Hacking the Process - Business Process Compromise

Sherwyn Moodley, *Offensive Security Zyston LLC*
James Hasewinkle, *Offensive Security Zyston LLC*

Abstract

The information security field focuses on preventing unauthorized access, use, disclosure, disruption, modification, inspection, recording, or destruction of information. This is accomplished by securing assets and people, using frameworks and methodologies such as the CIS top 18 and NIST. Hacking a Business Process requires hacking these assets or people first, but the business process still needs to be secured. Financial Processes such as Purchase Order process, HR processes such as employee termination pose a great risk to any organization because if the process is manipulated sufficiently any theft rising from it becomes much harder to identify.

1. Process Hacking

Unlike traditional theft, a cybersecurity breach can go unnoticed because nothing is physically taken. If data is taken from a database, the company would only find out if the attacker wants them to know or if the attacker triggered an alarm and created an incident.

Business Process Hacking is an even greater example of this disparity between IRL theft and cybercrime. If a process is hacked, and the attacker is good enough to hide all evidence of the hack, the process will be seen as legitimate by the company. It is possible that the victim will be believe a malicious bank transaction or purchase order or even salary cheque is a valid transaction.

Information Security practices and regulation show us how to secure an asset. We use configuration controls to secure or harden systems, periodic vulnerability assessments to check if there are known vulnerabilities on these systems, and penetration tests to exploit any vulnerabilities. These tests give critical information to assess the cybersecurity risk of a company, but even if a Penetration Test is taken to its fullest extent it usually goes so far as to put a malicious file on a server or gain access to a user's account, never to change a purchase order or create a fictional employee.

1.1. Previous Cases

In 2016 a group of hackers moved $81 million dollars from the Bangladesh Central Bank to four accounts. Hackers obtained valid credentials that banks use to conduct money transfers. The hack centered on dispatching fraudulent SWIFT messages and sabotaging the business process where the bank releasing the funds contacts the originating bank.

In 2013 hackers gained access to a container tracking system in Belgium and used the processes to smuggle drugs passed the port authorities.

1.2. Theoretical Cases

As a theoretical exercise, we can assume the role of a Blackhat malicious attacker who wants to profit from manipulating an employee onboarding process and payroll processes at an HR as a Service provider.

For this example, the entry point to the solution is not important, but it cannot trigger any alarms as manipulating a business process sufficiently requires that the outcomes of the process are not questioned. The attack vector with the highest level of success here would probably be Credential Harvesting.

Once we have access to an employee of the HR as a service company’s account, we can create employees at companies. The challenge here would be to create employees in small roles, or contract roles that would be a small enough income not to be noticed. It would also require business process enumeration, finding what supporting documents are needed, other processes that need to be triggered first, which forms need to be completed, and where these forms and files are stored.

If our attacker manages to manipulate the process to the extent it would pass a financial audit, how long could these fake employees exist on the payroll? How many fake employees across how many companies could be created?

What are the chances this hasn’t already happened?

2. Conclusion

The methodology of securing a business process needs to be as robust as the methodology of securing technology assets. There should be control compliance standards, such as document numbers that are dependent on upstream processes and validity checks that are monitored to create incidents in the event of an alarm.

Penetration tests of the process should also be recurring, to test the monitoring and incident response and to test the dependency checks and validity monitoring.