Análise do Incidente (Vazamento):

1. **Princípio da LGPD violado:**
   O princípio da LGPD diretamente violado neste incidente é o princípio da segurança (Art. 46). A LGPD exige que os dados pessoais sejam tratados de maneira segura, adotando medidas técnicas e administrativas para proteger os dados contra acessos não autorizados, vazamentos, ou qualquer outra forma de tratamento inadequado.
2. **Vazamento da senha_hash:**
   Sim, o vazamento da senha_hash é grave, mesmo que a senha não tenha sido vazada como texto puro. Embora o hash não seja o valor original da senha, ele pode ser revertido ou decifrado com ataques de força bruta ou dicionário, principalmente se o hash utilizado for fraco ou o atacante tiver acesso a recursos computacionais suficientes. Sem um "sal" adequado (salt), o hash pode ser vulnerável, permitindo que o atacante obtenha a senha original de forma relativamente fácil. Por isso, é fundamental que senhas sejam armazenadas com algoritmos de hashing fortes e técnicas de salting para dificultar a reversão do hash.

Medidas de Proteção:

3. **Técnicas de segurança para os campos Senha e CPF:**
   - **Senha:**
     A senha deve ser armazenada após ser processada por um algoritmo de hash seguro, como o bcrypt, Argon2, ou PBKDF2. Esses algoritmos geram hashes de senha que são resistentes a ataques de força bruta, pois incluem um fator de complexidade e demoram mais tempo para gerar um hash, dificultando ataques. Além disso, deve-se aplicar o salting, que adiciona um valor aleatório ao processo de hash, aumentando a segurança ao dificultar ataques de pré-computação (rainbow tables).
   - **CPF:**
     O CPF, por ser uma informação pessoal sensível, deve ser criptografado com um algoritmo robusto, como AES (Advanced Encryption Standard), para garantir que mesmo que o banco de dados seja acessado, os dados sensíveis permaneçam protegidos. A criptografia garante que apenas usuários ou sistemas autorizados possam descriptografar os dados.

Prevenção Técnica no Banco de Dados (Controle de Acesso):

4. **Princípio do "Least Privilege" (Menor Privilégio):**
   O princípio do "Least Privilege" (Menor Privilégio) significa que cada usuário, aplicação ou processo no banco de dados deve ter apenas as permissões necessárias para realizar suas funções específicas. Ou seja, não deve ser dado acesso mais amplo do que o necessário. Isso minimiza os danos em caso de comprometimento da conta, já que um atacante não poderá acessar ou modificar dados fora do seu escopo de permissão.

Para implementar esse princípio, o grupo poderia criar perfis de usuário e limitar rigorosamente as permissões. Por exemplo, a aplicação web só deve ter permissões para ler e modificar suas próprias tabelas, sem acesso aos dados sensíveis ou administrativos.

Proposta de Perfis de Usuário no SGBD:

1. **Aplicação Web:**
    - **Exemplo de quem usa**: Sistema de Cadastro/Login
    - **Permissões de DML essenciais**:
        - SELECT: Permite ler dados dos usuários na tabela de login, como nome de usuário e status de autenticação.
        - INSERT: Permite adicionar novos usuários ou dados ao banco (como novos cadastros).
        - UPDATE: Permite atualizar informações dos usuários, como alteração de senha ou dados de perfil (apenas nas suas próprias tabelas de cadastro e login).

    Justificativa: A aplicação deve ter acesso restrito apenas às tabelas que ela precisa para autenticação e gestão de usuários. Isso evita que o atacante acesse dados críticos, como CPF ou senhas de outros usuários.

2. **Analista de Dados:**
    - **Exemplo de quem usa**: Equipe de Business Intelligence (BI) ou analistas que criam relatórios estatísticos.
    - **Permissões de DML essenciais**:
        - SELECT: Permite consultar dados do banco para análise e geração de relatórios.
        - Sem permissões de INSERT, UPDATE ou DELETE: Isso impede que o analista modifique ou exclua dados.

    Justificativa: O analista de dados precisa apenas de permissão para consultar os dados, sem a capacidade de alterar ou excluir informações, garantindo que dados críticos não sejam modificados acidentalmente.

3. **Administrador DB:**
    - **Exemplo de quem usa**: Administradores de Banco de Dados (DBAs) que configuram e mantêm o banco.
    - **Permissões de DML essenciais**:
        - SELECT: Permite leitura dos dados para monitoramento de performance e auditoria.
        - INSERT, UPDATE, DELETE: Permite manutenção dos dados no banco, como ajustes de configurações e limpeza de registros temporários.

    Justificativa: O administrador do banco de dados precisa de acesso total para realizar a configuração e manutenção do sistema, mas essas permissões devem ser restritas e auditadas para garantir que o administrador não abuse de seus privilégios. Além disso, o uso de

autenticação multifatorial e a implementação de auditorias podem melhorar a segurança.