EXPERT OPINION

SECURITY DESIGN CONSIDERATIONS IN ROBOTIC AUTOMATIONS

Sridevi Kakolu*

Technical Architect in Information Technology Department, Boardwalk Pipelines, Houston, TX, USA

ISSN 0976-3104

ABSTRACT

Robotic Process Automation (RPA) is a new favorite among IT leaders. It can be quickly deployed to automate repetitive tasks, and it saves organizations time and money. RPA bots handle sensitive data, moving it across systems from one process to another. If the data is not secured, it can be exposed and can cost organizations millions of dollars. Security design considerations are vital in RPA implementation for its success. Based on my recent Robotic Process Automation project implementation and research, able to put together the design strategies and best practices to implement security in RPA. This paper focuses on RPA introduction, types of RPA, major security risks in RPA process and risk mitigate security design considerations in RPA process implementations for Organizations.

KEY WORDS
Robotic Process Automation, RPA, Unattended Bots, Security in RPA, Best practices for RPA

Received: 03 May 2023
Accepted: 04 July 2023
Published: 12 Aug 2023

INTRODUCTION

Robotic Process Automation is the event-driven software used to automate tasks and processes otherwise performed by humans. In robotic automation, a software robot mimics the work of human users to perform various tasks that are repetitive, high volume, and rules driven. Through automated processes, the software robot executes a workflow involving multiple steps and interactions with different enterprise applications. Robotic process automation remains a popular software market for improving operational efficiency with tactical automation.

TYPES OF RPA

Robotic Process Automations can be executed in two modes:

- Attended
- Unattended

An attended automation assists an agent/human in handling simple, repetitive tasks. In contrast, an unattended automation automates specific tasks which do not require agent/human intervention [Table 1]. Depending on the project or use case, the robotic automations can interact with enterprise applications, databases, or financial systems. RPA can help process the required tasks with or without presence of a human.

As per 2022 Gartner Magic Quadrant Evaluation Report, strategic planning assumptions: "By 2024, 95% of RPA vendors will offer automation via both API and UI integration. By 2024, 80% of enterprise customers who have deployed attended automation primarily on a desktop will pivot to wider UX covering web, mobile and voice interfaces." [1]

Table 1: Types of Robotic Automations.

Attended Automation	Unattended Automation
Executes workflow on an end user's desktop to assist in human work	Runs on a virtual or server and does not require human intervention
Works alongside agents to improve productivity and quality	Runs on a dedicated workstation to execute fully automated processes
Automates 20-90% of a given task	Automates 100% of given tasks

BENEFITS OF RPA

Over the past few years, robotic process automation (RPA) has become a popular technology. This is due to its ability to automate repetitive and high-volume tasks in order to reduce manual effort, eliminate error

***Corresponding Author**
E-mail:
Sridevi.Kakolu@gmail.com

and improve process productivity. With RPA, software bots can mimic human actions such as logging into various applications/systems. They can also navigate through user interfaces to perform tasks such as creating tickets and downloading data. Bots can provision and deprovision user access and respond to customer queries. RPA is versatile and flexible, allowing it to integrate easily with existing processes. It helps reduce cost, maintain consistent quality, improve delivery timelines, and enhance the customer experience [Fig. 1].

Fig.1: Benefits of RPA.

SECURITY RISKS WITH RPA

Organizations looking to implement RPA should be aware of the security-related risks [Fig. 2]. These include:

Compromise privileged access accounts

In terms of RPA security, the risks of privileged access abuse by RPA bots are similar to those by humans. For example, privileged access given to an RPA bot account may be used by attackers. They may break into the system and steal or misuse sensitive business information. For better auditing and troubleshooting, it is essential to distinguish bot activities from those of employees. Never use an employee's credentials for RPA implementation. Create unique identities for every bot in the system, and do not store passwords in the source code. Keep passwords in a centralized, encrypted location such as a password vault and change them frequently. Limit the number of employees who have access to RPA credentials. Configure a robust authentication method like two-factor authentication or token authentication for extra security.

Malfunctioning and system outage risks

System outage (or downtime) refers to the period when a system/network cannot perform its primary function. Downtimes can happen because of numerous reasons. The most frequent reasons for this issue are human error, outdated or unstable hardware/software, bugs in the server operating system, and integration/interoperability issues. In RPA, there are two potential risk scenarios related to system outage. First, unexpected network failure may disrupt the bot's operation leading to a significant loss in productivity. Second, a rapid sequence of bot activities may cause system failure or outage. For instance, in 2018 on Amazon Prime day, millions of shoppers faced a high-profile outage on the Amazon "Deals" page because its servers did not manage such a massive online traffic spike.[2]

Data breach

Confidential information is any information related to a company's business and affairs that is not available to the public and has commercial value. Unauthorized disclosure of financial information, marketing plans, upcoming projects, and other confidential materials may have devastating consequences. In RPA, disclosure of confidential information may occur with the intentional or negligent improper training of an RPA bot. This causes leakage of confidential data, such as payment or credit card data, to the web.

System vulnerabilities

Vulnerabilities are weaknesses in an information system that allows cyber attackers to illegally gain access to the system. One of the ways vulnerabilities may appear is when a malicious user behaves imprudently by visiting an unsafe website. In this case, the website is a threat resource that triggers vulnerability. Some of the most common examples of vulnerabilities are: missing data encryption, SQL injection, missing authorization, cross-site scripting and forgery, weak passwords, upload of infected software. Even though advanced RPA systems nowadays use encryption while transferring data, there are still low-security-level RPA tools. Here, non-encrypted data transfer may cause sensitive data leakage.

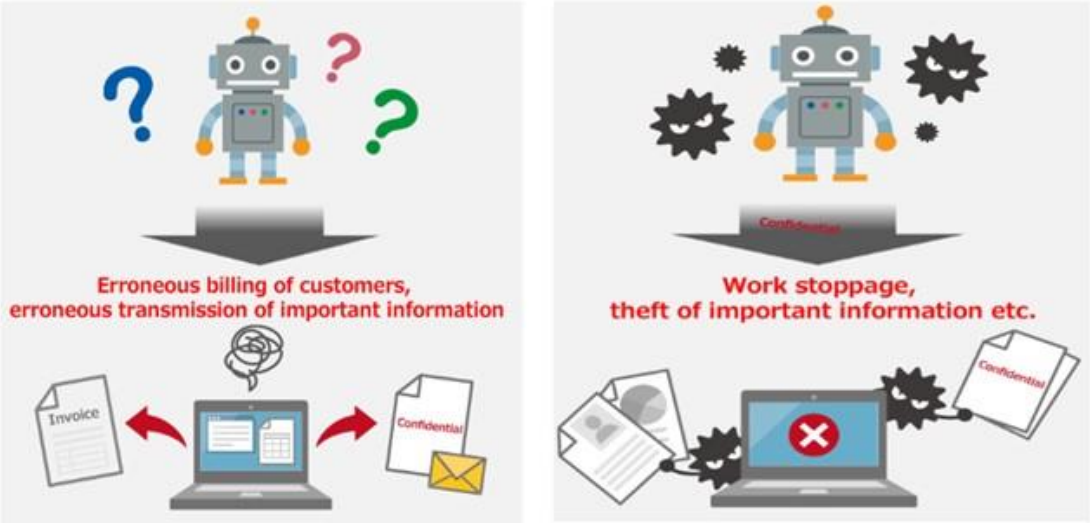

Fig. 2: Security risks in RPA.

Lack of Visibility on the Bots executions

Audit logs capture bot activity, and these logs are important to track bot health and effectiveness. For instance, if a bot stops working, the audit log helps identify the underlying reason. The reason may be improper use by an employee or malicious code. Bots need to be periodically monitored at various levels to ensure they do not misbehave. Misbehavior can lead to high error rates and potential damage. In some cases, bots may not perform as intended due to erroneous coding or inadequate testing. This will result in issues and errors during go-live. Besides RPA software out of the box logging, automation logic design needs to take care of detailed logging while executing set of actions. Audit logs can be monitored in dashboards also configure notifications, then there will not be much visibility on any of these issues.

SECURITY DESIGN CONSIDERATIONS IN RPA

Robotic automations execute the logic based on its design and implementation. This means it has the potential to touch every enterprise application within the organization and the confidential personal and customer data within it. Whether it is an attended automation or unattended automation, it is important to consider security standards within the Robot automation design. The cost of a security incident can be tremendous. Enterprise-ready RPA must assure both business and IT that the RPA deployment will not compromise security or compliance [Fig. 3].

Below five best practices are helpful for reliable and secured RPA design:

- Accountability for Bot Actions
- Automating Credential Management
- Strong Governance Framework
- Continuous Review and Change Control
- Logging, Auditing and Monitoring

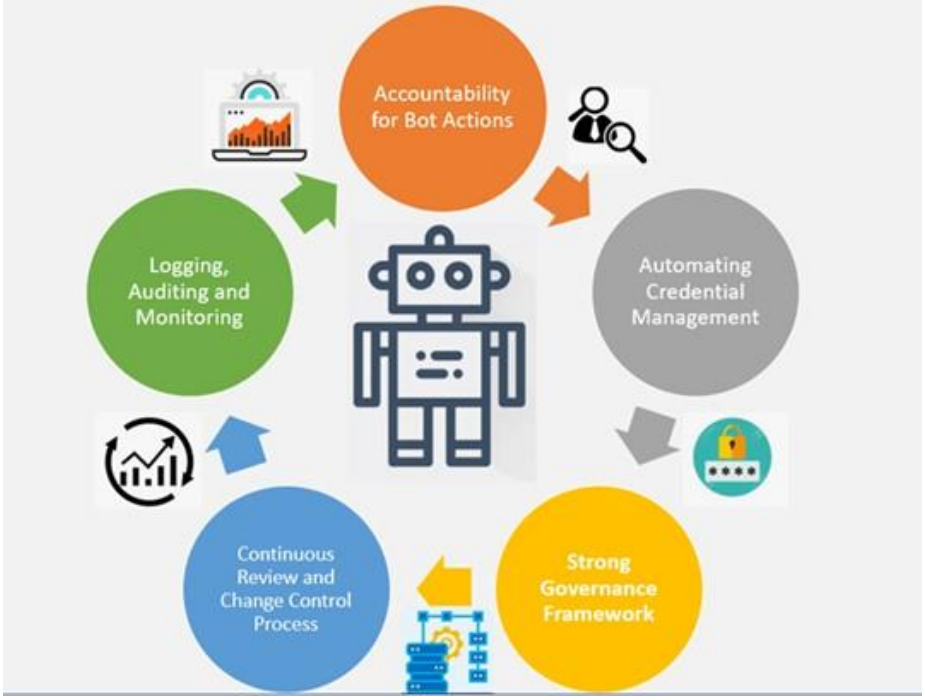

Fig. 3: Security design considerations in RPA.

Accountability for Bot Actions

For better auditing and troubleshooting purposes, it is essential to distinguish the activities of a bot from those of an employee. Never use an employee's credentials for RPA implementation. Create unique identities for every bot in the system, and do not store passwords in the source code. Keep passwords in a centralized, encrypted location such as a password vault and change them frequently. Limit the number of employees who have access to RPA credentials. Configure a robust authentication method like two-factor authentication or token authentication for extra security.

Automating Credential Management

Successful RPA deployments require automated credential management, including machine-generated passwords, automatic password rotation, identity verifications and just-in-time or time-limited credential access. RPA teams can save passwords in single password storage or vault without creating any security leaks. Never use an employee's credentials for RPA implementation. IT administrators can configure minimum access rights for a bot to access applications and databases.

Strong Governance Framework

It is very important to define rules and regulations to maintain security in RPA solutions. Without proper governance, RPA cannot ensure the security it is supposed to offer. Detailed criteria, development criteria, and business justification are some features that fall under an excellent governance framework.

- **Roles & Responsibility Management** - Build and implement a system with clear roles and responsibilities. Roles should include everyone in the department/team responsible for the automation process.
- **Strategy and Regulations** - The company should clearly elaborate the rules and requirements set out in their current security regulations. They should also provide adequate supervision to ensure compliance.

- **Awareness** - Top managers should raise awareness of RPA-related risks and the potential impacts. Awareness should be spread internally (within the responsible teams) and externally (among the RPA bots' creators). Regularly validate RPA scripts and audit logs to ensure a bot is working correctly.

Vendor and internal teams should work together to establish a robust governance framework. The framework must clearly define the automation scope, prioritize identified RPA candidates, and evaluate regulatory and business risks for each RPA candidate. The framework needs to define each team member's roles and responsibilities clearly. It is also advisable to update the company's Information Security Management System (ISMS) and Identity and Access Management (IAM) policies to incorporate RPA specific requirements.

Continuous review and change control

Create a transparent business continuity plan that specifies the backup procedures and data sources required to carry out every task. It is the responsibility of an internal audit team to check and review the documents in the business continuity plan to see if there is any information, like how to restart each process/activity even after failure. Build weekly or monthly review plans on overall RPA infrastructure in the company and review Bot performance. Implement framework-based design approaches to maintain the Bot's logic easily by the developers. Implement CI/CD pipeline process to deliver RPA software upgrades or patch updates and deliver fixes smoothly.

Logging, auditing and monitoring

Enforce proper regulations to monitor the performance of RPA bots and ensure that all bots function in accordance with the set rules. Periodic risk assessment is necessary to track the possibilities of new risks, mitigate, and review security risks in the RPA, to check if any restrictions have been lifted, and to determine if any RPA bot needs to be avoided. It is critical to monitor and log every transaction of an RPA script. Efficient security and risk management practices ensure consistent and accurate logging. It is a good practice to secure RPA logs in a separate system and encrypt sensitive data. Rapidly detect and respond to unauthorized or anomalous robot behavior by assigning human managers, enforcing least privilege, and making actions traceable.

RPA SECURITY DESIGN CHECKLIST

Above 5 best practices action plan helps Security and Risk Management leaders to mitigate RPA risks. Below are a few more comprehensive security checklists that can be useful when starting to design and implement RPA. [3]

- Enhance software development practices to include secure bot development/deployments.
- Implement bots based on input feed from a secured location.
- Treat a robot like a user and create a separate set of credentials.
- Implement integrations with a secret server using access token mechanism to get credentials.
- Maintain a password vault to store bot credentials and rotate bot credentials often.
- Establish mechanisms to find, avoid, and control bot abuse such as a provision to lock down bots.
- Do not leave any credentials in the source code.
- Use two-factor authentication for an extra layer of security.
- Follow the principle of least privilege and grant only the necessary permissions to the bots.
- Ensure that all transactions are correctly logged.
- Implement email notifications in the bot logic for failures and successful processing.
- Review RPA scripts and logs regularly.

CONCLUSION

Organizations adopting RPA to improve productivity should plan their implementations carefully to protect themselves from security breaches. RPA creates new application layers that are vulnerable to risk. Moreover, without constant supervision, bots may not work effectively, causing issues, errors, and potential damage. Since bots may need access to confidential information, it is imperative for organizations to institute the right security measures. Some of these measures include creating governance frameworks, audit logs, password vaults, and version controls. Establishing these processes will allow RPA to manage security risks by itself. This ensures best bot performance and reduced business risk.

CONFLICT OF INTEREST
Author declare no conflict of interest.

ACKNOWLEDGEMENTS
None.

FINANCIAL DISCLOSURE
None.

REFERENCES

[1] https://www.gartner.com/smarterwithgartner/4-steps-to-ensure-robotic-process-automation-security (Accessed on 5 April, 2023)
[2] https://www.uipath.com/resources/automation-analyst-reports/10-golden-rules-of-rpa-success-forrester-report (Accessed on 4 April, 2023)
[3] https://academy.pega.com/topic/pega-robotic-automation-introduction/v2/in/2976/5406 (Accessed on 3 April, 2023)
[4] https://www.xenonstack.com/blog/rpa-security-checklist (Accessed on 3 April, 2023)
[5] https://www.infosys.com/services/cyber-security/documents/rpa-security.pdf (Accessed on 4 April, 2023)
[6] https://electroneek.com/blog/security-concerns-in-rpa-4-step-guide-to-address-them/ (Accessed on 5 April, 2023)