Artigo final de segurança da informação - Análise da Criptografia Ponta-a-Ponta: Um Estudo sobre o Protocolo Signal e suas implementações

Luiz Antônio Lima de Freitas Leite[1], **Max José Lobato Pantoja Junior**[1], **Wesley Pontes Barbosa**[1]

[1]Instituto de Ciências Exatas e Naturais (ICEN) – Universidade Federal Pará
Belém, PA – Brasil

{luiz.freitas.leite,max.junior,wesley.pontes.barbosa}@icen.ufpa.br

Abstract.

Resumo.

1. Introdução

2. Fundamentação Teórica: O Protocolo Signal

O Protocolo Signal é um protocolo criptográfico não federado que provê confidencialidade ponta-a-ponta e integridade de mensagens. Ele combina primitivas criptográficas avançadas para garantir propriedades essenciais de segurança.

2.1. Estabelecimento de Sessão: X3DH

Para que duas partes (comumente chamadas de Alice e Bob) iniciem uma conversa segura, elas precisam estabelecer uma chave secreta compartilhada. O Protocolo Signal utiliza o algoritmo X3DH (*Extended Triple Diffie-Hellman*).

Diferente de uma troca Diffie-Hellman padrão que exige que ambas as partes estejam online, o X3DH permite comunicação assíncrona. O servidor armazena um conjunto de chaves públicas pré-geradas por Bob (Identity Key, Signed Pre Key e One-Time Pre Keys). Quando Alice quer enviar uma mensagem:

1. Alice solicita as chaves públicas de Bob ao servidor;
2. Alice realiza cálculos matemáticos combinando suas chaves privadas com as chaves públicas de Bob para derivar uma chave secreta compartilhada (*SK*);
3. Alice envia sua própria chave pública junto com a primeira mensagem cifrada.

Isso garante que, mesmo que Bob esteja offline, a chave possa ser acordada e a mensagem enviada [**?**].

2.2. Algoritmo Double Ratchet

Após o estabelecimento da sessão, o protocolo utiliza o algoritmo *Double Ratchet* para a troca de mensagens. Este é o componente que fornece as propriedades de *Forward Secrecy* (Sigilo Futuro) e *Post-Compromise Security*. O algoritmo opera atualizando as chaves de cifração a cada mensagem enviada, combinando dois tipos de "catracas"(ratchets):

- **Ratchet Simétrica (Cadeia de Hash):** Para cada mensagem enviada dentro de uma mesma "sessão"(enquanto o outro usuário não responde), uma função de derivação de chave (KDF) atualiza a chave de mensagem. Se um atacante descobrir a chave de uma mensagem, ele não consegue calcular as chaves anteriores (devido à irreversibilidade do hash).
- **Ratchet Diffie-Hellman (DH):** Quando a conversa "vira"(Bob responde a Alice), novas chaves públicas efêmeras são trocadas. Isso introduz nova entropia no sistema. Se um atacante comprometer as chaves atuais, ele não conseguirá derivar as chaves futuras assim que um novo passo do Ratchet DH ocorrer, garantindo a recuperação da segurança (autocura).

3. Análise Comparativa: Signal vs. WhatsApp

Embora ambos os aplicativos utilizem o mesmo protocolo base para cifrar o conteúdo das mensagens (texto, áudio, vídeo), as implementações e o modelo de negócios das empresas responsáveis introduzem diferenças críticas na superfície de ataque e na privacidade.

3.1. Gerenciamento de Metadados

A distinção mais significativa reside nos metadados os dados sobre os dados.

- **Signal:** Projetado para reter o mínimo de informação. O Signal implementa a tecnologia *Sealed Sender*, onde o remetente envia a mensagem de forma que nem o servidor do Signal sabe quem está enviando, apenas quem deve receber. O servidor armazena apenas a data de criação da conta e a data da última conexão.
- **WhatsApp:** Pertencente à Meta, o modelo de negócios envolve coleta de dados. Embora a Meta não possa ler o conteúdo (devido à E2EE), ela coleta metadados extensivos: quem fala com quem, frequência das conversas, horários, endereço IP (localização aproximada), dados do dispositivo e foto de perfil. Em uma análise de segurança, esses metadados são suficientes para traçar perfis de comportamento e redes de relacionamento.

3.2. Segurança de Backups

A E2EE protege os dados em trânsito. O armazenamento (dados em repouso) é outro vetor de ataque.

- **Signal:** Oferece apenas backups locais armazenados no dispositivo do usuário, criptografados por uma senha numérica (frase de acesso). Isso transfere a responsabilidade da custódia para o usuário, aumentando a segurança contra requisições legais a servidores, mas aumentando o risco de perda de dados caso o usuário esqueça a senha.
- **WhatsApp:** Tradicionalmente realiza backups em nuvem (Google Drive ou iCloud). Durante anos, esses backups não eram protegidos pela E2EE, sendo vulneráveis a requisições judiciais às provedoras de nuvem. Recentemente, implementou-se a opção de "Backup Criptografado de Ponta-a-Ponta", onde a chave fica com o usuário ou em um cofre de segurança (HSM). No entanto, esta opção não é ativa por padrão, deixando a maioria dos usuários vulnerável.

3.3. Código Fonte e Auditoria

A transparência é vital para a criptografia.

- **Signal:** Todo o código (cliente Android/iOS e servidor) é *Open Source*. Isso permite que pesquisadores de segurança auditem o código constantemente em busca de vulnerabilidades ou *backdoors*.
- **WhatsApp:** É um software proprietário de código fechado. Embora utilize bibliotecas criptográficas auditadas, não há garantia pública de que o código compilado na loja de aplicativos corresponda exatamente à implementação segura, exigindo confiança cega na empresa (Segurança por Obscuridade).

4. Conclusão

A análise realizada demonstra que o Protocolo Signal representa o estado da arte na segurança de mensageria, resolvendo problemas complexos como o sigilo futuro e a comunicação assíncrona através do X3DH e do Double Ratchet. Tanto o Signal quanto o WhatsApp oferecem confidencialidade robusta para o conteúdo das mensagens.

Entretanto, para cenários onde a modelagem de ameaça inclui a proteção contra análise de tráfego e metadados, o aplicativo Signal mostra-se superior. O WhatsApp, apesar de seguro no conteúdo, expõe uma superfície de metadados que pode comprometer a privacidade do usuário dependendo do contexto. Conclui-se que a segurança da informação não é apenas uma questão de algoritmos criptográficos, mas também de políticas de implementação e privacidade.

Referências