Journal of Vincentian Social Action

Volume 6 | Issue 2 Article 11

December 2022

Potential Risks Inherent in Robotic Process Automation

Colin L. Robinson
colinrobinson142@gmail.com

David Y. Chan
St. John's University, chand@stjohns.edu

POTENTIAL RISKS INHERENT IN ROBOTIC PROCESS AUTOMATION

Colin L. Robinson

David Y. Chan

ABSTRACT

Robotic process automation (RPA) uses automation technologies to perform tasks typically performed by humans. Although such technology has been instrumental in expediting business operations and lowering costs, it has also created several risks that warrant scrutiny. When discussing the drawbacks of automation, many will point to the number of jobs lost to the influx of automation. However, there are technology risks that organizations must consider such as fraud and cybersecurity. Fraudsters may utilize RPA to commit more novel and subtle technological and cyber security fraud. Organizations may implement internal control measures to prevent or mitigate such schemes, segregation of duties, and change management. RPA has many benefits, but the effective use of such technology will ultimately come down to how businesses adapt to risks in such an ever-changing business environment.

POTENTIAL RISKS INHERENT IN ROBOTIC PROCESS AUTOMATION

According to Deloitte, 58% of large organizations have embarked on the use of robotic process automation (RPA) and for the others it's a top strategic priority for transforming their business (AICPA, 2020, August 31). Robotic process automation is the use of automation technologies to mimic back-office tasks of human workers (IBM Cloud Education, 2020, October 22). Robotic Process Automation (RPA) can help businesses improve the efficiency and effectiveness of their operations, resulting in reduced labor costs and other related expenses. KPMG estimates that 47% of jobs will be replaced by automation over the next 10 to 20 years (KPMG, 2018).

"KPMG estimates that 47% of jobs will be replaced by automation over the next 10 to 20 years."

Robotic process automation applications can perform tasks such as merging data from multiple sources, making calculations, copying and pasting data, filling in forms, and extracting structured data (Lowes et al., 2017). Such functions are utilized from an accounting perspective for automating or semi-automating invoice processing, expense processing management, and other manual processes. For example, an RPA application can match purchase orders and receive information regarding incoming invoices. The application can then automatically send the matched electronic documents for approval or further investigation. For expense management, an RPA application can extract information from an employee submitted receipt, and the reimbursement request can then be automatically approved or sent for manual approval.

RPA can improve an organization's operations, but risks can arise with the development of such technology. Management and their auditors are traditionally concerned about the output from manual business tasks because humans can make unintentional or intentional mistakes when performing such tasks. In contrast,

RPA applications perform business tasks designed or coded by the programmer. An unintended output indicates either a design flaw in the application or an unintentional programming error. Alternatively, a programmer can intentionally program an application to perform a nefarious action or create a cybersecurity incident. The risk of RPA applications making an unintentional error or performing a rogue action can be mitigated with proper internal controls over the software development life cycle and the change management process.

Salami slicing is an example of a nefarious action where a programmer intentionally modifies the program of an application to take a minute fraction of all financial transactions (Romney & Steinbart, 2018). For example, a programmer may include unauthorized programming code in an RPA application to take a minute slice (.00001 of a penny) from every financial transaction in addition to performing its intended task. Taking .00001 of a penny off every transaction may not seem material, but over time, this can add up to millions of dollars if undetected. For example, $0.00001 of a transaction for one billion transactions is ten thousand dollars (Table 1). Big banks can have billions of financial transactions annually, and $0.00001 for each transaction can add up if the fraud goes undetected for long periods.

Table 1

Example of Skimming

Transactions	Times	Portion of a penny per transaction	Total stolen per transaction
100,000	X	0.00001	$ 1
500,000	X	0.00001	$ 5
1,000,000	X	0.00001	$ 10
250,000,000	X	0.00001	$ 2,500
500,000,000	X	0.00001	$ 5,000
750,000,000	X	0.00001	$ 7,500
1,000,000,000	X	0.00001	$ 10,000

In addition to an RPA code modification designed to perform unintended tasks, management should consider RPA's ramifications on cybersecurity. Today, most applications rely on access to the internet or a company network with internet access. Management should consider the risk that RPA applications may allow unauthorized network access to outsiders due to unauthorized code. An RPA application may facilitate data theft by transferring sensitive data or information to outsiders. A backdoor trojan is a perfect example of such a subtle cybersecurity scheme. By circumventing security measures, cybercriminals can use a backdoor to steal personal or essential information by bypassing security measures, leading to ransomware, spyware, malware, and data theft. A programmer can, for example, include an unauthorized code (backdoor trojan) in an RPA application that will allow an outsider to gain access and control an organization's computer or network (Inspired eLearning, 2018, May 22).

According to the Malwarebytes Labs 2020 State of Malware report, backdoors trojans were the fourth most common threat detection in 2018 for both consumers and businesses (Malwarebytes Labs, 2020). Backdoors can impact government

agencies as well. For example, a backdoor in the network of the United States Commission on International Religious Freedom was found by researchers (Bagwe, 2021, December 17). The researchers discovered that the attackers utilized the backdoor to gain visibility into the government agency's network. Ultimately, it gave attackers complete control of their systems, which enabled them to intercept all local network traffic within the organization. A backdoor could be the first step in an attack designed to penetrate the organization's network and subsequently a third-party partner's network. Therefore, such a cybersecurity incident can lead to significant financial losses and reputational damage without adequate controls.

Proper internal controls can help mitigate risks related to unauthorized code. Internal controls regarding the segregation of duties on software development and change management are relevant. The Software Development Life Cycle (SDLC) identifies six stages of software or application development: 1) requirement gathering and analysis, 2) designing the software, 3) implementation and coding the software, 4) testing the software, 5) deploying the software, and 6) maintaining and managing the software (Wegrzynowicz & Stein, 2009). In stages three and six, the risk of unauthorized code in an RPA application is primarily a concern. A programmer may include unauthorized code during the development of an RPA application or when an existing RPA application is maintained or updated. The objective of a Software Development Life Cycle is to have proper segregation of duties, where the person(s) reviewing the code is independent of those who developed the RPA application.

Change management is a systematic set of processes that are executed to manage enhancements, updates, installations, implementations, incremental fixes, and patches to production systems (Taylor, 2005). A poor change management program may expose the organization to risks associated with an unauthorized code. One tool designed to augment change management is a code repository. A code repository enables organizations to manage software updates. The code is stored in a securely located repository that requires programmers to check out the code they are assigned to change. The change management system will be notified once the changes are complete, and the code is then checked back in for review. Such a method ensures documentation and version control. Like having proper segregation of duties in the Software Development Life Cycle, management should ideally have an independent programmer review and verify the intentions of any modified code before its implementation in production.

Internal controls, such as segregation of duties, are essential for managing the risk of unauthorized code. However, such risk reduction measures may be further enhanced through a layered approach known as the three lines of defense model developed by The Institute of Internal Auditors (The Institute of Internal Auditors, 2020). The first line of defense holds management responsible for managing risks associated with internal controls. The second line of defense ensures that an individual in the organization monitors the risks and effectiveness of the internal controls implemented. For example, implementing the segregation of duties internal control is insufficient. Management should have an individual review the application and effectiveness of the segregation of duties. Finally, the third line of defense ensures that an organization establishes an internal audit function to monitor the first two lines to determine their effectiveness and provide recommendations for improvement. Taken together, the three lines of defense model provide a comprehensive approach to continuously improve the management of risks associated with the rise of robotic process automation and other application technologies.

CONCLUSION

Technology such as robotic process automation assists organizations in becoming more effective and efficient in their operations. Ultimately, risks will invariably arise as organizations adopt such new technologies. However, through proper use of the segregation of duties and the three lines of defense model, effective governance can be achieved over the change management system and the Software Development Life Cycle to minimize financial and reputational risks associated with an unauthorized code modification.

REFERENCES

AICPA. (2020, August 31). Why CPAs should care about RPA. https://blog.aicpa.org/2020/08/why-cpas-should-care-about-rpa.html#sthash.JZVhViAA.dpbs

Bagwe, M. (2021, December 17). *Backdoor discovered in US federal agency network*. Gov Info Security. https://www.govinfosecurity.com/backdoor-discovered-in-us-federal-agency-network-a-18147

IBM Cloud Education. (2020, October 22). *Robotic process automation*. IBM. https://www.ibm.com/cloud/learn/rpa#toc-rpa-use-ca-k1vYYvH9

Inspired eLearning. (2018, May 22). *Trojan horse examples and how to defend yourself*. https://inspiredelearning.com/blog/trojan-horse-examples-defend/

KPMG. (2018). *Robotic process automation (RPA) - On entering an age of automation of white-collar work through advances in AI and robotics*. https://assets.kpmg/content/dam/kpmg/jp/pdf/jp-en-rpa-business-improvement.pdf

Lowes, P., Cannata, F. R. S., Chitre, S., & Barkham, J. (2017). *Automate this - The business leader's guide to robotic and intelligent automation, service delivery transformation*. Deloitte. https://www2.deloitte.com/content/dam/Deloitte/us/Documents/process-and-operations/us-sdt-process-automation.pdf

Malwarebytes Labs. (2020). *2020 state of malware report*. Malwarebytes. https://www.malwarebytes.com/resources/files/2020/02/2020_state-of-malware-report.pdf

Taylor, J. R. (2005). Global technology audit guide (GTAG) 2: Change and patch management controls: critical for organizational success (2nd ed.). The Institute of Internal Auditors. The Institute of Internal Auditors. (2020). *The IIA'S three lines model - An update of the three lines of defense*. https://www.theiia.org/globalassets/site/about-us/advocacy/three-lines-model-updated.pdf

Romney, M. B., & Steinbart, P. J. (2018). *Accounting information systems* (14th ed.). Prentice Hall.

Wegrzynowicz, K., & Stein, S. (2009). *Global Technology Audit Guide (GTAG) 12: Auditing IT Projects*. The Institute of Internal Auditors. https://www.theiia.org/en/products/bookstore/global-technology-audit-guide-gtag-12-auditing-it-projects/

ABOUT THE AUTHORS

Colin L. Robinson

crobinson@jd25.law.harvard.edu

Mr. Robinson received his Bachelor of Science in Accounting with a minor in Business Law in May 2022 from the Peter J. Tobin College of Business at St. John's University. Last summer, he was an SEO (Sponsors for Educational Opportunity) Law Fellow at Paul, Weiss, Rifkind, Wharton & Garrison LLP in their New York City office for the summer of 2022. Mr. Robinson is a first-year law student at Harvard Law School as a J.D. Candidate for the class of 2025. He is interested in the intersection of business, law, technology, and public policy and how regulatory standards meet the needs of businesses and the larger community.

David Y. Chan

chand@stjohns.edu

Dr. Chan earned a Ph.D. in Management with a concentration in Accounting Information Systems from Rutgers, The State University of New Jersey. He also holds a Master of Science in Accounting and Bachelor of Science in Finance from St. John's University. Dr. Chan is a licensed Certified Public Accountant (CPA) in the state of New York, Certified Internal Auditor (CIA), Certified Information Systems Auditor (CISA), and Certified Fraud Examiner (CFE). Dr. Chan's teaching interest includes financial auditing, internal auditing, fraud examination, and information technology auditing. Dr. Chan's research interests include auditing, auditing technology, and the application of technology in accounting and auditing.