INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA

DO SERTÃO PERNAMBUCANO

PROJETO ICPEdu

(INSTALAÇÃO APACHE2 + SSL + CERTIFICADOS)

Petrolina/PE

2016

CONCEITO

Definição 1: O Apache-SSL é uma iniciativa de Ben Laury, desenvolvedor que participa dos projetos Apache e OpenSSL. O Apache-SSL é um servidor web capaz de fornecer, por padrão, criptografia baseada no protocolo SSL, utilizando o OpenSSL e SSLeay. A licença é BSD-Style e pode ser utilizado para fins comerciais e não comerciais.

Definição 2 : OpenSSL é uma ferramenta livre para a implementação de protocolos para Conexões Seguras (SSL - Secure Sockets Layer) e transporte seguro (TLS - Transport Layer Security) de dados em uma rede, e mecanismos de criptografia. Com essa ferramenta é possível criar certificados, chaves sumarizadas, chaves públicas e privadas e criptografar arquivos.

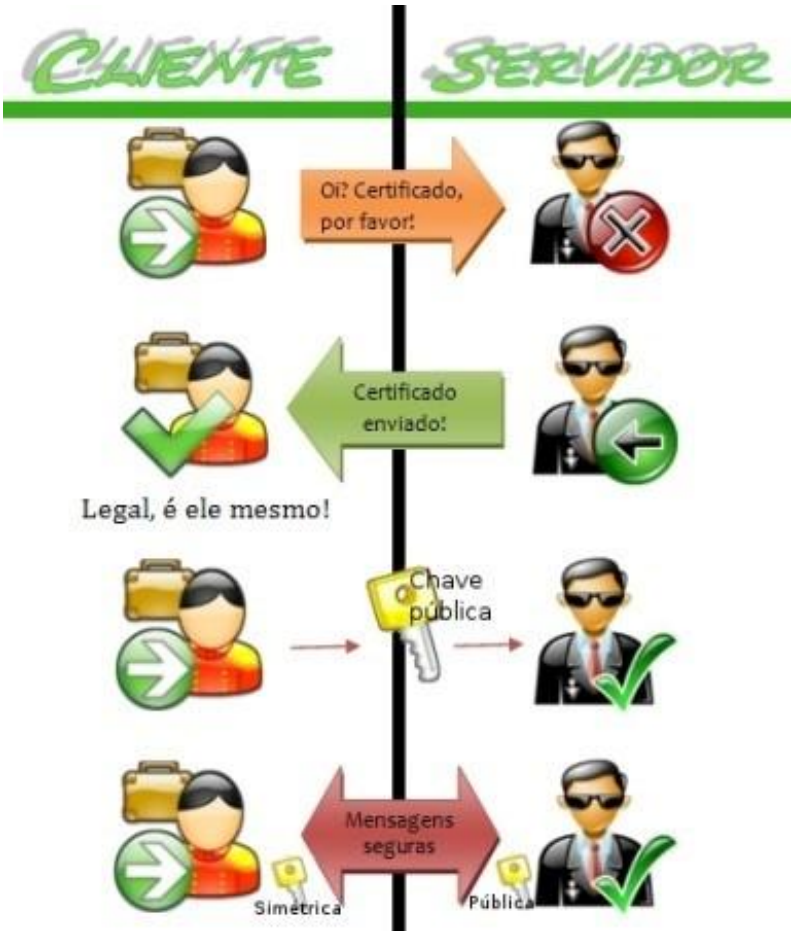
Entendendo como funciona

Uma conexão utilizando SSL é sempre iniciada pelo cliente. Quando um usuário solicita a conexão com  um site seguro, o navegador web (Firefox, Internet Explorer, Opera, Chrome, etc.) solicita o envio do Certificado Digital e verifica se:

- O certificado enviado é confiável.
- O certificado é válido.
- O certificado está relacionado com o site que o enviou.

Uma vez que as informações acima tenham sido confirmadas, a chave pública é enviada e as mensagens podem ser trocadas. Uma mensagem que tenha sido criptografada com uma chave pública somente poderá ser decifrada com a sua chave privada (simétrica) correspondente.

Pense na mensagem como sendo uma fechadura e que ela possui duas chaves, umas para trancar (criptografar) e outra para destrancar (decifrar) a porta. O desenho abaixo explica melhor.



Um servidor web protegido pelo protocolo SSL possui uma URL que começa em "https://", onde o S significa *"secured"* (seguro, protegido).

Alguns algoritmos famosos de criptografia utilizam o protocolo SSL. Veja abaixo alguns deles:

- DES e DAS - algoritmo de criptografia usado pelo governo americano.
- KEA - usado para a troca de chaves pelo governo americano.
- MD5 – muito usado por desenvolvedores de software para que o usuário tenha certeza que o aplicativo não foi alterado.
- RSA - Algoritmo de chave pública para criptografia e autenticação.
- SHA-1 - também usado pelo governo americano.

A versão 3.0 do SSL exige à autenticação de ambas as partes envolvidas na troca de mensagens. Ou seja, tanto cliente quanto servidor deve fazer autenticação e afirmar que são que dizem ser.

FASE 1 (SERVIDOR WEB SEM SSL - HTTP)

1. **Instalação do Apache2 com suporte a SSL**

apt-get install apache2 openssl

2. **Ativar o módulo SSL no seu servidor**

a2enmod ssl

/etc/init.d/apache2 restart

OBS: Ao reiniciar o Apache poderá aparecer a seguinte mensagem "Could not reliably determine the server's fully qualified domain name, using 127.0.1.1 for ServerName"
Solução:

- Ajustando o arquivo de configuração do Apache

cp /etc/apache2/apache2.conf /etc/apache2/backup.apache2.conf

- Em seguida, abra o arquivo de configuração e acrescente (pode ser no final):

ServerName localhost

- Reinicie o Apache:

/etc/init.d/apache2 restart

Após terminar acesso HTTP://IP-DA-MÁQUINA-SERVIDOR

FASE 2 (SERVIDOR WEB COM SSL - HTTPS)

 # Criação do certificado auto-assinado

1. **Criar um novo diretório onde iremos armazenar a chave do servidor e o certificado SSL.**

mkdir /etc/apache2/ssl

2. **Geração da Chave Privada e da CSR (Certificate Signing Request) que contém o certificado que será fornecido aos clientes**

openssl req -x509 -nodes -days 3650 -newkey rsa:2048 -keyout /etc/apache2/ssl/ifsertao.key -out /etc/apache2/ssl/ifsertao.crt

cd /etc/apache2/sites-available

rm defaultt-ssl

cp default default-ssl

vim default-ssl

- Alterar a porta 80 para 443
- Adicionar as seguintes linhas de commando:

SSLEngine on
SSLCertificateFile /etc/apache2/ssl/ifsertao.crt
SSLCertificateKeyFile /etc/apache2/ssl/ifsertao.key

- Criar um link Simbolico

ln -s /etc/apache2/sites-available/default-ssl /etc/apache2/sites-enabled/000-default-ssl

/etc/init.d/apache2 restart

Após terminar acesso HTTPS://IP-DA-MÁQUINA-SERVIDOR e verificar as informações do certificado.

FASE 3 (Certificado ICPEdu – RNP)

1. **Gerando um Pedido CSR (O utilitário "OpenSSL" é usado para criar ambos Chave Privada (chave) e Solicitação de Assinatura de Certificado (CSR))**

Gerar chave privada com senha

openssl genrsa -des3 -out teste.reitoria.ifsertao-pe.edu.br.key 2048

Retirar senha da chave privada

#openssl rsa -in teste.reitoria.ifsertao-pe.edu.br.key -out teste.reitoria.ifsertao-pe.edu.br.key.insecure

mv teste.reitoria.ifsertao-pe.edu.br.key.insecure teste.reitoria.ifsertao-pe.edu.br.key

OBS: Caso não tire a senha, toda vez que reiniciar o servidor será solicitada a senha da chave privada.

2. **Gerar Certificado CSR (Certificate Signing Request)**

openssl req -nodes -newkey rsa:2048 -keyout teste.reitoria.ifsertao-pe.edu.br.key -out teste.reitoria.ifsertao-pe.edu.br.csr

OBS: Introduza a informação para a Solicitação de Assinatura de Certificado. A mesma será exibida no certificado.

Nota: Os seguintes caracteres não são aceitos: < > ~ ! @ # $ % ^ * / \ ( ) ?.,&

- Nome do País (código de 2 letras) [AU]:GB
- Estado ou Provícia (nome completo) [Algum Estado]:London
- Localidade (ex: cidade) []:London
- Nome da Organização (ex: empresa) [Internet Widgits Pty Ltd]:Global Sign
- Nome da Unidade Organizacional (ex: setor) []:IT
- Nome Popular (ex: SEU nome) []:www.globalsign.net (Precisa ser o NDTQ – Nome de Domínio Totalmente Qualificado)

Nota: Não insira os seguintes itens:

- Endereço de E-mail []:
- Uma senha difícil []:
- Um nome opcional da empresa []:

3. **Verifica se CSR as informações estejam corretas**

openssl req -noout -text -in teste.reitoria.ifsertao-pe.edu.br.csr

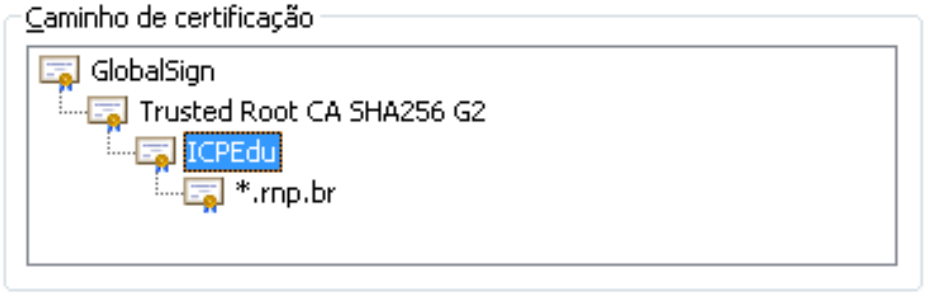
4. **Informar o conteúdo do arquivo servidorweb.csr para Gerencia de Rede da Reitoria. Informando o Domínio do servidor e a funcionalidade dele**

E-MAIL : redes@ifsertao-pe.edu.br

5 - Instalando um certificado gerado com CSR convencional

A instalação dos certificados é simples e feita com apenas alguns arquivos e comandos a mais nos arquivos de configuração do servidor web.

A cadeia de certificação de um certificado emitido pelo serviço de **AC SSL Corporativa da ICPEdu** fica da seguinte maneira:



Neste caso nosso certificado que foi emitido para o domínio ***.rnp.br** possui:

- Os certificados ***ICPEdu*** e ***Trusted Root CA SHA256 G2*** como **certificados intermediários** e
- O certificado ***GlobalSign*** como o **certificado raiz** da nossa cadeia de certificação.

Certificado	Função
gs_root.pem	**GlobalSign Root R3** - Certificado raiz da cadeia de certificação da AC SSL Corporativa da ICPEdu
intermediate.pem	**Trusted Root CA SHA256 G2** + **ICPEdu**: Arquivo com dois certificados intermediários concatenados formando um único o certificado intermediário da cadeia de certificação da AC SSL Corporativa da ICPEdu

1. Faça aqui o download do certificado raiz da GlobalSign (GlobalSign Root R3) e salve-o como **gs_root.pem**.

2. Faça aqui o download do certificado intermediário (Trusted Root CA SHA256 G2 + ICPEdu) e salve-o como **intermediate.pem**.

3. Faça o download do seu certificado emitido através da AC SSL Corporativa da ICPEdu como <**seudomínio>.crt (Enviado pela gerencia de rede da Reitoria).**

4. Salve todos os 3 arquivos juntamente com sua private key (**suachave.key**) no diretório que você planeja guardar seus certificados (neste exemplo utilizamos /etc/apache2/ssl/).

5. Abra seu arquivo de configuração do apache em um editor de textos, encontre o virtual host que você deseja configurar e complemente com os comandos abaixo:

   - **SSLCACertificateFile**: Esta diretiva deve apontar para o certificado da AC Raiz da GlobalSign.
   - **SSLCertificateChainFile**: Esta diretiva deve apontar para o certificado concatenado, contendo os certificados Trusted Root CA SHA256 G2 e o certificado da ICPEdu.
   - **SSLCertificateFile**: Esta diretiva deve apontar para o seu certificado pelo portal, através do serviço ICPEdu.
   - **SSLCertificateKeyFile**: Esta diretiva deve apontar para o seu arquivo contendo a chave privada associada ao seu certificado.

   Um exemplo de como ficaria a configuração:

   SSLCACertificateFile /etc/apache2/ssl/icpedu/gs_root.pem

   SSLCertificateChainFile /etc/apache2/ssl/icpedu/intermediate.pem

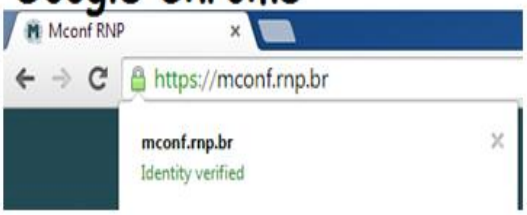
   SSLCertificateFile /etc/apache2/ssl/icpedu/seudominio.crt

   SSLCertificateKeyFile /etc/apache2/ssl/icpedu/suachave.key

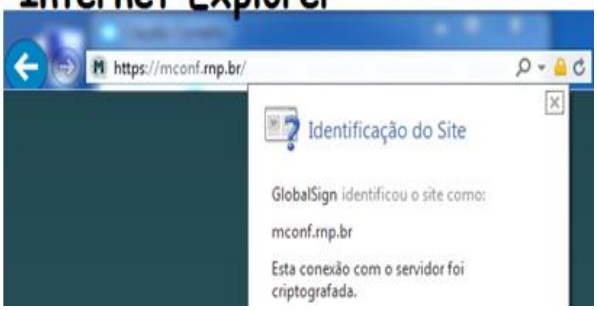
6. Reinicie o serviço do apache e faça o teste de acesso: **sudo service apache2 restart**

Mozilla Firefox

Google Chrome



Internet Explorer



Comandos openssl e certificados digitais

O que é um CSR (Certificate Signing Request)?

O CSR é um arquivo de texto criptografado, gerado pelo servidor web do seu site, contendo as informações para a solicitação do seu Certificado Digital.

O CSR contém as informações da sua empresa (nome, departamento, cidade, estado, país) e a URL onde o certificado SSL será utilizado (Common Name).

Ao gerar o CSR no seu servidor web utilize as seguintes informações:

- **Common name (server domain name):** url onde o certificado vai ser utilizado.
- **Organization:** Nome oficial da empresa, igual ao existente no cartão do CNPJ.
- **Organizational Unit:** Departamento ou setor da empresa.
- **City or Locality:** Sua cidade.
- **State or Province:** Seu estado por extenso e sem abreviações.
- **Country:** País com 2 caracteres, BR para Brasil.

OBS 1: Informações para a geração do CSR

- Ao gerar o CSR não utilize caracteres especiais, acentos e cedilha (" ' ! @ # $ % ¨ & * _ - + = § ¬ ¢ £ ³ ² ¹ ` ´ [ ] { } ( ) ª º ^ ~ ? / \ ; : . , < > |).
- Gere o CSR com uma chave de 2048 bits, para que os navegadores utilizem uma criptografia de até 256 bits.
- Na geração do CSR, o seu servidor web cria um par de chaves: uma chave pública (CSR) e outra privada. Faça uma cópia de segurança do par de chaves em local seguro.
- Configure o ítem "Common Name" com a exata url onde o certificado vai ser utilizado. Por exemplo, se deseja utilizar o certificado em https://www.teste.com.br utilize como "Common Name" www.teste.com.br Lembre-se que www.teste.com.br é diferente de teste.com.br (sem o www).

OBS 2 : CRT (Certificado de Segurança)

1. **Gerar chave privada com senha**

openssl genrsa -des3 -out teste.reitoria.ifsertao-pe.edu.br.key 2048

Descrição do comando:

- **genrsa –** opção utilizada para gerar uma chave privada;
- **-des3 –** opção utilizada para diminuir o trafego da rede;
- **-out teste.reitoria.ifsertao-pe.edu.br.key –** opção que informa o nome do arquivo gerado (“teste.reitoria.ifsertao-pe.edu.br.key”);
- **2048 –** Opção utilizada para informar o tamanho da chave privada, em bits.

OBS: Será pedida uma senha para a chave privada. Como resultado da execução do comando acima, teremos um arquivo **"teste.reitoria.ifsertao-pe.edu.br.key"** que conterá uma chave privada, criptografada com o algoritmo DES3, de 2048 bits de tamanho.

2. Criando a requisição do certificado

Comando:

openssl req -new -key ca-key.pem -out ca-csr.pem -config openssl.cnf -days 3650

Descrição do comando:

- **req –** opção para gerar certificados e requisições;
- **-new –** opção para criar um novo certificado;
- **-key ca-key.pem –** informa a chave privada utilizada para gerar a requisição;
- **-out ca-csr.pem** – diz para salvar a requisição do certificado em um arquivo "ca-csr.pem";
- **-config openssl.cnf –** informa o arquivo de configuração que conterá as informações do certificado;
- **-days 3650 –** gera a requisição com uma validade de 3650 dias;

3. Criando o certificado auto assinado

Comando:

openssl x509 -req -in ca-csr.pem -signkey ca-key.pem -out ca-cert.pem

Descrição do comando:

- x509 – opção para gerar certificados e requisições do tipo x509;
- -req – opção para gerar o certificado a partir de uma requisição;
- -in ca-csr.pem – informa o arquivo "ca-csr.pem"(requisição de certificado) utilizado como base para gerar o certificado;
- -signkey ca-key.pem – informa a chave privada "ca-key.pem" utilizada para assinar o certificado;
- -out ca-cert.pem – diz para salvar o certificado em um arquivo chamado "ca-cert.pem".

O comando acima irá criar um certificado digital, assiná-lo com a chave privada informada e armazená-lo em um arquivo chamado ca-cert.pem. Se julgar

necessário, este arquivo poderá ser renomeado para ca-cert.crt ou ca-cert.cer para melhor integração com aplicativos. No final, teremos, como principais arquivos, o certificado da Autoridade Certificadora e a sua chave privada. O arquivo ca-csr.pem não é mais necessário e pode ser apagado

Redirecionar a solicitações de http para https no Apache2

vim /etc/apache2/sites-available/default

Insira as seguintes linhas

<VirtualHost *:80>

ServerName servername.dominio.com.br

DocumentRoot /var/www/site

Linhas que vão fazer o redirecionamento RedirectPermanent e UseCanonicalName

RedirectPermanent / https://servername.dominio.com.br

UseCanonicalName Off

</VirtualHost>

Saia e salve o arquivo. Reinicie o serviço do apache2 como administrador:

/etc/init.d/apache2 restart

Comentários:

- servername-->Directive sets the hostname and port that the server uses to identify itself.
- Documentroot --> This directive sets the directory from which httpd will serve files.
- Redirectpermanent --> Sends an external permanent redirect asking the client to fetch a different URL.
- UseCanonicalname --> Apache will form self-referential URLs using the hostname and port supplied by the client if any are supplie