### **Data leak worksheet**

**Incident summary:** A customer success representative received access to a folder of internal documents from a manager. It contained files associated with a new product offering, including customer analytics and marketing materials. The manager forgot to unshare the folder. Later, the representative copied a link to the marketing materials to share with a customer during a sales call. Instead, the representative copied a share link to the entire folder. During the sales call, the customer received the link to internal documents and posted it to their social media page.

|  |  |
| --- | --- |
| **Control** | **Least privilege** |
| **Issue(s)** | The factors that contributed to the information leak are that first the customer analytics and marketing materials were combined together, and it should’ve been anticipated that something could be wrong and should be separated. Another obvious factor is that the principle of least privilege (POLP) wasn’t adhered to, like when the manager forgot to remove the reading privilege to the customer analytics file to unauthorized users. So even if it was still accidentally shared the customer wouldn’t be able to access it and when they come back saying they weren’t able to access it you would know that a data leak was averted. |
| **Review** | For scenarios like this, the manufacturer uses implemented controls to protect against data leaks defined in NIST SP 800-53—a set of guidelines for securing the privacy of information systems. In the category of access controls, SP 800-53 lists least privilege sixth (SP 800-53: AC-6) and its purpose is that minimal access and authorization required to complete a task or function to users. It should be implemented within processes, user accounts, and roles should be enforced as necessary to achieve least privilege. |
| **Recommendation(s)** | Some improvements that may have prevented the data leak could be of Restrict access to sensitive organizational resources based on user role. Another one could be to automatically revoke access to information after a period of time. |
| **Justification** | For example, if you restrict the access to sensitive data/ resources based on the user then this current scenario would’ve been avoided. Such as when the customer representative inadvertently shared the customer analytics with the customer. If it was previously defined that privileges for the user role “customer” should not be given any privileges for the customer analytics file, then they wouldn’t have been able to access it. Also if access is automatically revoked after a period of time, then it would somewhat automate rescinding privileges instead of a person having to do it manually. This can somewhat act as a fail-safe too, like how the customer rep accidentally shared a file they weren’t supposed to |