

Windows Internals: Advanced Exploit Mitigation

Fabian Nguyen

Abstract

We take a look at general techniques used by attackers to compromise Windows systems and some fundamental defense mechanisms against them. This paper will provide an overview of Microsoft's latest additions to the security concept of the Windows Operating System [OS], analyze inherent flaws in their design and take a brief look at already existing attacks.

1 Introduction

In an increasingly digitalized world an overwhelming amount of private and/or safety-critical information and data is stored on computers. Windows is by far the most used operating system and therefore the main target of attackers to compromise data or computer systems. One common intent of attackers is to steal an individual's passcode for a website, e.g. an online-banking website. Naturally, as the amount and complexity of attacks rises, OS vendors are forced to put an increasingly high amount of effort into mitigating existing weaknesses and deny attackers of further possibilities to compromise their OS. Even though this is the case, the amount of potentially abusable vulnerabilities in Windows has been increasing, instead of decreasing. 1

Year	# of Vulnerabilities	DoS	Code Execution	Overflow	Memory Corruption	XSS	Bypass something	Gain Information	Gain Privileges
2015	57	4	19	6	6		10	5	26
2016	172	6	47	23	7		19	31	82
2017	268	32	50	16	2	1	18	108	19
2018	257	21	45	19	1	1	39	72	1
2019	357	28	124	101	6	1	10	73	2
Total	1111	91	285	165	22	3	96	289	130
% Of All		8.2	25.7	14.9	2.0	0.3	8.6	26.0	11.7

Figure 1: Amount of documented vulnerabilities in the Windows Operating System.

Note that the spike in "Gain Information" vulnerabilities in 2017 is inflated by a family of attacks widely known as Meltdown/Spectre

We can see that the three largest groups of vulnerabilities consist of "Gain Information", "Code Execution" and "Overflow". Of course these three categories aren't entirely separated from each other. An attacker that is able to execute Code on a machine

often does so in order to gain information and overflows are often the reason why an attacker can execute code in the first place.

2 Overflows

An overflow happens when a program writes data to memory beyond the limits of the intended data structure.

3 Related Work

4 Approach

5 Conclusion

References

- [1] cvedetails.com. Windows 10 security vulnerabilities.