|  |  |
| --- | --- |
| ataques cibernéticos  PESQUISA | Resumo  Este arquivo contém dois exemplos de ataques cibernéticos que ocorreram no decorrer dos últimos cinco anos.  Lauanda Jones  Sistemas Computacionais e Segurança – SCS |

**Ataque de Ransomware à JBS Foods**

* **Data aproximada:** Maio de 2021
* **Tipo de ataque:** Ransomware (sequestro de dados)
* **Descrição do ataque:** O grupo cibercriminoso REvil invadiu a rede da **JBS**, uma das maiores empresas de processamento de carne do mundo. Eles criptografaram os dados da empresa e exigiram um resgate milionário para restaurar o acesso. O ataque forçou a interrupção das operações de produção em diversas fábricas nos Estados Unidos, Canadá e Austrália, afetando a cadeia de suprimentos de alimentos globalmente.
* **Vulnerabilidade explorada:** A JBS nunca divulgou oficialmente a vulnerabilidade específica. No entanto, o FBI suspeita que o ataque tenha explorado uma vulnerabilidade em um sistema de controle de acesso (VPN) ou em sistemas de gerenciamento de arquivos. Muitos ataques de ransomware semelhantes, incluindo os do REvil, exploraram vulnerabilidades conhecidas, como as relacionadas a sistemas de RDP (Remote Desktop Protocol) desprotegidos ou falhas de segurança em firewalls e outros dispositivos de borda. No caso de VPNs, uma vulnerabilidade possível seria a **CVE-2021-21985**, que afeta a VMware vCenter Server e foi explorada por grupos de ransomware na época.
* **Impactos e/ou prejuízo:** A JBS pagou um resgate de **US$ 11 milhões em Bitcoin** para o grupo REvil para recuperar seus dados e restaurar as operações. Além do prejuízo financeiro direto, a empresa sofreu com a interrupção da produção, que afetou a oferta de carne e causou um impacto significativo na cadeia de abastecimento.
* **Tipo de proteção que poderia ter sido aplicada:**
  + **Gerenciamento de patches:** Manter todos os sistemas e softwares atualizados para corrigir vulnerabilidades conhecidas.
  + **Autenticação multifator (MFA):** Exigir um segundo fator de verificação para acesso a sistemas críticos, como VPNs e redes corporativas.
  + **Segmentação de rede:** Isolar partes da rede para que, caso uma área seja comprometida, o ataque não se espalhe para toda a infraestrutura da empresa.
  + **Backups:** Manter backups de dados offline e de forma segura, o que permite a recuperação sem a necessidade de pagar o resgate.

**Ataque de Injeção de SQL à Microsoft**

* **Data aproximada:** Agosto de 2020
* **Tipo de ataque:** Injeção de SQL (SQL Injection)
* **Descrição do ataque:** Um grupo de hackers chamado "Lapsus$" usou uma técnica de **injeção de SQL** para roubar dados de repositórios privados de código fonte da Microsoft. A injeção de SQL é uma técnica em que o atacante insere comandos maliciosos em campos de entrada de um site para manipular a base de dados subjacente. Nesse caso, os hackers exploraram uma falha em um sistema de rastreamento de bugs da Microsoft.
* **Vulnerabilidade explorada:** Embora a Microsoft não tenha divulgado um CVE específico, a injeção de SQL é uma vulnerabilidade de segurança muito comum. Ela é categorizada pelo CWE (Common Weakness Enumeration) como **CWE-89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')**. A injeção de SQL ocorre quando um aplicativo não valida ou "limpa" corretamente a entrada de dados do usuário antes de usá-la em uma consulta de banco de dados, permitindo que o atacante execute comandos SQL não autorizados.
* **Impactos e/ou prejuízo:** O principal impacto foi o roubo de dados de código fonte de produtos da Microsoft, incluindo o Bing, Cortana e Azure DevOps. O grupo Lapsus$ divulgou os dados roubados em canais do Telegram, expondo a propriedade intelectual da empresa e potencialmente permitindo que outros hackers identificassem e explorassem novas vulnerabilidades. A Microsoft teve que investir recursos significativos para analisar o código vazado e reforçar suas defesas.
* **Tipo de proteção que poderia ter sido aplicada:**
  + **Consultas parametrizadas:** Usar consultas preparadas (prepared statements) que separam a lógica do SQL dos dados de entrada, impedindo a execução de comandos maliciosos.
  + **Validação de entrada:** Implementar validação rigorosa para todas as entradas de usuário, garantindo que os dados correspondam ao tipo esperado e não contenham caracteres especiais ou comandos perigosos.
  + **Princípio do menor privilégio:** As contas de usuário do banco de dados não devem ter permissões excessivas, limitando o que um atacante pode fazer mesmo se conseguir injetar um comando.
  + **Web Application Firewall (WAF):** Um WAF pode inspecionar e filtrar o tráfego HTTP, bloqueando tentativas de injeção de SQL antes que cheguem ao servidor web.