

Risk assessment and impact analysis for updates

Risk is the future uncertainty between potential events and the potential for loss. Classic software documentation states that every deployment process should undergo a thorough inspection. This may increase test costs but reduce risks of failure. Some guidelines to mitigate risks of software updates include:

- Use legitimate sources of updates. Only download software updates from trusted sources to avoid introducing malicious software
- Implement continuous integration and delivery. Continuous integration and delivery (CI/CD) is a set of practices and tools that automate the building, testing, and deploying of a system. CI/CD enables you to deliver frequent and reliable updates to the system, as well as to detect and fix errors quickly. CI/CD also reduces the risk of conflicts and inconsistencies among the system components, as well as the need for manual interventions. Some common CI/CD tools are Jenkins, Travis CI, and CircleCI.
- Test thoroughly. Testing is a crucial step in ensuring the quality and functionality of a system. Testing should be done at different levels, such as unit testing, integration testing, system testing, and user acceptance testing. The system should also be tested in different environments, such as development, staging, and production. Testing will help you verify the system requirements, identify and resolve defects, and evaluate the system performance and usability. You should also use automated testing tools and frameworks, such as Selenium, JUnit, and TestNG, to speed up and simplify the testing process.
- Monitor and evaluate. Once you deploy the system or make changes to it, you should monitor and evaluate its performance and behavior. You should use tools and metrics that can track and measure the system availability, reliability, scalability, security, and user satisfaction. You should also collect and analyze feedback from the stakeholders, users, and customers, as well as from the logs and reports generated by the system. Monitoring and evaluating will help you identify and troubleshoot any issues or anomalies, as well as to improve and optimize the system over time.
- Consider taking a backup of the software before deployment
- Changes should be deployed outside of office hours and users should be informed of the changes

Revision #1

Created 8 October 2025 10:07:00 by RISA

Updated 8 October 2025 10:08:02 by RISA