# Avaliação da Confiança por Análise Comportamental para Autorização Evolutiva em Aplicações Seguras na Web

**Luiz Fernando Rust da Costa Carmo, Carlos Augusto Reis Júnior**

Núcleo de Computação Eletrônica – Universidade Federal do Rio de Janeiro (UFRJ)
Caixa Postal 2324 – 20.010-974 – Rio de Janeiro – RJ – Brasil

{rust,carlosjunior}@nce.ufrj.br

***Abstract*** *This paper deals with a joint use of a trust evaluation approach and access control mechanisms for improving security in Web-usage. Trust evaluation is achieved by means of both behavioral evaluation and credentials exchange, in such way that transitions among different access polices are automatically fired whenever a user behavior is validated. Behavioral analysis uses machine-learning techniques to gain knowledge about users navigation tracks, creating a user signature to be compared with a current behavior of the respective user. This mechanism is validated through the use of different simulation scenarios.*

***Resumo.*** *Esse artigo trata do problema de estabelecimento da confiança em aplicações seguras na Web e seu uso integrado em mecanismos de controle de acesso. Basicamente, é proposta uma estratégia de avaliação da confiança que combina avaliação comportamental e troca de credenciais. As transições entre diferentes políticas de acesso são feitas automaticamente quando é possível validar o comportamento do usuário. O mecanismo de avaliação comportamental usa uma máquina de aprendizagem por análise da trilha de navegação do usuário, que confronta o comportamento atual com uma assinatura histórica. O mecanismo é validado por simulações que atestam a viabilidade da sua utilização no contexto de diferenciação comportamental.*

## 1. Introdução

Atualmente existe uma grande preocupação quanto à segurança dos aplicativos Web. A Infra-estrutura de chaves públicas (ICP) é sem dúvidas uma grande aliada na resolução de muitas das questões relativas à segurança, especificamente no que se refere à autenticação. Em [Lopez et al. 2004] discute-se a diferença entre autenticação e autorização: (i) um serviço de autenticação prova que a identidade de um objeto/sujeito é de fato a que ele diz ter; enquanto que (ii) a autorização significa a concessão de permissão baseada na identificação autenticada. Esta última definição pode ser alterada com a introdução do conceito de "confiança": normalmente uma entidade pode dizer que "confia" em outra, quando ela assume que essa segunda entidade vai se comportar exatamente como ela espera. Dessa forma a autorização pode ser reescrita como "concessão de permissão baseada na confiança depositada". O fato é que, talvez mais importante do que saber com quem está se relacionando, é saber como essa pessoa/objeto se comportará. Por outro lado, uma autenticação sem fraudes é ainda a melhor forma de se antever um perfil de comportamento anunciado e, a combinação da

autenticação com análise comportamental contínua permite verificar gradualmente a coerência da união destes fatores, gerando uma consolidação da confiança.

Verifica-se atualmente um sensível aumento de propostas para incorporar mecanismos baseados em confiança, tanto em aplicações Web, como em serviços Web e computação ubíqua. A maioria destes mecanismos propõe o uso de credenciais digitais para uma gerência eficaz do estabelecimento da confiança. O objetivo é, após uma etapa inicial de autenticação, gerenciar as demandas do usuário e evoluir o nível de confiança em função da troca de credenciais em momentos pré-determinados. Para isso, a gerencia de confiança está atrelada a um mecanismo de controle de acesso de forma granular (ex. RBAC [Lopez et al. 2004]) de forma que a evolução da relação de confiança implica numa alteração dos privilégios de acesso atribuídos a este usuário.

Um exemplo clássico deste tipo de abordagem é a relação entre um usuário e um *site* de compras [Skogsrud et al. 2004]: após a autenticação, o usuário (i) navega no *site* e seleciona um dado objeto para compra; neste momento ele deve mudar de privilégios para (ii) consolidar o pagamento; esta transição é controlada por um número de cartão de afinidade (exemplo de credencial) que, em caso de sucesso, implica num aumento do nível de confiança e na atribuição de um novo "papel" da política de acesso, proporcionado os privilégios necessários para a (iii) continuidade da operação.

Este artigo basicamente propõe uma estratégia de avaliação da confiança que não seja apenas guiada pela revelação gradativa de credenciais, mas faça uso também de um mecanismo de avaliação comportamental por análise continua da conduta contemporânea de um usuário vis-à-vis das atividades regressas, gerando subsídios para uma possível evolução do nível de confiança. No exemplo anterior do *site* de compras, se o usuário é um cliente antigo, pode-se gradativamente estabelecer uma certa assinatura quanto a sua forma de navegação, de forma que, a trajetória de navegação em relação a essa assinatura seja analisada em cada nova operação, promovendo um aumento automático do nível de confiança, sem uma necessidade de uma troca de credenciais (cartão de afinidade).

A idéia básica é desenvolver um sistema contínuo de avaliação comportamental do usuário, onde confiança e restrições de acesso podem ser deduzidas automaticamente em aplicações Web. Porém o mecanismo não está restrito a aplicações Web, sendo que a visão de computação ubíqua [Langheinrich 2003] parece reforçar ainda mais o papel da confiança, não somente pela alta descentralização inerente, mas também pelo propagado *modus operandi* não-intrusivo. A confiança, neste tipo de aplicação, substituiria integralmente os métodos de autenticação tradicionais, suportando o conceito de usuários livres (sem certificados, logins e senhas).

Serviços Web também vêm ganhando mais e mais importância como tecnologias habilitadoras para o desenvolvimento de aplicações distribuídas orientadas a serviço. E conseqüentemente, com o crescente número de serviços, especialmente em redes corporativas, cresce também a complexidade para autenticar e administrar os privilégios dos usuários, criando um ambiente favorável para emprego do conceito de confiança [Plaltzer 2004].

## 2. Trabalhos Relacionados

Abordagens unindo modelos de políticas para controle de acesso e gerenciadores de confiança são relativamente recentes e visam o estabelecimento da confiança de uma forma gradual e interativa, de forma a atualizar dinamicamente privilégios de acesso [Bacon et al. 2003].

Em [Skogsrud et al. 2003] é proposto um *framework* para negociação da confiança (*Trust-Serv*) para serviço Web. A relação de confiança evolui no tempo via troca de credenciais controladas por uma máquina de estados associada à aplicação. Alguns exemplos de credenciais são cartões de crédito, passaporte e cartões de afinidades. Os atributos destas credenciais são avaliados para habilitar, ou não, a troca de estado e a respectiva alteração do perfil de acesso do usuário.

[Tatyana R. et al. 2005] propõe um *framework* para controle de acesso adaptativo e negociação da confiança que combina uma API de autorização e controle de acesso e um gerenciador de confiança (*TrustBuilder*), com o intuito de regular quando, e como, informações sensíveis podem ser reveladas. Esta proposta se caracteriza por uma análise reativa face à ocorrência de falhas, i.e, uma falha (ex. credenciais erradas) implica no aumento do nível de suspeição sobre o usuário, que por sua vez implica em restringir privilégios de acesso.

Os dois trabalhos anteriores fazem uso de credenciais para inferir quanto à segurança: a diferença básica é que o primeiro é pró-ativo — aumenta a confiança no sucesso, e o segundo é reativo — aumenta a suspeição (diminui a confiança) no insucesso. A abordagem proposta neste trabalho não descarta o uso de credenciais, mas sugere o uso em conjunto de técnicas de análise comportamental para a evolução da confiança.

[Platzer 2004] propõe um mecanismo para avaliação da confiança em serviços Web, baseado em análise comportamental através da rastreabilidade contínua de um usuário. O objetivo é prover um mecanismo auto-gerenciável para controle de acesso em ambiente composto por federações de serviços Web. O nível de confiança é modelado exponencialmente em função dos serviços requisitados pelo usuário: incrementos para certas classes de serviços esperados e decrementos para outras. Em [Véras e Ruggiero 2005] é empregado um modo de avaliação comportamental análogo, porém com o intuito de re-autenticar usuários.

Uma das diferenças principais dessas abordagens em relação ao trabalho proposto é a forma com que se avalia o comportamento do usuário. Basicamente as propostas anteriores partem de um mapeamento inicial dos serviços/funções disponíveis no provedor em caminhos pré-determinados que, se seguidos, permitem aumentar a confiança depositada no usuário. A nossa proposta avalia o comportamento do usuário em função do seu passado (*logs* de utilização) com a finalidade de aumentar a confiança sobre a identidade do usuário (se ele é mesmo quem diz ser), fazendo uso de uma técnica de análise comportamental por aprendizado.

Sistemas de segurança baseados em análise comportamental por aprendizado podem ser classificados em duas categorias quanto ao tipo de "comportamento" estudado: (i) *Comportamento físico* – busca-se aprender alguma característica pessoal do usuário, ex: forma de teclar ou usar o mouse; e (ii) *comportamento contextual* - procura-se aprender o perfil de utilização de serviços do usuário, ex: comandos Unix, navegação Web, etc.

A primeira categoria está fortemente relacionada a mecanismos complementares de autenticação, enquanto a segunda é disseminada no domínio de Detecção de Intrusão.

Uma abordagem bastante explorada na literatura, na categoria "comportamento físico", consiste em gerar uma assinatura a partir da dinâmica individual de utilização do teclado [Monrose e Rubin 1997], [Guven e Sogukpinar 2003], [Peacock et al. 2004]. Basicamente este método não usa a informação do que está sendo teclada, mas sim o ritmo em que esta informação é teclada — distância temporal entre duas tecladas e duração de uma teclada. Os dados coletados são modelados em vetores de tamanho fixo, e para cada nova autenticação, cria-se um novo vetor que é comparado ao vetor inicial, gerando um índice de similaridade. Este método tem como principal desvantagem a atual tendência de obsolescência de interfaces textuais em prol de interfaces via mouse.

[Pusara, e Brodley 2004] propõem um mecanismo de re-autenticação via movimentos do mouse. Este mecanismo captura informações do mouse, (posição instantânea, clique, duplo clique, etc) e, após criar um modelo de comportamento normal, usa um classificador por árvore de decisão para validar o atual comportamento e re-autenticar o usuário.

Um exemplo da abordagem de análise comportamental contextual é o trabalho de [Lane e Brodley 1999] para detecção de anomalia, englobando tanto a detecção de intrusão, como a identificação de comportamentos hostis de usuários autenticados. O foco deste trabalho é a análise de linhas de comando em função do histórico de utilização do usuário, através de um mecanismo de aprendizado por instâncias — *Instance-based Learning* (IBL) [Aha et al. 1991]. Diversamente à proposta deste artigo, não existe uma preocupação em caracterizar o comportamento individual de cada usuário, mas sim, em classificá-lo como "normal".

Um outro tipo de abordagem de análise comportamental de usuários, diretamente relacionados com aplicações Web, é direcionado a personalização da navegabilidade de um *site*. [El-Ramly e Stroulia 2004] propõem o uso de técnicas de mineração de dados sobre *Web logs*, de forma a inferir sobre diferentes perfis de acesso dos usuários e, automaticamente, adaptar opções de navegação de um *site*.

Concluindo, o caráter inovador da abordagem apresentada neste trabalho está basicamente apoiado em dois fatores principais:

1. O uso de um mecanismo de avaliação comportamental por aprendizado como provedor de subsídios para evolução da confiança;
2. Proposta de um mecanismo de avaliação comportamental baseada em análise de trilha de navegação Web, superposta a uma assinatura contextual histórica.

## 3. Avaliação da Confiança

O conceito de confiança usado neste trabalho pode ser informalmente definido como uma medida de quão certo o provedor de uma aplicação está a respeito da identidade de um usuário e, conseqüentemente, da forma com que o usuário irá se comportar.

Em [Skogsrud et al. 2004] a evolução da confiança é controlada por uma máquina de estados finita (Modelo *Trust-Serv*) previamente especificada, onde os estados representam o nível atual de confiança de uma relação. Cada estado está associado a uma política de acesso específica (ex. papel em um modelo RBAC), sendo que a

transição entre estados é controlada por troca de credenciais, previsões/obrigações (serviços que devem ser executados antes) e *timeouts*. A figura 1 descreve um pequeno extrato de um exemplo de máquina de estados usada na negociação da confiança em *Trust-Serv*. Esta máquina possui dois estados, Revisor e Comprador, cada um requerendo um determinado nível de confiança e controlado por uma política de acesso específica (A & B). A transição entre os dois estados é salvaguardada pela troca das credenciais endereço e cartão de crédito.

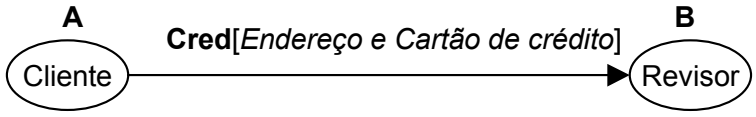


**Figura 1: Modelo de negociação da confiança**

Na proposta descrita neste trabalho, o conceito de nível de confiança/política de acesso é capturado pela definição de um macro-estado, sendo que os estados representam as páginas de uma aplicação Web. A mudança de um macro-estado para outro (implicando em mudança de política de acesso) é feita automaticamente pelo resultado de uma avaliação comportamental, ou, em caso de insucesso, por uma negociação explícita via troca de credenciais (figura 2).

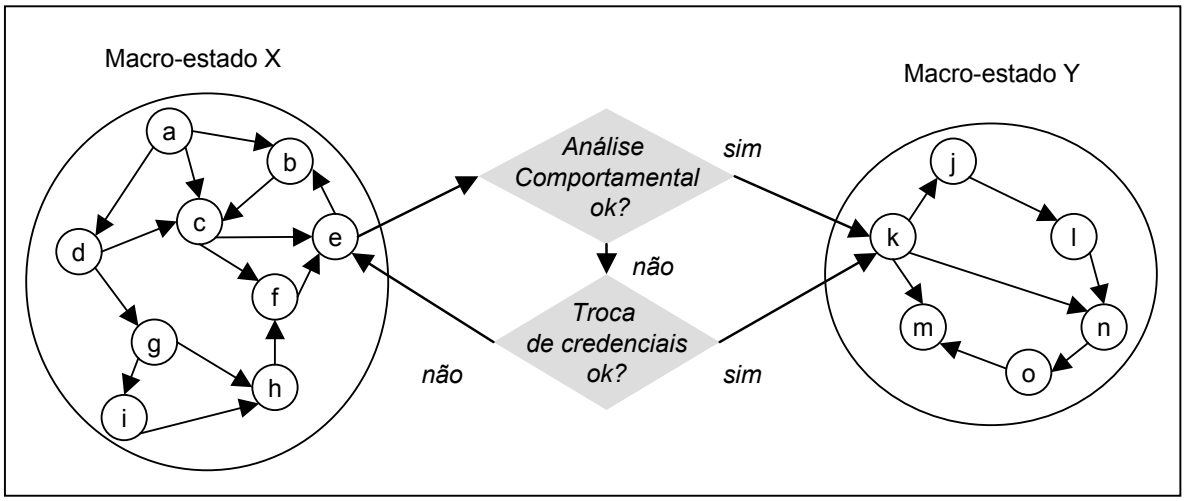


**Figura 2: Macro-estados x Avaliação da Confiança**

**Definição 3.1.** Um cenário de avaliação de confiança C é definido pela 6-upla:

$(MacroEstados^C, Estados^C, Transições^C, Perfis^C, \varphi^C, \omega^C)$

- $MacroEstados^C$ é o conjunto de macro-estados de C, onde cada *macro-estado* M é um subconjunto de $Estados^C$
- $Estados^C$ é o conjunto de estados de C (páginas)
- $Transições^C$ é conjunto de *transições* de C
- *Perfis* é conjunto de perfis de acesso (papéis) associados a C

- $\varphi^C$ é a *função de atribuição de transições*, associando cada transição a um estado origem e a um estado destino:

$$\varphi^C: Transições \rightarrow Estados^C \; x \; Estados^C$$

- $\omega^C$ é a *função de atribuição de perfis de acesso*, associando cada perfil a um conjunto de macro-estados:

$$\omega^C: Perfis^C \rightarrow Conjunto\ das\ partes\ de\ Macro\text{-}Estados^C$$

**Definição 3.2.** As transições específicas entre macro-estados podem ser capturadas pelo conjunto denominado:

$$MacroTransições^C = \{t \in Transições^C \mid \varphi(t) = (a,b) \wedge (a \in m_1,\ b \in m_2 :\ m_1 \neq m_2)\}$$

**Definição 3.3.** Seja $V^C$ o subconjunto de $MacroTransições^C$ experimentado durante a seção de usuário em um cenário C, sejam *ConfiançaComportamental* e *Credencial* funções com domínios $V^C$ e o contradomínios booleanos, a seção do usuário é dita conforme a estratégia de avaliação proposta sse a seguinte condição é satisfeita:

$$\forall t \in V^C : ConfiançaComportamental(t) \vee Credencial(t) = verdadeiro$$

É importante ressaltar que existe uma relação de dependência entre o resultado da Avaliação de confiança comportamental e a execução da avaliação por credenciais; a segunda só é disparada no caso de uma avaliação negativa da primeira conforme a figura 2.

Este tipo de abordagem, praticamente estabelecendo a troca de credenciais como uma redundância, tem impactos significativos nos requisitos impostos aos mecanismos de análise comportamental. A preocupação com falsos negativos praticamente desaparece, pois a não identificação de um comportamento de um usuário não causa nenhuma degradação da sessão em curso, apenas implicando na realização da uma etapa "redundante" de troca de credenciais.

Por outro lado, o uso isolado da avaliação comportamental seria capaz de identificar as seguintes condutas anômalas: (i) um usuário autenticado do sistema que faz um uso legítimo para o abuso de recursos de sistema, (ii) uso esporádico por um colega de trabalho "que pede emprestada" uma estação de trabalho, (iii) ataque automatizado lançado por um usuário relativamente ingênuo através de uma seqüência típica de ataque. Com o uso combinado de um mecanismo de troca de credenciais, a condição (i) naturalmente perde sua eficácia, pois é muito provável que um usuário autenticado também tenha sucesso numa troca de credenciais.

## 4. Avaliação comportamental

Nesta seção, é analisado o problema da avaliação comportamental por aprendizagem, de forma a caracterizar, e diferenciar, o comportamento de um indivíduo/sistema em termos de seqüências temporal de dados discretos. Embora o foco dado seja em trilhas de navegação para aplicações Web, os métodos apresentados são suficientemente genéricos para cobrir outras aplicações. A caracterização de um comportamento típico de um usuário é um grande desafio, pois, certamente, além da variabilidade inerente, existe uma alteração do padrão de uso normal como conseqüência natural da absorção de novos conhecimentos pelo usuário.

O uso de uma máquina de aprendizado permite treinar um classificador com os dados históricos de usuários, para que seja possível distinguir os diferentes comportamentos, levando em conta tanto a variabilidade, como o caráter evolutivo. Nesta seção são examinados métodos para coletar uma assinatura comportamental do usuário baseados em aprendizagem, e uma respectiva definição de similaridade apropriada ao contexto em questão.

### 4.1 Coleta de Assinatura comportamental

O problema de coleta de assinatura pode ser formulado como uma tarefa de aprendizado para caracterizar o comportamento típico de um usuário (assinatura), em termos de uma seqüência de dados discretos. Para tratar a avaliação comportamental como uma tarefa de aprendizado, é necessário definir tanto o modelo da aprendizagem, como o formato representacional dos dados de entrada.

Muitas das abordagens tradicionais de aprendizado não são aplicáveis ao domínio da diferenciação comportamental devido à especificidade dos tipos de dados disponíveis: elementos discretos com valores nominais. As redes neurais [Lane 2000] provaram ser úteis para séries contínuas de valores numéricos, empregando tipicamente a *Distância Euclideana* para cálculo de similaridade. Uma outra limitação, associada ao uso das redes neurais em diferenciação comportamental, é a necessidade do re-treinamento destas para cada novo usuário da aplicação [Monrose e Rubin 1997].

Uma classe popular e bastante genérica de técnicas de máquina de aprendizagem é a baseada em instâncias – *Instance-based Learning* (IBL). Neste modelo, um conceito é representado implicitamente por um conjunto de instâncias que exemplificam este conceito (dicionário do exemplo). Uma instância previamente desconhecida é classificada de acordo com sua relação às instâncias armazenadas. Um esquema típico é a classificação do k-ésimo vizinho, onde é dado para uma nova instância, a etiqueta da maioria das instâncias do dicionário de k mais próximas a ela, onde "o mais perto" é uma medida específica do domínio em questão. Em domínios contínuos, para o exemplo, a medida da similaridade é feita freqüentemente pela *Distância Euclideana*.

No nosso caso, pode-se aplicar diretamente um método bastante simplificado do modelo de aprendizagem IBL, sendo que cada instância comportamental (figura 3) é diretamente classificada de acordo com o usuário gerador. Assim, a assinatura comportamental é representada pelo conjunto de instâncias comportamentais de um usuário específico, sendo gerada para cada macro-transição.

Um possível problema desta abordagem está relacionado à necessidade de armazenamento de um jogo completo de instâncias comportamentais por assinatura. Em um ambiente dinâmico, tal como a diferenciação comportamental, o tamanho do dicionário de instâncias pode crescer consideravelmente, requerendo técnicas de redução de dados, que não serão abordadas neste artigo.

**Definição 4.1.** Seja I um conjunto de índices; os conceitos de instância comportamental *ic* e assinatura comportamental *ac* podem ser definidos como:

$$ic = \{e_i \in Estados^C \mid i \in I\}$$

$$ac = \{ic_i \mid i \in I\}$$

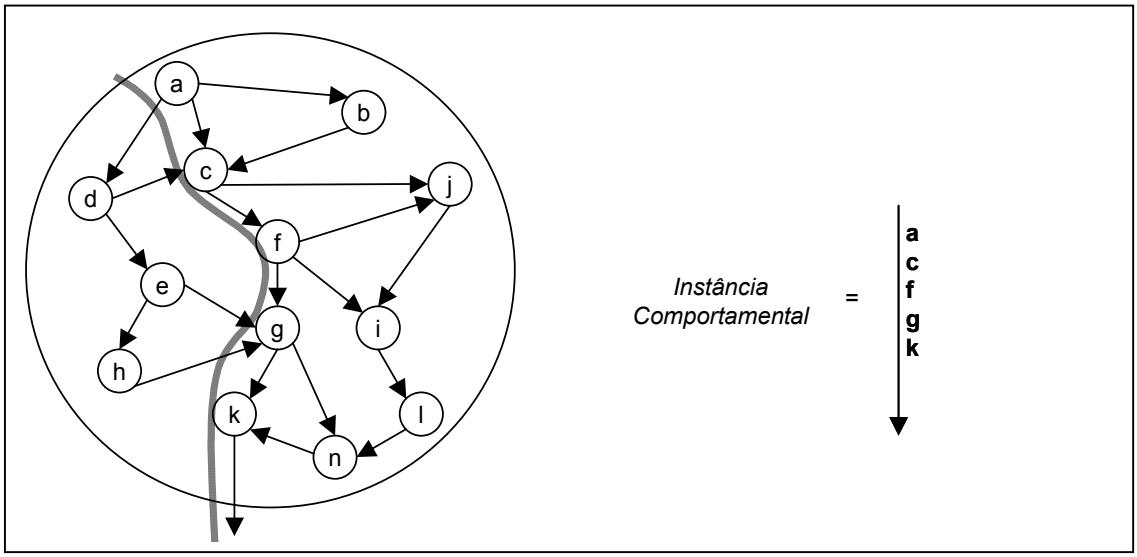


**Figura 3: Exemplo de instância comportamental**

## 4.2 Índice de similaridade

O Índice de similaridade *S* é definido como uma função de duas instâncias comportamentais que calcula uma medida de quão parecidos são estes comportamentos. Para o cálculo de *S*, adotou-se como base o procedimento proposto por [Lane e Brodley 1999] para o cálculo de similaridade entre duas seqüências de comandos de tamanhos iguais. Basicamente este procedimento pontua pares de elementos idênticos e usa um cálculo acumulativo para dar um peso maior aos pares idênticos encadeados (subseqüências iguais). O procedimento adotado basicamente realiza duas alterações sobre esta proposta: (i) a normalização do resultado deste cálculo (equação 3) e (ii) adaptação para entradas com seqüências de tamanhos diferentes (ex:{a,b,c,d} & {a,g,d}). Para o caso (ii), definiu-se um procedimento de homogeneização entre um par de instâncias comportamentais (que "estica" a menor até atingir o comprimento da maior) a ser aplicado antes do cálculo de similaridade, conforme a definição 4.2, ilustrado pela figura 4.

**Definição 4.2.** Sejam duas instâncias comportamentais *c* e *d,* o processo de homogeneização *H* é capturado pela função:

$$H(c,d) = (\{a_0, a_1 ... a_{m-1}\}, \{b_0, b_0 ... b_{m-1}\}) = \begin{cases} (c, \tau(d,c)) & se \quad card(c) > card(d) \\ (\tau(c,d), d) & se \quad card(c) < card(d) \\ (c,d) & se \quad card(c) = card(d) \end{cases} \qquad (1)$$

Onde o comportamento da função $\tau(x,y)$ é expresso por:

$\tau(x, y)$:
Seja $z$ um vetor do mesmo tamanho de $x$, preenchido com valores de estados inválidos
Seja $p_i$ uma posição qualquer de um vetor $i$
Faça $p_z$ igual à primeira posição de $z$
1) Para cada estado $e_y$ de $y$, faça
    Para cada $p_x$ em {$p_z$ até última posição de $x$}, faça
    Se $e_y$ for igual ao estado $e_x$ na posição $p_x$, então
    Insere $e_y$ em $z$ na posição $p_z$
    Faça $p_z$ igual a $p_x$
    Volte para 1
Faça $y$ igual a $z$

**Listagem 1: Pseudocódigo da função τ(x,y)**

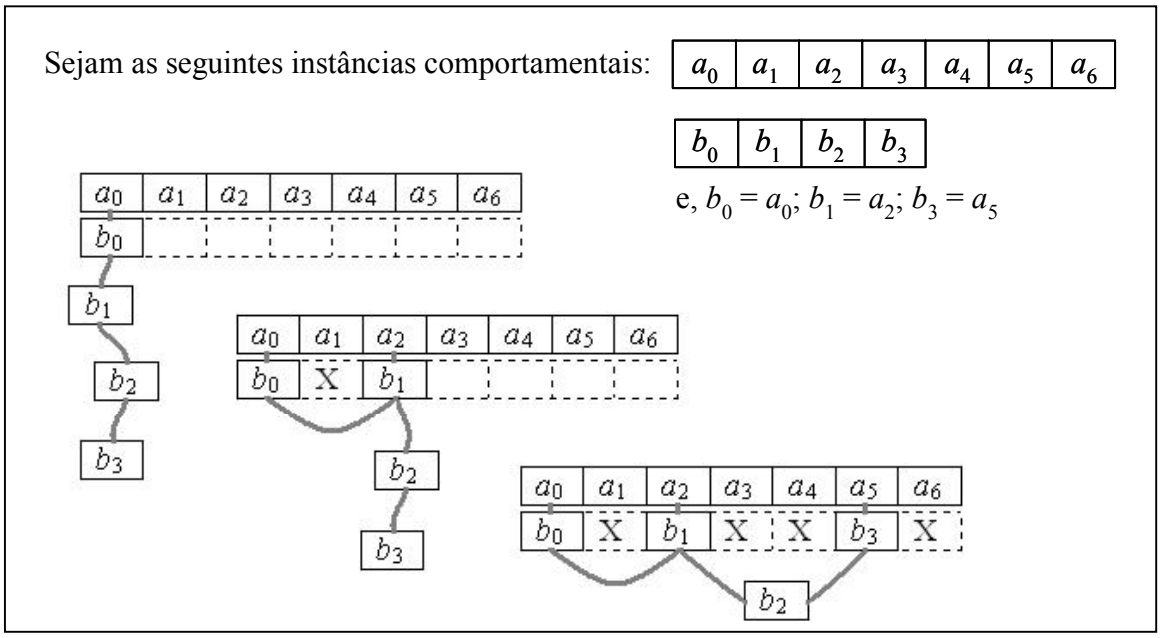


**Figura 4: Idéia gráfica do algoritmo de homogeneização**

**Definição 4.3.** O índice de similaridade $S$ entre $a = (a_0, a_1, \ldots a_{m-1})$ e b = $(b_0, b_1, \ldots b_{m-1})$ é dado pelo seguinte trio de funções:

$$\Omega(a_i, b_i) = \begin{cases} 0 & se \quad a_i \neq b_i \vee i < 0 \\ 1 + \Omega(a_{i-1}, b_{i-1}) & se \quad a_i = b_i \end{cases} \tag{2}$$

$$Som(a,b) = \sum_{i=0}^{m-1} \Omega(a_i, b_i) \tag{3}$$

$$S(a,b) = \frac{Som(a,b)}{Som(a,a)} \tag{4}$$

### 4.3 Cálculo da Confiança Comportamental

Após o estabelecimento de uma assinatura comportamental a um usuário (associada a uma macro-transição), o próximo passo consiste em estabelecer uma heurística para atribuir uma medida de confiança a uma dada instância comportamental em função de sua assinatura. Fundamentalmente, existem três fatores interdependentes que devem ser considerados: *Similaridade comparativa*, *Intra-similaridade* e *Inter-similaridade*.

*Similaridade comparativa* (*Scomp*) - representa a similitude entre a instância coletada atual e o conjunto de instâncias que compõe a assinatura. Essencialmente, o seu valor espelha o quanto este comportamento se aproxima dos demais previamente capturados. Para este cálculo, aplica-se a função *similaridade* entre a instância comportamental atual e cada uma das instâncias que compõem a assinatura, retendo o valor máximo obtido:

$$Scomp^{M} = \max\{S(ic_{atual}^{M}, ic_{i}^{M}), \forall ic_{i}^{M} \in ac^{M}\} \quad (5)$$

*onde $ic_{atual}$ denota a instância computacional atual, e $ac^{M}$ denota a assinatura comportamental da macro-transição M.*

*Intra-similaridade (Sintra)* - está relacionada à qualidade da assinatura do usuário, sendo completamente independente da amostra atual de instância comportamental. Este fator representa se um usuário possui um comportamento bem formado (quando as instâncias pertencentes à assinatura são repetidas, ou levemente diferentes), ou o contrário (quando todas as instâncias da assinatura são bem diferentes entre si). Naturalmente, uma assinatura mal-formada dificulta o processo de validação do comportamento do usuário, implicando em uma confiança menor. Para o cálculo de *Sintra*, realiza-se a média entre todos os valores resultantes da aplicação da função de similaridade entre todos os arranjos 2 a 2 de instâncias de uma assinatura:

$$Sintra^{M} = \frac{\sum_{\forall ic_{n}^{M} \in ac^{M}} \sum_{\forall ic_{m}^{M} \in ac^{M}} S(ic_{m}^{M}, ic_{n}^{M})}{A_{card(ac^{M}),2}} \quad (6)$$

*Inter-similaridade (Sinter)* - traduz a qualidade assinatura de um usuário em função do conjunto completo de assinaturas (de diferentes usuários) associado a uma mesma macro-transição. Uma dada instância comportamental pode ser altamente similar a uma assinatura bem-formada, mas mesmo assim, não expirar confiança devido a uma possível similaridade dela com outras assinaturas existentes (dos diversos usuários do mesmo cenário). A similaridade das assinaturas dificulta o processo de diferenciação dos usuários. *Sinter* reflete a similaridade entre uma dada assinatura e a assinatura mais "parecida" do conjunto total de assinaturas, e é expressa pelo seguinte par de funções:

$$\Phi(ac^{M}, ac_{i}^{M}) = \frac{\sum_{\forall ic_{n}^{M} \in ac^{M}} \max\{S(ic_{n}^{M}, ic_{m}^{M}), \forall ic_{m}^{M} \in ac_{i}^{M}\}}{card(ac^{M})} \quad (7)$$

$$Sinter(ac^{M}) = 1 - \max\{\Phi(ac^{M}, ac_{i}^{M}), \forall ac_{i}^{M} \in U^{M} - \{ac^{M}\}\} \quad (8)$$

*onde $U^M$ denota o conjunto total de assinaturas associadas a M de um cenário C.*

Concluindo, o cálculo da confiança (Trust) é expresso pelo produto destes três fatores:

$$Trust^M = Scomp^M * Sintra^M * Sinter^M \quad (9)$$

Dado que a medida de confiança pode ser quantificada, basta agora estabelecer um patamar de aceitação mínimo *TrustRef* para avaliar a função *ConfiançaComportamental* (definição 3.3):

$$ConfiançaComportamental(M) = \begin{cases} verdaeiro & se \quad Trust^M \geq TrustRef \\ Falso & se \quad Trust^M < TrustRef \end{cases} \quad (10)$$

## 5. Avaliação da análise comportamental

O desempenho de um mecanismo deste porte é fortemente influenciado pelo tipo de aplicação e pela variedade de comportamentos intrínsecos. A maioria das propostas similares faz uso de análises empíricas, realizadas em cima de exemplos concretos de aplicações como ambientes de teste, por exemplo: [Pusara e Brodley 2004], [Lane 2000]. O grande problema deste tipo de abordagem é o perigo de selecionar um ambiente extremamente impróprio, gerando uma possível falsa avaliação negativa do mecanismo; ou, ao contrário, extremamente apropriado, o que também levaria a uma conclusão errônea de uma questionável eficácia, certamente desaconselhável a ser generalizada. Portanto, decidiu-se pelo desenvolvimento de um método de avaliação por simulações, que permita um controle efetivo dos principais parâmetros envolvidos nos cálculos realizados durante o processo de análise comportamental. Basicamente, a variável em exposição é o resultado final do *Trust*, já que um conhecimento maior da sua variabilidade permite colher subsídios para a posterior definição do *TrustRef.*

O primeiro passo é controlar o espaço usado para uma geração pseudo-aleatória de assinaturas dos usuários com o intuito de observar o comportamento do mecanismo face aos diferentes perfis de assinaturas, ou seja, até quanto as assinaturas de dois usuários devem ser diferentes para que de fato o mecanismo de análise comportamental seja conclusivo. Para isso, considera-se um macro-estado composto de n estados totalmente interconectados e selecionam-se dois diferentes subconjuntos de estados (*A* e *B*) com 5 elementos, representando dois usuários diferentes, que vão servir como "sementes" de geração das respectivas assinaturas. Inicialmente escolhem-se dois conjuntos sem nenhuma interseção, que vão gradativamente se intersecionando de 1 elemento, gerando 6 diferentes cenários $(AB^0, AB^1, AB^2, \ldots, AB^5)$. Para cada um destes cenários, é realizado o conjunto de procedimentos definidos nos próximos itens. A figura 5 ilustra a composição de dois diferentes cenários: sem interseção (i) e com interseção de 2 elementos (ii);

*(i)* $AB^0 = (\{e^{23}, e^{32}, e^{33}, e^{34}, e^{43}\}, \{e^{35}, e^{44}, e^{45}, e^{46}, e^{55}\})$

*(ii)* $AB^2 = (\{e^{23}, e^{32}, e^{33}, \underline{\boldsymbol{e^{34}}}, \underline{\boldsymbol{e^{43}}}\}, \{\underline{\boldsymbol{e^{34}}}, \underline{\boldsymbol{e^{43}}}, e^{44}, e^{45}, e^{54}\})$

O segundo passo compreende o processo de geração das assinaturas dos usuários de *A* e *B*. O objetivo é conseguir uma varredura representativa de valores de intra-similaridade *Sintra* e para isso são realizados os seguintes procedimentos:

- geram-se exaustivamente todas as possíveis assinaturas no subconjunto *A* e no subconjunto *B* com 5 estados (esta limitação em 5 estados foi decorrência das dificuldade de simulação, uma vez que a rede é totalmente interconectada);
- calcula-se *Sintra* (equação 6) para cada assinatura;
- escolhem-se 5 assinaturas (para A e para B) que possuam valores de *Sintra* com espalhamento bem representativo (ex: 0,1 , 0,2 , 0,4 , 0,6...);
- para cada par de assinaturas escolhidas de A e B com o mesmo valor aproximado de *Sintra*, calcula-se *Sinter* (equação 8).

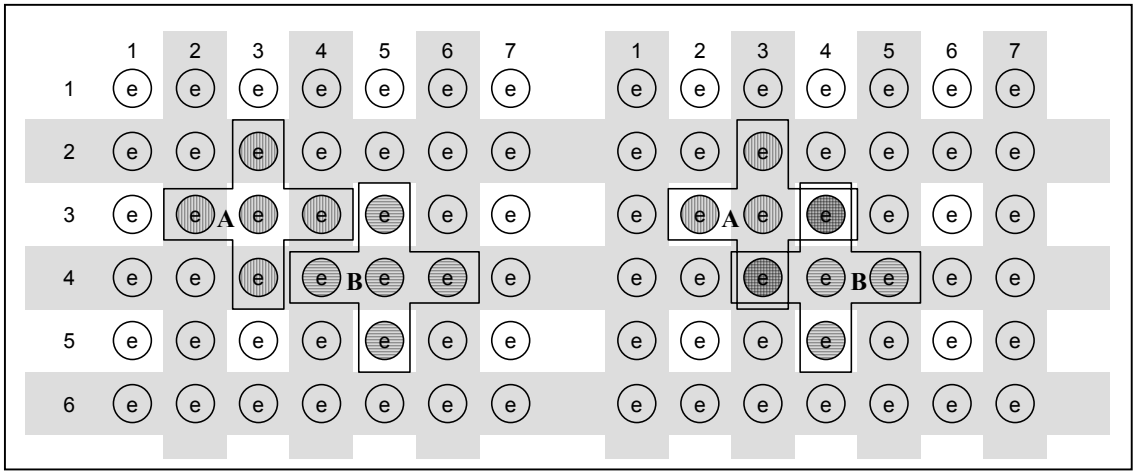


**Figura 5: (i) A∩B = 0 (ii) A∩B = 4**

O terceiro, e último passo, consiste em gerar um conjunto de instâncias comportamentais controladas, de forma a avaliar a efetividade do mecanismo em função das assinaturas geradas via os procedimentos do item anterior. O objetivo é gerar um conjunto de amostras (instâncias) representativas de possíveis comportamentos reais, mas que ao mesmo tempo sejam suficientemente bem caracterizadas para possibilitar avaliações conclusivas. Deste modo, utiliza-se o mesmo artifício de variar o subconjunto de estados usados para a geração da assinatura de um usuário (*A*), incluindo gradativamente elementos do subconjunto usado para o outro usuário (*B*) para compor os conjuntos geradores das instâncias comportamentais. A tabela seguinte descreve os conjuntos geradores usados para A.

| Com 5 estados de A |
|---|
| Com 4 estados de A e 1 de B |
| Com 3 estados de A e 2 de B |
| Com 2 estados de A e 3 de B |
| Com 1 estado de A e 4 de B |
| Com 5 estados de B |

**Tabela 1: Conjuntos geradores das instâncias comportamentais de A**

Sendo que, para cada conjunto gerador, gera-se 5 instâncias comportamentais aleatórias e calcula-se *Scomp* (equação 5) para cada uma das 10 assinaturas do usuário correspondente. Para cada assinatura de usuário, calcula-se média de *Scomp* e aplica-se este valor na expressão do cálculo do *Trust* (equação 9). Finalmente os valores gerados são usados para traçar (por usuário e por assinatura) os gráficos *Trust* versus *conjuntos geradores das Instâncias comportamentais* [5/0, 4/1, 3/2, 2/3, 1/4, 0/5]. Foram escolhidos dois grupos de gráficos obtidos para os conjuntos geradores $AB^0$ (nenhuma interseção), figura 6, e $AB^4$ (interseção de 4 elementos entre A e B), figura 7. Cada gráfico corresponde a um valor diferente de *Sintra* (assinatura diferente). Os valores de *Sintra* utilizados foram aproximadamente (0,1; 0,2; 0,3; 0,4; 0,5; 0,6; 0,7; 0,8; 0,9). Os gráficos estão associados a valores decrescentes de *Sintra*, i.e, os gráfico com valores maiores de *Trust*, em ambas as figuras, representam um valor de *Sintra* de 0,9. Isso significa uma confiança maior para assinaturas mais diferenciadas.

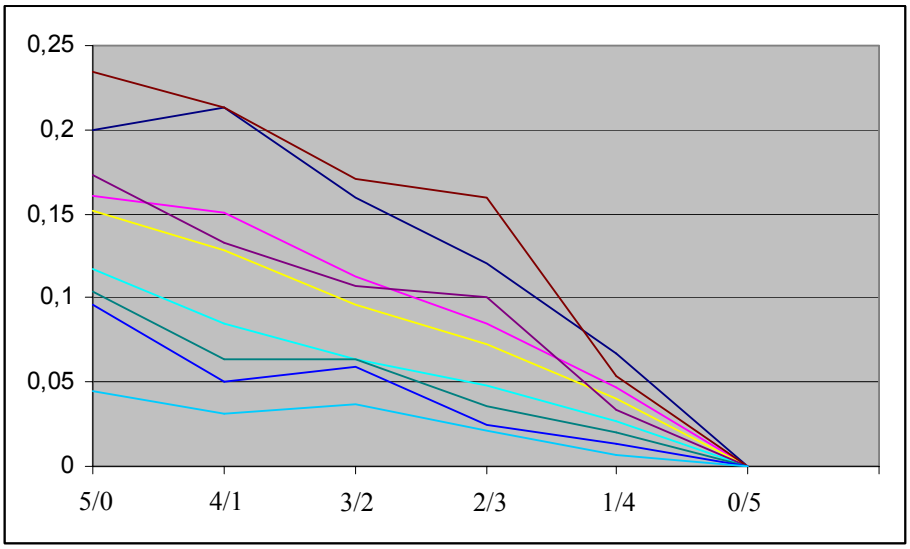


**Figura 6: Trust x Conjuntos Geradores para o cenário $AB^0$**

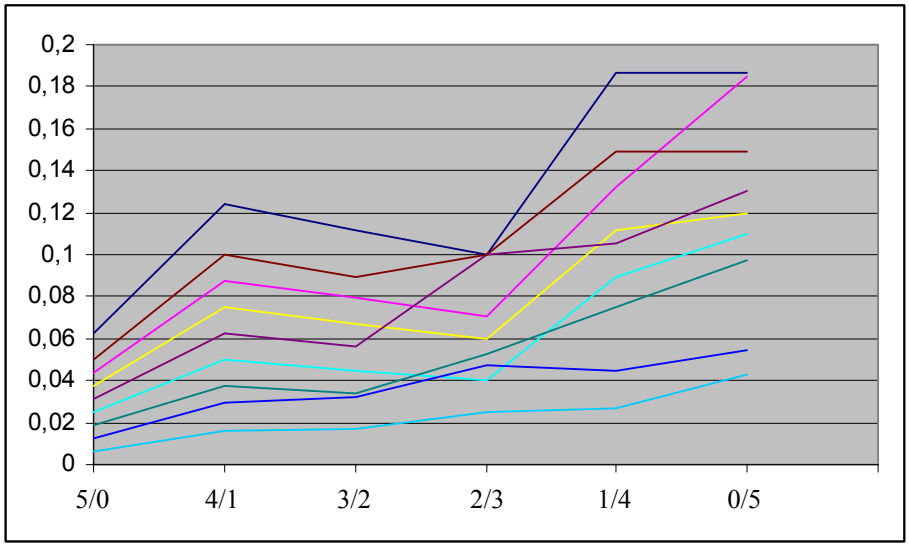


**Figura 7: Trust x Conjuntos Geradores para o cenário $AB^1$**

Visualmente pode-se verificar que os gráficos são relativamente bem comportados e contemplam os requisitos necessários para diferenciação comportamental. No primeiro conjunto de gráficos (figura 6), para nenhuma interseção entre A e B, encontram-se valores de *Trust* mais elevados para instâncias comportamentais geradas apenas com elementos de A, que vão decaindo à medida que são usados elementos de B. No segundo conjunto de gráficos (figura 7) a situação se inverte, pois as assinaturas geradas possuem maior interseção com *B*, fazendo com que os maiores valores de *Trust* sejam encontrados na região onde as instâncias comportamentais são geradas com mais elementos de *B*.

## 6. Conclusões e Trabalhos Futuros

Este artigo descreveu uma proposta de emprego integrado do conceito de confiança e gerência de controle de acesso em aplicações seguras na Web. O ineditismo da abordagem apresentada neste trabalho reside no uso de um mecanismo de avaliação comportamental do usuário (pela trilha de navegação Web) através de uma máquina de aprendizado. O resultado desta análise é usado no processo de evolução da confiança para substituir, ou complementar, o uso habitual de mecanismos de troca de credenciais.

Outra importante contribuição deste trabalho é o procedimento proposto para o cálculo de similaridade entre duas amostras representativas do comportamento do usuário que, diferentemente do usual, emprega um processo de homogeneização entre seqüências comportamentais de diferentes comprimentos.

Destaca-se ainda a abrangência da heurística empregada no cálculo final do nível de confiança de uma instância comportamental, que leva em contra três diferentes fatores: (i) similaridade comparativa – relação entre a instância comportamental atual e a assinatura (similitude do comportamento), (ii) intra-similaridade – relação entre as instâncias comportamentais que formam a assinatura (qualidade da assinatura) e (ii) inter-similaridade – relação entre as diversas assinaturas existentes (diferenciabilidade das assinaturas).

Para a avaliação do processo de análise comportamental fez-se uso de uma metodologia credenciadora de um controle individualizado dos principais parâmetros envolvidos nos cálculos da *Confiança*. Conseqüentemente, o desempenho do mecanismo proposto é bem caracterizado por simulações que, além de atestarem a viabilidade da sua utilização no contexto de diferenciação comportamental, permitem colher subsídios para o estabelecimento de um patamar de confiança mínimo *TrustRef*.

Uma questão em aberto da abordagem proposta, e alvo de trabalhos em andamento, é quanto a necessidade do emprego de técnicas de redução de dados, uma vez que as assinaturas armazenam todas as instâncias comportamentais passadas de um usuário. Para determinadas aplicações, que permitam uma grande variabilidade de comportamentos, o tamanho da assinatura pode crescer consideravelmente. Estuda-se a viabilidade de substituição de um grupo de instâncias comportamentais similares de uma assinatura por modelos genéricos que capturem um certo grau de variabilidade. Um outro trabalho em andamento tenta caracterizar uma evolução no tempo do comportamento de um certo usuário, permitindo o expurgo de comportamentos antigos de sua assinatura que não deverão mais se repetir.

## 7. Referências


Aha, D. W., Kibler, D., AlbertT, M. K. (1991). "Instance-based learning algorithms", Machine Learning, Vol. 6, No 1, January, Kluwer Academic Publishers, pp. 37–66.

Chivers H., Clark J.A. (2004), Smart dust, friend or foe?—Replacing identity with congiguration trust", Computer Networks 46, Elsevier, pp.723–740.

Bacon J., Moody K., and Yao, W. (2003). "Access Control and Trust in The Use of Widely Distributed Services", Software-Practice Experience, 33, pp.375–394.

El-Ramly, M., Stroulia, S. (2004). "Analysis of Web-usage behavior for focused Web sites: a case study", Journal of Software Maintenance and Evolution: Research and Practice, 16, pp.129–150.

Guven, A., Sogukpinar, I. (2003). "Understanding Users' Keystroke Patterns for Computer Access Security", Computers & Security, Vol. 22, No 8, Elsevier, pp 695-706.

Lane, T., Brodley, C. (1999). "Temporal Sequence Learning and Data Reduction for Anomaly Detection", ACM Transactions on Information and System Security, Vol. 2, No. 3, August, Pages 295–331.

Lane, T. (2000), "Machine learning techniques for the computer security". Ph.D. thesis, Purdue University.

Langheinrich, M, (2003). "When Trust Does Not Compute – The Role of Trust in Ubiquitous Computing", Workshop on Privacy at UBICOMP 2003, Seattle, Washington.

Lopez, J., Oppliger, R., Pernul, G. (2004). "Authentication and authorization infrastructures (AAIs): a comparative survey", Computers & Security, 23 - 2004, Elsevier, pp. 578-590.

Monrose, F. e Rubin, A. (1997). "Authentication via Keystroke Dynamics", In Fourth ACM Conference on Computer and Communication Security - CCS 97, pp. 48-56, Zurich, Switzerland.

Peacock, A., Ke, X., Wilkerson, M. (2004). "Typing Patterns: A Key to User Identification", IEEE Security & Privacy, September/October 2004,1540-7993/04, pp. 40-47.

Platzer C. (2004) "Trust-based Security in Web Services", Master's Thesis, Information Systems Institute, Technical University of Vienna, Austria.

Pusara, M., Brodley, C.E. (2004). "User Re-Authentication via Mouse Movements", *In:* CCS Workshop on Visualization and Data Mining for Computer Security - VizSEC/DMSEC'04, October, ACM press, Washington, DC, USA.

Skogsrud, H., Benatallah, B., Casati, F. (2004). Security and privacy: "Trust-serv: model-driven lifecycle management of trust negotiation policies for web services**",** In: 13th international conference on World Wide Web, ACM Press.

Skogsrud, H., Benatallah, B., Casati, F. (2003). "Model-Driven Trust Negotiation for Web Services", IEEE Internet Computing, 1089-7801/03, November/December 2003, pp. 45-52.

Tatyana R., Zhou L, Neuman, C., Leithead, T., Seamons, K.E. (2005), "Adaptive trust negotiation and access control", In tenth ACM symposium on Access control models and technologies, ACM Press, Stockholm, Sweden.

Véras, L.M.A e Ruggiero, W.V. (2005). "Autenticação Contínua de Usuários em Aplicações Seguras na Web", In: V Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais, Florianópolis.