**DevSecOps Policy Creation**

**Provide an existing company which has a strategic goal - i.e., Increase Revenue, Optimize Value to Clients etc - List the pain points or the drivers to this requirement**

One example of a company with a strategic goal is Amazon. The company's strategic goal is to "become the earth's most customer-centric company". Here are some pain points or drivers to this requirement:

1. Increased competition: With more and more e-commerce players entering the market, Amazon needs to constantly optimize its value to clients in order to stay ahead of the competition.

2. Changing consumer expectations: Consumers are becoming increasingly demanding when it comes to convenience, speed, and personalized experiences. Amazon needs to constantly adapt to changing consumer expectations in order to stay customer-centric.

3. Rapidly growing product offerings: Amazon's product offerings are constantly growing and evolving, which can make it challenging to optimize value to clients and provide a seamless customer experience across all product categories.

4. Delivery logistics: With millions of products being sold every day, Amazon needs to ensure that its delivery logistics are flawless and efficient in order to optimize value to clients.

5. Profit margins: While Amazon is known for its high revenue, profit margins can be a challenge. In order to increase revenue and optimize value to clients, Amazon needs to strike a balance between offering competitive prices and maintaining healthy profit margins.

Planning - understand what are the current process and tooling that they have i.e. Planning, ALM, Integrated Development Environment, Deployment Tools, Monitoring Tools Find the gaps of the Automation Tools - Plan, Build, Test and Release Expectations

Sure, here are the steps to understand the current processes and tooling, identify gaps in automation tools, and determine expectations for automation:

Understanding the current processes and tooling:

1. Schedule meetings with stakeholders involved in planning, ALM, development, deployment, and monitoring to gain an understanding of their current processes and tooling.
2. Create a detailed inventory of the tools currently being used, including planning tools, ALM tools, integrated development environments, deployment tools, and monitoring tools.
3. Analyze the current processes and tooling to identify pain points, bottlenecks, and areas for improvement.
4. Document the current workflows and the tools used at each stage of the process.

Identifying gaps in automation tools:

Identifying gaps in automation tools:

1. Evaluate the current processes to identify areas that can be automated, such as testing, deployment, and monitoring.

2. Assess the current tooling to identify gaps in automation tools that could help streamline the process.

3. Prioritize the automation tools that will have the greatest impact on productivity, speed, and quality of output.

Determining expectations for automation:

1. Schedule meetings with stakeholders to determine their expectations for automation.

2. Define the goals and objectives of the automation project, such as improving process efficiency, reducing errors, and improving collaboration.

3. Determine the scope of the automation project, including which processes will be automated and which tools will be used.

4. Develop a timeline and budget for the automation project.

5. Identify the metrics that will be used to measure the success of the automation project, such as reduced cycle time, increased productivity, and improved quality.

By following these steps, you can gain a better understanding of the current processes and tooling,

identify gaps in automation tools, and define expectations for automation.

1.is to be able to fill the gaps in tooling to provide a full automation and improve speed, acceleration and release of software 2.how much improvement has been imposed with the new proposed system against the current

1. To fill the gaps in tooling and provide full automation, it is important to first identify the specific tools that are needed to automate each stage of the software development lifecycle. This may include tools for planning, ALM, integrated development environments, testing automation, deployment automation, and monitoring automation. Once the necessary tools have been identified, it is important to integrate them into a cohesive system that provides end-to-end automation. By doing so, software development tea

2. ms can accelerate the delivery of software, reduce errors, and improve the quality of their output.

3. To measure the improvement that has been imposed with the new proposed system against the current, it is important to define metrics that can be used to quantify the impact of the automation. These metrics may include cycle time, time to market, productivity, defect rate, and customer satisfaction. Once these metrics have been defined, it is important to measure them both before and after the introduction of the new proposed system. By doing so, it is possible to quantify the improvement that has been achieved and identify areas where further improvements can be made.