



Using Drivers

Caleb McGary



About Me

Red Teamer at MSFT

I am bad at twitter; I follow you, you shouldn't follow me

2 dogs:

- <https://aka.ms/ludwig>
- <https://aka.ms/wolfgang>

I enjoy coding, reading, astronomy, and occasionally my job



Opinions are **my own** and not representative of my employer.

Code

Slides and code samples here: GITHUB

Capabilities pushed to Faction: <https://github.com/FactionC2/>

- Made by Jared Haight: <https://twitter.com/jaredhaight>, <https://c2.lol>

Additional GitHub references:

<https://github.com/Microsoft/Windows-driver-samples>

DEMO

Let's do this live!



Brass Tacks

Evaluated our surface area from past attacks

We were not really doing much space with drivers

Decided to put some effort into playing with them as a result

This talk is generally **Windows heavy**, but I will cover a few aspects of Linux; no MacOS (ask someone smarter than me for MacOS)

Serpent Process

Look at what attackers are doing

See where we can replicate

Try it out

Fail often / fail fast

20 percent time project (for most innovations)

In the News

Mysterious Avatar rootkit with API, SDK, and Yahoo Groups for C&C communication

[Home](#) > [Malware](#)

TODAY'S TOP STORIES

What is Stuxnet,

Chinese-speaking APT LuckyMouse uses malicious NDISProxy drivers to distribute Trojans

September 11, 2018 | Threat Actors



[Blog](#) >

Duqu Malware Techniques Uncovered

Duqu 2.0 is a [really impressive](#) piece of malware,

Kernel Write-What-Where in Qualcomm Driver == LPE

APT REPORTS

Shedding Skin – Turla's Fresh Faces

'Screwed Drivers': Driver Vulnerabilities Affect Intel, AMD, Other Vendors

These newly disclosed vulnerabilities can allow attackers to install malware directly on device firmware, giving malicious software the ability to remain on the device, even after the operating system has been reinstalled, according to Intel-backed security startup Eclipsium.

Attackers



Actions on Objectives

- Process Injection, Lateral Movement, etc

Privilege Escalation

- User to System

Persistence

EDR Evasion

- Disable or disarm defensive measures present

Where **we** can replicate

Persistence

- This helps us; but requires substantial effort (we can't leave a serious bug around)
- We are testing this option out at the moment, using internally developed tooling

EDR Evasion

- This is much more useful, as we can limit our exposure (driver only needs to be loaded for short period of time)
- Does remove telemetry from box, but doesn't necessarily weaken security posture of system

Choose Your Own Adventure

Use an existing driver

- Discovery required
- More challenging to control
- Stealthier
- Tougher to remediate / be evicted

Build your own driver

- Need a code sign certificate
- Need to write one
- Powerful, full control over features and functionality
- Potentially easier to detect

Use an existing driver

Possibly the easiest and most common thing

Use an existing driver

Look for:

- Stuff that is SPC signed
 - WQHL vs SPC
- Things **outside** of systemroot/system32/drivers (if 64 bit)
- Things for older versions of the OS

Enumeration

What drivers are installed on my system?

(Windows Demo) – driverquery /?

- driverquery /S system /U domain\user /P password /FO LIST
- pnputil /enum-drivers
- Double Driver Freeware to back stuff up:
<http://boozet.org/download.htm>
- Device Tree – legacy tool

(Linux Demo) – lsmod or cat /proc/modules

Double Driver [Backup]

Name	Version	Date	Provider	Class	Setup Information	Setup Section	Hardware ID
<input type="checkbox"/> Local Print Queue	10.0.18362.1	6-21-2006	Microsoft	PrintQueue	printqueue.inf	NO_DRV_LOCAL	PRINTENUM\LocalPrintQueue
<input type="checkbox"/> USB xHCI Compliant Host Controller	10.0.18362.207	6-19-2019	Microsoft	USB	usbhcd.inf	Generic.Install.NT	PCI\CC_0C0330
<input type="checkbox"/> Standard Enhanced PCI to USB H...	10.0.18362.1	6-21-2006	Microsoft	USB	usbport.inf	BHCI.Dev.NT	PCI\CC_0C0320
<input type="checkbox"/> USB Root Hub	10.0.18362.1	6-21-2006	Microsoft	USB	usbport.inf	ROOTHUB.Dev.NT	USBROOT_HUB20
<input type="checkbox"/> USB Root Hub (USB 3.0)	10.0.18362.1	3-18-2019	Microsoft	USB	usbhub3.inf	Generic.Install.NT	USBROOT_HUB30
<input type="checkbox"/> USB Mass Storage Device	10.0.18362.1	6-21-2006	Microsoft	USB	usbstor.inf	USBSTOR_BULK.NT	USB\Class_0886SubClass_068Prot_50
<input type="checkbox"/> USB Composite Device	10.0.18362.1	6-21-2006	Microsoft	USB	usb.inf	Composite.Dev.NT	USB\COMPOSITE
<input type="checkbox"/> Generic USB Hub	10.0.18362.1	6-21-2006	Microsoft	USB	usb.inf	StandardHub.Dev.NT	USB\Class_09
<input type="checkbox"/> ACPI x64-based PC	10.0.18362.1	6-21-2006	Microsoft	Computer	hal.inf	ACPI_MODEL_HAL	acpi.inf
<input type="checkbox"/> Disk drive	10.0.18362.1	6-21-2006	Microsoft	DiskDrive	disk.inf	disk_instal.NT	GenDisk
<input type="checkbox"/> Disk drive	10.0.18362.1	6-21-2006	Microsoft	DiskDrive	disk.inf	disk_instal_VHD_drive.NT	SCSI\DiskMft_Virtual_Disk_
<input checked="" type="checkbox"/> 67EF:CF	22.19.162.4	4-24-2017	Advanced Micro De...	Display	oem6.inf	ati2mtag_Polaris1105	pci\ven_1002&dev_67ef&rev_cf
<input type="checkbox"/> Standard SATA AHCI Controller	10.0.18362.1	6-21-2006	Microsoft	HDC	mshdc.inf	msahci_inst	PCI\CC_010601
<input type="checkbox"/> HID Keyboard Device	10.0.18362.1	6-21-2006	Microsoft	Keyboard	keyboard.inf	HID_Keyboard.Inst.NT	HID_DEVICE_SYSTEM_KEYBOARD
<input checked="" type="checkbox"/> AMD High Definition Audio Device	10.0.1.6	11-16-2017	Advanced Micro De...	MEDIA	oem9.inf	HDAudio\install	HDAUDIO\FUNC_01&VEN_1002&DEV_AA01&SUBSYS_00AA0100&REV_1007
<input type="checkbox"/> USB Audio Device	10.0.18362.1	3-18-2019	Microsoft	MEDIA	wdma_usb.inf	USBAudio	USB\Class_01
<input checked="" type="checkbox"/> Realtek Audio	6.0.1.6.111	8-12-2016	Realtek Semicondu...	MEDIA	oem4.inf	AzAudModelASio.NTamd64	hdaudio\func_01&ven_10ec&dev_0280&subsys_10280617
<input type="checkbox"/> Microsoft Streaming Clock Proxy	10.0.18362.1	6-21-2006	Microsoft	MEDIA	ksfilter.inf	MSPCLOCK.NT	SW\{97ebacc-95bd-11d0-a3ea-00a0c9223196}
<input type="checkbox"/> Microsoft Streaming Service Proxy	10.0.18362.1	6-21-2006	Microsoft	MEDIA	ksfilter.inf	MSISRV.NT	SW\{96E080C7-143C-11D1-840F-00A0C9223196}
<input type="checkbox"/> Microsoft Streaming Tee/Link Mana...	10.0.18362.1	6-21-2006	Microsoft	MEDIA	ksfilter.inf	MTSrv.NT	SW\{6d669f1-9ac3-11d0-8299-0000f822efba}
<input type="checkbox"/> Microsoft Streaming Quality Mana...	10.0.18362.1	6-21-2006	Microsoft	MEDIA	ksfilter.inf	MSPCM.NT	SW\{D0F4358E-8B3C-11D0-A42F-00A0C9223196}
<input type="checkbox"/> Microsoft Trusted Audio Drivers	10.0.18362.1	3-18-2019	Microsoft	MEDIA	wdmaudio.inf	WDM_DRMKAUD	SW\{EEC12D86-AD9C-4168-8658-803DAEF417FE}
<input type="checkbox"/> Generic PnP Monitor	10.0.18362.207	6-21-2006	Microsoft	Monitor	monitor.inf	PnPMonitor.Install	*PNP09FF
<input type="checkbox"/> Generic Non-PnP Monitor	10.0.18362.207	6-21-2006	Microsoft	Monitor	monitor.inf	NonPnPMonitor.Install	MONITOR\Default_Monitor
<input type="checkbox"/> HID-compliant mouse	10.0.18362.1	6-21-2006	Microsoft	Mouse	msmouse.inf	HID_Mouse.Inst.NT	HID_DEVICE_SYSTEM_MOUSE
<input type="checkbox"/> Microsoft Kernel Debug Network ...	10.0.18362.1	6-21-2006	Microsoft	Net	kdnic.inf	root\kdnic	root\kdnic
<input checked="" type="checkbox"/> Intel(R) Ethernet Connection I21...	12.13.17.4	8-4-2015	Intel	Net	oem11.inf	E153A.10.0.1	pci\ven_8086&dev_153a&subsys_06171028
<input type="checkbox"/> Hyper-V Virtual Switch Extension ...	10.0.18362.1	6-21-2006	Microsoft	Net	vwms_rnp.inf	VMSVMP.nd	vms_vmp
<input type="checkbox"/> Hyper-V Virtual Ethernet Adapter	10.0.18362.1	6-21-2006	Microsoft	Net	vwms_rnp.inf	VMSMP.nd	vms_vmp
<input type="checkbox"/> Communications Port	10.0.18362.1	6-21-2006	Microsoft	Ports	msports.inf	ComPort.NT	*PNP0501
<input type="checkbox"/> Standard iWMM Express Controller	10.0.18362.1	6-21-2006	Microsoft	SCSIAdapter	stormme.inf	Stormme.Inst	PCI\CC_010802
<input type="checkbox"/> Microsoft Storage Spaces Controller	10.0.18362.1	6-21-2006	Microsoft	SCSIAdapter	spaceport.inf	Spaceport.Install	Root\Spaceport
<input type="checkbox"/> Microsoft VHD Loopback Controller	10.0.18362.356	6-21-2006	Microsoft	SCSIAdapter	vhdmnp.inf	vhdmnp_inst	{8e7d593-6e6c-4c52-86a6-77175494dd8e}\VtVhdRba
<input type="checkbox"/> Microsoft Hyper-V Virtualization I...	10.0.18362.387	6-21-2006	Microsoft	System	vvid.inf	Vid_Device_Client.NT	ROOT\VID
<input type="checkbox"/> Composite Bus Enumerator	10.0.18362.329	6-21-2006	Microsoft	System	compositebus.inf	CompositeBus_Device.NT	ROOT\CompositeBus
<input type="checkbox"/> UMBS Root Bus Enumerator	10.0.18362.329	6-21-2006	Microsoft	System	umbus.inf	UmBusRoot_Device.NT	root\umbus
<input type="checkbox"/> NDIS Virtual Network Adapter En...	10.0.18362.1	6-21-2006	Microsoft	System	ndsvirtualbus.inf	NdisVirtualBus_Device.NT	ROOT\NdisVirtualBus
<input type="checkbox"/> Plug and Play Software Device En...	10.0.18362.329	8-27-2019	Microsoft	System	svenum.inf	SWENUM	ROOT\SWENUM
<input type="checkbox"/> Remote Desktop Device Redirect...	10.0.18362.1	6-21-2006	Microsoft	System	rdpbus.inf	RDPBUS	ROOT\RDPBUS
<input type="checkbox"/> Microsoft ACPI-Compliant System	10.0.18362.329	6-21-2006	Microsoft	System	acpi.inf	ACPI.Inst.NT	*PNP0C08
<input type="checkbox"/> PCI Bus	10.0.18362.418	6-21-2006	Microsoft	System	pci.inf	PCI_ROOT	*PNP0A03
<input type="checkbox"/> ACPI Module Device	10.0.18362.267	6-21-2006	Microsoft	System	machine.inf	NO_DRV_X_PNP	*ACPI0004
<input type="checkbox"/> Microsoft Windows Management I...	10.0.18362.1	6-21-2006	Microsoft	System	wmicapi.inf	WMIQM_Hst.NT	*PNP0C14
<input type="checkbox"/> PCI Express Root Complex	10.0.18362.418	6-21-2006	Microsoft	System	pci.inf	PCI_ROOT	*PNP0A08
<input type="checkbox"/> ACPI Power Button	10.0.18362.267	6-21-2006	Microsoft	System	machine.inf	NO_DRV	*PNP0C0C
<input type="checkbox"/> ACPI Fixed Feature Button	10.0.18362.267	6-21-2006	Microsoft	System	machine.inf	NO_DRV	ACPI\FixedButton
<input checked="" type="checkbox"/> Intel(R) Xeon(R) E7 v4/Xeon(R) E...	10.1.2.19	1-26-2016	INTEL	System	oem5.inf	Needs_NO_DRV	PCI\VEN_8086&DEV_6F00
<input checked="" type="checkbox"/> Intel(R) Xeon(R) E7 v4/Xeon(R) E...	10.1.2.19	1-26-2016	INTEL	System	oem5.inf	Needs_PCI_DRV	PCI\VEN_8086&DEV_6F02
<input checked="" type="checkbox"/> Intel(R) Xeon(R) E7 v4/Xeon(R) E...	10.1.2.19	1-26-2016	INTEL	System	oem5.inf	Needs_PCI_DRV	PCI\VEN_8086&DEV_6F03
<input checked="" type="checkbox"/> Intel(R) Xeon(R) E7 v4/Xeon(R) E...	10.1.2.19	1-26-2016	INTEL	System	oem5.inf	Needs_PCI_DRV	PCI\VEN_8086&DEV_6F04
<input checked="" type="checkbox"/> Intel(R) Xeon(R) E7 v4/Xeon(R) E...	10.1.2.19	1-26-2016	INTEL	System	oem5.inf	Needs_PCI_DRV	PCI\VEN_8086&DEV_6F05

Scan Current System Scan Other System

PS D:\demo>

Example

The image shows a Windows Device Manager window with the properties of a device named `\Device\WinRing0_1_2_0`. The device is of type `0x9c40`. The driver is `\Driver\WinRing0_1_2_0`. The device object is `0xFFFF91801A51AD20` and the driver object is `0xFFFF91801A519D40`. The next device is `0x0000000000000000`. The device type is `0x9c40` and the stack size is `1`. The DPC importance is `0x0`, the DPC routine is `0x0000000000000000`, and the DPC number is `0x0`.

Below the device properties, there is a code editor showing the following code:

```
84 #define IOCTL_OLS_READ_MEMORY \
85     CTL_CODE(OLS_TYPE, 0x841, METHOD_BUFFERED, FILE_READ_ACCESS)
86
87 #define IOCTL_OLS_WRITE_MEMORY \
88     CTL_CODE(OLS_TYPE, 0x842, METHOD_BUFFERED, FILE_WRITE_ACCESS)
```

The device properties window also shows the following fields:

- Instance Id:
- Vendor:
- Hardware Ids:
- Compatible Ids:
- Device Capabilities: ☐ DeviceD1, ☐ DockDevice, ☐ RawDeviceOK, ☐ WakeFromD0, ☐ HardwareDisabled, ☐ DeviceD2, ☐ UniqueID, ☐ SurpriseRemovalOK, ☐ WakeFromD1, ☐ NonDynamic, ☐ LockSupported, ☐ Removable, ☐ SilentInstall, ☐ WakeFromD2, ☐ WarmEjectSupported, ☐ EjectSupported, ☐ WakeFromD3, ☐ NoDisplayInUI
- Address:
- SystemWake:
- UI Number:
- Device State:

A security dialog box is open, showing the security settings for the device. The dialog box is titled `\Device\WinRing0_1_2_0 Properties` and has a `Security` tab. It shows a list of users and groups, including `Administrators (DESKTOP-AJQ04M9\Administrators)`. The dialog box also has `Add...` and `Remove` buttons. Below the list, there is a table showing permissions for `Everyone`, `Allow`, and `Deny`.

	Allow	Deny
Read Access	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Modify Access	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Delete Access	<input checked="" type="checkbox"/>	<input type="checkbox"/>
All Access	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Special permissions	<input checked="" type="checkbox"/>	<input type="checkbox"/>

At the bottom of the dialog box, there is a button labeled `Advanced` and a note: `For special permissions or advanced settings, click Advanced.` The dialog box also has `OK`, `Cancel`, and `Apply` buttons.

What you need to know

```
bResult = DeviceIoControl(hDevice,           // device to be queried
                          IOCTL_DISK_GET_DRIVE_GEOMETRY, // operation to perform
                          NULL, 0,           // no input buffer
                          pdg, sizeof(*pdg), // output buffer
                          &junk,             // # bytes returned
                          (LPOVERLAPPED) NULL); // synchronous I/O

CloseHandle(hDevice);
```

How to call Device

```
NULL,
OPEN_EXISTING,
FILE_ATTRIBUTE_NORMAL,
NULL
);

&isb,
FILE_SHARE_READ | FILE_SHARE_WRITE,
FILE_NON_DIRECTORY_FILE
);
```


Tooling to help

Device Tree: <http://www.osronline.com/article.cfm%5earticle=97.htm>

IRPTracker: <http://www.osronline.com/article.cfm%5earticle=199.htm>

IOCTL Fuzzer: <https://code.google.com/archive/p/ioctlfuzzer/>

IOCTLBF: <https://github.com/koutto/ioctlbf>

WinObj: <https://docs.microsoft.com/en-us/sysinternals/downloads/winobj>

Windows Object Explorer 64bit: <https://github.com/hfiref0x/WinObjEx64>

Driver capabilities you want

Read / Write Memory

Allocate Memory

Accessible by Everyone

- If you're hunting privilege escalation 0-day

Existing Useful Drivers



WinRingO_1_*

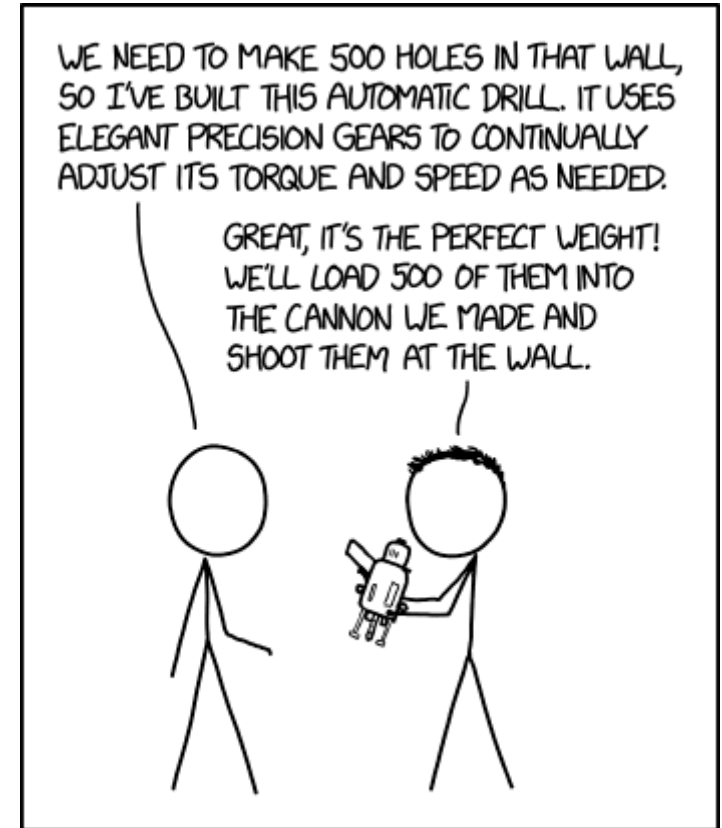
ProcessHacker

- Many other types of “older system / perf monitoring software”

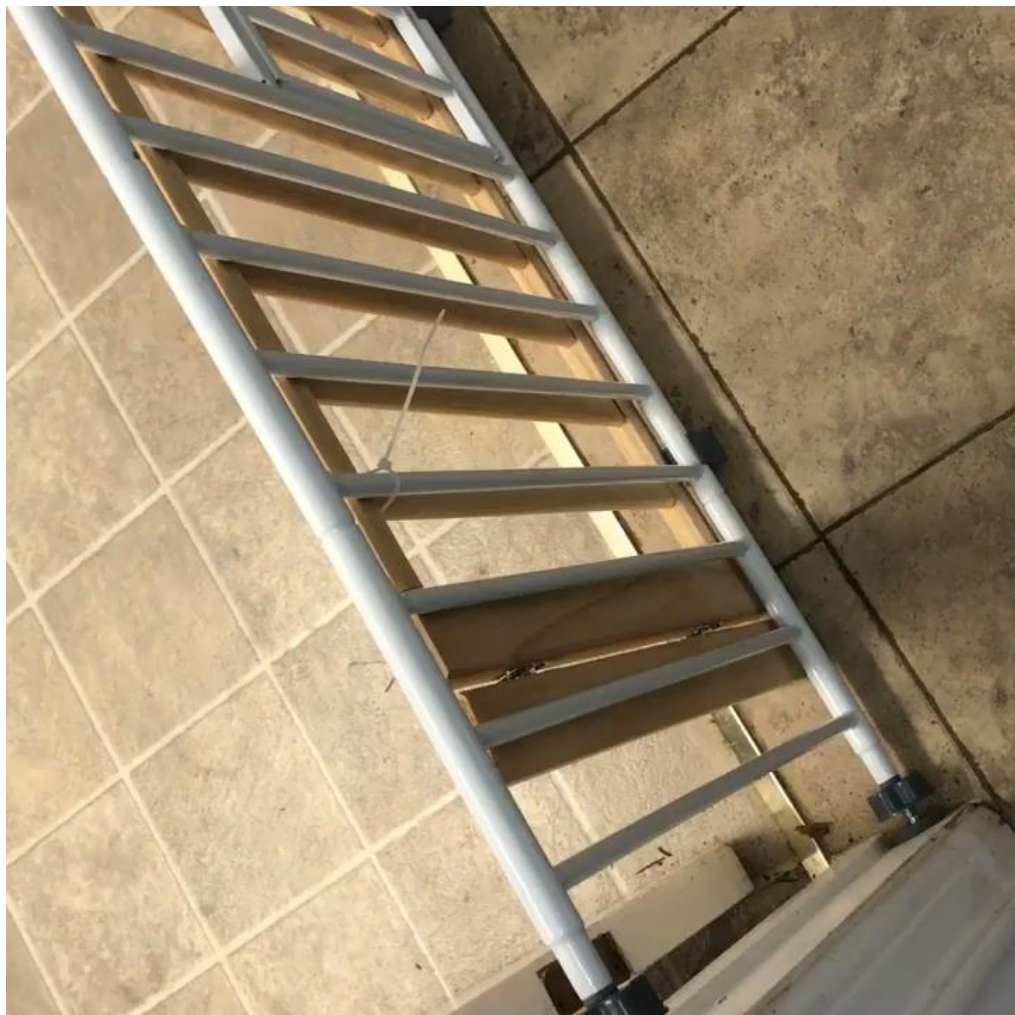
VirtualBox (older versions)

BYOD

Disclaimer: this is NOT a comprehensive course



HOW SOFTWARE DEVELOPMENT WORKS



My preferred
route

Windows: Drivers

Kernel Mode Drivers (Windows Driver Model - WDM)

- Plug & Play, I/O (filter drivers), Security, OS stuff
- Chaining -> Higher Level Drivers (NTFS) call lower level drivers (filter drivers) call lowest level drivers (hardware / legacy)

User Mode Drivers (UMDF)

- User-mode environment ☹️
- File system, display (full devices) and print drivers **CANNOT** be UMDF

Windows: Drivers

Signing:

- Not required for 32-bit versions of Windows
- [Cross Certificates Overview](#) (short)
- Buy one, don't spend more than \$200.
- Timestamp Certificates are also useful
- Pay **attention** to who is in the valid list; in 2021 this changes

Driver Signing Certificate

Starting at

\$199.99 /yr

\$199.99/yr when you renew⁴

Add to Cart

Required for all Microsoft[®] hardware drivers on Windows Vista[®] and Windows 7

Validates and secures your code

Eliminates security warnings during download and installation

Provides high-grade SHA-2 encryption

✓ Reinforces security with cross-certificate validation



Linux: Drivers

Generally part of Kernel

- Often a kernel module
 - [Walkthrough](#)
- Often proprietary ☹️
- Can be written in a variety of things, just **not C++** (Rust, C, Assembler, etc)

Install Shim Required

Demo

- Loading Process Hacker KMDF
 - [CreateService](#)
 - [Start Service](#)
 - [NTOpenFile](#)
 - This is **noisy**, but often undetected (loading a USB drive loads a driver)
- Code sample on github; you can probably do it without creating a service btw, [pnputil](#) + rundll32 is an option

Features you will want

Read / Write / Allocate Memory

- Possibly from nonpaged pool (rather than NX pool)
- Buffered I/O -> for working with user land

Windows: Process Context Switching

Windows: File System operations

Windows: Security - <https://github.com/Microsoft/SymCrypt>

Generic: System / method call hooking

Tooling

WDF Verifier: debugging drivers

WDK (DDK): this is your core toolset if you write your own

Device Fundamentals Tests: black box tests; need a testing box

- This finds lot of things; many are not security specific
- Penetration Tests (Fuzz Tests and I/O Spy + I/O Attack)

Visual Studio (worth it)

Detections

Talk about EventLog

Event ID: 7045

Service was installed on your system

- One-time event
- Child devices don't always show up if you find a vulnerable pnpdevice
- Unload does **not** generate an event

ELAM **should** catch you

- If it has your driver signed

Event 7045, Service Control Manager

General Details

A service was installed in the system.

Service Name: ProcessHacker

Service File Name: C:\Users\camcg\AppData\Local\Temp\\kprocesshacker.sys

Service Type: kernel mode driver

Service Start Type: demand start

Service Account:

Log Name: System

Source: Service Control Manager

Event ID: 7045

Level: Information

User: ALNILAM\camcg

OpCode: Info

More Information: [Event Log Online Help](#)

Logged: 11/7/2019 12:15:14 PM

Task Category: None

Keywords: Classic

Computer: Alnilam

Conclusion

That's it y'all

Being inside the inner ring walls

EDR evasion / destruction / what have you

- You may have to win a race condition

Persistence

- Load on boot (call DriverEntry routine)
- Reference binary from hidden filesystem, etc
- Go to firmware and hide there

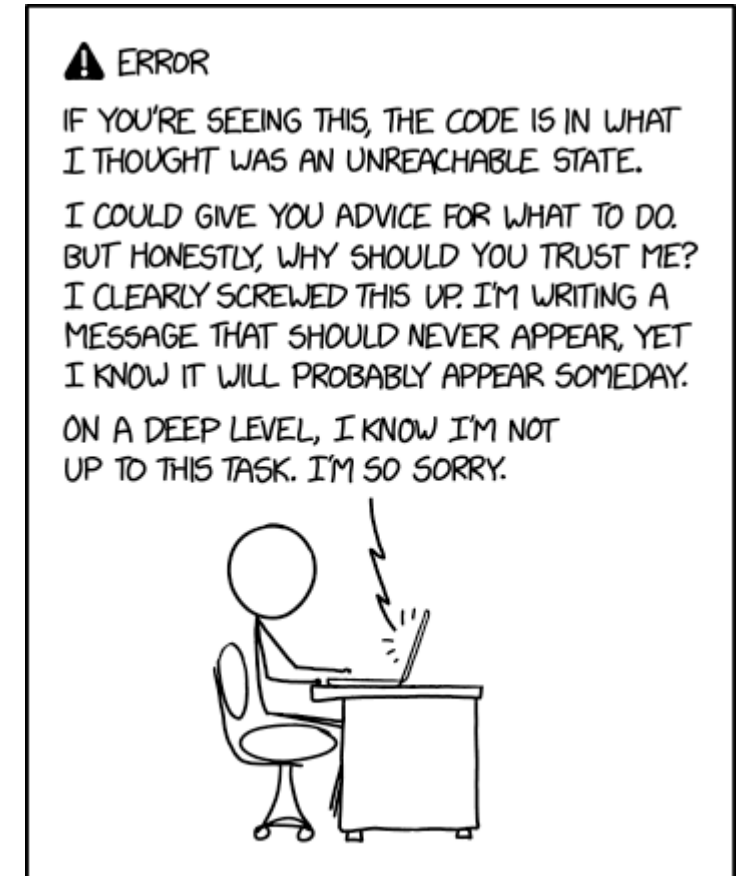
Actions on Objectives

- This may or may not meet your testing model

Happy Hacking

Slides and source: <https://aka.ms/UsingDrivers>

Contact: caleb.mcgary@gmail.com



NEVER WRITE ERROR MESSAGES TIRED.

Backup Screenshots

In case of demo fail



Filter agents..

41_JXcetcin8

Admin

FactionWin10\caleb

FactionWin10 (PID: 1308)

DIRECT Transport

11/07 8:02:25 PM

ping	Pings a computer or list of computers	winlib	true
portscan	Scan port(s) on computer(s)	winlib	true
powershell	Execute a PowerShell command through .NET. Doesn't launch powershell.exe and by default will attempt to bypass AMSI and ScriptBlock logging	winlib	true
reg	Work with the registry	winlib	true
services	List Services on local or remote computers	winlib	true
wmi	Execute a WMI query	winlib	true
driver	Work with Kernel Drivers	winlib	true
assembly	Loads and executes a .NET assembly	stdlib	true
cd	Change the directory you're operating out of	stdlib	true
download	Downloads a file from Faction to the Agent	stdlib	true
ls	List the contents of a directory or path	stdlib	true
mkdir	Create a directory	stdlib	true
ps	List processes	stdlib	true
pwd	Returns the directory that the agent is currently	stdlib	true

Send

Filter agents..

41_JXcetcin8

Admin

👤 FactionWin10\caleb

🖥️ FactionWin10 (PID: 1308)

🌐 DIRECT Transport

🕒 11/07 8:07:37 PM

127.0.0.1true

[7:32:00 PM] [admin]

F2> driver /Operation:"Enumerate"

[7:32:06 PM] [41_JXcetcin8] [#31]

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
\	S	y	s	t	e	m	R	o	o	t	\	s	y	s	t	e	m	3	2	\	n	t	o	s	k	r
\	S	y	s	t	e	m	R	o	o	t	\	s	y	s	t	e	m	3	2	\	h	a	l	.	d	l
\	S	y	s	t	e	m	R	o	o	t	\	s	y	s	t	e	m	3	2	\	k	d	.	d	l	l
\	S	y	s	t	e	m	R	o	o	t	\	s	y	s	t	e	m	3	2	\	m	c	u	p	d	a
\	S	y	s	t	e	m	R	o	o	t	\	S	y	s	t	e	m	3	2	\	d	r	i	v	e	r
\	S	y	s	t	e	m	R	o	o	t	\	S	y	s	t	e	m	3	2	\	d	r	i	v	e	r
\	S	y	s	t	e	m	R	o	o	t	\	S	y	s	t	e	m	3	2	\	d	r	i	v	e	r
\	S	y	s	t	e	m	R	o	o	t	\	S	y	s	t	e	m	3	2	\	d	r	i	v	e	r
\	S	y	s	t	e	m	R	o	o	t	\	S	y	s	t	e	m	3	2	\	d	r	i	v	e	r
\	S	y	s	t	e	m	R	o	o	t	\	s	y	s	t	e	m	3	2	\	P	S	H	E	D	.

Send