Sistema para Autenticação entre Clientes, Técnicos e ISPs

Vagner E. Quincozes[1,3], Daniel Temp[1,2,3], Silvio E. Quincozes[4], Diego Kreutz[1,2,3], Rodrigo B. Mansilha[1,2,3]

[1] Laboratório de Estudos Avançados (LEA)

[2]Programa de Pós-Graduação em Engenharia de Software (PPGES)

[3]Universidade Federal do Pampa (UNIPAMPA)

[4]Universidade Federal Fluminense (UFF)

Resumo. *O processo de suporte dos Internet Service Providers (ISPs) regionais ainda costuma utilizar protocolos frágeis de autenticação, como as carteirinhas físicas (ou virtuais) baseadas em dados estáticos não autenticáveis. Essas carteirinhas podem ser facilmente roubadas, clonadas, ou reproduzidas, comprometendo o processo de autenticação entre as entidades envolvidas com suporte técnico, presencial e remoto, do ISP (clientes, técnicos, e gestores). Neste trabalho, propomos um sistema composto por mecanismos e protocolos para autenticar e identificar de forma segura as referidas entidades. A ferramenta Scyther foi utilizada para realizar uma verificação automática do protocolo proposto.*

1. Introdução

A expansão local e regional dos ISPs, ou provedores de serviços de Internet, vem sendo motivada por diferentes fatores, como a proximidade com o cliente, a personalização no atendimento e a agilidade no suporte técnico [Bettio 2016]. No entanto, os investimentos em agilidade e economicidade são priorizados em relação aos investimentos em segurança.

Ao observarmos o modo de operação de ISPs regionais[1], identificamos que o processo de autenticação das entidades envolvidas em suporte (*i.e.,* cliente, técnico e gestor de atendimento do ISP) tem sido realizado empregando protocolos e mecanismos frágeis de autenticação, como as carteirinhas físicas (ou virtuais) baseadas em dados estáticos não autenticáveis (*e.g.*, foto, nome e código de barras estático). Em primeiro lugar, essas carteirinhas podem ser roubadas, clonadas, ou reproduzidas, comprometendo o processo de autenticação entre as entidades mencionadas anteriormente. Em segundo lugar, um cliente do ISP, atualmente, não dispõe de nenhuma forma para autenticar o técnico, o qual chega até sua casa para realizar o suporte técnico especializado. A carteirinha do técnico é simples, contendo apenas uma foto, um nome e, em alguns casos, um código de barras estático contendo um dado identificador do técnico, como seu CPF.

Na literatura existem propostas de protocolos de autenticação de múltiplas entidades, como *Needham-Schroeder* (NS, α), *Needham-Schroeder-Lowe* (NSL, β) e *Bilateral Key Exchange* (BKE, β^*) [Cremers and Mauw 2006]. Em particular, por questões de eficiência, o protocolo BKE emprega criptografia simétrica em vez de criptografia assimétrica em uma das etapas da autenticação. Entretanto, esses protocolos tradicionais assumem a existência de uma infra-estrutura de chaves públicas (do inglês, *Pu-*

[1]Nós omitimos os nomes dos ISPs por questões de privacidade.

blic Key Infrastructure – PKI). Tal abordagem pode ser complexa, indesejada ou desnecessária em cenários como o da gestão da segurança de infra-estruturas de redes programáveis [Kreutz et al. 2019].

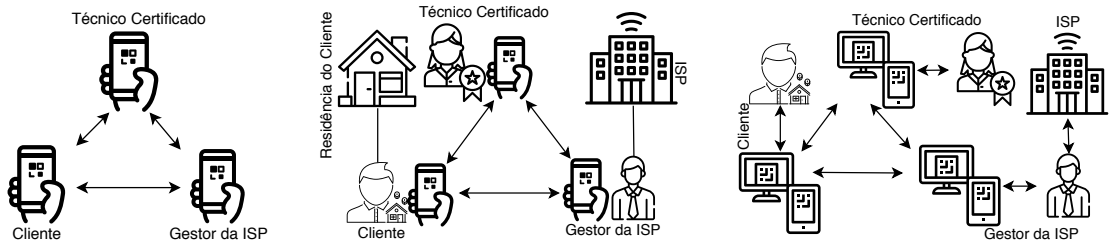
Soluções de autenticação mais recentes, voltadas para aplicativos de dispositivos móveis, como a 2FMA-NetBank [Pratama and Prima 2016], também dependem de uma PKI e, além disso, adicionam múltiplos fatores de autenticação priorizando aspectos de segurança em detrimento de aspectos de usabilidade. Recentemente, propomos a Auth4App [Kreutz et al. 2020], que consiste em um conjunto de protocolos projetados para a autenticação de duas entidades quaisquer (*e.g.*, duas pessoas) utilizando um aplicativo para dispositivos móveis. A Auth4App destaca-se por: (i) não depender de uma PKI para realizar o processo de vinculação e autenticação de entidades e (ii) não necessitar de múltiplos fatores tradicionais de autenticação (*e.g.*, login/senha e um SMS com código de verificação) para garantir níveis mais elevados de segurança.

O principal objetivo deste trabalho é apresentar uma solução de autenticação para múltiplas entidades (*i.e.,* clientes, técnicos e gestores) de ISPs regionais (Seção 2), combinando as características essenciais de protocolos como o BKE e de soluções recentes como a Auth4App. O protocolo principal da solução proposta, denominado BKE-Auth4ISP, foi formalmente verificado utilizando a ferramenta Scyther, como apresentado na Seção 3. Por fim, apresentamos conclusões e trabalhos em andamento na Seção 4.

2. Sistema de Autenticação

2.1. Casos de Uso

O cenário de autenticação mútua ilustrado na Figura 1(a) pode ser aplicado em dois casos de uso: ambientes físicos e ambientes virtuais. O primeiro caso trata-se de autenticação presencial, isto é, os dois indivíduos principais (*e.g.*, cliente e técnico) estão no mesmo ambiente físico, como a residência do cliente, gerenciados por um gestor remoto. A Figura 1(b) ilustra a visita de técnico certificado para prestar atendimento a um cliente do ISP. Ao chegar no local do atendimento, o técnico deve apresentar o seu código de identificação, que pode ser representado através de um *QR Code* contendo uma imagem de baixa resolução, dados pessoais do técnico, dados do atendimento e um código de autenticação. O cliente do ISP utilizará o seu *smartphone* para ler o *QR Code* e verificar se a pessoa é de fato o técnico certificado indicado por um gestor do ISP. Similarmente, o técnico poderá ler o *QR Code* do cliente para verificar se ele é realmente o cliente correto.



(a) Autenticação da Tríade (b) Ambiente Físico (c) Ambiente Virtual

Figura 1. Casos de Uso da solução de autenticação proposta.

A segunda forma de autenticação mútua, conforme ilustrado na Figura 1(c),

refere-se ao acesso remoto de suporte (*e.g.*, técnico do ISP acessando o roteador WiFi do cliente). Nesse caso, o técnico certificado irá prestar um atendimento remoto a um cliente do ISP. De maneira análoga ao primeiro cenário, o cliente e o técnico poderão autenticar-se mutuamente. Contudo, diferentemente do primeiro caso, os dados de verificação serão lidos através da tela do computador (ou outro dispositivo) com acesso à Internet, dispensando que os indivíduos estejam no mesmo ambiente físico. Vale ressaltar que, em ambos os casos, nós assumimos que todos os participantes possuem conexão com a Internet.

2.2. Modelo de Ataque

Neste trabalho, assumimos que os gestores do ISP são responsáveis por identificar e cadastrar novos usuários (clientes, técnicos e gestores). Essa premissa é razoável uma vez que, por exemplo, o cliente (tipicamente) estabelece uma relação de confiança ao assinar o contrato de prestação de serviços com o seu ISP. Como implicação, na solução proposta (Seção 2.3), consideramos que os clientes, técnicos e gestores já estão devidamente identificados pelo ISP. A seguir são discutidas as premissas particulares para cada um dos casos de aplicação da solução investigada: ambientes físico e virtual.

No ambiente físico, presumimos que o *smartphone* utilizado pelo usuário, seja cliente ou técnico, é confiável. Essa premissa se justifica pois em caso de substração do dispositivo autenticado, o usuário deverá notificar o ISP. Em seguida, o ISP deverá revogar a identificação do dispositivo e, assim, impedir que um atacante possa se beneficiar do dispositivo subtraído para efetuar ataque de personificação.

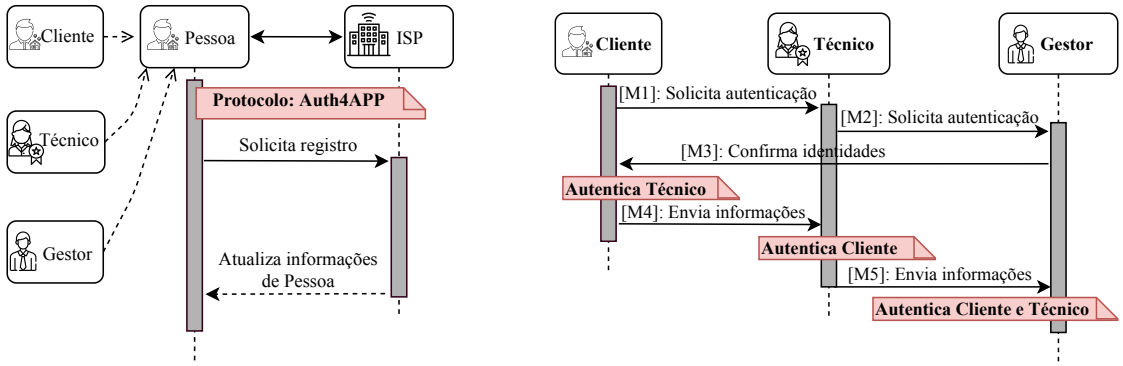
No ambiente virtual, o atacante pode realizar ataques de personificação mais sofisticados a partir de acesso à rede. Especificamente, modelamos as capacidades do atacante seguindo o modelo Dolev-Yao [Dolev and Yao 1983]. Assim, um intruso poderá controlar a rede e descriptografar as mensagens se e somente se conhecer a chave de descriptografia. Além disso, o atacante poderia modificar, atrasar e inserir comunicação nas mensagens entre usuários [Cremers and Mauw 2006].

2.3. Autenticando Clientes, Técnicos e Gestores de ISPs

A Figura 2 ilustra o sistema proposto para autenticar o Cliente, Técnico e Gestor do ISP. Tais entidades são consideradas instâncias da entidade `Pessoa`. Cada uma dessas entidades deve ser devidamente identificada no ISP. Posteriormente, a cada atendimento é realizado um processo de autenticação entre cada uma dessas três instâncias.

Para identificar as pessoas que futuramente serão envolvidas em atendimento, especializamos o protocolo apresentado em [Kreutz et al. 2020], conforme ilustrado na Figura 2(a). A pessoa pode ser qualquer entidade da tríade cliente-técnico-gestor, dependendo do cenário. A Figura 2(b) resume o protocolo de autenticação, denominado *Bilateral Key Exchange for ISP* (BKE4ISP). Esse protocolo é uma instância do BKE* – o menos computacionalmente custoso proposto em [Cremers and Mauw 2006] e suficientemente seguro para o modelo de ataque considerado neste trabalho. Além de especificar as entidades em termos de quantidade ($n = 3$) e papéis, BKE4ISP emprega o Auth4App como mecanismo alternativo a uma infra-estrutura PKI clássica.

No Protocolo 1, realizamos duas modificações fundamentais em relação ao protocolo BKE original. Primeiro, utilizamos *One Time Authentication Codes* (OTACs), como proposto na solução Auth4App, e assim eliminamos a necessidade de uma infra-estrutura



(a) Mecanismo Auth4ISP. (b) Protocolo BKE-Auth4ISP.

Figura 2. Sistema proposto para autenticação de atendimento de ISP.

PKI clássica. Com OTACs, podemos utilizar apenas algoritmos de cifra simétrica, aumentando o desempenho e assegurando a resistência a ataques de computadores quânticos, como pode ser visto em trabalhos anteriores [Kreutz et al. 2019]. Segundo, adicionamos um código de autenticação de mensagem para acelerar o processo de verificação da autenticidade de cada mensagem. O custo computacional de um HMAC é significativamente inferior ao de um algoritmo de cifra (*e.g.*, ver gráfico de desempenho das comunicações com apenas HMAC ou com a adição de algoritmos de cifra [Kreutz et al. 2019]). O desempenho pode impactar na resiliência do protocolo em situações de ataques de negação de serviço (do inglês, *Denial of Service* – DoS), por exemplo.

Protocolo 1. BKE-Auth4ISP

1	Cliente → Técnico	[E(Cli, Isp, nonce-cli)]HMAC
2	Técnico → Gestor do ISP	[E(Tec, Cli, nonce-cli, nonce-tec)]HMAC
3	Gestor do ISP → Cliente	[E(nonce-cli, nonce-tec, nonce-isp, Tec, Isp)]HMAC
4	Cliente → Técnico	[E(nonce-isp,nonce-tec)]HMAC
5	Técnico → Gestor do ISP	[E(nonce-isp)]HMAC

Na prática, para utilizar os OTACs, um cliente pode ler os dados de identificação contidos em um *QR Code* do aplicativo do técnico. O processo é análogo a uma verificação manual, pouco confiável, dos documentos de identificação do técnico, como carteirinha física ou RG. Os OTACs são definidos aos pares, ou seja, entre o cliente e o técnico, entre o técnico e o gestor do ISP e entre o gestor do ISP e o cliente. A inicialização do gerador desses códigos dinâmicos e únicos ocorre durante o cadastro e a vinculação ao aplicativo de cada entidade junto ao ISP.

Podemos observar no Protocolo 1 (BKE4ISP) que todas as mensagens entre duas entidades quaisquer (*e.g.*, cliente-técnico e técnico-gestor do ISP) são cifradas utilizando uma função E (do inglês, **E**ncrypt), cuja chave secreta é um OTAC. O protocolo inicía com o cliente enviando (linha 1) uma mensagem para o técnico contendo um nonce de autenticação (`nonce-cli`), sua identificação (`Cli`) e a identificação do gestor do ISP (`Isp`). O técnico acrescenta o seu identificador (`Tec`) e o seu nonce de autenticação (`nonce-tec`) e envia para o gestor do ISP (linha 2). O gestor do ISP envia uma mensagem ao cliente (linha 3) contendo os nonces de autenticação recebidos, confirmando assim as respectivas identidades, e o nonce `nonce-isp` do ISP, que é utilizado para fi-

nalizar o processo de autenticação mútua entre o cliente e o técnico. O cliente recebe a mensagem do gestor do ISP (linha 3), autentica o técnico e envia os nonces `nonce-isp` e `nonce-tec` para o técnico. O técnico verifica o nonce de autenticação recebido, autenticando o cliente, e envia o nonce `nonce-isp` para o gestor do ISP, finalizando o processo de autenticação.

3. Avaliação

Como primeiro passo para validar a proposta conceitual, avaliamos o protocolo BKE-Auth4ISP. Para tanto, utilizamos a ferramenta *Scyther* [Cremers 2006] em sua versão *Standard* mais recente (1.1.3)[2] para a plataforma Microsoft Windows. As configurações são as que seguem: `--auto-claims` (que gera automaticamente os testes a serem avaliados) e `--max-runs= 6` (que determina o número de rodadas realizadas) [Cremers 2008]. A implementação proposta é apresentada no Algoritmo 1 e pode ser encontrada em repositório digital[3]. Há três papéis definidos, cada um em um bloco distinto: Cliente (linhas 4-10), Técnico (linhas 11-18) e Gestor do ISP (linhas 19-25). Para cada mensagem enviada (*e.g.*, linha 7) há uma mensagem recebida correspondente (linha 14).

Algoritmo 1: BKE4ISP na linguagem Scyther.

```
1  usertype String, MessageKey; const ISP-CLI, ISP-TEC,
     palavraVazia: String;
2  secret otac: MessageKey;
3  protocol BKE-Auth4-ISP(Cli, Tec, Isp){
4  role Cli {
5     var nonce-tec, nonce-isp : Nonce;
6     fresh nonce-cli : Nonce;
7     send_1(Cli, Tec, {nonce-cli, Cli, Isp}otac);
8     recv_3(Isp, Cli, {nonce-cli, nonce-tec, nonce-isp, Tec,
      Isp}otac);
9     send_4(Cli, Tec, ({nonce-isp}nonce-tec));
10 }
11 role Tec {
12    var nonce-cli, nonce-isp : Nonce;
13    fresh nonce-tec : Nonce;
14    recv_1(Cli, Tec, {nonce-cli, Cli, Isp}otac);
15    send_2(Tec, Isp, {nonce-cli, nonce-tec, Tec, Cli}otac);
16    recv_4(Cli, Tec, ({nonce-isp,nonce-tec}otac));
17    send_5(Tec, Isp, ({nonce-isp}otac));
18 }
19 role Isp {
20    var nonce-tec, nonce-cli : Nonce;
21    fresh nonce-isp : Nonce;
22    recv_2(Tec, Isp, {nonce-cli, nonce-tec, Tec, Cli}otac);
23    send_3(Isp, Cli, {nonce-cli, nonce-tec, nonce-isp, Tec,
      Isp}otac);
24    recv_5(Tec, Isp, ({nonce-isp}otac));
25 } }
```

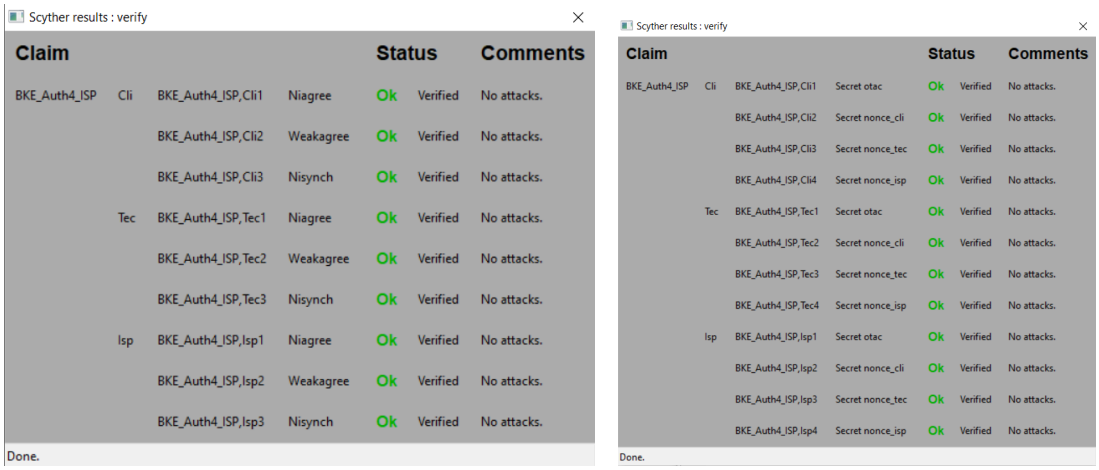
A Figura 3 apresenta um relatório que aponta os resultados da avaliação. A coluna ***Claim*** apresenta o protocolo testado (*BKE_Auth4_ISP*), os indicadores analisados (*e.g., Cli, Tec e Isp*), o indicador único para cada evento (*e.g., BKE_Auth4_ISP, Cli1*) e um evento de afirmação (*e.g., Secret nonce_cli*). Nas colunas ***Status*** e ***Comments*** são reportados possíveis ataques [Jenuario et al. 2020]. Observe-se que, para todas as linhas resultantes, o campo *Status* mostra *OK Verified* e que o campo *Comments* apresenta *No Attacks*, o que indica que o protocolo é seguro segundo a metodologia adotada.

4. Conclusão e Trabalhos em Andamento

Neste trabalho, propomos e avaliamos um sistema para autenticação entre clientes, técnicos e gestores de ISPs. O sistema é composto por uma instanciação de um protocolo de autenticação combinado com um mecanismo recente, eficiente e seguro para identificação e autenticação. Realizamos a verificação formal do protocolo de segurança utilizando a ferramenta *Scyther* e constatamos que o protocolo não apresenta falhas.

[2]Disponível em `https://people.cispa.io/cas.cremers/scyther/`

[3]Disponível em `https://github.com/vagnerereno/autenticacaoparaisps.git`



(a) Propriedades Niagree, Weakgree, Nisynch. (b) Propriedades secretas.

Figura 3. Resultado da verificação do protocolo *BKE_Auth4_ISP* com *Scyther*.

Atualmente, estamos implementando um protótipo da solução, incluindo um aplicativo para *smartphone* integrado com o sistema proposto. Além disso, vislumbramos a elaboração de uma plataforma para aproximar e facilitar o contato entre técnicos terceirizados credenciados por ISPs para agilizar atendimento dos clientes, especialmente em cenários de enxurradas de demandas. Por fim, pretendemos aplicar novas instâncias da solução em outros mercados, como entregas de produtos e serviços à domicílio.

Agradecemos ao apoio da FAPERGS (PROBITI - Bolsa de Iniciação Tecnológica - Edital nº 102/2020), que beneficiou o acadêmico Vagner Ereno Quincozes do curso de Engenharia de Software.

Referências

Bettio, L. W. d. (2016). O crescimento da internet no Brasil, serviços e regulamentação.

Cremers, C. and Mauw, S. (2006). A family of multi-party authentication protocols. In *First Benelux Workshop on Information and System Security (WISSec)*.

Cremers, C. J. (2008). The Scyther Tool: Verification, Falsification, and Analysis of Security Protocols. In *International conference on computer aided verification*, pages 414–418. Springer.

Cremers, C. J. F. (2006). *Scyther: Semantics and verification of security protocols*. Eindhoven university of Technology Eindhoven, Netherlands.

Dolev, D. and Yao, A. (1983). On the security of public key protocols. *IEEE Transactions on information theory*, 29(2):198–208.

Jenuario, T., Chervinski, J. O., Paz, G., Beltran, R., Fernandes, R., and Kreutz, D. (2020). Verificação Automática dos Protocolos de Segurança Needham-Schroeder, WMF e CSA com a ferramenta Scyther. *Revista Eletrônica Argentina-Brasil de Tecnologias da Informação e da Comunicação*, 3(1).

Kreutz, D., Fernandes, R., Paz, G., Jenuario, T., Mansilha, R., Immich, R., and Miers, C. C. (2020). Auth4App: Protocols for Identification and Authentication using Mobile Applications. In *SBC 20th International Brazilian Symposium on Information and Computational Systems Security (SBSeg)*, pages 1–14. SBC.

Kreutz, D., Yu, J., Ramos, F. M. V., and Esteves-Verissimo, P. (2019). ANCHOR: Logically centralized security for software-defined networks. *ACM Transactions on Privacy and Security*, 22(2):8:1–8:36.

Pratama, A. and Prima, E. (2016). 2FMA-NetBank: A Proposed Two Factor and Mutual Authentication Scheme for Efficient and Secure Internet Banking. In *2016 8th International Conference (ICITEE)*, pages 1–4. IEEE.