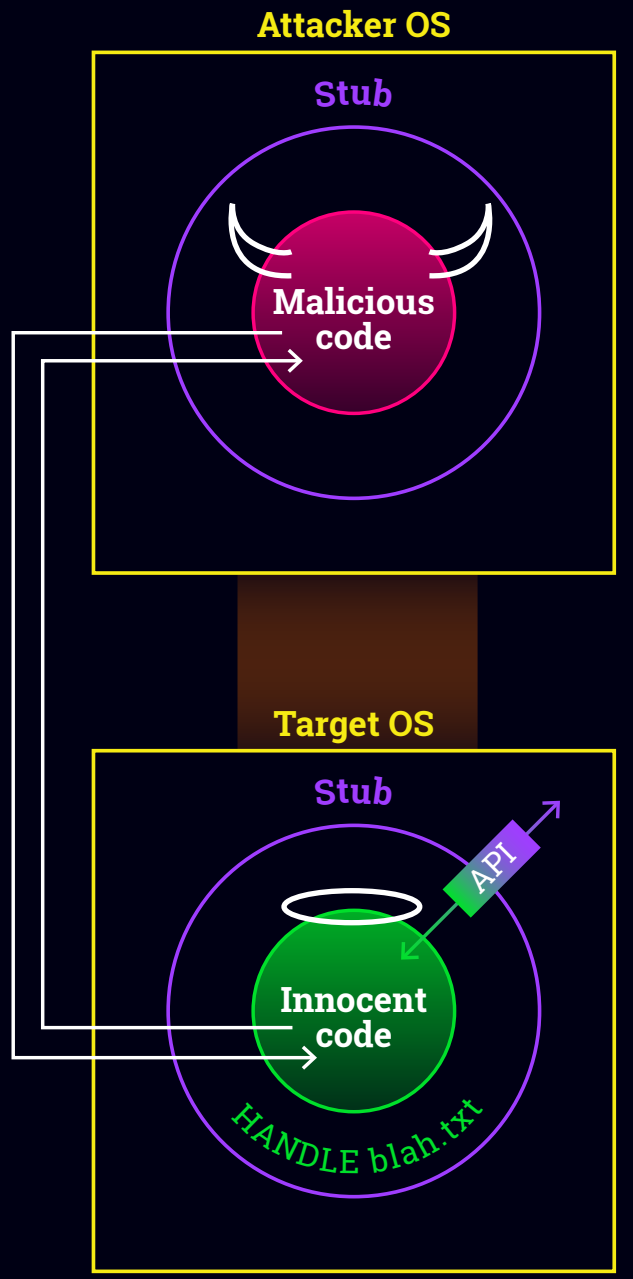
MALPROXY

- Target & attacker stubs
- Load the PE file and hook system API functions
- Execution flow – hook, serialize, send, execute, serialize, send, return. Repeat.



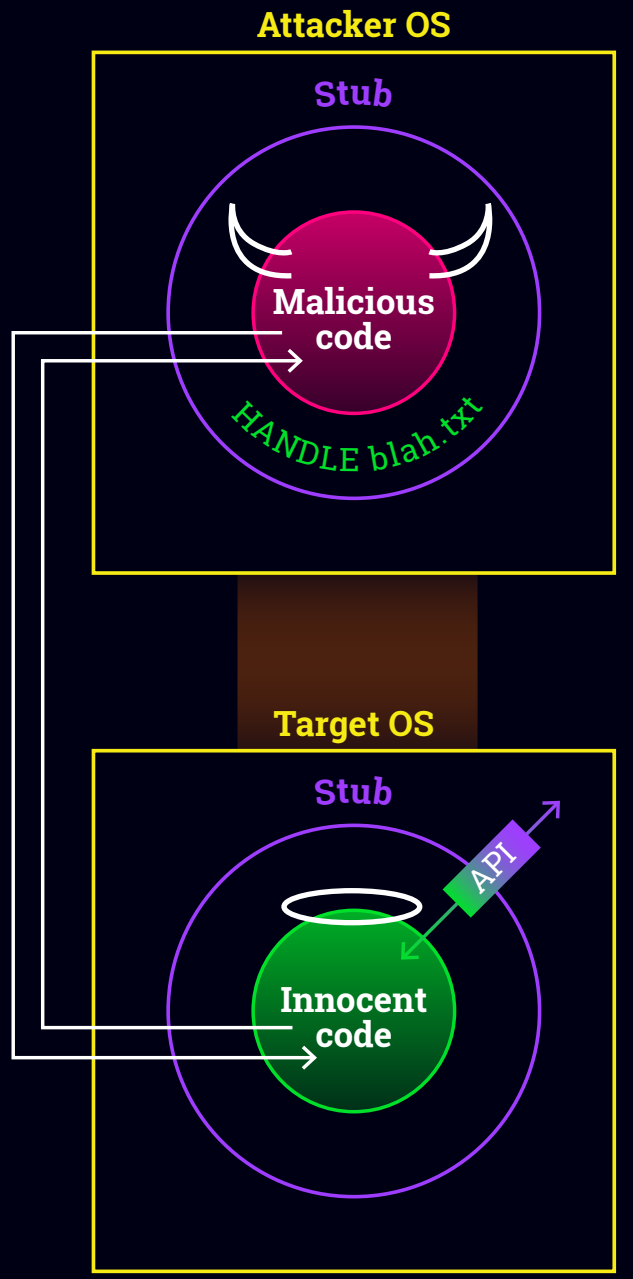
MALPROXY

- Target & attacker stubs
- Load the PE file and hook system API functions
- Execution flow – hook, serialize, send, execute, serialize, send, return. Repeat.



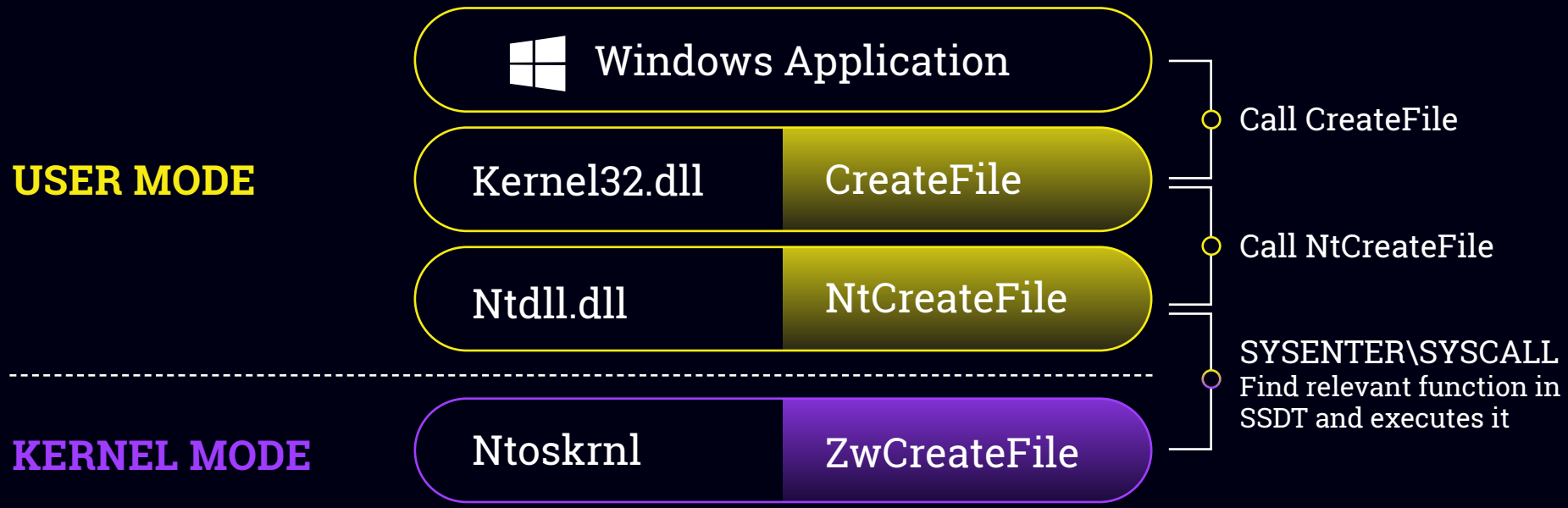
MALPROXY

- Target & attacker stubs
- Load the PE file and hook system API functions
- Execution flow – hook, serialize, send, execute, serialize, send, return. Repeat.



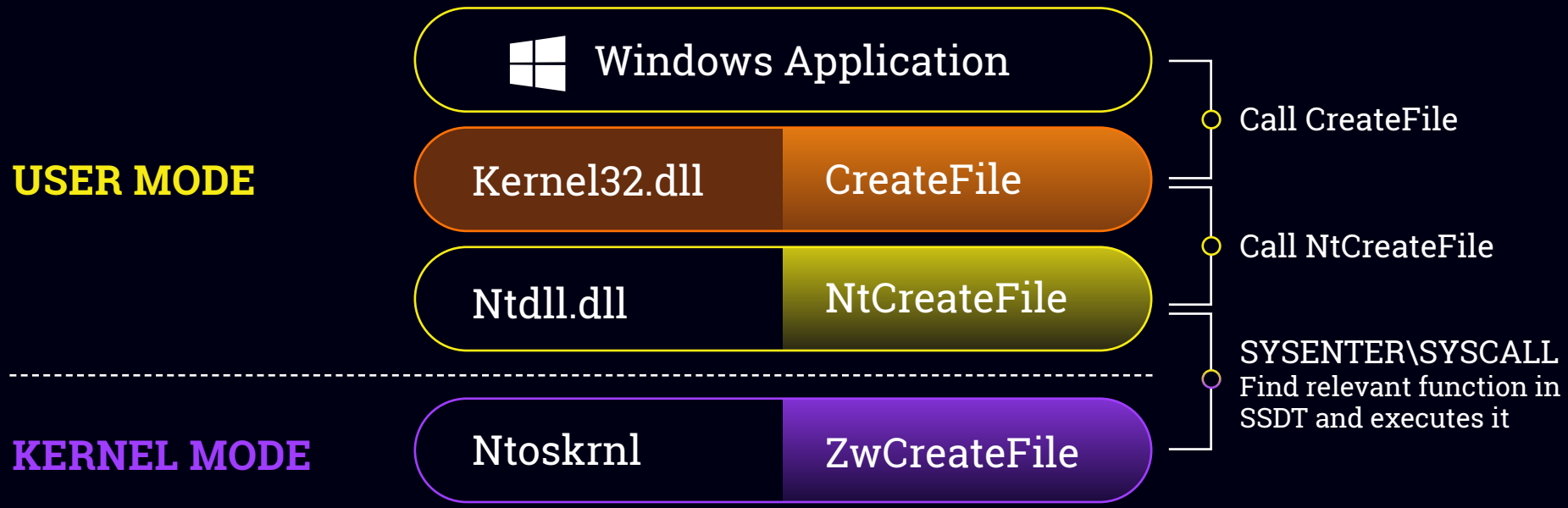
Key terms:

SYSTEM CALLS OVERVIEW



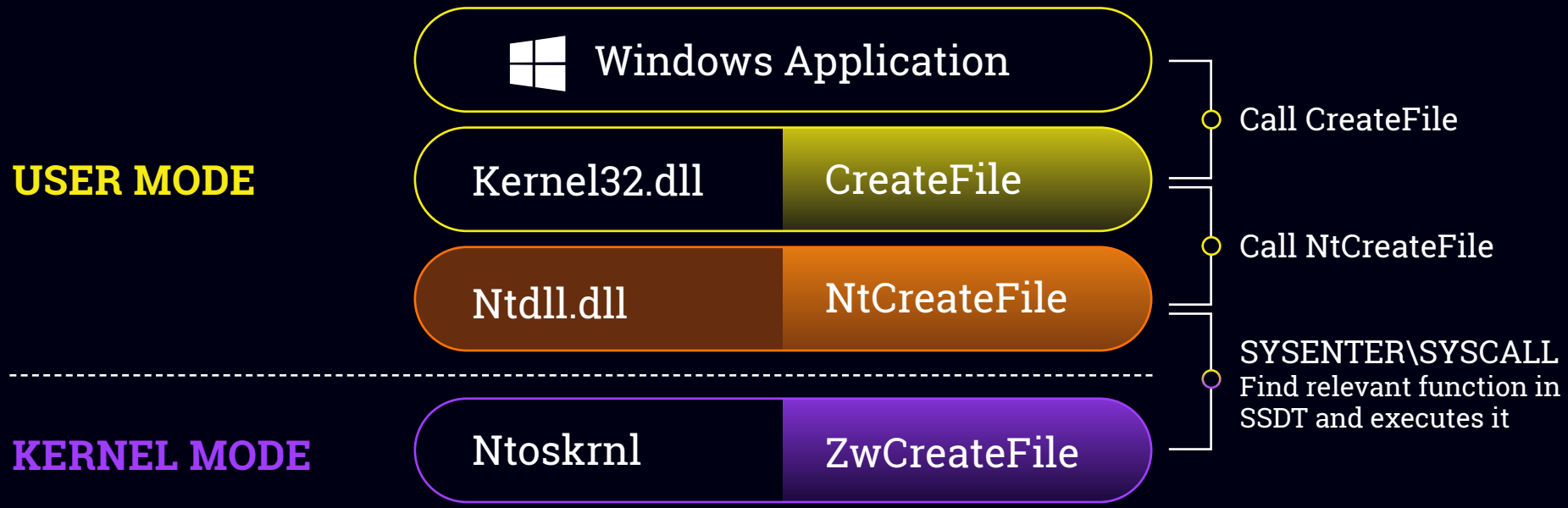
Key terms:

SYSTEM CALLS OVERVIEW



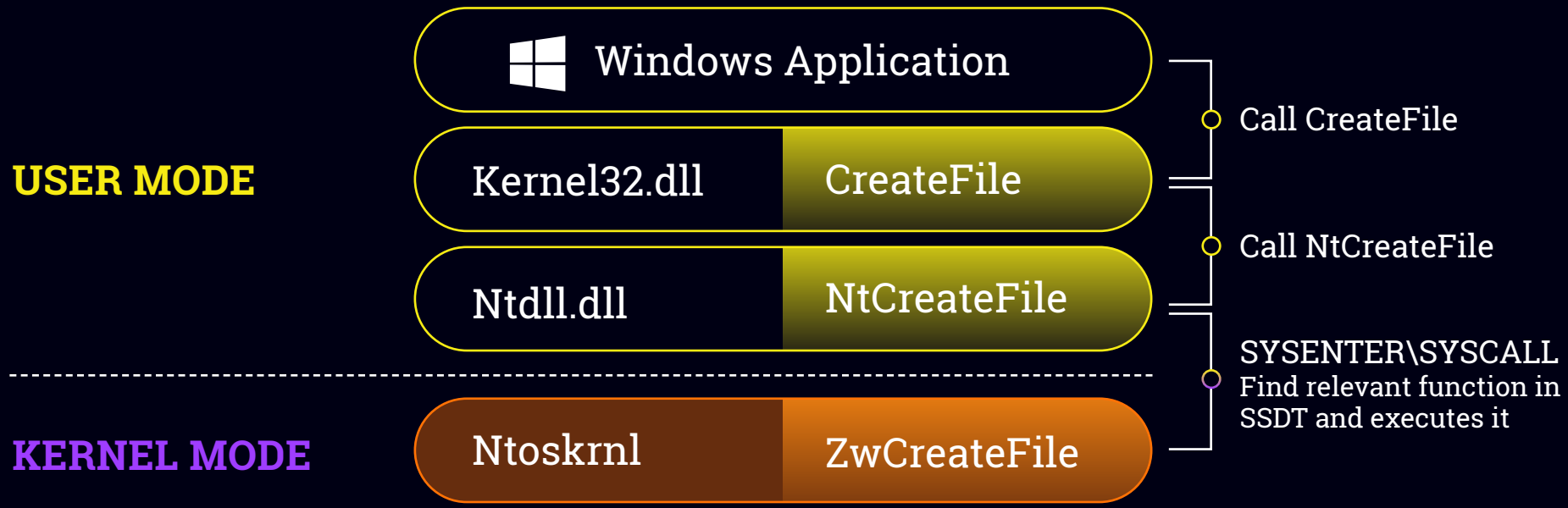
Key terms:

SYSTEM CALLS OVERVIEW



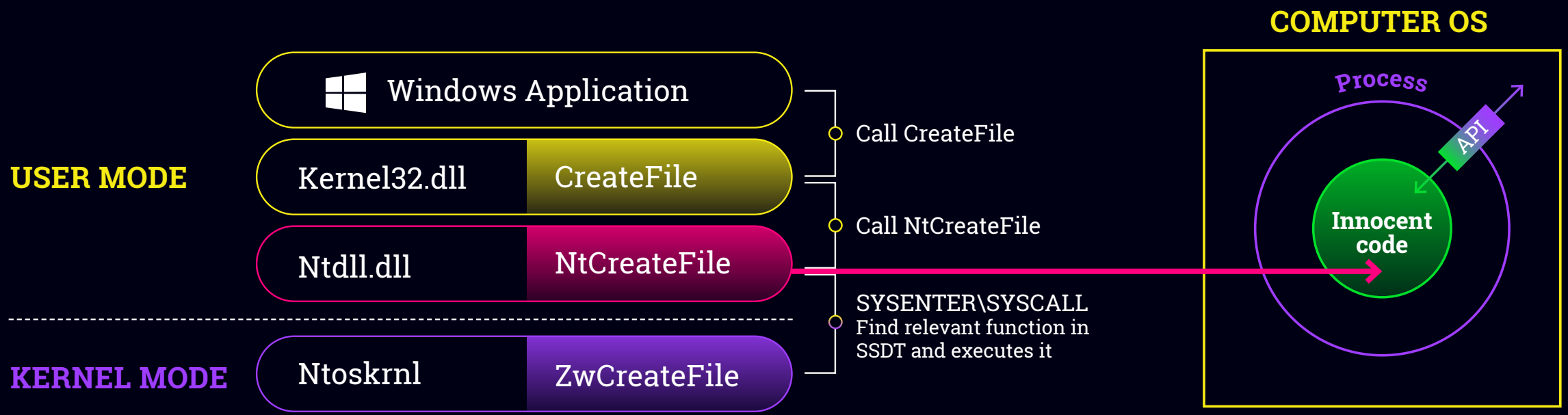
Key terms:

SYSTEM CALLS OVERVIEW



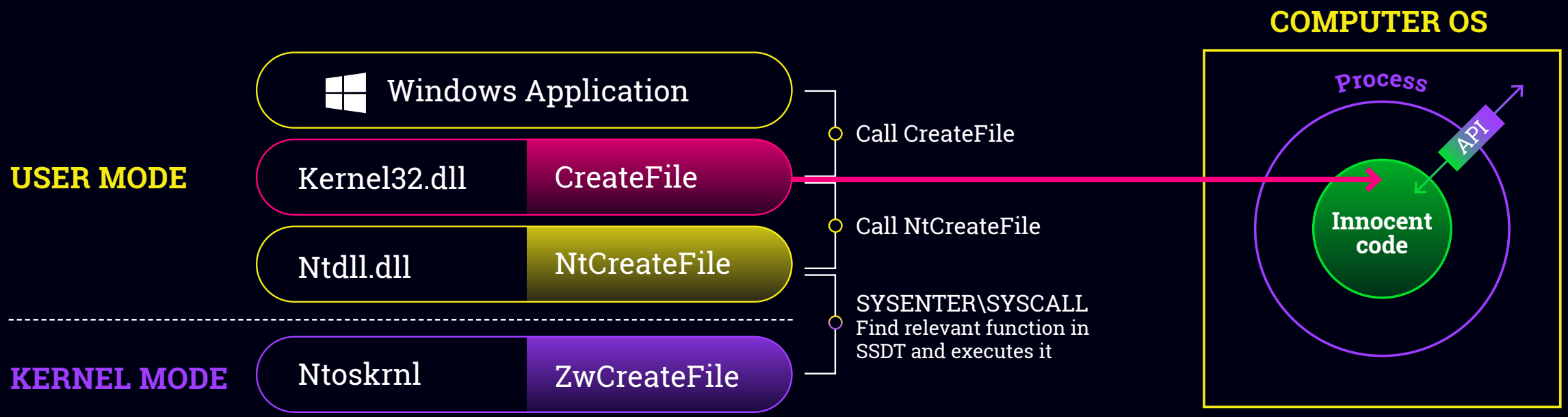
Key terms:

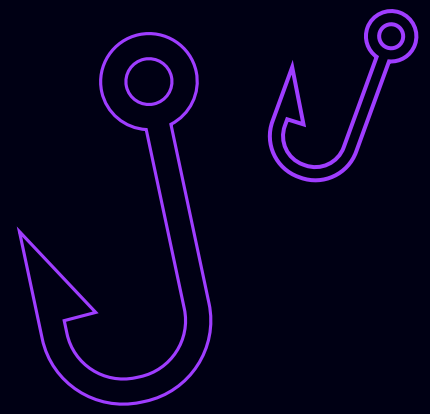
SYSTEM CALLS OVERVIEW



Key terms:

SYSTEM CALLS OVERVIEW





Key terms:

HOOKING

- Redirect system API calls to our code
- Imported system API function addresses are resolved during PE load process and can be overridden later – IAT hooking
- Control all arguments & return value
- This allows us to separate the code's logic from its interaction with the OS

IMPORT ADDRESS TABLE

NtQuerySystemInformation	Malproxy
OpenProcess	Malproxy
ReadProcessMemory	Malproxy
BCryptGenerateSymetricKey	Bcrypt.dll
ConvertSidToStringSidW	Advapi32.dll
...	...
RtlAdjustPrivilege	Malproxy
NtQueryInformationProcess	Malproxy
RtlEqualUnicodeString	Ntdll.dll

Key terms:

FUNCTION PROTOTYPE

```
BOOL  stdcall ReadProcessMemory(HANDLE hProcrss, LPCVOID lpBaseAddress, LPVOID lpBuffer, SIZE_T nSize, SIZE_T *lpNumberOfBytesRead);
```

Return Type

Calling Convention

Function arguments

Proxying Win32 API

Dealing with all aspects of different prototypes

- Calling convention – same for all Win32API and Native API calls
- **Input Arguments:**
  - Primitives
  - Pointers to primitives
  - User-allocated buffers
- **Output Arguments:**
  - User-allocated output buffer
  - System-allocated output buffer
- Return values

Handling ARGUMENTS

```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                   ProcessHandle,
    IN PROCESSINFOCLASS         ProcessInformationClass,
    OUT PVOID                   ProcessInformation,
    IN ULONG                    ProcessInformationLength,
    OUT PULONG                  ReturnLength
);
```

Handling ARGUMENTS

ATTACKER SIDE

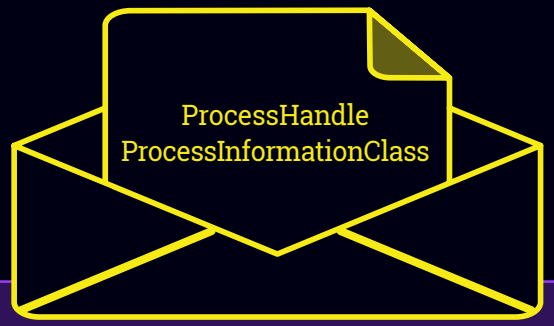
Request Message


```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                   ProcessHandle,
    IN PROCESSINFOCLASS         ProcessInformationClass,
    OUT PVOID                   ProcessInformation,
    IN ULONG                    ProcessInformationLength,
    OUT PULONG                  ReturnLength
);
```

Handling ARGUMENTS

ATTACKER SIDE

Request Message

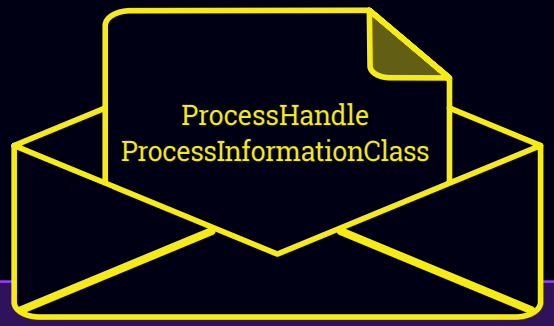



```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE               ProcessHandle,
    IN PROCESSINFOCLASS     ProcessInformationClass,
    OUT PVOID               ProcessInformation,
    IN ULONG                ProcessInformationLength,
    OUT PULONG              ReturnLength
);
```

Handling ARGUMENTS

ATTACKER SIDE

Request Message

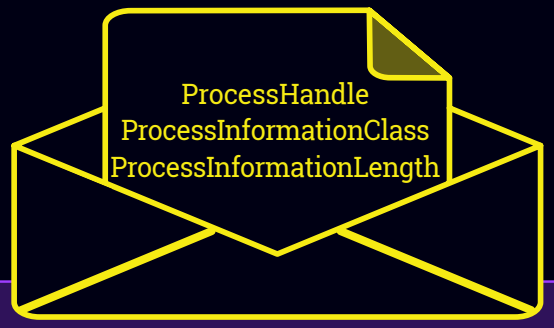



```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                   ProcessHandle,
    IN PROCESSINFOCLASS         ProcessInformationClass,
    OUT PVOID                   ProcessInformation,
    IN ULONG                    ProcessInformationLength,
    OUT PULONG                  ReturnLength
);
```

Handling ARGUMENTS

ATTACKER SIDE

Request Message

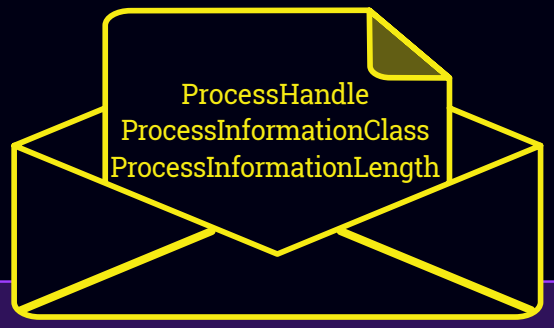



```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE               ProcessHandle,
    IN PROCESSINFOCLASS     ProcessInformationClass,
    OUT PVOID               ProcessInformation,
    IN ULONG                ProcessInformationLength,
    OUT PULONG              ReturnLength
);
```

Handling ARGUMENTS

ATTACKER SIDE

Request Message



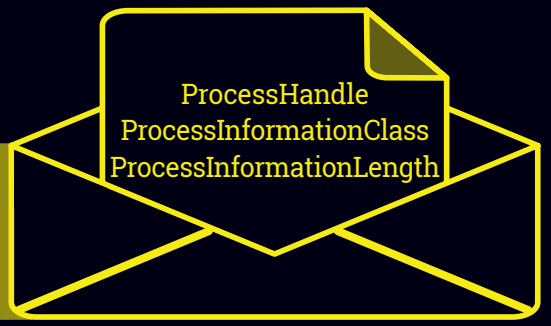

```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE               ProcessHandle,
    IN PROCESSINFOCLASS     ProcessInformationClass,
    OUT PVOID               ProcessInformation,
    IN ULONG                ProcessInformationLength,
    OUT PULONG              ReturnLength
);
```

Handling ARGUMENTS

ATTACKER SIDE

TARGET SIDE

Request Message




```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE               ProcessHandle,
    IN PROCESSINFOCLASS     ProcessInformationClass,
    OUT PVOID               ProcessInformation,
    IN ULONG                ProcessInformationLength,
    OUT PULONG              ReturnLength
);
```

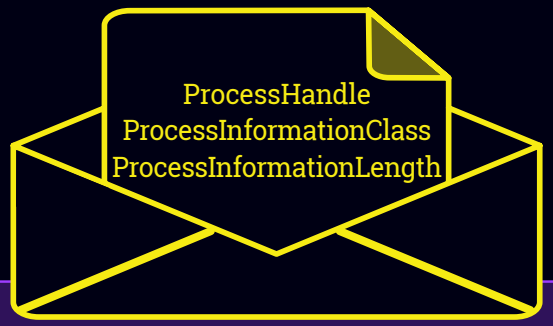
Handling ARGUMENTS

ATTACKER SIDE

```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                   ProcessHandle,
    IN PROCESSINFOCLASS         ProcessInformationClass,
    OUT PVOID                   ProcessInformation,
    IN ULONG                    ProcessInformationLength,
    OUT PULONG                  ReturnLength
);
```

TARGET SIDE

Request Message




```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                   ProcessHandle,
    IN PROCESSINFOCLASS         ProcessInformationClass,
    OUT PVOID                   ProcessInformation,
    IN ULONG                    ProcessInformationLength,
    OUT PULONG                  ReturnLength
);
```

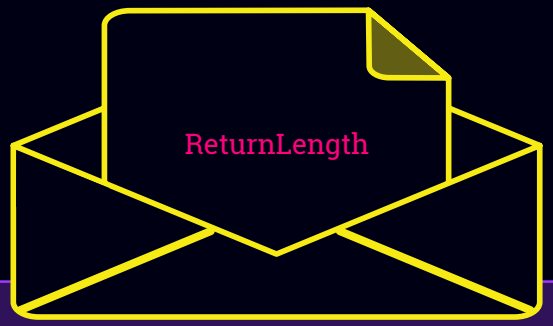
Handling ARGUMENTS

ATTACKER SIDE

```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                   ProcessHandle,
    IN PROCESSINFOCLASS         ProcessInformationClass,
    OUT PVOID                   ProcessInformation,
    IN ULONG                    ProcessInformationLength,
    OUT PULONG                  ReturnLength
);
```

TARGET SIDE

Response Message



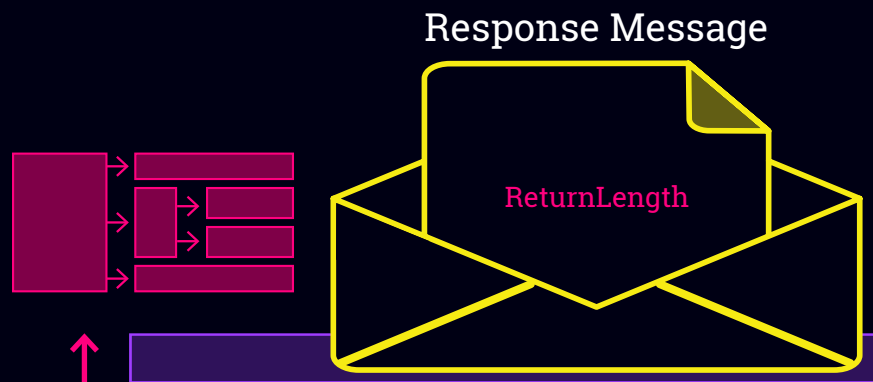

```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                   ProcessHandle,
    IN PROCESSINFOCLASS         ProcessInformationClass,
    OUT PVOID                   ProcessInformation,
    IN ULONG                    ProcessInformationLength,
    OUT PULONG                  ReturnLength
);
```

Handling ARGUMENTS

ATTACKER SIDE

```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                   ProcessHandle,
    IN PROCESSINFOCLASS         ProcessInformationClass,
    OUT PVOID                   ProcessInformation,
    IN ULONG                    ProcessInformationLength,
    OUT PULONG                  ReturnLength
);
```

TARGET SIDE




```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                   ProcessHandle,
    IN PROCESSINFOCLASS         ProcessInformationClass,
    OUT PVOID                   ProcessInformation,
    IN ULONG                    ProcessInformationLength,
    OUT PULONG                  ReturnLength
);
```

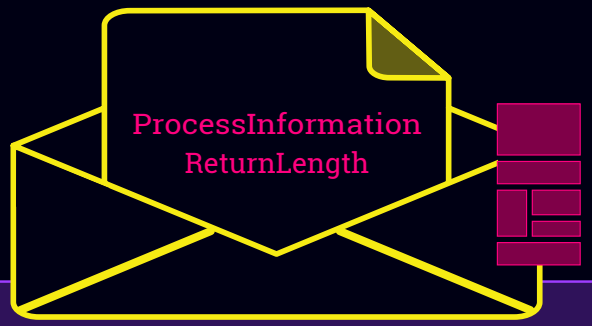
Handling ARGUMENTS

ATTACKER SIDE

```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                   ProcessHandle,
    IN PROCESSINFOCLASS         ProcessInformationClass,
    OUT PVOID                   ProcessInformation,
    IN ULONG                    ProcessInformationLength,
    OUT PULONG                  ReturnLength
);
```

TARGET SIDE

Response Message




```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                   ProcessHandle,
    IN PROCESSINFOCLASS         ProcessInformationClass,
    OUT PVOID                   ProcessInformation,
    IN ULONG                    ProcessInformationLength,
    OUT PULONG                  ReturnLength
);
```

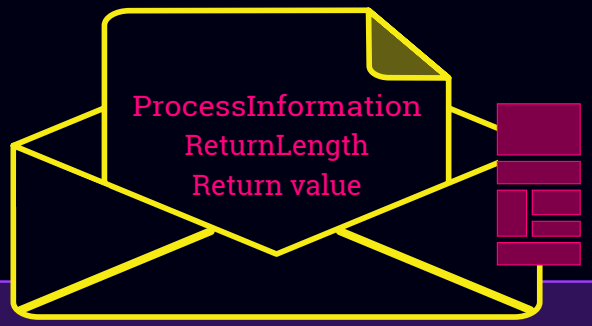
Handling ARGUMENTS

ATTACKER SIDE

```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                   ProcessHandle,
    IN PROCESSINFOCLASS         ProcessInformationClass,
    OUT PVOID                   ProcessInformation,
    IN ULONG                    ProcessInformationLength,
    OUT PULONG                  ReturnLength
);
```

TARGET SIDE

Response Message

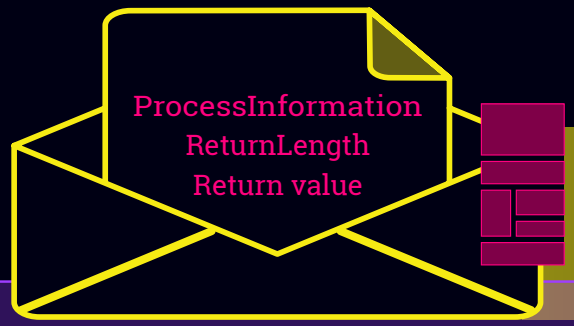



```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                   ProcessHandle,
    IN PROCESSINFOCLASS         ProcessInformationClass,
    OUT PVOID                   ProcessInformation,
    IN ULONG                    ProcessInformationLength,
    OUT PULONG                  ReturnLength
);
```

Handling ARGUMENTS

ATTACKER SIDE

Response Message




```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                ProcessHandle,
    IN PROCESSINFOCLASS      ProcessInformationClass,
    OUT PVOID                ProcessInformation,
    IN ULONG                 ProcessInformationLength,
    OUT PULONG               ReturnLength
);
```

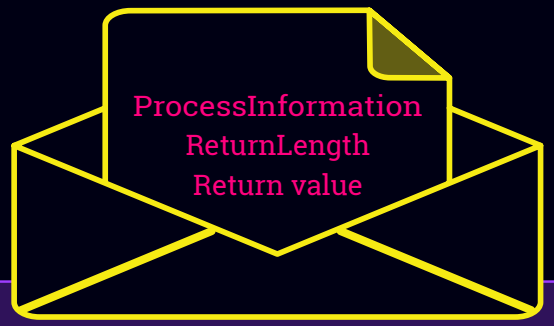
TARGET SIDE

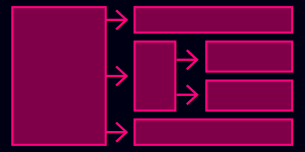
```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                ProcessHandle,
    IN PROCESSINFOCLASS      ProcessInformationClass,
    OUT PVOID                ProcessInformation,
    IN ULONG                 ProcessInformationLength,
    OUT PULONG               ReturnLength
);
```

Handling ARGUMENTS

ATTACKER SIDE

Response Message





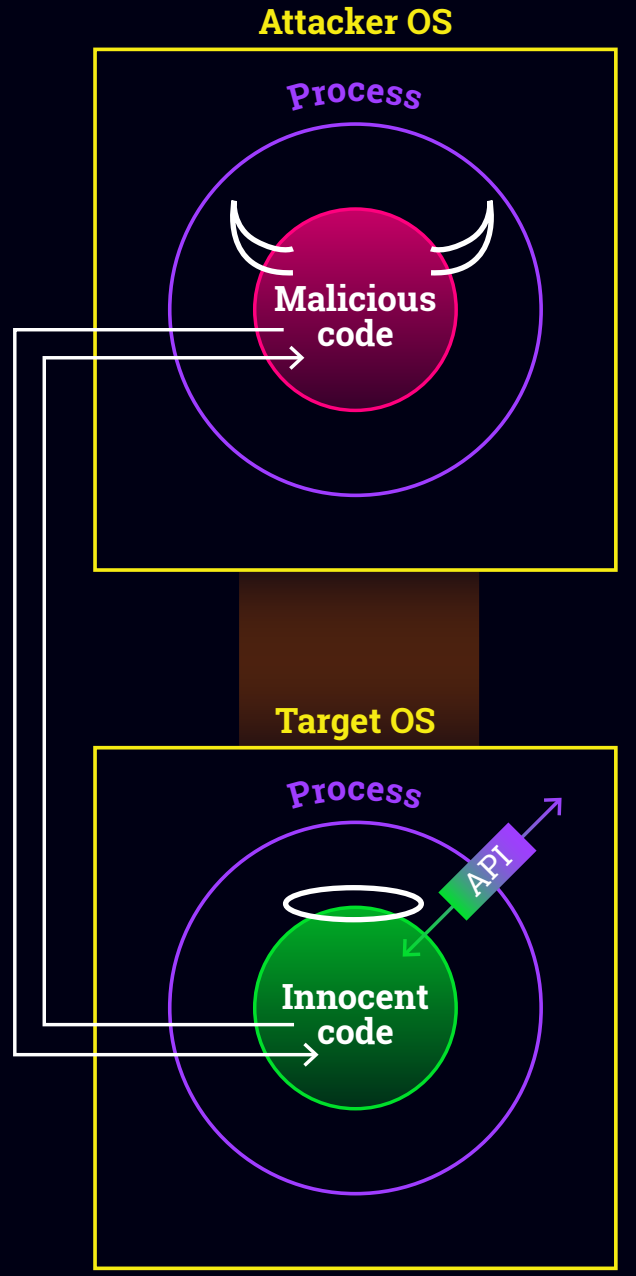
```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                ProcessHandle,
    IN PROCESSINFOCLASS      ProcessInformationClass,
    OUT PVOID                ProcessInformation,
    IN ULONG                 ProcessInformationLength,
    OUT PULONG               ReturnLength
);
```

TARGET SIDE

```
NTSTATUS NtQueryInformationProcess(
    IN HANDLE                ProcessHandle,
    IN PROCESSINFOCLASS      ProcessInformationClass,
    OUT PVOID                ProcessInformation,
    IN ULONG                 ProcessInformationLength,
    OUT PULONG               ReturnLength
);
```

RECAP

- Target & attacker stubs
- Load the PE file and hook system API functions
- Execution flow – hook, serialize, send, execute, serialize, send, return. Repeat.



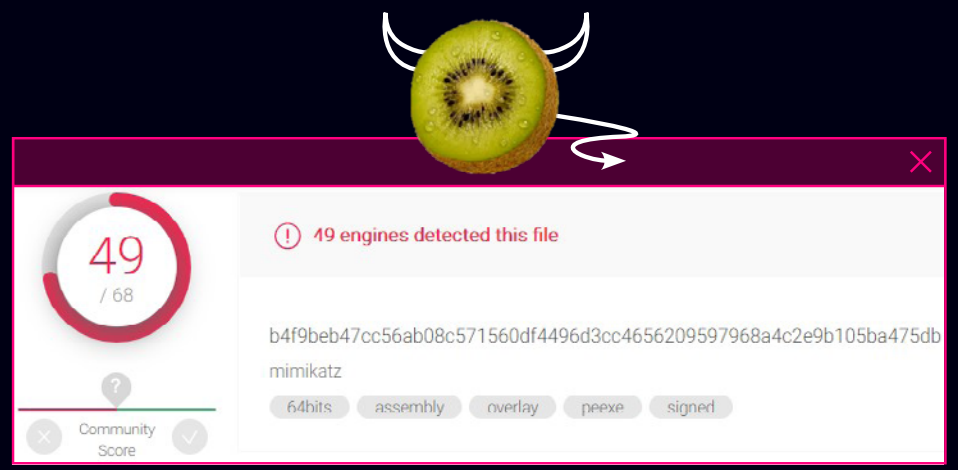
Running MALPROXY

ATTACKER SIDE

TARGET SIDE

Running MALPROXY

ATTACKER SIDE



TARGET SIDE

Running MALPROXY

ATTACKER SIDE

IMPORT ADDRESS TABLE

NtQuerySystemInformation	Kernel32.dll
OpenProcess	Kernel32.dll
ReadProcessMemory	Ntdll.dll
BCryptGenerateSymetricKey	Bcrypt.dll
ConvertSidToStringSidW	Advapi32.dll
...	...
RtlAdjustPrivilege	Ntdll.dll
NtQueryInformationProcess	Ntdll.dll
RtlEqualUnicodeString	Ntdll.dll

TARGET SIDE

Running MALPROXY

ATTACKER SIDE

IMPORT ADDRESS TABLE

NtQuerySystemInformation	Malproxy
OpenProcess	Malproxy
ReadProcessMemory	Malproxy
BCryptGenerateSymetricKey	Bcrypt.dll
ConvertSidToStringSidW	Advapi32.dll
...	...
RtlAdjustPrivilege	Malproxy
NtQueryInformationProcess	Malproxy
RtlEqualUnicodeString	Ntdll.dll

TARGET SIDE

Running MALPROXY

ATTACKER SIDE

RtlAdjustPrivilege
NtQuerySystemInformation
RtlEqualUnicodeString
OpenProcess
NtQueryInformationProcess
ReadProcessMemory
BCryptDecrypt

TARGET SIDE

Running MALPROXY

ATTACKER SIDE

RtlAdjustPrivilege
NtQuerySystemInformation
RtlEqualUnicodeString
OpenProcess
NtQueryInformationProcess
ReadProcessMemory
BCryptDecrypt

TARGET SIDE

RtlAdjustPrivilege

Running MALPROXY

ATTACKER SIDE

RtlAdjustPrivilege
NtQuerySystemInformation
RtlEqualUnicodeString
OpenProcess
NtQueryInformationProcess
ReadProcessMemory
BCryptDecrypt

Chrome.exe, explorer.exe
Calc.exe, lsass.exe

TARGET SIDE

RtlAdjustPrivilege
NtQuerySystemInformation

Running MALPROXY

ATTACKER SIDE

RtlAdjustPrivilege
NtQuerySystemInformation
RtlEqualUnicodeString
OpenProcess
NtQueryInformationProcess
ReadProcessMemory
BCryptDecrypt

TARGET SIDE

RtlAdjustPrivilege
NtQuerySystemInformation

Running MALPROXY

ATTACKER SIDE

RtlAdjustPrivilege
NtQuerySystemInformation
RtlEqualUnicodeString
OpenProcess
NtQueryInformationProcess
ReadProcessMemory
BCryptDecrypt

PID 1234

Handle 0x00000080

TARGET SIDE

RtlAdjustPrivilege
NtQuerySystemInformation

OpenProcess

Running MALPROXY

ATTACKER SIDE

RtlAdjustPrivilege
NtQuerySystemInformation
RtlEqualUnicodeString
OpenProcess
NtQueryInformationProcess
ReadProcessMemory
BCryptDecrypt

Handle 0x00000080

PEB at 0xdeadbeef

TARGET SIDE

RtlAdjustPrivilege
NtQuerySystemInformation

OpenProcess
NtQueryInformationProcess

Running MALPROXY

ATTACKER SIDE

RtlAdjustPrivilege
NtQuerySystemInformation
RtlEqualUnicodeString
OpenProcess
NtQueryInformationProcess
ReadProcessMemory
BCryptDecrypt

Read 0xdeadbeef

[0x12, 0x34, 0x56, 0x78]

TARGET SIDE

RtlAdjustPrivilege
NtQuerySystemInformation

OpenProcess
NtQueryInformationProcess
ReadProcessMemory

Running MALPROXY

ATTACKER SIDE

RtlAdjustPrivilege
NtQuerySystemInformation
RtlEqualUnicodeString
OpenProcess
NtQueryInformationProcess
ReadProcessMemory
BCryptDecrypt

TARGET SIDE

RtlAdjustPrivilege
NtQuerySystemInformation

OpenProcess
NtQueryInformationProcess
ReadProcessMemory

Running MALPROXY

ATTACKER SIDE

RtlAdjustPrivilege
NtQuerySystemInformation
RtlEqualUnicodeString
OpenProcess
NtQueryInformationProcess
ReadProcessMemory
BCryptDecrypt

TopSecretPassword

TARGET SIDE

RtlAdjustPrivilege
NtQuerySystemInformation
OpenProcess
NtQueryInformationProcess
ReadProcessMemory

PWNED!


DEMO

You came off as a naive idiot. and naive idiots are not a threat

Endpoint protections

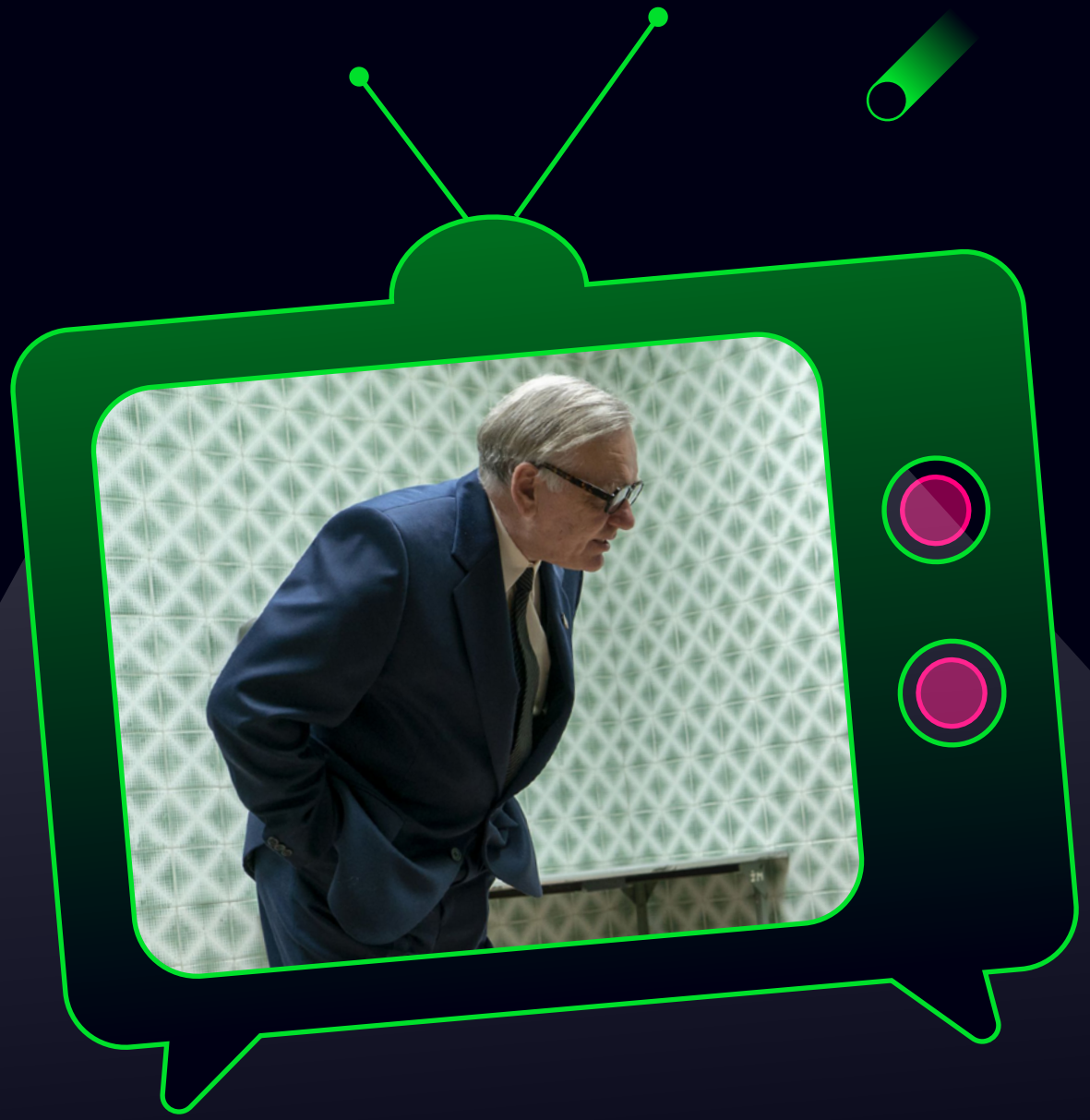
BYPASS

Bypassing
Static Signatures

Bypassing
Heuristic Rules

Behavioral
Signatures

Security Solution	Mimikatz sekurlsa::logonpasswords
Microsoft Defender	Malproxied!
Symantec Norton Security	Malproxied!
Kaspersky Internet Security	Blocks ReadProcessMemory without a verdict
ESET Smart Security	Malproxied!
Avast Free Antivirus	Blocks OpenProcess on lsass.exe without a verdict
Bitdefender Total Security	Malproxied!
McAfee Total Protection	Malproxied!



Why worry about something that isn't going to happen?

MITIGATIONS

- Hunt and sign the target-side proxy stub
- Improve the behavioral signature engines to handle their known weaknesses
- Any more ideas?

MITIGATIONS

- Hunt and sign the target-side proxy stub
- Improve the behavioral signature engines to handle their known weaknesses
- Any more ideas?
- /dev/null

CREDITS

The Crazy Ideas Section - Remote Syscalls by Yaron Shani:
http://breaking-the-system.blogspot.com/2016/06/the-crazy-ideas-section-remote-syscalls.html

Syscall Proxying - Simulating remote execution by Maximiliano Caceres:
http://www.vodun.org/papers/exploits/SyscallProxying.pdf

Syscall Proxying || Pivoting Systems by Filipe Balestra and Rodrigo Rubira Branco:
https://www.kernelhacking.com/rodrigo/docs/H2HCIII.pdf


Questions?