📖 What is BYOVD?

Even though Windows enforces **Driver Signature Enforcement (DSE)** and protections like **PatchGuard** and **HVCI**, attackers exploit **digitally signed but vulnerable drivers** to:

- Kill EDR processes.
- Access kernel memory.
- Bypass user-mode security protections.

These drivers are "**brought in**" (not already present), hence the term "**Bring Your Own**".

⚙️ How It Works (Simple Flow)

1. Attacker drops a **vulnerable driver** (e.g., TfSysMon.sys) onto disk.

Loads it into kernel mode using:

```
sc.exe create <ServiceName> binPath= <PathToDriver> type= kernel
sc.exe start <ServiceName>
```

2.
3. Uses the driver's features or bugs (often **IOCTL codes**) to:
    - Kill protected EDR processes.
    - Access or tamper with kernel memory.
    - Perform privilege escalation.

🔍 Real-World Examples (Observed Cases)

✅ Case 1: TfSysMon.sys (ThreatFire System Monitor)

- Dropped by: WatchMgrsCore.exe
- Attempted EDR kill via IOCTL.
- Masqueraded as: C:\Windows\WatchMgrsCore.sys
- Blocked due to Falcon's IOCTL protection.

✅ Case 2: RTCore64.sys

Loaded using:

```
 pgsql
CopyEdit
sc.exe create RTCore64 binPath=C:\Windows\Temp\RTCore64.sys
type=kernel
```

-
- Observed use for terminating protected processes.

✅ Case 3: szkg64.sys

Command used:

```
 pgsql
CopyEdit
sc.exe create szkg64 binPath=C:\Windows\Temp\szkg64.sys type=kernel
```

-
- Attempted to install as a kernel service.

🚩 Red Flags & Tactics

🔧 Suspicious IOCTL Use

- IOCTL (Input Output Control) is used to communicate with drivers.
- Adversaries send **crafted IOCTL requests** to vulnerable drivers to perform unauthorized actions.
- Example: Kill EDR agent processes.

🎭 Masquerading Drivers

- Drivers are renamed to look like legitimate ones.
    - TfSysMon.sys → WatchMgrsCore.sys
    - TrueSight.sys → truepath.sys
- Hides malicious intent from analysts and security tools.

👁️ Detection Logic

🧪 KQL (Microsoft Defender 365 or Sentinel)

```
DeviceProcessEvents
| where FileName =~ "sc.exe"
| where ProcessCommandLine has_all ("create", "type=kernel") or
ProcessCommandLine has ".sys"
| project Timestamp, DeviceName, InitiatingProcessFileName,
ProcessCommandLine
```

```
kql

DeviceFileEvents
| where FileName endswith ".sys"
| where InitiatingProcessFileName =~ "explorer.exe"
| project Timestamp, DeviceName, FileName, FolderPath,
InitiatingProcessFileName
```

📌 Sigma Rule

```
title: Suspicious Kernel Driver Creation via sc.exe
id: f23495c8-2025-45b4-b3e4-bc11cb02c7f5
description: Detects attempts to load kernel drivers using sc.exe
with type=kernel
status: stable
logsource:
  category: process_creation
  product: windows
detection:
  selection:
    Image|endswith: '\sc.exe'
    CommandLine|contains:
      - 'create'
      - 'type= kernel'
  condition: selection
fields:
  - CommandLine
  - Image
  - ParentImage
falsepositives:
  - Admin or IT team actions (baseline required)
level: high
tags:
  - attack.defense_evasion
  - attack.t1068
  - attack.t1216
```