Autenticação Contínua de Usuários em Aplicações Seguras na Web

Alisson L. M. Véras, Wilson V. Ruggiero

LARC – Laboratório de Arquitetura e Redes de Computadores – Escola Politécnica da Universidade de São Paulo (POLI-USP)
Av. Prof. Luciano Gualberto, travessa 3, n.158, sala C1-46 – 05.508-900 – São Paulo – SP – Brasil

{alisson, wilson}@larc.usp.br

Abstract. *Secure web applications are high reliant of their user authentication. The access data, in general "username" and "password", can be easily stolen from inadvertent users, common practice nowadays. Using these authentication data, the attacker passes the initial authentication phase as a legitimate user, what turns the task to detect this intrusion in a non trivial mission. In this scenario, a continuous analysis of the application use is extremely important and a complement to the initial authentication. This work presents a method of continuous authentication based in a metric of confidence for secure* web *applications.*

Resumo. *Aplicações seguras na web são fortemente dependentes da autenticação do usuário. Os dados de acesso a essas aplicações, em geral "nome de usuário" e "senha", podem ser facilmente roubados de usuários inadvertidos, prática cada vez mais comum. Utilizando esses dados, o fraudador passa pela fase inicial de autenticação sendo identificado como um usuário legítimo, o que torna não trivial a tarefa de detectar essa intrusão. Dessa forma, uma análise contínua da utilização da aplicação para confirmar a identidade do usuário é fundamental e complementar à autenticação inicial. Este trabalho apresenta um método de autenticação contínua baseada em uma métrica de confiança para aplicações seguras na web.*

1. Introdução

A segurança de um sistema computacional é vital para a sua utilização. Aplicações seguras na web são implementadas com utilização de serviços fornecidos pelo protocolo HTTP (HyperText Transfer Protocol) [IETF RFC2616 1999], acessíveis a partir de um navegador web e com restrições de segurança bem definidas, como caracterizado pelos seguintes requisitos de segurança: Confidencialidade, Integridade, Disponibilidade e Legitimidade [Stallings 1996]. Lojas virtuais e aplicações bancárias on-line são exemplos mais comuns dessas aplicações.

Para acesso a essas aplicações, os usuários necessitam passar por um processo de autenticação criterioso. Os dados utilizados nesse processo, por sua vez, devem ser protegidos de forma que não possam ser conhecidos por usuários ilegítimos. Por outro lado, os desenvolvedores dessa categoria de aplicações estão cada vez mais preocupados com a segurança dos sistemas e atuando fortemente para reforçá-la,

principalmente dentro do ambiente da aplicação. Como conseqüência dessa atitude, nos últimos tempos, o foco de ataques voltou-se para o elo mais fraco, ou seja, o usuário final. Conforme discutido por [Schneier 2001], senhas são facilmente roubadas de usuários inadvertidos, bastando para isso, verificar a grande quantidade de fraudes ocorridas no sistema bancário nos últimos anos através da Internet com utilização de dados roubados [IFCC 2002].

Todo o processo de autenticação pode ser visualizado como sendo baseado em confiança, ou seja, uma vez autenticado, o usuário é considerado confiável para a aplicação. Em um cenário onde os dados de autenticação não são totalmente confiáveis, não se pode confiar cegamente no usuário, portanto, deve-se encontrar uma técnica complementar para comprovar a veracidade das informações de autenticação e a própria validade dessa autenticação continuamente.

Este trabalho apresenta uma proposta de autenticação contínua baseada em uma métrica de confiança para aplicações seguras na web. O seu desenvolvimento é evoluído a partir do conceito de confiança apresentado na sessão 2, em conjunto com o mapeamento de uma aplicação segura na web, conforme apresentado na sessão 3. Posteriormente é apresentada a proposta de autenticação contínua, que permite em função da monitoração do comportamento de um usuário e da sua integração com a aplicação, verificar o valor do seu indicador de confiança, e dessa forma, manter ou revogar a sua autenticação quando certos limites de confiança ou desconfiança forem ultrapassados.

2. Confiança

Confiança em sistemas computacionais tem sido alvo de investigações mais intensas com interesse em segurança como em [Ruggiero 2002], [Jones e Marsh 1997] e [Platzer 2004] ou em métricas e formalizações como em [Marsh 1994] e [Gambetta 2000].

Para discussão sobre o conceito de confiança, devem ser tomados por base os indivíduos e o contexto onde as interações entre eles ocorrem. A confiança evolui a partir de uma confiança inicial, que pode ser derivada de experiências anteriores e pode mudar conforme o resultado de novas experiências. Há ainda a confiança em um contexto específico, onde um indivíduo pode apresentar um nível de confiança diferente de outras situações. A evolução da confiança também depende da importância da interação em um dado contexto. E, finalmente, a confiança pode ser variada conforme o sucesso ou fracasso obtido por certo indivíduo em suas interações com o sistema.

Conforme apresentado por [Jones e Marsh 1997] e posteriormente por [Gambetta 2000], além da descrição de confiança e sua evolução, pode-se extrair algumas variáveis importantes a partir das quais esses conceitos podem ser quantificados. O resultado desse trabalho, juntamente com as especificidades de aplicações seguras web, é um índice de confiança de uma aplicação segura web em um usuário e que pode ser utilizado para produzir um processo contínuo de autenticação.

Alguns conceitos devem ser definidos para que as variáveis de confiança possam ser extraídas a partir da descrição anterior, e permitir a formulação de uma métrica que possa ser associada ao conceito de confiança. Esses conceitos são:

Indivíduos: Entidades identificadas em uma relação de confiança;

Contexto: Situação em que certas interações entre indivíduos ocorrem (para este caso, usuário e aplicação);

Confiança inicial: Toda confiança é derivada de uma confiança inicial. Confiança de um indivíduo em outro antes da sua primeira interação;

Conhecimento: Representação do conhecimento de outros indivíduos como indicador de confiança a ser utilizado pelos demais indivíduos em uma relação de confiança;

Confiança contextual: Quando indivíduos se encontram em determinado contexto, há uma confiança contextual, ou seja, para uma determinada situação um indivíduo pode ser mais confiável do que em outra situação;

Importância: A interação em um dado contexto pode possuir certo nível de importância através dos indivíduos envolvidos. Por exemplo, em uma aplicação de Internet Banking, o efeito de consulta ao saldo é menos crítico que o causado por uma transferência de valores. Dessa forma, níveis de importância diferenciados podem ser identificados para contextos diferentes em uma aplicação;

Utilidade: Indivíduos esperam algum ganho por uma cooperação. Esse ganho esperado representa a utilidade de uma cooperação para um indivíduo.

Aplicações seguras na web fornecem um ambiente propício para utilização da confiança como critério para autenticação contínua. Em aplicações com suporte a trabalho cooperativo, pode-se analisar as interações de usuários com usuários e interações entre os usuários com a própria aplicação, para se extrair dados relevantes para evolução da confiança. A confiança, por sua vez, pode ser a confiança de um indivíduo Y no indivíduo Z e a do indivíduo Z no indivíduo Y (confiança bidirecional) e pode ser simples, de Y em Z (confiança unidirecional). Um exemplo de confiança unidirecional é a confiança de uma aplicação segura na web em um usuário.

3. Aplicações seguras na web

Aplicações web são baseadas no protocolo HTTP. Este é um protocolo de requisição/resposta e não foi originalmente desenvolvido para servir como base para as aplicações complexas existentes atualmente. Ele não mantém estado, o que torna cada requisição independente das demais. Essa é uma das principais causas para problemas de ataques às aplicações web em nível de protocolo, como por exemplo, roubo de sessão ou alteração de dados de estado no lado do cliente.

As aplicações seguras na web se diferenciam de aplicações web simples pela grande importância e restrições de segurança. O conjunto de regras e procedimentos de segurança aplicáveis a todas as atividades relacionadas com a aplicação segura na web dentro de um domínio, define a política de segurança da aplicação. Ela formaliza as necessidades de segurança, e como mantê-la. Determina as expectativas para segurança fornecendo direção e suporte ao gerenciamento da mesma.

Para a utilização da confiança em aplicações web, é necessário entender e conhecer a aplicação, sua política de segurança, seus fluxos de interação e padrão de requisições para cada funcionalidade. Daí, advem a necessidade de se realizar um mapeamento da aplicação em uma máquina de estados finitos, utilizada para o acompanhamento criterioso dos usuários nas tentativas de operar as suas funcionalidades durante uma sessão web.

3.1. Navegação e mapeamento da aplicação

Com o acompanhamento da navegação do usuário na aplicação, informações podem ser obtidas a partir do protocolo HTTP. Essas informações são de grande importância para a evolução do indicador de confiança.

Tabela 1. Dados utilizados para acompanhamento do usuário

Variável	Significado
Endereço IP origem da requisição	Endereço de rede responsável pela requisição
User-Agent	Identificador do cliente que iniciou a requisição
Identificador de Sessão	Pode ser autenticação do usuário no servidor, cookies, uma variável, endereço de rede , etc, ou combinação dos mesmos
URI	Identificador do recurso requisitado
Método da requisição	Pode ser GET, POST. Os métodos PUT, DELETE, HEAD, TRACE e CONNECT não são comumente observados.
Parâmetros	Dados recebidos por certo recurso. São utilizados para computação de dados pelo recurso requisitado e formulação da resposta.
Método de passagem de parâmetros	Parâmetros podem ser passados através de método GET (diretamente na URL) ou POST (no cabeçalho da requisição)
HTTP_REFERER	Informação pelo cliente do endereço do recurso a partir do qual o URI da requisição atual foi obtido
Tempo entre requisições	O tempo entre requisições em uma sessão pode ser utilizado para identificar uma possível utilização de robôs para acessar a aplicação

Durante a utilização de uma aplicação web, um usuário realiza inúmeras requisições ao servidor da aplicação. Um sistema de acompanhamento deve guardar os dados importantes das sessões da aplicação de forma a confrontá-los com o mapeamento realizado da aplicação e com a próxima transição de estados esperada.

Para cada requisição intencional do usuário, há outras requisições secundárias para busca de informações complementares, como aplicativos, estilos de formatação, imagens, etc. O mapeamento de uma aplicação segura na web deve utilizar as requisições realizadas para os recursos principais. Através do mapeamento, a aplicação pode ser visualizada em dois níveis principais de abstração:

Organizacional: São consideradas as unidades macro visualizadas pelo navegador. Faz-se o levantamento dos recursos requisitados na aplicação, bem como parâmetros utilizados para entrada e computação de dados;

Semântico: Neste nível é realizada a correlação entre as unidades identificadas no nível anterior, com o contexto de negócio ou funcionalidade da aplicação em que estão

inseridos. Serve para diferenciar o uso de uma mesma unidade organizacional que possa estar presente em dois processos de negócio distintos.

Cada nível de abstração fornece informações diferentes para a análise do comportamento de navegação do usuário. O mapeamento organizacional é importante para identificar transições inválidas ou inesperadas entre estados próximos. Já o mapeamento em nível semântico dá indicativas dos estados a serem analisados em nível organizacional. Indica se as transições realizadas e mapeadas em nível organizacional seguem um caminho, não somente válido por existência das mesmas em situações normais na aplicação, mas também considera o histórico das transições, seguindo o caminho percorrido pelo usuário e a sua validade dentro de um processo de negócio.

3.2. Validação de requisições em uma aplicação mapeada

A validação de requisições pode ser efetuada com análise de uma gama de variáveis do protocolo HTTP. No exemplo a seguir, essa análise será realizada de forma simplificada para demonstrar o processo proposto, com levantamento dos recursos e métodos de requisição. A figura 1 representa duas funcionalidades diferentes em uma aplicação web com algumas requisições em comum {/A, /A3 e /B}.

Para validação das requisições, deve ser realizado o mapeamento da aplicação e ajuste das requisições em transições de estado. Um estado no diagrama é identificado pela composição dos dados recebidos (parâmetros e dados obtidos a partir de especificidades do protocolo HTTP) e pelo recurso na requisição atual.

$$EstadoAtual = \{URI_n; Dados_n\}$$

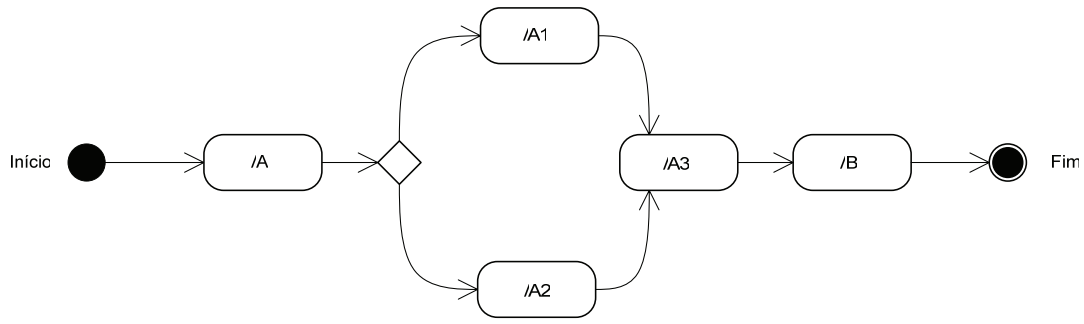

Figura 1. Exemplo de mapeamento de aplicação web

Na Tabela 2 está exemplificado o mapeamento completo das funcionalidades de negócio F1 e F2.

Tabela 2. Requisições mapeadas

Processo	Página Inicial	Página destino	Método
F1	/A	/A1	GET
	/A1	/A3	POST
	/A3	/B	POST
F2	/A	/A2	GET
	/A2	/A3	GET
	/A3	/B	POST

Para a funcionalidade F1, a requisição de /A3 é feita a partir de /A1 através do método POST. Já para a funcionalidade F2, a requisição de /A3 é feita a partir de /A2 através do método GET. De posse dessas informações, não se pode validar ou invalidar alguma dessas requisições para a navegação de um usuário, pois ambas as requisições de /A3 são inicialmente válidas. Para finalmente validar a requisição, é necessário recorrer ao mapeamento em um nível mais alto de abstração, o mapeamento semântico. Com ele, pode-se localizar a requisição de /A3 como pertencente à funcionalidade F1 ou F2, para então validá-la dentro da funcionalidade de negócio utilizada pelo usuário.

O resultado da validação da requisição deve ser utilizado para aumentar ou diminuir o indicador de confiança da aplicação no usuário, dependendo da interação ser normal (pode aumentar a confiança) ou suspeita (pode diminuir a confiança). Outras variáveis podem ser utilizadas para análise como o próprio endereço IP, dados do protocolo como o cabeçalho HTTP_REFERER, tempo entre requisições, dados esperados e passados pelo usuário, etc., como apresentado na Tabela 1.

4. Autenticação contínua baseada em confiança

A confiança segundo [Shankar e Arbaugh 2002] é baseada na identificação do indivíduo na sua identidade. Informações obtidas anteriormente sobre um indivíduo só podem ser aplicadas no futuro se o indivíduo for identificado como sendo o mesmo.

Para a definição da confiança inicial, supõe-se que todos os usuários são iguais no início das interações, e dessa forma, a confiança inicial de todos os usuários é a mesma. Esse comportamento foi assumido como premissa para o indicador de confiança, uma vez que não se pode garantir que um usuário utilizando certa credencial seja o mesmo usuário que a utilizou em uma sessão anterior.

A confiança, como explicado na sessão 2, pode ser tratada de maneira unidirecional ou bidirecional. Para a utilização do conceito de confiança na implementação da autenticação contínua, a confiança deve ser quantificada unidirecionalmente, ou seja, deve ser quantificada a confiança da aplicação no usuário. É a confiança que fornece os critérios para classificar uma seqüência de interações (navegação) como normal ou suspeita, base para o processo de autenticação contínua.

A confiança segundo [Gambetta 2000] pode ser quantificada com valores entre ZERO e UM, onde ZERO representa total desconfiança e UM a confiança cega e total em um outro indivíduo. É de se esperar que esses valores não sejam alcançados para o bom funcionamento matemático do sistema. Comparando com o comportamento humano, espera-se que:

- Quanto mais confiável o usuário, mais o valor se aproxima de UM, sem alcançá-lo;
- Quanto menos confiável, mais o indicador se aproxima de ZERO, sem alcançá-lo;
- A alteração do indicador depende da importância do contexto;
- Depende da quantidade de interações, assim, uma interação de um evento comum não interfere tanto quanto um evento raro.

Pelas propriedades acima é de se esperar que o comportamento do indicador de confiança seja similar ao de uma exponencial. A composição dos dois comportamentos é utilizada para representar as interações confiáveis e não confiáveis. Esse comportamento é apresentado em [Platzer 2004] juntamente com a demonstração das equações utilizadas para representar o comportamento totalmente confiável e o totalmente não confiável.

$$T_x = e^{-\frac{\ln(T_{x-1})}{S*(1-I_x)*\ln(T_{x-1})-1}}$$

Equação 1. Variação da confiança em comportamento confiável

$$T_x = 1 - e^{-\frac{\ln(1-T_{x-1})}{S*I_x*\ln(1-T_{x-1})-1}}$$

Equação 2. Variação da confiança em comportamento não confiável

Na tabela 3 seguem os índices utilizados nas equações para cálculo do indicador de confiança.

Tabela 3. Índices para cálculo do indicador de confiança

Descrição	**Representação**	**Valores possíveis**
Contexto	α, β, ...	
Variável temporal representada em eventos discretos	X	
Confiança Inicial	T_0	]0; +1[
Confiança Contextual	$T_x(\alpha)$	]0; +1[
Confiança mínima	T_m	]0; +1[
Importância do estado atual	$I_x(\alpha)$	]0; +1[
Incremento. Velocidade de variação do indicador de confiança em uma interação.	S	]0; ...[

A figura 2 representa o comportamento esperado para o comportamento confiável com confiança inicial 0,1 e o comportamento não confiável com confiança inicial 0,9. Percebe-se que a confiança é alterada sem atingir os valores limite 0 e 1. O valor da importância é igual para todos os contextos.

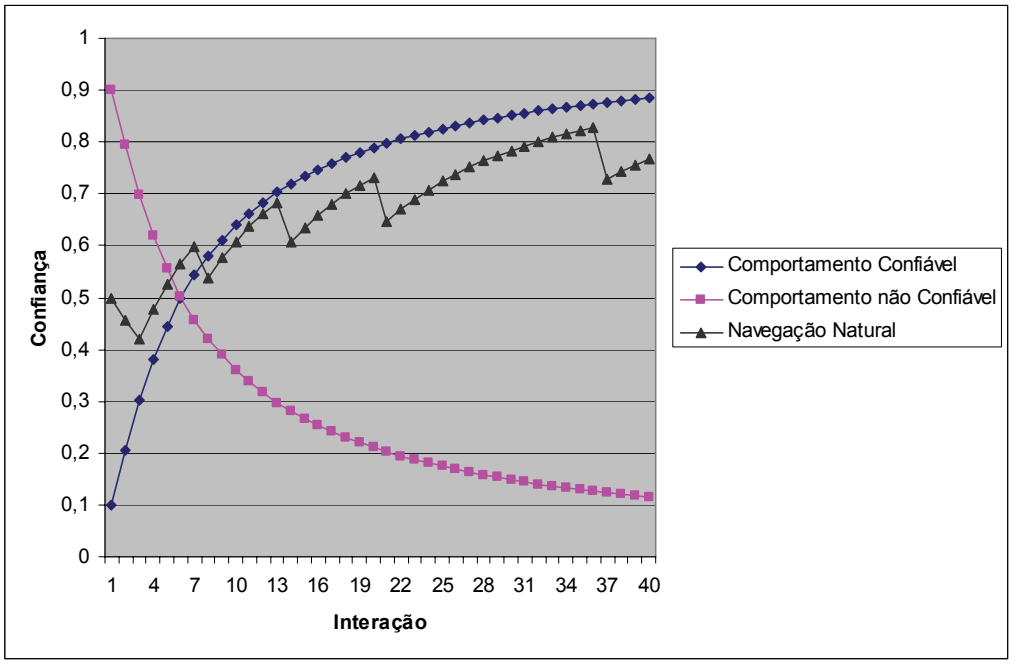

Figura 2. Confiança e Comportamento

As curvas de comportamento confiável e não confiável são simétricas em relação ao valor médio 0,5. Para um sistema de autenticação contínua, espera-se que o efeito de uma ação não esperada (indevida), que venha a diminuir o indicador de confiança, seja maior do que o efeito de uma ação esperada. Esse comportamento é natural, pois em situações normais, é de se esperar um número maior de ações classificadas como normais, em relação às ações não esperadas. Há duas possibilidades para implementar esse comportamento a partir das equações propostas:

- **Ajuste do valor da importância do contexto**: As duas equações são complementares. Para o comportamento confiável, quanto maior a importância, menor a variação. Para o comportamento não confiável, quanto maior a importância, maior a variação. Esse comportamento é realizado pelos termos $(1- I_x)$ e I_x, respectivamente;
- **Utilização de valores diferentes para S**: Quanto maior o valor de S, maior a sua variação.

A linha de navegação natural foi obtida a partir das três primeiras interações com uma aplicação fictícia durante a autenticação. O usuário realizou duas ações inesperadas ou inválidas e, portanto, a confiança depositada nele pela aplicação foi reduzida. Na terceira tentativa, o usuário se autenticou corretamente, o que restaurou parte da confiança da aplicação. Durante a utilização, algumas ações do usuário foram entendidas como incorretas, mas a grande maioria foi aceita normalmente, daí o formato em serra apresentado pelo indicador de confiança nesse usuário.

Percebe-se pelo gráfico da figura 2 o efeito de uma ação indevida em níveis diferentes de confiança. Quanto maior a confiança depositada no usuário, maior o efeito na confiança de uma ação indevida. O comportamento apresentado é bastante similar ao da confiança entre os seres humanos.

O conceito de autenticação contínua pode ser aplicado, negando o acesso do usuário caso a confiança caia abaixo de certo valor predeterminado. Outra possibilidade é a requisição de uma autenticação complementar, diferente da autenticação inicial quando esse valor for alcançado. Isto serve para possibilitar o incremento da sua confiança e o retorno do acesso às funcionalidades da aplicação ou então para revogar a autenticação inicialmente realizada.

O processo de autenticação pode ser utilizado também para realizar controle de acesso a funcionalidades. Pode-se configurar a aplicação de forma que certas funcionalidades estejam disponíveis somente a partir de certo nível de confiança, negando acesso às mesmas caso um usuário esteja com baixa confiança pela aplicação.

5. Arquitetura do sistema e apresentação de resultados

A autenticação contínua baseada em confiança pode ser implementada em um monitor de estados para uma aplicação segura na web. Para o sistema de acompanhamento manter independência do servidor de aplicação, o mesmo foi concebido como módulo externo do servidor. A localização ideal é como proxy reverso, conforme apresentada na figura 3, para receber as requisições, inspecioná-las e tomar a ação necessária.

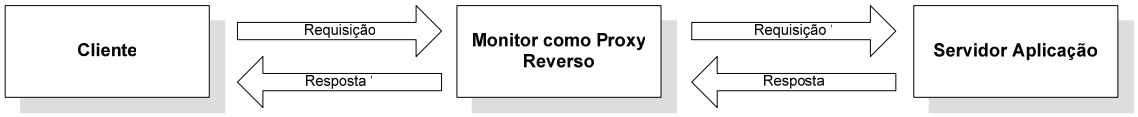

Figura 3. Arquitetura para acompanhamento de usuários

A ação realizada pelo monitor decorrente de uma violação das regras de confiança para a aplicação é chamada de ação restritiva. Pode ser a finalização da sessão, ou conforme o mapeamento realizado, a requisição de uma autenticação auxiliar de forma a restaurar o grau de confiança da aplicação no usuário.

5.1. Processos de análise

O processo de análise é iniciado com toda requisição de negócio realizada pelo usuário. Cada uma dessas requisições é traduzida em uma requisição HTTP, que por sua vez, é capturada pelo monitor, ou seja, a cada nova requisição HTTP o processo de análise é iniciado. Ele é composto pelos seguintes sub-processos:

- **Identificação do usuário na aplicação:** Análise da requisição para identificação da sessão de aplicação dentre todas as requisições recebidas pelo servidor de aplicações. A técnica de identificação da sessão utilizada pelo monitor deve ser a mesma utilizada pela aplicação;
- **Análise semântica**: Processo baseado na análise das transições do usuário na aplicação, em contraposição com o mapeamento realizado. Tem como objetivo a identificação da linha de processamento[1] em que o usuário se encontra. Enquanto ela não puder ser identificada (por

[1] Linha de processamento: Conjunto de estados organizados em seqüência para finalização de uma funcionalidade de negócio.

ambigüidade), as requisições são guardadas sequencialmente à sua chegada (em uma lista ligada).

- **Análise individual da requisição**: A análise individual da aplicação só pode ser realizada após a identificação da linha de processamento. Compreende a comparação da requisição esperada para aquela linha de processamento em função da requisição anterior, com aquela efetivamente realizada (recurso e dados). Comparação das transições de estados;
- **Atualização do indicador de confiança**: Compreende o processo de atualização do valor do indicador conforme o resultado das análises anteriores, utilizando as equações apresentadas na sessão 4.

Somente após a decisão tomada com base nessas análises, é que o monitor realiza a transição de estados, direcionando a requisição para a aplicação ou intervindo através de uma medida restritiva.

5.2. Descrição dos ensaios

Para a validação da metodologia de mapeamento e arquitetura propostos, foram realizados ensaios monitorando o comportamento de usuários em uma aplicação comercial de Internet Banking instalada em um ambiente controlado.

Os acessos à aplicação em questão foram realizados a partir de um navegador web, sem injeção de parâmetros com intuito de manipulação da aplicação. Para uma simplificação na realização dos testes, o navegador foi configurado para utilizar o monitor como proxy (a arquitetura proposta propõe a monitoração como proxy reverso).

O ensaio principal foi realizado para a funcionalidade de transferência entre contas, acessada a partir da funcionalidade de login, seguida da requisição do mapa de serviços (funcionalidade não mapeada) e requisição de extrato (também não mapeada). Como exemplo, o diagrama de transições de estados para a funcionalidade de transferência encontra-se na figura 4.

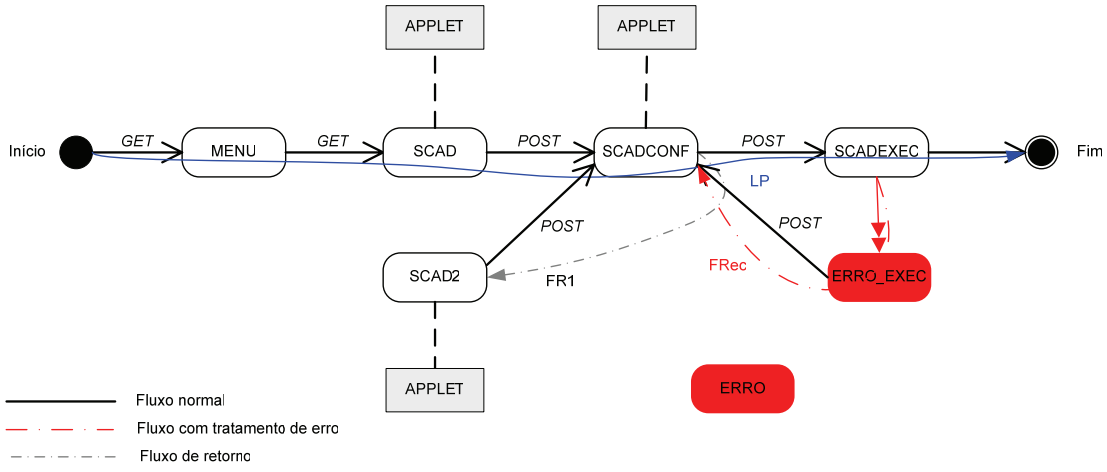

Figura 4. Transferência - Diagrama de transições de estados

Seguem as descrições dos estados:

- **MENU**: Apresenta duas opções de transferência, para contas cadastradas e não cadastradas;
- **SCAD**: Requisita os dados para a transferência: agência, conta, dígito e valor a ser transferido;
- **SCAD2**: Funcionalidade igual ao estado SCAD, mas acessado via fluxo de retorno (FR1). A transição para este estado pode ser identificada pelo método de requisição diferente, neste caso o método POST;
- **SCADCONF**: Exibe os dados recebidos do cliente, nome da agência e nome do cliente a receber o crédito da transferência. Requisita a senha do cartão de débito para autorização da operação e gera um número de controle específico para a transferência em curso;
- **SCADEXEC**: Efetua a transferência em si, representa o fim do caso de uso;
- **ERROR_EXEC**: Representa erro na execução da transferência e só pode ser alcançado (identificado pelo monitor) na próxima interação, após a verificação de um retorno ao estado SCADCONF a partir do estado SCADEXEC. Como o monitor só observa o comportamento pelas requisições, se essa última requisição (retorno ao estado SCADCONF) não fosse realizada, o monitor consideraria a transição para o estado SCADEXEC como sucesso, consequentemente, não diminuiria o valor do indicador de confiança.

Para cada estado mapeado na funcionalidade deve ser atribuído um valor de confiança. Valores maiores devem ser dados aos estados mais importantes, de forma que uma ação em um estado de pouca importância não interfira no indicador de confiança tanto quanto se ocorressem em um estado muito importante.

Para a configuração do monitor, os valores utilizados nos testes encontram-se na tabela 4. O valor inicial do indicador de confiança foi escolhido como a média entre os valores de confiança total e desconfiança total, para representar uma situação neutra. Os valores para o incremento foram escolhidos de forma a reforçar que uma ação indesejada tenha um efeito consideravelmente maior que uma ação reconhecidamente normal. O valor de confiança mínima foi escolhido entre a confiança inicial e o valor de total desconfiança.

Tabela 4. Transferência - Índices utilizados na configuração do monitor

Descrição	Representação	Valores possíveis
Confiança Inicial	T_0	0,5
Confiança mínima	T_m	0,3
Incremento	S	0,2(Confiável)/0,8(Não Confiável)

A navegação foi iniciada a partir do login com todas as requisições conforme esperadas. Entre a finalização do login e a entrada na funcionalidade de transferência, o

usuário passou por duas funcionalidades não mapeadas da aplicação e, portanto, o indicador de confiança permanece inalterado (não houve como determinar a validade ou não validade das transições).

O usuário inicia um processo de levantamento de informações de agência/conta e clientes, realizando transferências para diversas contas, uma a uma, sem finalizá-las, fazendo uso constante do fluxo de retorno FR1. Na segunda entrada dos dados de agência/conta, foi passada uma combinação inexistente, o que levou a aplicação a apresentar uma mensagem de erro no estado SCADCONF. Após esse incidente, a pesquisa de contas continuou normalmente. Segue na tabela 5 a evolução do indicador de confiança em função do mapeamento apresentado e os valores de confiança no usuário para as interações com a aplicação.

Tabela 5. Transferência – Indicador de confiança durante a navegação

Interação	Estado	Importância	Indicador de confiança
0			0,500000000
1	INICIAL	0,2	0,535824273
2	LOGIN	0,5	0,555823640
3	LOGINCHK	0,9	0,559626437
4	LINKS	0,5	0,577736076
5	não mapeado		0,577736076
6	não mapeado		0,577736076
7	MENU	0,3	0,600793797
8	SCAD	0,5	0,615818723
9	SCADCONF	0,8	0,621524381
10	SCAD2	0,7	0,467001681
11 (erro)	SCADCONF	0,8	0,477630769
12	SCAD2	0,7	0,378864848
13	SCADCONF	0,8	0,392859492
14	SCAD2	0,7	0,322950809
15	SCADCONF	0,8	0,339131800
16	SCAD2	0,7	0,285530494

Deve-se notar a diferença entre os valores da importância dos estados SCAD e SCAD2. Embora sejam mapeados para o mesmo recurso, os seus papéis são bastante diferentes no diagrama. Enquanto SCAD está localizado no fluxo natural da linha de processamento, SCAD2 está localizado em um fluxo de retorno (fluxo não natural), e portanto, é classificado como um estado de atenção. Ao final de 16 interações o indicador de confiança T_{16} assumiu um valor menor que T_m, e, portanto, o usuário sofreu uma medida restritiva (finalização da sessão).

6. Conclusões e trabalhos futuros

Este trabalho procurou apresentar a utilização da autenticação contínua para identificar comportamentos maliciosos de usuários, com utilização de dados de autenticação roubados ou não, e tentativas para burlar a política de segurança da aplicação.

O processo se baseia no mapeamento das funcionalidades da aplicação e no acompanhamento do usuário tendo como base o resultado desse mapeamento, que interfere diretamente no comportamento do monitor durante os processos de análises. O monitor considera transições (sintaticamente corretas) no decorrer da linha de processamento como sendo o comportamento normal, e qualquer desvio, mesmo que permitido pela aplicação, interfere negativamente no indicador de confiança do usuário.

A validade do processo, como demonstrada na sessão 5 deste artigo, foi reforçada por testes de navegação simultânea, com dois navegadores utilizando a mesma sessão (procurando simular um roubo de sessão) e diversas navegações durante a funcionalidade de login da aplicação, para assim, verificar o comportamento do indicador de confiança. O comportamento apresentado pelo monitor nos testes demonstra a sua capacidade de detectar comportamentos anômalos de usuários em uma aplicação web e a validade da escolha do indicador de confiança para representar o comportamento do usuário. A capacidade de identificar situações em que os usuários estejam utilizando a aplicação de maneira indesejada, ou indevida, foi apresentada fortemente nos resultados do ensaio na funcionalidade de transferência.

A técnica proposta não invalida a utilização das funcionalidades por usuários legítimos, ou ilegítimos, desde que sejam permitidas e o comportamento seja considerado confiável. Por exemplo, se um usuário Z, acessa um serviço bancário on-line com dados de autenticação do usuário Y e realiza transferência para a conta de um usuário X, todo o processo é validado, pois nenhuma regra da política de segurança da aplicação foi ferida diretamente. O acesso a essa funcionalidade é autorizado por dados válidos e a transferência autorizada por ser uma operação acessível ao usuário Z.

O mapeamento das funcionalidades pode ser facilitado se o próprio monitor for utilizado para gerar as saídas a partir das transições a serem utilizadas no acompanhamento dos usuários, ou seja, através de um processo de auto-configuração. Dessa forma, seria necessário que todas as linhas de processamento fossem percorridas, assim como no mapeamento realizado manualmente.

A análise de navegação insere um aumento no tempo de resposta da aplicação percebido pelo usuário. Esse tempo pode ser minimizado ao tempo de manipulação das conexões (recebimento e transmissão) pelo monitor, se o processo de análise da navegação for realizado em paralelo aos de manipulação de conexão. Dessa forma, o valor do indicador referente a uma interação só estaria disponível na interação seguinte, o que não inviabiliza o processo, desde que nas transições para estados com uma grande importância, a finalização desse cálculo seja aguardada, para evitar que uma situação crítica e indevida seja permitida.

O processo de identificação do usuário foi baseado em variáveis como endereço de rede e variáveis do protocolo HTTP, como cookies e controle de sessão da aplicação. Aplicações fornecidas por um monitor similar ao apresentado neste trabalho, podem ser executadas na estação do cliente com a possibilidade de levantar diversas informações

sobre a estação de trabalho e relacioná-las com o usuário da aplicação. Um aplicativo dessa natureza seria capaz de identificar algumas variáveis tanto de hardware como de software de forma que em uma próxima visita, essas informações pudessem ser utilizadas para alterar o valor da confiança inicial. Um usuário acessando uma aplicação a partir de um ambiente já conhecido, por exemplo, o seu computador pessoal doméstico, pode receber inicialmente uma confiança maior do que o mesmo usuário acessando a aplicação a partir de um novo ambiente.

Outra situação interessante é a utilização de diversas formas de autenticação. Dessa maneira, um usuário que acesse a aplicação bancária com autenticação simples só teria acesso a realizar operações de consulta, enquanto se a autenticação for realizada com a utilização de token (método mais seguro), mais funcionalidades podem ser disponibilizadas ao mesmo. O mesmo pode ser realizado no processo de comparação da confiança do usuário com uma confiança mínima para executar certa funcionalidade.

Referências

GAMBETTA, D. "Trust: Making and Breaking Cooperative Relations", capítulo 13. Disponível em <http://www.sociology.ox.ac.uk/papers/trustbook.html>. Acesso em: Fev. 2005.

THE INTERNET ENGINEERING TASK FORCE (IETF), "RFC 2616 - Hypertext Transfer Protocol - HTTP/1.1", Jun.1999, Disponível em: <http://www.ietf.org/rfc/rfc2616.txt>. Acesso em: Nov.2004.

INTERNET FRAUD COMPLAINT CENTER, "IFCC Annual Internet Fraud Report", Dec. 2002. Disponível em: < http://www.ifccfbi.gov/strategy/2002_IFCCReport.pdf>. Acesso em Jun. 2004.

JONES, S. and MARSH S. "Human-Computer-Human Interaction: Trust in CSCW". ACM SIGCHI Bulletin, V.29, n.3, p.36-40, Jul. 1997.

MARSH, S. "Formalising Trust as a Computional Concept". Ph.D.Thesis. Department of Mathematics and Computer Science, University of Stirling. 1994.

PLATZER, C. "Trust-Based security in web services". Master's Thesis – Technical University of Vienna. Austria 2004.

RUGGIERO, W. V. "Modelo de Segurança para redes Ad.Hoc". 97p. Tese (Livre-Docência) – Escola Politécnica, Universidade de São Paulo. São Paulo, 2002.

SCHNEIER, B., "Segurança.com: Segredos e mentiras sobre a proteção na vida digital". Tradução Daniel Vieira. Rio de Janeiro: Campus, 2001. 385 p.

SHANKAR, N. e ARBAUGH, C. "On Trust for Ubiquitous Computing". Workshop on Security on Ubiquitous Computing (UBICOMP'02): Göteborg, Sweden. Set. 2002.

STALLINGS, W., "Data and Computer Communications". Fifth Ed. Prentice Hall, 1996. 798p.