

Evading Antivirus on Disk



William Hardy
Security Engineer



Module Overview



Custom shellcode runner

Metasploit exe and exe-only formats

**Hiding the stager using an encoder or
encryptor**

Modifying the stager shellcode

Confusing AI/ML models

Demo



Custom Shellcode Runner



The stager can be embedded inside the shellcode runner

Stealthy



Metasploit exe Format



Can be used by passing the `-f exe` flag to `msfvenom`

If the `-x` flag is not specified, the default templates are used

Makes use of a small stub of shellcode and `VirtualAlloc`.

It copies the stager inside the section at runtime



Metasploit exe-only Format



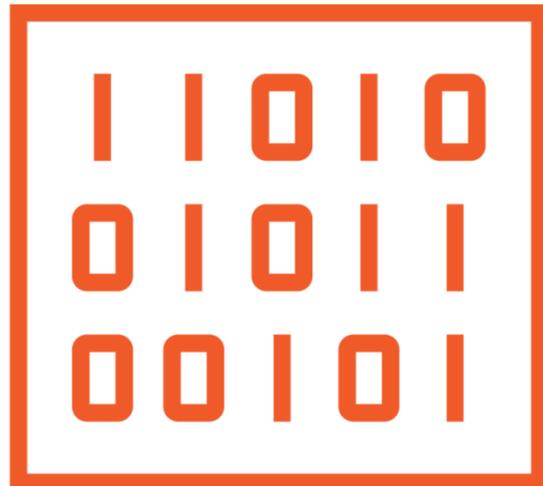
Can be used by passing the `-f exe-only` flag to `msfvenom`

If the `-x` flag is not specified, the default templates are used

Inserts the stager at the entry-point of the executable file



Encoders and Encryptors



Stagers contain hardcoded shellcode which is ideal for developing signatures

Metasploit ships with various encoders and encryptors

Encoders can be used by passing the `-e` flag to `msfvenom`

Encryptors can be used by passing the `--encrypt` and `--encrypt-key` flags

Some are polymorphic, such as the `Zutto_Dekiru` encoder which are hard to detect



```
push    rbx
mov     r14, 'crypt32'
push    r14
mov     rcx, rsp
mov     r10, 0xABCDEF
call   rbp
```

Stager is compiled

```
534C8B3425000000041564
889E149C7C2EFCDAB00FFD5
```

Compiled stager is sent to the encoder

Random 8-byte key

Zutto-Dekiru encoder

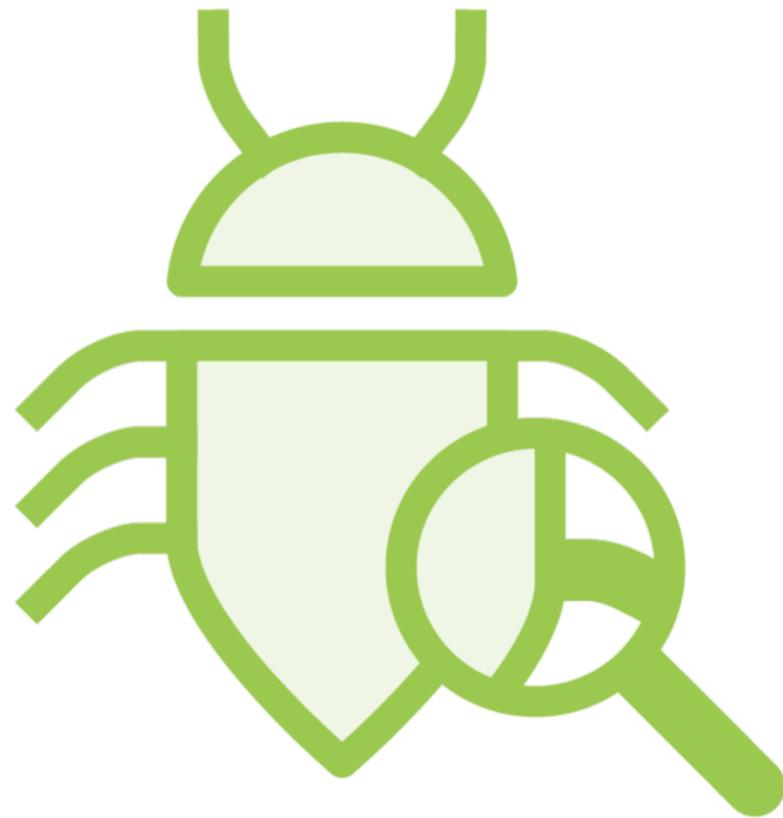
Zutto-Dekiru will perform an XOR operation 8-bytes at a time with a random 8-byte key

```
1E5A840041F3BCE932AEBDC
50504831E14C21F1415D415
C4C21F1415D415C504831E1
4C21F1415D
```

A polymorphic decoder stub is added in front of the encoded stager



Modifying the Stager Shellcode



An alternative to using an encoder or encryptor

Not as generic as an encoder

More complex

More time-consuming



Confusing AI/ML Models



Avoid having too few imports in the executables IAT



Avoid having frequently-abused APIs in the executables IAT



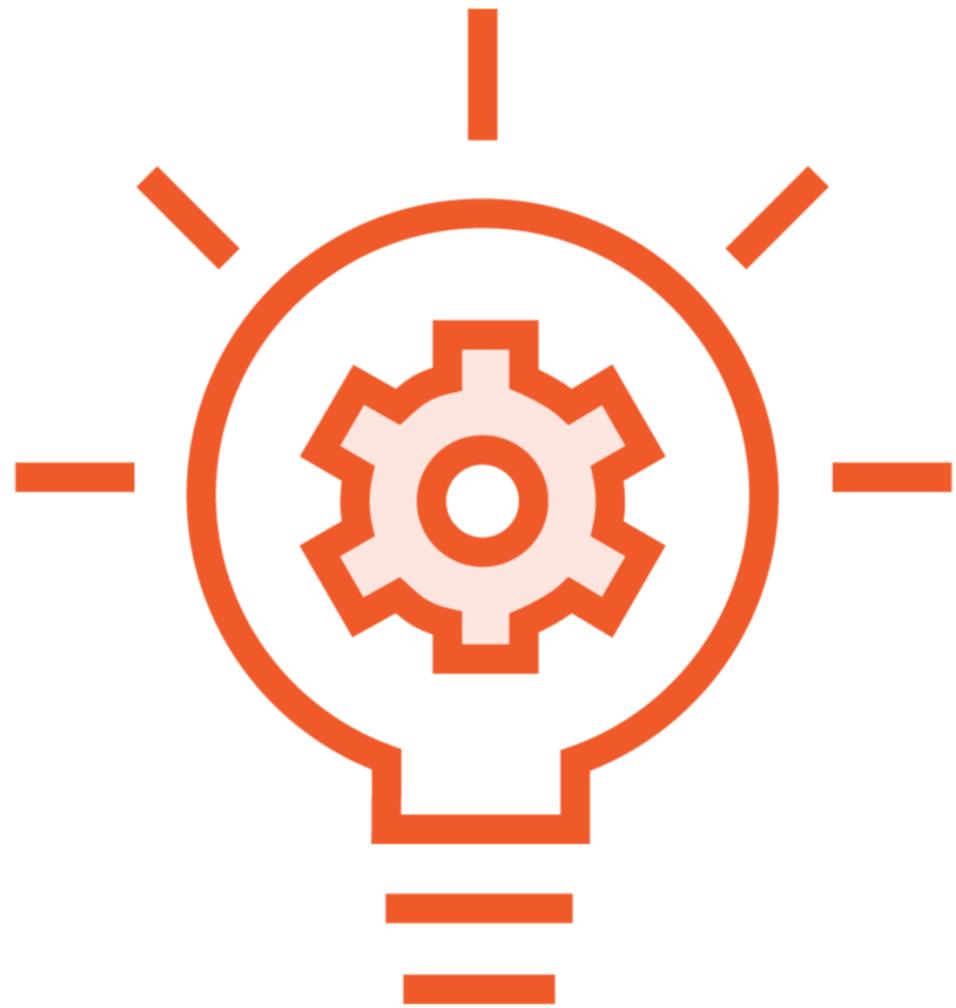
Avoid RWX permissions on memory sections



Avoid high-entropy inside RWX memory sections



Writing Your Own Stager



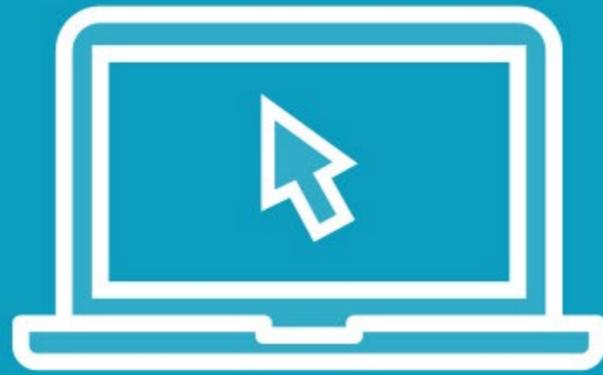
Using known tools to build your stagers can make them easier to detect

Writing your own stagers is relatively simple

Gives you more control



Demo



Evading Windows Defender

- Writing a custom stager
- Evading antivirus on disk

