Projeto/Release: Identidade Digital	
Requisitos para integração ao sistema Identidade Digital	
Relator do Documento: Benícia Simone Hentges	**Data da Preparação:** 03/02/06
Versão: v02-030206	

Arquitetura lógica, componentes e funções do Sistema Identidade Digital

1 Objetivo

Apresentação da arquitetura lógica, componentes, métodos de autenticação, protocolos de comunicação, entre outros da solução Identidade Digital para integração de aplicações.

Serão mostrados os requisitos necessários para a viabilização de integrações de aplicações web e client ao sistema Identidade Digital.

2 Arquitetura Lógica

A definição da arquitetura lógica tem como objetivo mostrar uma visão geral do framework de autenticação (iChain e eDirectory) e o acesso do usuário.

Entende-se por proteger: Autenticar, autorizar, acelerar, habilitar cache, criptografar através de SSL e habilitar single sign-on.

3 Métodos de autenticação

A grande maioria das aplicações utilizam apenas o método senha simples ou seja, é informado nome do usuário e senha para autenticação. Autenticação com outros métodos tais como: Tokens, Smartcards, Dispositivos biométricos, certificados X.509 etc, são aplicados apenas em casos de necessidade/solicitação da aplicação.

4 Aplicações WEB

Entende-se por aplicação WEB: Qualquer aplicação que pode ser acessada via Browser, que utiliza o protocolo HTTP ou HTTPS e cujo conteúdo que chega ao usuário está nos formatos padrão web (HTML, DHTML, JavaScript, XML).

De acordo com a figura abaixo, o ciclo que teremos em qualquer aceleração de aplicação WEB será:

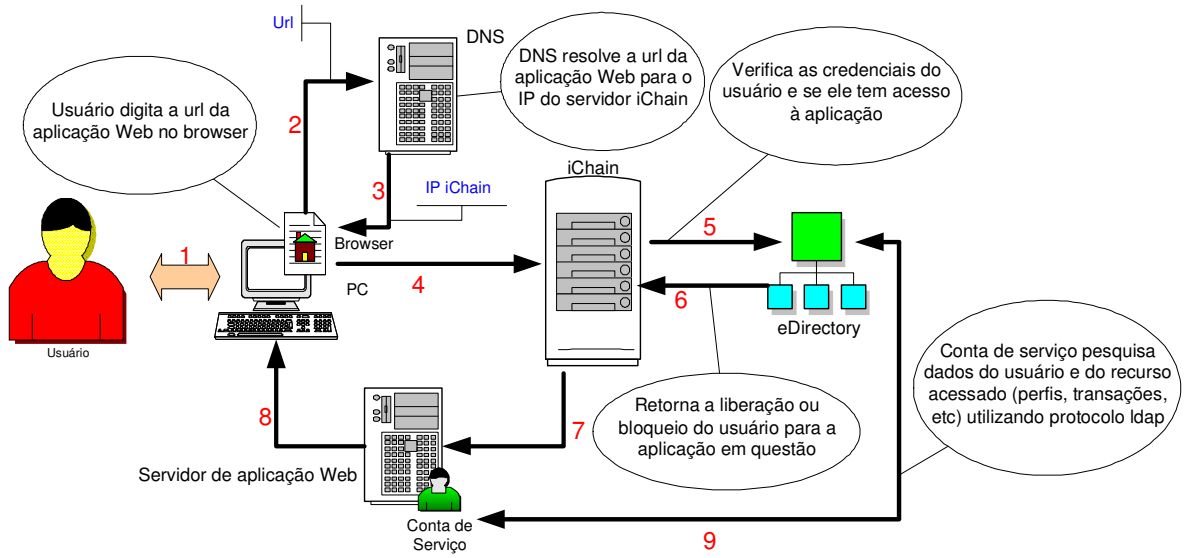

Figura 1 - Desenho lógico do framework de segurança (iChain) para aplicações web

Item	Descrição
1	O usuário digita no browser (ou simplesmente clica em um link) que aponta para uma URL. O sistema operacional envia a requisição de resolução de nomes para o servidor DNS.
2 e 3	A partir do momento que iChain estiver customizado para a aplicação, o DNS deve retornar seu endereço IP e não mais o do servidor de aplicação.
4	O iChain retorna para o usuário uma tela de autenticação, caso a url acessada seja dos tipos restrita ou segura (Obs.: o iChain também pode acelerar páginas públicas). O usuário deve então, fornecer as suas credenciais, que podem ser o seu nome, senha, e-mail, tokens, certificados digitais ou qualquer outro atributo contido no diretório.
5	O iChain acessa o diretório para saber se o usuário e suas credenciais são válidas. Caso a verificação seja positiva o processo prossegue. Caso não seja, o diretório bloqueia o acesso e o iChain apresenta uma tela de erro de acesso ao usuário.
6	O diretório retorna para iChain a validação do usuário e a ACL contendo as URL's que este usuário pode ou não acessar.
7	No caso de sucesso na validação de usuário e as Acls a que tem acesso, o iChain encaminha a requisição de acesso do usuário para o servidor de aplicação onde a aplicação Web está hospedada. Caso a aplicação necessite de algumas informações do usuário, como nome, email, etc (exceto perfis), estes podem ser repassados para a aplicação através do header http.
8	O usuário consegue atingir a aplicação web.
9	Se a aplicação trabalha com perfis e transações, e estes estiverem armazenados no diretório, deverão ser consultados pela aplicação (utilizando o protocolo ldap), através de uma conta de serviço. Para esta consulta, será criada uma conta de serviço para a aplicação e serão atribuídas as permissões necessárias para a realização das mesmas.

Tabela 1 - Processo de autenticação para aplicações web

5 Aplicações Client

Entende-se por aplicação Client: Qualquer aplicação que funcione no esquema client-server.

De acordo com a figura abaixo, o ciclo que teremos em qualquer integração de aplicação Client será:

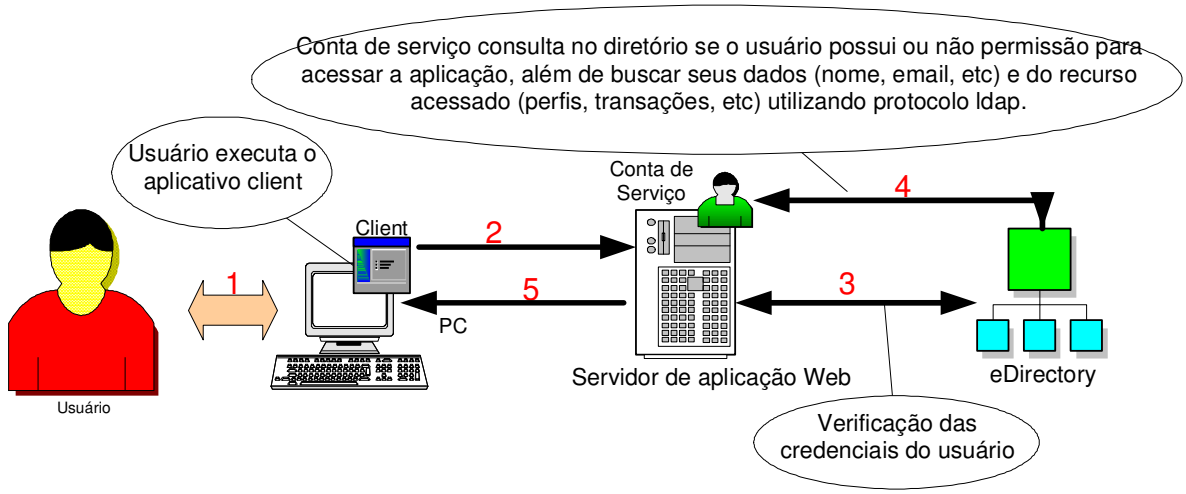

Figura 2 - Desenho lógico do framework de segurança para aplicações client

Item	Descrição
1	O usuário executa a aplicação client, que devolve uma tela de autenticação para acessar o sistema. O usuário deve fornecer suas credenciais, que podem ser o seu nome, senha, e-mail, tokens, certificados digitais ou qualquer outro atributo contido no diretório, para poder acessar a aplicação.
2	A aplicação client acessa o servidor de aplicação, que está configurado para realizar a autenticação do usuário na base do diretório, e não em sua base local.
3	O servidor da aplicação acessa o diretório (utilizando o protocolo ldap) com as credenciais fornecidas pelo usuário para saber se este é um usuário válido e se suas credenciais são aceitas. Caso a verificação seja positiva o servidor de aplicação executa o passo 4, caso não seja, o servidor de aplicação deve bloquear o acesso e apresentar no client uma tela de erro de acesso ao usuário.
4	Se a validação do usuário for positiva, a aplicação deve realizar uma outra consulta (utilizando o protocolo ldap) ao diretório utilizando uma conta de serviço para pesquisar se o usuário que está tentando acessar a aplicação pode ou não acessar esta aplicação, através de atributos determinados para cada aplicação. Caso a aplicação trabalhe utilizando perfis, estes deverão ser consultados também, através da conta de serviço. Para estas consultas, será criada uma conta de serviço para a aplicação e serão atribuídas as permissões necessárias para estas consultas.
5	As telas customizadas da aplicação são liberadas ou não para o usuário, de acordo com suas permissões e perfis.

Tabela 2 - Processo de autenticação aplicações client

6 Componentes de segurança e suas funções

Este item tem como finalidade apresentar os componentes lógicos pertencentes e necessários ao framework de segurança para se fazer a autenticação, autorização, caching, SSLizer e single sign-on.

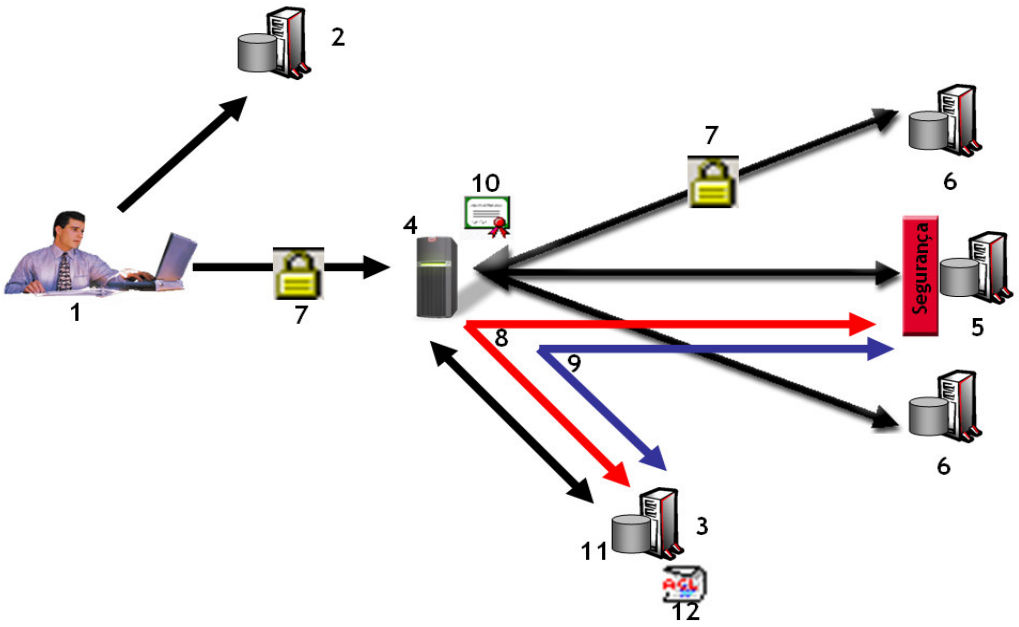

Figura 3 - Componentes de segurança e suas funções

Item	Componente	Descrição
1	Browser do usuário	Todas as aplicações suportadas pelo framework de autenticação (iChain) devem ser do tipo WEB e como tal, deve ser acessada através de um web browser.
2	Servidor DNS	O servidor DNS deve ser capaz de resolver o nome da infra-estrutura de segurança. Todas as aplicações que já possuem uma URL resolvida devem solicitar uma alteração no servidor DNS que as contém para ser direcionadas para a nova infra-estrutura de segurança. Não é necessária a implementação de uma nova arquitetura de DNS e este servidor DNS pode ser qualquer um, de qualquer fabricante em qualquer sistema operacional.

Item	Componente	Descrição
3	Diretório	A solução do framework de segurança requer um diretório (Novell eDirectory obrigatoriamente) como fonte de autenticação e autorização. Esse diretório deve permitir acesso LDAP(S). Este diretório já está disponível: trata-se do diretório corporativo da Brasil Telecom. Neste diretório, podem ser armazenados dados relativos ao acesso da aplicação, tais como permissões de acesso, bem como os perfis e as transações pertencentes a uma determinada aplicação.
4	Proxy reverso	A solução de framework de segurança necessita de uma infra-estrutura de proxy reverso. Estes servidores farão o trabalho de SSLizer, caching, autenticação, autorização e single sign-on das aplicações e sites existentes. Este framework já está disponível: trata-se do iChain da Brasil Telecom.
5	Aplicações WEB Aplicações Client	Para se caracterizar uma aplicação web, esta deve ser acessada via browser, utilizar o protocolo HTTP(S) e transmitir o conteúdo que chega ao usuário nos formatos padrão web (HTML, DHTML, JavaScript, XML). As aplicações web podem ter uma camada de autenticação já inerente à própria aplicação. Para se caracterizar como uma aplicação Client, esta deve funcionar no esquema client-server.
6	URL's	Para se caracterizar uma URL, esta deve ser acessada via browser, ser estática *, utilizar o protocolo HTTP(S) e transmitir o conteúdo que chega ao usuário nos formatos padrão web (HTML, DHTML, JavaScript, XML). As URL's são públicas e não possuem banco de dados de usuários ou camadas de autenticação ou segurança. * A url base da aplicação deve ser estática. Por exemplo: http://intranet.brasiltelecom.com.br/xpto. Isto não impede a utilização de diretórios (pastas) na url da aplicação, desde que a url até o contêiner da aplicação sejam fixos.
7	Certificados Digitais SSL	O proxy reverso (iChain) manterá os certificados digitais SSL para que o browser do usuário consiga manter uma conexão segura na camada de aplicação. Estes certificados podem ser gerados pelo próprio framework de segurança, por uma CA (Certificate Authority) interna ou mesmo uma externa. Existe a possibilidade de se suportar uma comunicação segura entre o proxy reverso e a aplicação web, mantendo o canal seguro em toda a extensão.
8	OLAC (Object-Level Access Control)	O OLAC representa uma funcionalidade do proxy reverso capaz de ler informações diversas do diretório e repassar essas informações para a aplicação web, caso esta necessite destes dados (Ex: Basic Authentication).

Item	Componente	Descrição
9	Form Fill	O form fill representa uma funcionalidade do proxy reverso (iChain) capaz de ler informações diversas do diretório e repassar essas informações para a aplicação web, caso esta necessite destes dados para uma autenticação através de "forms". Ele pode ser utilizado para transmitir para a aplicação web credenciais diferentes das do diretório ou armazenadas no Secret Store, permitindo o single sign-on. ** ** O uso de OLAC ou Form Fill vai depender da natureza da aplicação web.
10	Trusted Root	Caso a aplicação WEB possua Certificados Digitais SSL, o trusted root desse certificado deve ser exportado e adicionado ao proxy reverso para que este consiga estabelecer o canal seguro e confiável.
11	Secret store	Repositório de dados no diretório capaz de armazenar credenciais de qualquer aplicação. O proxy reverso (iChain), juntamente com o OLAC ou Form Fill, podem acessá-lo para informar as credenciais para a aplicação web.
12	ACLs	As ACLs são as listas de controle de acesso que garantem quais serão as aplicações ou URL's que os usuários poderão ou não acessar. Elas são armazenadas no diretório e acessadas pelo proxy reverso (iChain) no momento da autenticação.

Tabela 3 – Componentes de segurança e suas funções