Criptografia de Dados em Sistemas de Arquivos em Linux

1. Objetivo

Investigar modelos e tecnologias de criptografia aplicadas a sistemas de arquivos em ambientes Linux, comparando níveis (criptografia de bloco vs criptografia a nível de ficheiro) e mecanismos implementados (por exemplo LUKS/dm-crypt, fscrypt, eCryptfs).

---

2. Questões de pesquisa

1. Quais são as diferenças técnicas e operacionais entre criptografia em nível de bloco (LUKS/dm-crypt) e em nível de ficheiro (fscrypt, eCryptfs) no Linux?
2. Quais são as melhores práticas para gestão de chaves e proteção do volume root em sistemas Linux?
3. Quais são os impactos de desempenho (latência e throughput) ao ativar criptografia de disco em servidores e estações de trabalho Linux?
4. Quais ataques conhecidos afetam a criptografia em Linux?

---

3. Referências-chave

- Linux Kernel Documentation. Disponível em: https://docs.kernel.org/
- NIST SP 800-111 – **Guide to Storage Encryption Technologies for End User Devices**. Disponível em: https://csrc.nist.gov/publications/detail/sp/800-111/final
- NIST SP 800-57 – **Recommendation for Key Management**. Disponível em: https://csrc.nist.gov/publications/detail/sp/800-57-part-1/rev-5/final
- - Cloudflare – **Speeding Up Linux Disk Encryption**. Disponível em: https://blog.cloudflare.com/speeding-up-linux-disk-encryption/
- - ArchWiki – **dm-crypt/Encrypting an Entire System**. Disponível em: https://wiki.archlinux.org/title/Dm-crypt/Encrypting_an_entire_system

---

4. Achados

- Existem dois níveis principais de criptografia aplicáveis a Linux: criptografia de bloco (LUKS/dm-crypt) e criptografia a nível de ficheiro (fscrypt, eCryptfs), que protege dados por ficheiro/diretório dentro do filesystem.
- LUKS/dm-crypt é amplamente usado para proteção de volumes inteiros (incluindo root); fornece gerenciamento de cabeçalhos e slots de chave; amplamente suportado por distribuições e ferramentas (cryptsetup).

- Se tratando de solução integrada, temos o fscrypt para criptografia a nível de arquivo em filesystems como ext4, f2fs e UBIFS, oferecendo controle por diretório e menores requisitos de setup para alguns cenários.
- eCryptfs é uma solução legada, cujo uso diminuiu devido a limitações de desempenho e manutenção, sendo substituído por fscrypt e LUKS em cenários modernos (por exemplo: Wright, C. et al., Linux filesystem encryption analysis, USENIX ;login:, 2016).
- Gestão de chaves e integração com TPM, systemd, e initramfs são pontos cruciais para garantir que sistemas criptografados inicializem de forma segura sem expor chaves em disco.

---

5. Próximos passos

1. Adicionar diagramas do fluxo de inicialização do LUKS e do funcionamento interno do fscrypt.
2. Comparação entre LUKS / fscrypt / eCryptfs.