From what I’ve read online, there are a five basic steps required in making a cyber security risk assessment.

The first step is to determine the scope of the risk assessment. Are you looking specifically at data storage & access? Or maybe application access and authorization roles? Maybe service to service communication and network protocols. Maybe you want to assess a vertical slice of all those aspects of the system, but only as it pertains to a single service, like a payment processing service in a larger e-commerce ecosystem. It’s good to understand the security posture of the system as a whole, but in a lot of cases that might be too large of an undertaking to commit to at once.

The next step is to identify cybersecurity risks within your defined scope from step one. You can start by identifying assets, then identifying potential risks to those assets. You can also look at those assets, and identify potential areas of failure within them that could lead to additional risks (a glaring example would be a SQL injection vulnerability in one of your services)

The third step is to analyze those risks and determine the potential impact. This can be helpful in multiple ways, it could help with prioritizing resolving those risks with other work, and it could also help explain to stakeholders what the risk really entails and it’s severity. The article from TechTarget suggests ranking security vulnerabilities on two different axis, 1-5 likelihood to be exploited, and 1-5 severity. This allows for a simple risk matrix to prioritize risks, which leads directly into step four.

Step four suggest prioritizing risks. Many of the benefits to this were already discussed in step three, but the article offers a few ways to treat risks that are above the organization’s severity tolerance. You can avoid the functionality of that part of the system if the risk outweighs the benefits. Especially if it’s old and outdated software, and there is already a replacement for it. Or you could transfer the risk to third parties. So, if your system is susceptible to something like a DDoS attack, maybe your cloud infrastructure has built in ability to mitigate that risk (Azure has some tools for this). The third option is to explicitly mitigate the risk. This might entail making code changes, process changes, etc. to ensure the risk isn’t able to be exploited.

The final step I’ve noticed tends to be ignored the most. Document the risk and how you went about resolving it! Documentation is the most under-appreciated aspect of software development and the IT field as a whole. It always seems to me like the companies I’ve worked for either don’t prioritize it, or they don’t have a great way to maintain documentation. This is a very important piece of the technological development domain and should be upheld as such. For this documentation, TechTarget suggests including the risk scenario, identification date, existing security controls, current risk level, treatment plan, progress status, residual risk (what is the risk level after treatment), and the risk owner.

A document like this would be incredibly helpful to stakeholders, especially if it can be easily kept up to date with version control. That way stakeholders can easily understand the risk found in the context of the standardized priority scheme, and can see what the status is of the risk and how it’s being treated.

Technology around risks and vulnerabilities has been here for a while, and continues to emerge every year. There is a lot of tech out there that will automatically detect vulnerabilities in code. These tools include analyzers like Trivy, Lynis, and Tiger, and they’re all incredibly useful and should be implemented in all organizations in my opinion. As useful as these tools are, they can lead to a false sense of security for your organization. It's important to understand that one solution can't catch all potential vulnerabilities, and you should be implementing multiple security methods and measures around your technological ecosystem.

As for how I would conduct a risk assessment, I think after reading this article it had a lot of impact on what steps I would take to assess a system. I would prefer to work in vertical slices of a system unless instructed to do a horizontal slice of data security, or network security, and so on. And I'd like to present my findings in the way outlined by this article. I think it would be best presented in a way that's easy to digest by the stakeholder. For my individual project, I think that's how I'll plan on providing my own risk assessment. Vertical slices of each aspect of the system, probably centered around the individual services domain functionality.

Cobb, M. (2024, January 18). How to perform a cybersecurity risk assessment in 5 steps: TechTarget. TechTarget | Security. https://www.techtarget.com/searchsecurity/tip/How-to-perform-a-cybersecurity-risk-assessment-step-by-step