Autenticação Mútua TLS

Ou: acesso a serviços web TLS mediante uso de certificados de cliente.

Objetivo deste documento:

Explicar fundamentos da conexão TLS, da criptografia de chave pública e autenticação mútua.

Para quem é este documento?

Para equipes de TI – desenvolvimento, infraestrutura e segurança de sistemas computacionais – envolvidas no acesso a serviços web TLS.

O que é serviço web?

É um serviço oferecido por um **dispositivo eletrônico** para outro dispositivo eletrônico por meio de um protocolo da web (http ou https). Ou seja, diferencia-se de um site web por não ser utilizado por pessoas diretamente.

TLS ou SSL? Com ou Sem?

[v. https://en.wikipedia.org/wiki/Transport_Layer_Security]
Transport Layer Security – é a evolução do SSL. Os termos são intercambiáveis e o TLS será usado daqui pra frente. Essa camada garante que o **canal de comunicação entre cliente e servidor é seguro**, criptografado.

Certificados digitais

[v. https://en.wikipedia.org/wiki/Public_key_certificate]

Um certificado digital serve para provar a identidade do seu proprietário.

É assinado por uma Autoridade Certificadora (AC ou CA). As ACs são terceiros que garantem a autenticidade dos certificados mediante processos de validação. Numa analogia com os documentos em papel, as ACs seriam os cartórios, e um certificado seria um documento autenticado.

ACs confiáveis

Todos os navegadores – Firefox, Internet Explorer, Chrome, Safari, etc. – vêm de fábrica com um conjunto de ACs confiáveis. No caso de IE e Chrome no Windows, usam as ACs confiáveis do sistema operacional – que podem ser vistas no programa certmgr.msc. Também os sistemas operacionais e VM's de linguagem, como a máquina virtual do Java, vêm de fábrica com um conjunto de ACs confiáveis (o trusted store).

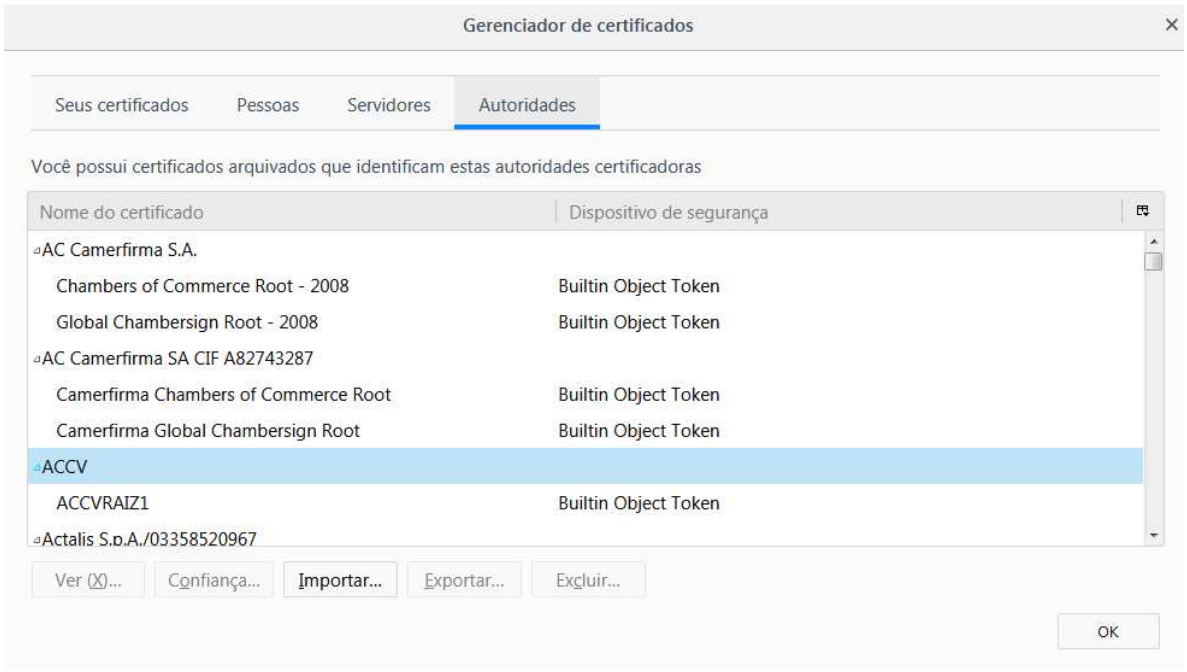

Figura 1: AC's confiáveis do Firefox (fonte: próprio sistema)

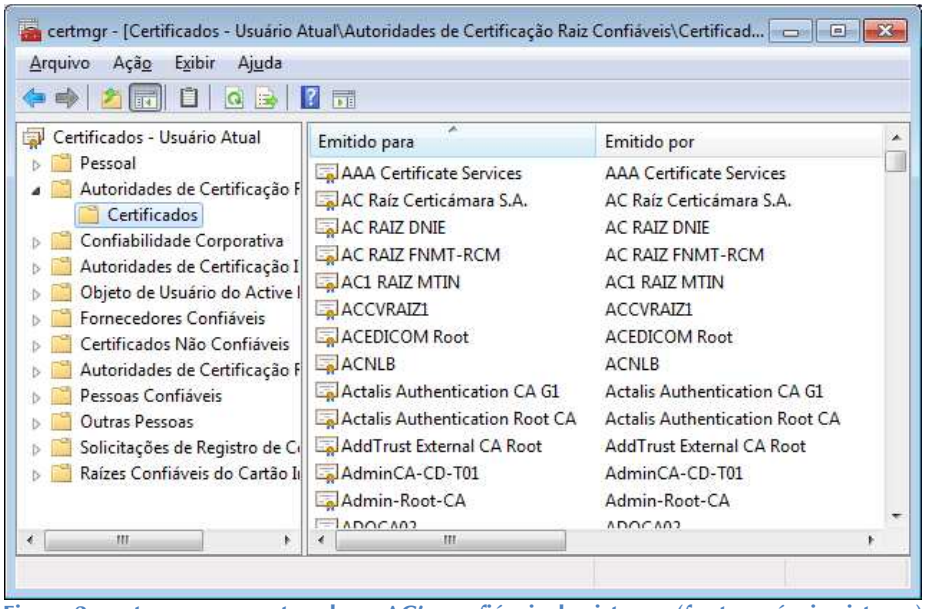

Figura 2: certmgr.msc mostrando as AC's confiáveis do sistema. (fonte: próprio sistema)

Cadeia de certificados

Normalmente, por razões de segurança da raiz, um **certificado fim** ou certificado de entidade é emitido por um **certificado intermediário** (que geralmente também é gerenciado pela AC). Assim, o certificado fim é emitido pelo intermediário que é emitido pelo **certificado raiz**.

O uso de um certificado intermediário é comum. Mas há casos em que não há certificado intermediário na cadeia – ou seja, o certificado fim é emitido diretamente pelo certificado raiz. Bem como pode haver mais de um certificado intermediário.

Aquele que deseja se autenticar com um certificado deve enviar a **cadeia completa** e não somente o certificado fim.

O formato é o seguinte:

- C-FIM , depois o
- C-INT-1 (emissor do C-FIM), depois o ...
- C-INT-2 (emissor do C-INT-1), depois ...
- ...
- C-INT-n (emissor do C-INT-(n-1)), e finalmente o...
- C-RAIZ (emissor do C-INT-(n-1)). Obs.: o envio do C-RAIZ é **OPCIONAL**, pois este certificado deve constar na lista de confiáveis para que a validação seja possível pela outra parte.

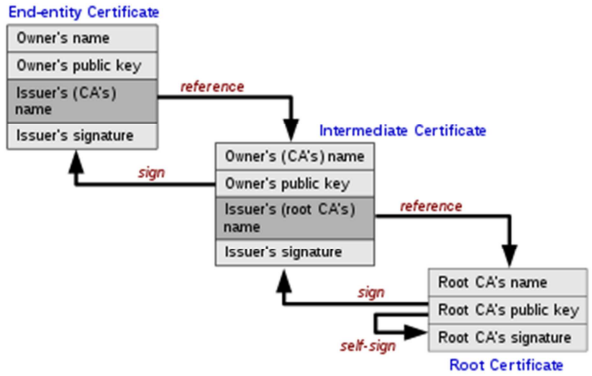

Figura 3: Cadeia de certificados (fonte Wikipedia https://en.wikipedia.org/wiki/Public_key_certificate).

Autenticação de certificado

Quando um usuário acessa um site https, ocorre uma autenticação de certificado. Não se trata ainda da autenticação mútua.

No início da conexão TLS, ocorre o "handshake TLS", ou seja, quando os participantes da conversa definem os parâmetros daquela conversa. Entre os parâmetros estão por exemplo qual o algoritmo criptográfico que será utilizado, bem como seus parâmetros numéricos.

Durante o handshake TLS, em uma das mensagens o servidor **envia seu certificado** (a cadeia de certificados, na realidade) cuja autenticidade poderá ser verificada pelo cliente. A Figura 4 mostra em detalhes como isso ocorre.

Obs: na figura, a palavra "certificate" é uma simplificação; no lugar, considerar "cadeia de certificados".

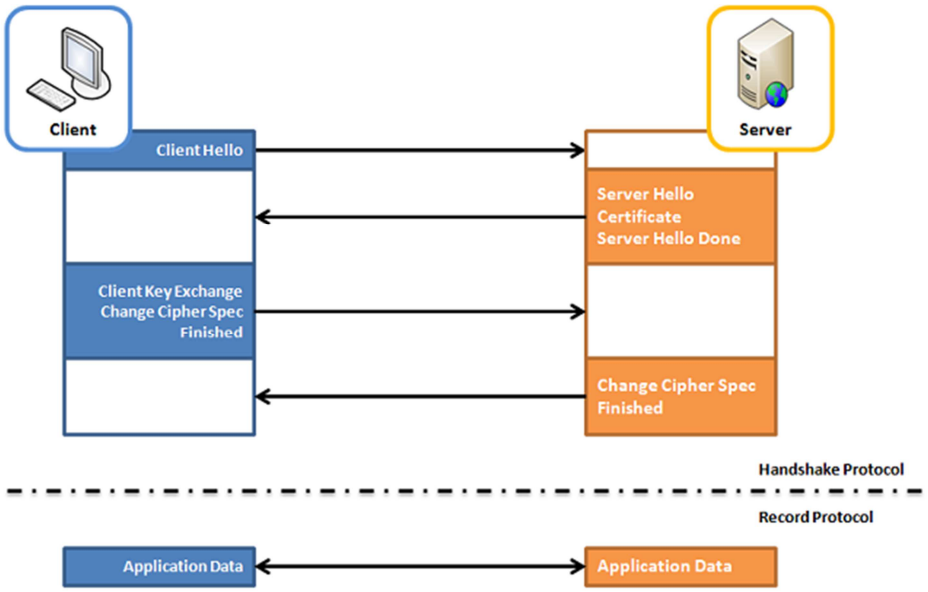

Figura 4: Autenticação TLS comum – cliente valida o certificado do servidor. Note que apenas o servidor envia o certificado. [v. https://www.codeproject.com/Articles/326574/An-Introduction-to-Mutual-SSL-Authentication].

Exemplo:

1. Conforme Figura 5, Firefox tem várias raízes incluídas na distribuição;
2. Usuário visita https://www.bndes.gov.br/ ;
3. O site envia para o navegador os seus certificados (a cadeia completa) conforme Figura 6;
4. O navegador, de posse dos certificados recebidos do servidor, pode verificar que o certificado fim foi assinado pelo intermediário. E como ele já possui o certificado raiz, ele pode verificar que o intermediário é assinado por uma raiz na qual ele confia; ou seja, o certificado fim deste servidor é confiável e o navegador pode ativar o cadeado na barra.

Por exemplo, se o certificado selecionado na Figura 5 fosse removido, o site não mais seria validado como confiável, e o navegador exibiria uma tela de aviso de segurança.

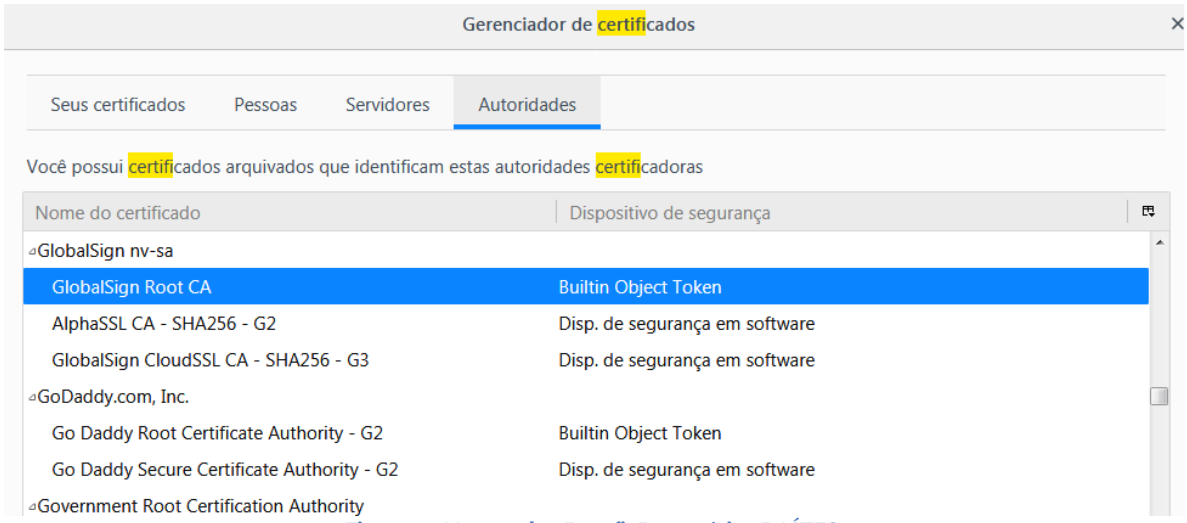

Figura 5: Navegador "confia" em várias RAÍZES.

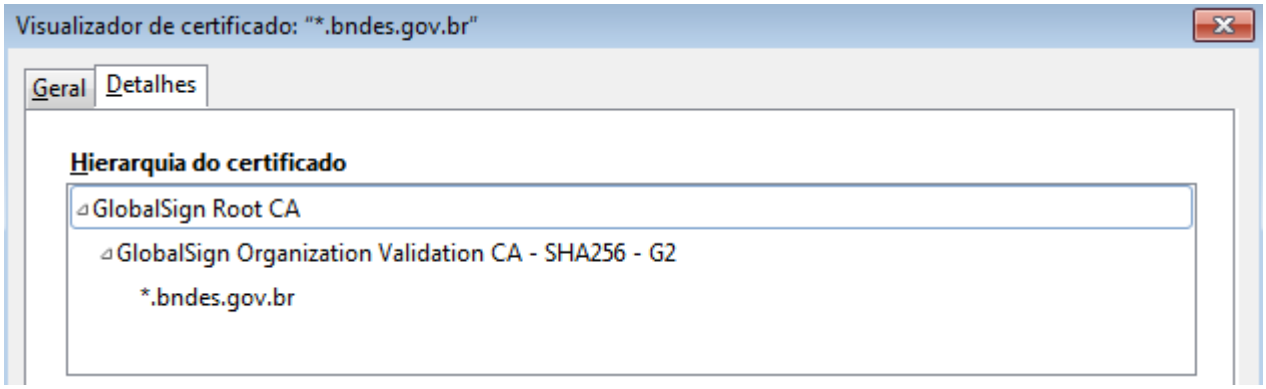

Figura 6: Cadeia de certificados do site https://www.bndes.gov.br/

Autenticação mútua

Há casos em que o acesso a um recurso do servidor só pode ser liberado mediante a validação de um certificado digital do cliente – seja ele um usuário ou um dispositivo eletrônico.

Chamamos de **autenticação mútua quando ambos cliente e servidor apresentam certificados para serem validados pelo par**.

Na Figura 7, é mostrado o handshake da autenticação mútua. Note que tanto o servidor quanto o cliente enviam seus certificados (na realidade, suas cadeias de certificados). O cliente envia seus certificados depois que o servidor envia um "*certificate request*".

Embora não fique claro na figura abaixo, junto deste *certificate request*, o servidor envia também a lista de ACs nas quais ele confia, para que o cliente possa escolher qual certificado enviar.

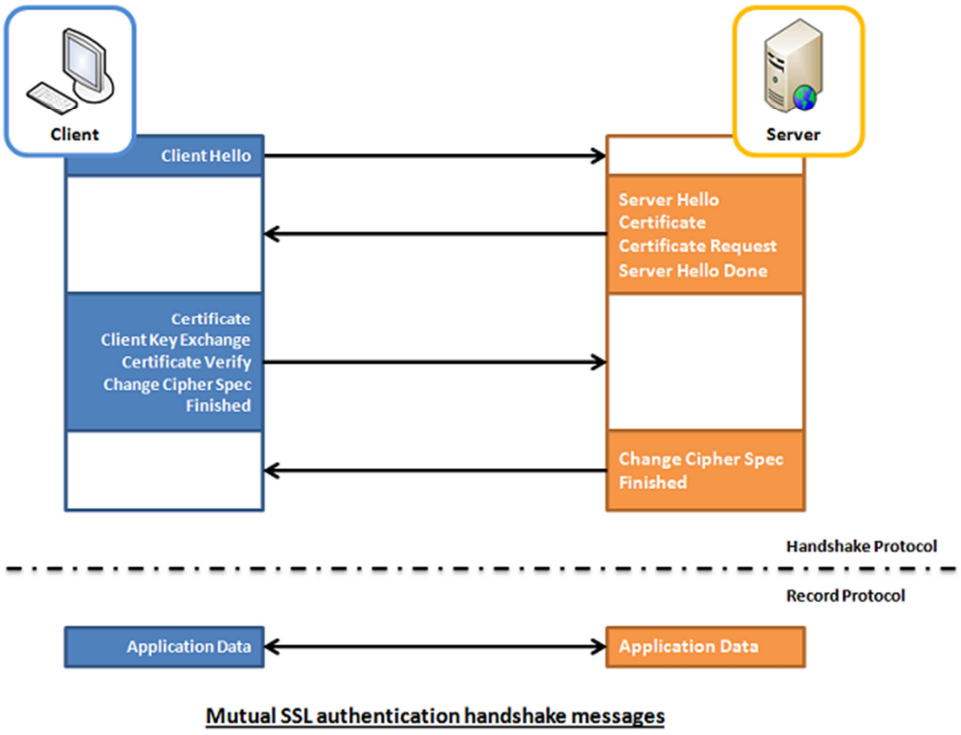

Figura 7: Autenticação Mútua. Note que Cliente e Servidor enviam um certificado.
[de https://www.codeproject.com/Articles/326574/An-Introduction-to-Mutual-SSL-Authentication]

ICP-Brasil

Em síntese, a ICP-Brasil é a cadeia hierárquica de emissão de certificados vinculada ao governo brasileiro. Os certificados raízes da ICP-Brasil podem ser encontradas em http://iti.gov.br/repositorio/84-repositorio/143-repositorio-ac-raiz .

Para validar um certificado fim da ICP-Brasil, aquele que está validando deve verificar se a cadeia de certificados leva a uma das raízes da ICP-Brasil.

Exemplos de sites/serviços que exigem certificados ICP-Brasil:

- o site da eCAC da Receita Federal exige um certificado eCPF ou eCNPJ para alguns serviços https://cav.receita.fazenda.gov.br/autenticacao/login/index/2 ;
- web service Infoconv da Receita;
- eSocial [v. https://login.esocial.gov.br/login.aspx];
- processo judicial eletrônico [v. http://www.cnj.jus.br/tecnologia-da-informacao/processo-judicial-eletronico-pje/certificacao-digital]

Testando autenticação

O objetivo do roteiro seguinte é prover ferramentas para testar a autenticação mútua de uma aplicação.

1. Gerar um certificado auto assinado para o servidor simples

```
$ openssl req -x509 -newkey rsa:4096 -sha256 -nodes -keyout key.pem -out server.pem -days 365
(preencher os dados com valores quaisquer)
```

2. Rodar o servidor simples (precisa do pacote OpenSSL)

```
$ openssl s_server -accept 4433 -Verify 3 -CAfile trusted-cas -cert server.pem -key key.pem –www

(mensagens de debug vão aparecer aqui)
```

Ao acessar https://localhost:4433/ pelo navegador e não apresentar um certificado de cliente, você receberá o erro “SSL_ERROR_HANDSHAKE_FAILURE_ALERT” no navegador e a mensagem abaixo vai aparecer na tela do servidor:

```
4294956672:error:140890C7:SSL routines:ssl3_get_client_certificate:peer did not return a certificate:s3_srvr.c:3330:
ACCEPT
```

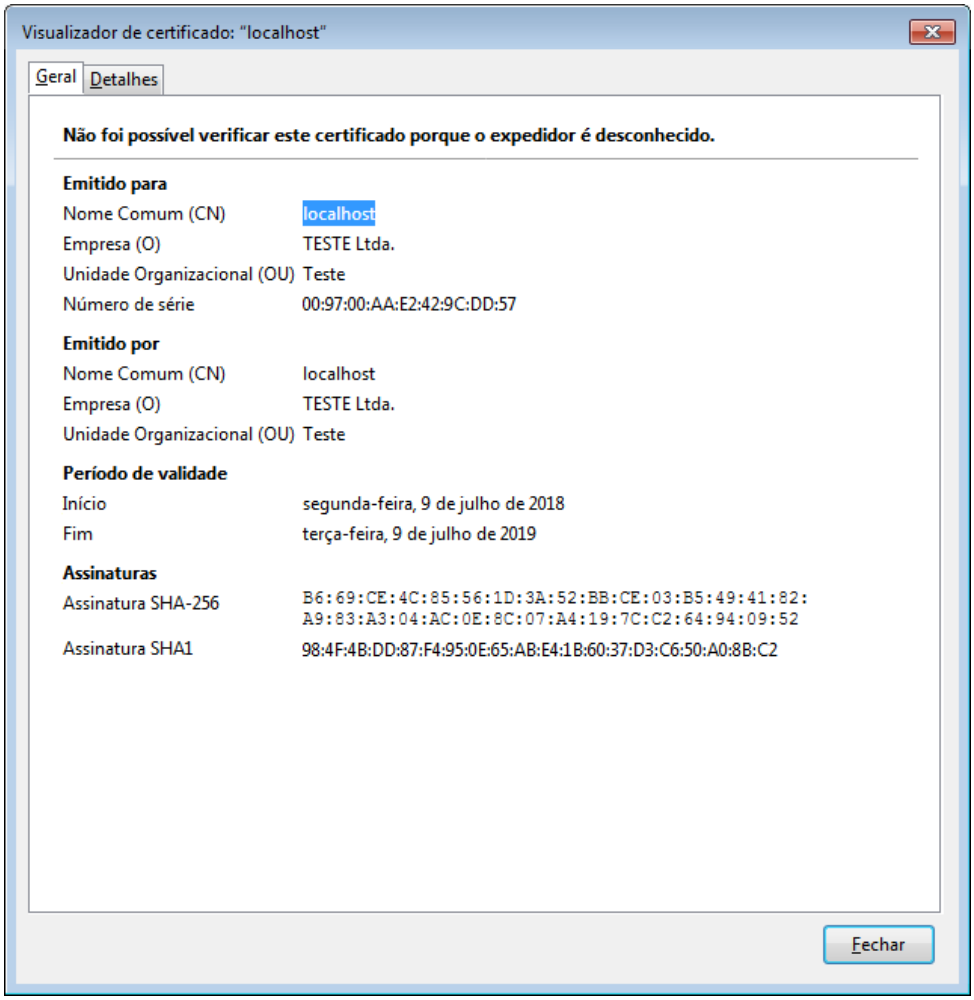

Figura 8: Certificado server.pem gerado no OpenSSL visto no Firefox.

Instalação do Cygwin

Caso os testes sejam rodados em Windows, será necessária a instalação do Cygwin conforme abaixo:

1. Visitar o site http://cygwin.com/
2. Baixar o aplicativo de setup-x86_64.exe (ou _x86, se desejar)
3. Executar o comando abaixo:

```
setup-x86_64.exe --quiet-mode --verbose --no-admin --local-package-dir D:\Temp\cygwin64 --root D:\cygwin64 --site
http://linorg.usp.br/cygwin/ --packages openssl,wget
```

substituindo os caminhos para a raiz do sistema Cygwin e para o diretório temporário como preferir. Também é possível simplesmente clicar duplo no setup e seguir as orientações pedindo para instalar os pacotes openssl e wget.