How secure am I with EMET?

Benedikt Tröster
btroester@ernw.de

ERNW GmbH

- IT-Security Service Provider
- Vendor-independent
- Based in Heidelberg
- Founded in 2001
- 40 Employees

- Troopers (www.troopers.de)
  - We invite you to come to Heidelberg ;)

ERNW
providing security.

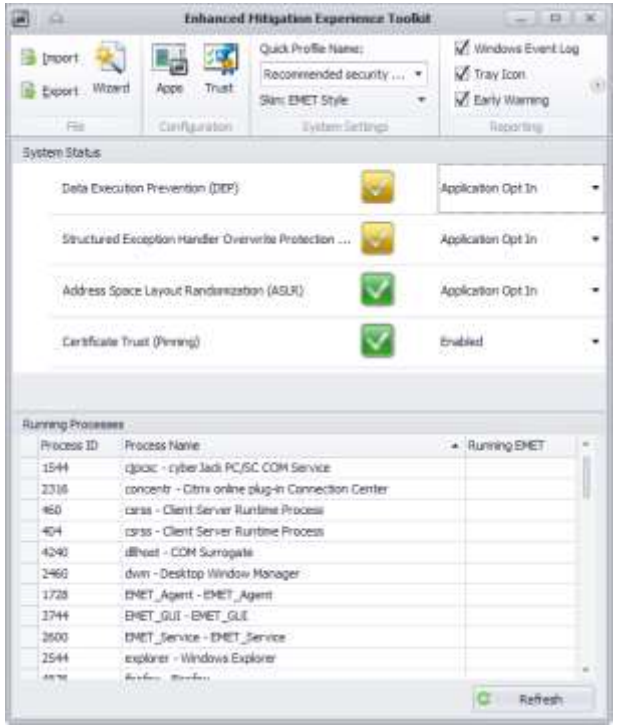
Agenda

- About EMET
- Mitigation Techniques
- Demo
- EMET (5.1) Bypassing
- Demo
- Wrap-up

What is EMET?

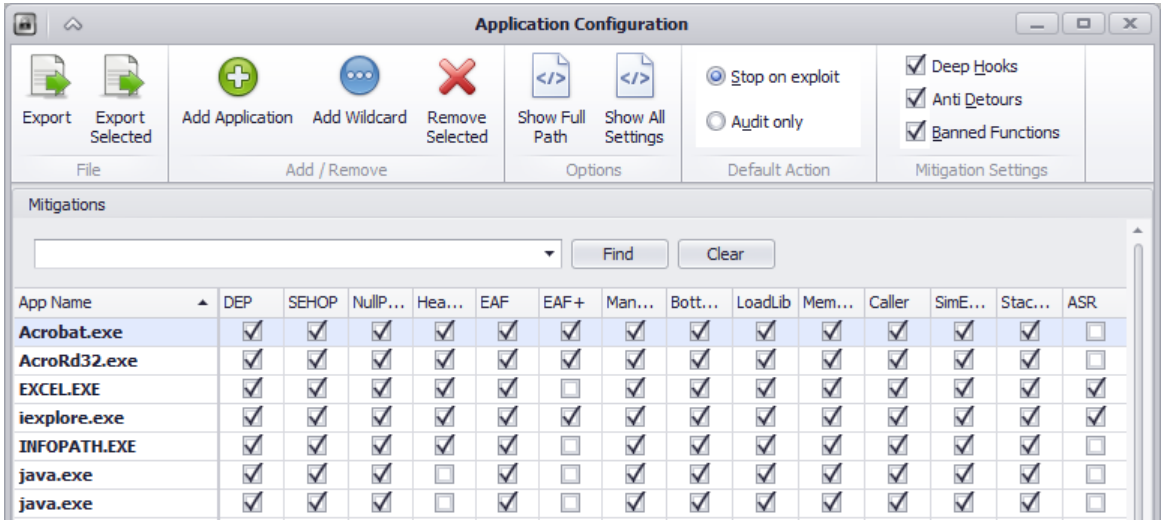
- Application to further harden a Windows system (XP+)
- (Additional) GUI for system mitigation techniques
- Can be used to enforce certificate rules
  - Binding of SSL certificates to legitimate Root CAs
- Enables exploit protection features for applications
  - Stops program on detected exploit (or lists details)

What is EMET?



- Recommended settings are a good starting-point
- Applications that process untrustworthy data should be hardened:
  - Adobe Reader, MS Office, Wordpad, Java, Browser, E-Mail Clients, Instant Messanger, ZIP-Packer
- (Small) compatibility List is available online:
  - http://support.microsoft.com/kb/2909257

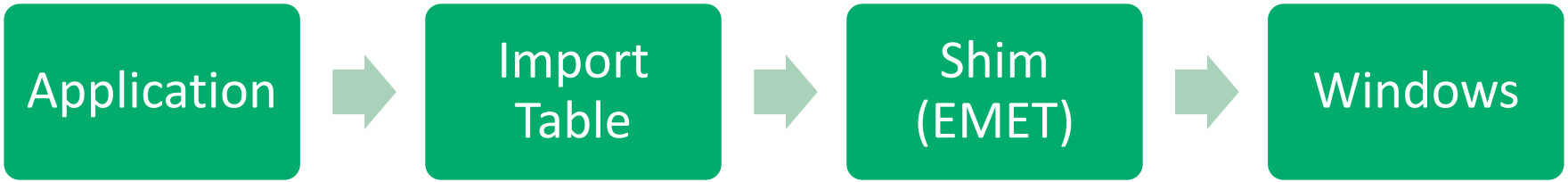
What is EMET?



- Applications can be added by:
  - Executable name/path
  - Wildcard (for different names)
- Protection settings can be configured per application

How it works

- Works via Application Compatibility Framework
- Intercepts function calls
- Monitors and modifies the process



EMET – Mitigations (1)

- Data Execution Prevention (DEP)
  - Marks Stack and Heap as non-executable to prevent shellcode execution
- Structured Exception Handling Overwrite Protection (SEHOP)
  - Prevents exploitation of the exception handler to execute exploit code
  - Validates Exception Handlers

EMET – Mitigations (2)

- Address Space Layout Randomization (ASLR)
  - Randomizes where modules will be loaded in memory to prevent prediction of mappings
- Bottom Up ASLR Security Mitigation
  - Bottom up allocations (e.g. VirtualAlloc(), VirtualAllocEx()) are now randomized
- NullPage Security Mitigation
  - Prevents code accessing to a NULL page

EMET – Mitigations (3)

- ASR (Attack Surface Reduction)
  - A way to stop DLL plugins from loading into processes (e.g. block Flash from loading into Excel)
- Heapspray Allocation Security Mitigation
  - Prevents allocation of reoccurring typical code byte patterns (by preoccupying them)

Export Address Filtering(+)

- Prevents reading of critical APIs in the EAT
  - Kernel32.dll
  - Ntdll.dll
  - Kernelbase.dll
- +
  - Detects mismatch of stack and frame pointer registers
  - Detection of read access to MZ/PE header

ROP Mitigations

- Load Library Check
  - Monitors LoadLibrary calls
  - Prevents loading dlls from UNC paths (i.e. \\evil\my.dll)
- Memory Protection
  - Checks if VirtualProtect marks stack memory as executable
- Caller checks (32-bit only)
  - Critical functions can only be called via CALL (not RET)
- Stack Pivot
  - Checks if the stack pointer is within the threads upper and lower specified stack limit

ROP Mitigations (2)

- Deep hooks
  - Protection of related functions of critical API calls
    - Kernel32!VirtualAlloc
      - Kernelbase!VirtualAlloc
      - Ntdll!NtAllocateVirtualMemory
- Anti detours
  - Prevents code from taking detours around hooked functions
- Banned functions
  - Allows to ban the use of API calls

Demo

EMET 5.1 Bypassing – What to do?

```
struct FRAME {
    size;
    CONFIG;
    writable;
}
```

- Props to offensive-security.com!
- Analysis of the EMET.dll shows:
  - Encoded Pointer to a structure "FRAME"
  - FRAME holds the enabled mitigations as a struct (CONFIG)
  - Memory area is write protected

EMET 5.1 Bypassing – What to do?

- Pointer to the structure is encoded
  - Decoding needed to resolve the address! (Can be found at EMET+0x67372)
- We can't write directly to the CONFIG/FRAME structure
  - Enable write access to the memory area first!

EMET 5.1 Bypassing – How to do it

- Find EMET.dll base address
- Call the decoding code at address EMET+0x67372
- Return into EMET+0x67372 and obtain the CONFIG address (EDX register)

EMET 5.1 Bypassing – How to do it

- Call ntdll!NtProtectVirtualMemory at CONFIG+0x1b8
  - Make CONFIG writeable
- Disable the EMET mitigations switch at CONFIG+0x558

ERNW
providing security.

Demo

Conclusions

- EMET should be used in corporate environments
  - Can be deployed fairly easy (e.g. via CMD)
  - Manageable with low effort (ruleset updates)
  - Adds findings to Windows Event Log
  - (Mostly) reliable 0-day protection
  - Protect outdated environments
  - Cheap way to “raise the bar”

Conclusions

- Drawbacks:
  - May raise compatibility issues with applications
  - Additional workload
  - Surely no “perfect” solution
- Social engineering attacks impose a bigger threat than 0-day exploits

There's never enough time...

THANK YOU...

...for yours!

btroester@ernw.de

ERNW
providing security.

Further links

- http://www.microsoft.com/emet
- https://www.offensive-security.com/vulndev/disarming-and-bypassing-emet-5-1/
- https://prezi.com/z0kjt1wi_9nl/ruxcon-2014-emet-50-armor-or-curtain/
- http://casual-scrutiny.blogspot.in/2015/03/defeating-emet-52-protections-2.html

ERNW
providing security.

Disclaimer

All products, company names, brand names, trademarks and logos are the property of their respective owners!