Um sistema de criptografia de broadcast para navegadores de internet

Hilder Vitor Lima Pereira

Instituto de Matemática e Estatística da Universidade de São Paulo

hilder.pereira@usp.br

Motivação

Junto com as recentes notícias que revelaram um esquema de espionagem internacional praticado pela *National Security Agency* (NSA)[TG13], agência estadunidense de segurança, houve o surgimento (e o reaparecimento) de muitos sistemas ligados à privacidade[TM13]. Dentre eles, destacam-se as soluções livres[LM13], de código aberto, como os sistemas listados no site do *PRISM Break*[PB], que vão desde criptografia de e-mails, como o sistema *Mailvelope*, até sistemas de navegação privada, como o *Tor Browser Bundle*.

Estas ferramentas, em conjunto, oferecem um bom nível de privacidade, mas, em geral, demandam conhecimento técnico e, em alguns casos, o próprio usuário é quem fica responsável pelas chaves usadas para criptografar as mensagens. Além disto, muitas destas ferramentas são feitas para casos muito específicos (um para e-mail, um para chat, um para HTTPS, etc), o que acaba acarretando na necessidade de se utilizar vários sistemas em conjunto. Isto tudo dificulta o acesso para pessoas mais leigas.

Por isto, este trabalho propõe um sistema que criptografa campos das páginas web, pois isto torna o sistema mais geral, funcionando para uma gama grande de sites, que faça o próprio gerenciamento de chaves (criação, armazenamento, distribuição), pois isto elimina boa parte das exigências de conhecimentos técnicos e facilita a sua utilização.

Além disto, ele é baseado em esquemas de criptografia para sistemas de *broadcast*, o que possibilita seu uso em redes sociais, fóruns e outros sites onde mensagens são enviadas para um grupo de pessoas, ao invés de enviadas para um único destinatário.

Características do sistema

§ Criptografa uma mensagem para um grupo de contatos inteiro.

§ Reconhece os campos editáveis nas páginas e as criptografa (e descriptografa) na própria página.

§ Os dados são criptografados antes que eles sejam enviados ao servidor.

§ Protege contra ataques do tipo ”Homem no meio”.

§ Impede que os provedores de serviços tenham acesso aos dados dos usuários.

§ Faz o gerenciamento das chaves dos usuários.

§ Gerencia os grupos de contatos dos usuários.

A imagem abaixo mostra o princípio de funcionamento do sistema. Ela mostra a situação em que um usuário, digamos, João (o homem no canto superior esquerdo), envia uma mensagem a uma rede social (figura com fundo azul) e esta rede social redistribui a mensagem para os contatos que João tem nesta rede social (as pessoas na parte inferior). Além disto, a rede social fornece a mensagem a um terceiro (homem a direita).

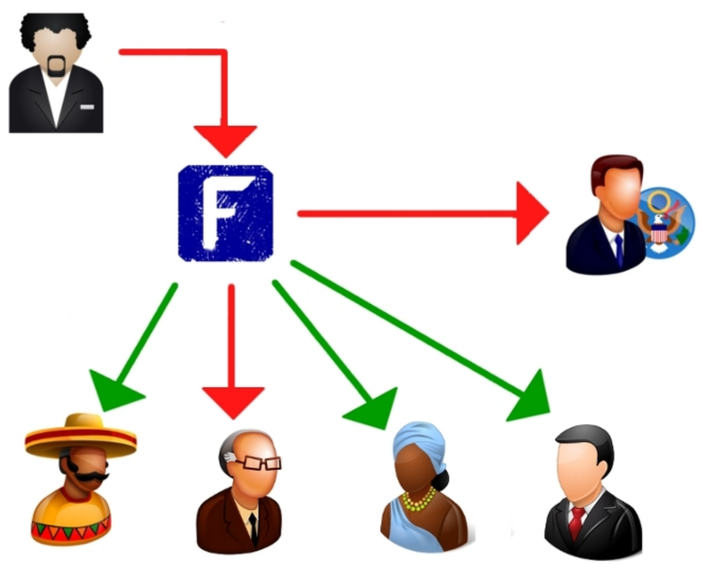


Figura 1: *Mensagem sendo enviada a rede social e sendo redistribuida*

As setas vermelhas indicam que o destino só terá acesso à mensagem criptografada e as setas verdes indicam que o destino terá acesso à mensagem original.

Note que entre os contatos de João na rede social, há um que não tem acesso à mensagem original (senhor de óculos). O que indica que ele não foi colocado por João no grupo de pessoas com permissão para descriptografar as mensagens deste.

Algoritmos de criptografia

1. *Advanced Encryption Standard*: é um algoritmo de criptografia simétrica, criado pela *National Institute of Standards and Technology* (NIST), para ser o sucessor do algoritmo DES (*Data Encryption Standard*). Ele efetua a criptografia de blocos de 128 bits usando uma chave de 128, 192 ou 256 bits[AES]. Pelo fato da chave não ser tão grande e efetuar a criptografia por substituição, ao invés de manipulações numéricas, este algoritmo é mais rápido do que algoritmos de chave pública.
2. *ElGamal*: é um algoritmo de criptografia assimétrica. Seu funcionamento é o seguinte:
   (a) Criptografia: Considere uma tripla (g, a, x), com $(g, a) \in G^2$ e $x \in \mathbb{Z}$, onde $(G, \cdot)$ é um grupo e $g^x = a$. Tanto o grupo quantos os valores g e a são públicos. O valor x é chamado de chave privada. Então, uma mensagem m é criptografada para o par (A_0, A_1), onde

   $$A_0 = g^k$$

   e

   $$A_1 = m \cdot a^k$$

   sendo k uma chave de sessão (é gerado um valor aleatório k para cada mensagem criptografada).
   (b) Descriptografia: Dado (A_0, A_1) como acima e a chave privada x: calcula-se $A_2 = A_0^x$, e então

   $$A_1 \cdot (A_2)^{-1} = (mA_0^k) \cdot (A_0^x)^{-1} = (mg^{xk}) \cdot (g^{xk})^{-1} = m$$

   onde A_2^{-1} é o inverso de A_2 no grupo $(G, \cdot)$.

Os dois algoritmos apresentados acima foram combinados em um esquema de criptografia baseado no sistema apresentado em [GW89], conforme explicado a seguir:

Para cada usuário U, definimos $C(U) = \{c_1, c_2, .., c_M\}$ como o conjunto de todos os contatos de U e $N_U \in \mathbb{N}^*$, tal que N_U é coprimo de N_V, para todo $U \neq V$.

Ou seja, para cada usuário há um valor natural positivo tal que o máximo divisor comum entre quaisquer dois destes valores seja igual a 1.

Inicialização: No início da sessão de um usuário U, o sistema gera uma chave K_U, que é usada para criptografar as mensagens, criptografa esta chave, usando ElGamal e as chaves públicas dos contatos e resolve um sistema de equações modulares, para usar o resultado como um *locker*.

Ou seja, considerando que cada contato c_i tem como chave pública o par (g_i, a_i), ele faz

$$\forall c_i \in C(U), R_i = ElGamal(g_i, a_i, K_U)$$

e então acha X tal que

$$\begin{cases} X \equiv R_1 (mod N_1) \\ X \equiv R_2 (mod N_2) \\ ... \\ X \equiv R_M (mod N_M) \end{cases}$$

Note que, pelo Teorema Chinês do Resto, tal X existe e é único módulo $\prod_{i=1}^{M} N_i$.

Criptografia: para criptografar uma mensagem m escrita pelo usuário U, o sistema utiliza o algoritmo AES com a chave de sessão K_U e concatena com o *locker* X e o *id* de U. Ou seja, faz

$$E = AES(K_U, m)$$

e envia

$$(id, X, E)$$

Descriptografia: digamos que um usuário c_i que seja um contato de U, isto é, $c_i \in C(U)$, recebe a mensagem (id, X, E), então, o sistema verifica que c_i está no grupo de contatos de U usando o *id*, consegue a chave de sessão K_U a partir de X e obtém a mensagem original a partir de E, usando K_U.

Para conseguir a chave de sessão a partir de X, é calculado R_i

$$R_i \equiv X (mod N_i)$$

e então R_i é descriptografado com a chave privada de c_i e o algoritmo ElGamal, ou seja,

$$K_U = ElGamal(x_i, R_i)$$

e com K_U se obtém a mensagem m a partir de E

$$m = AES(K_U, E)$$

Utilização

O sistema implementado é uma extensão para o navegador *Mozilla Firefox*. Cada usuário deve instalar a extensão no seu navegador, efetuar um cadastro simples (digitando o nome de usuário, um e-mail e uma senha em uma janela da própria extensão) e cadastrar os contatos (digitando os nomes de usuário das pessoas que poderão descriptografar suas mensagens).

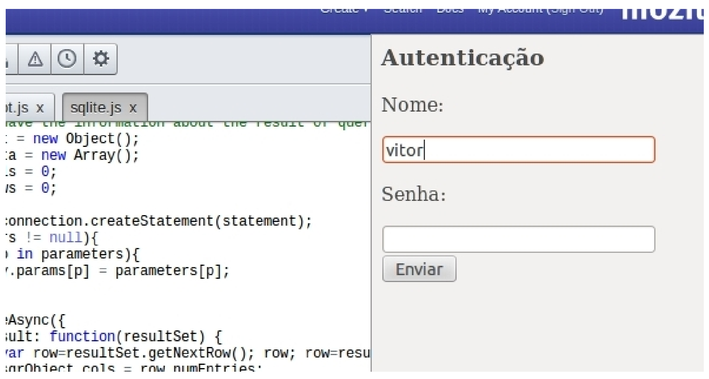


Figura 2: Pop-up *para realizar autenticação do usuário*

Para criptografar um campo qualquer, basta apertar as teclas de atalho (CRTL + ESPAÇO). O sistema automaticamente pega o texto do campo que está com o foco e o criptografa.

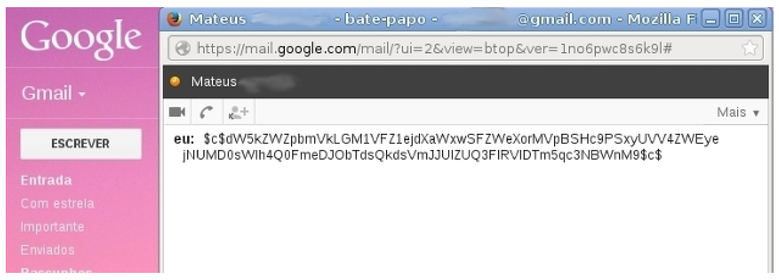


Figura 3: *Conversa do GTalk criptografada*

Toda vez que uma página é carregada, o sistema procura, dentro da página, por algum texto que tenha sido criptografado. Caso encontre, verifica se o texto foi criptografado por algum usuário cuja lista de contatos inclui o usuário que está carregando a página. Em caso afirmativo, o texto é descriptografado e a mensagem original é mostrada. Em caso negativo, o texto criptografado é carregado na página.



Figura 4: *Conversa do GTalk descriptografada*

Referências

[AES] FEDERAL INFORMATION PROCESSING STANDARDS, The ADVANCED ENCRYPTION STANDARD *http://csrc.nist.gov/publications/*

[GW89] GUANG-HUEI CHIOU E WEN-TSUEN CHEN, Secure Broadcasting Using the Secure Lock *IEEE Transactions on Software Engineering*, Volume 5, nº 8.

[LM13] LE MONDE, PRISM – Comment passer entre les mailles de la surveillance d'Internet ? *http://bigbrowser.blog.lemonde.fr/2013/06/11/prism-comment-passer-entre-les-mailles-de-la-surveillance-dinternet/*

[PB] PRISM BREAK, PRISM Break home page *https://prism-break.org/*

[TG13] THE GUARDIAN, NSA paid millions to cover Prism compliance costs for tech companies. *http://www.theguardian.com/world/2013/aug/23/nsa-prism-costs-tech-companies-paid*, 2013.

[TM13] TIME, The Anonymous Internet: Privacy Tools Grow in Popularity Following NSA Revelations. *http://business.time.com/2013/06/20/the-anonymous-internet-privacy-tools-grow-in-popularity-following-nsa-revelations/*