Tutorial para a leitura do campo subject dos certificados

Foi proposto um desafio para a vaga de estágio na empresa BRy Tecnologia. Pediu-se a leitura do campo subject dos certificados passados por e-mail. Há a presença de dois certificados em formatos distintos. O primeiro estava com a extensão .crt: “certificado-ac-raiz-bry-v3” e o segundo com a extensão .cer: “certificado-verisign”. Primeiramente precisamos que os certificados estejam em formato DER para conseguirmos utilizar as funções da biblioteca OpenSSL. Já que o primeiro certificado mencionado não estava em formato DER, foi necessário fazer a conversão do formato PEM para formato DER, adequado para a utilização das funções da biblioteca necessárias para a apresentação do campo subject dos certificados.

Desenvolvi uma aplicação C++ para a exibição do campo subject. Mas para isso foi necessário fazer a instalação da biblioteca OpenSSL. O sistema operacional utilizado foi o Ubuntu. Para a instação da biblioteca utilizou-se o comando: “sudo apt-get install libssl-dev”. Agora com a biblioteca instada, fomos para o desenvolvimento da aplicação em C++. Resumidamente nós abrimos o arquivo com a função fopen e obtemos o tamanho deste arquivo. Alocamos memória dinamicamente para o beffer a ser utilizado. Usamos a função fread para a leitura do arquivo. Esta função lê os dados do arquivo e armazena em nosso buffer, a mesma função retorna o tamanho do nosso buffer. Foi verificado se o tamanho do buffer retornado e o tamanho do nosso arquivo são iguais, como forma de tratamento de erros. Quando tentamos utilizar a função d2i_X509(NULL, &dadosCertificado,result), tivemos um problema de compilação: “undefined reference to `d2i_x509'”. Isto quer dizer que o compilador GCC não estava com o link com a biblioteca lcrypto. Toda a compilação estava sendo feita direto no terminal do Ubuntu. O link do compilador com a biblioteca lcrypto pode ser feito com este comando no terminal: g++ arquivo.cpp -o arquivo-lcrypto

Agora com o link da biblioteca funcionando, segui as instruções presentes no documento repassado por e-mail:

> Ler certificado da memória (em formato DER) para estrutura OpenSSL X509:
> X509* certificado = d2i_X509(NULL,&dadosCertificado,tamanhoCertificado);
> Obter estrutura com informações do titular do certificado (X509_NAME):
> X509_NAME* subject = X509_get_subject_name(certificado);
> Obter representação ASCII de uma estrutura X509_NAME:
> X509_NAME_oneline(subject,buffer,tamanhoBuffer);

Os tipos das variáveis adequados para cada função presente na imagem acima foi retirado no site da biblioteca OpenSSL: https://www.openssl.org/. E sua implementação de maneira adequada também foi retirado do mesmo website.

Agora vou fazer as instruções para a compilação do programa:

1- Você deve criar uma posta para colocar o seu arquivo Cpp e os certificados que serão utilizados na aplicação. Para esta aplicação você deve converter os arquivo .crt para formato DER. Depois da conversão você vai utilizar o mesmo certificado, agora, em formato DER
2- Agora você tem uma pasta e lá temos o arquivo em Cpp e os certificados a serem utilizados
3- Para você fazer a compilação do programa via terminal, você precisa primeiramente acessar o diretório que a sua aplicação Cpp se encontra.
4- Abra o terminal e digite: “cd ..”
5- Após o comando anterior digite:”cd”
6- Agora você pode listar as pastas com o comando: “ls”
7- No meu caso a minha aplicação estava da pasta Desktop, então devo digitar cd Desktop, ou se no seu caso ela estiver em outra pasta você pode digitar:”cd outra_pasta”.
8- Agora você entrou na pasta que sua aplicação está
9- Para fazer a compilação você pode digitar o seguinte comando no terminal:” g++ nome_arquivo.cpp -o nome_arquivo-lcrypto”. Este comando vai compilar seu código C++ utilizando o GCC e fazendo o link com a biblioteca lcrypto para não ter problema de referência das funções desejadas da biblioteca lcrypto.
10- Após a compilação é preciso executar o código. Você pode executar o código com o comando: “./nome_arquivo”.

Como resultado da minha aplicação, obtive o campo subject dos certificados:

impressão do subject representação ASCII
/C=US/O=VeriSign, Inc./OU=VeriSign Trust Network/OU=(c) 2006 VeriSign, Inc. - For authorized use only/CN=VeriSign Class 3 Public Primary Certification Authority - G5
impressão do subject representação ASCII
/C=BR/O=BRy Tecnologia SA/OU=Infraestrutura de Chaves Publicas BRy Tecnologia/CN=Autoridade Certificadora Raiz BRy Tecnologia v3

Referências

OpenSSL. Disponível em: < https://www.openssl.org/>. Acesso em: 11 de jul. de 2022.

Stack Overflow. Disponível em: < https://stackoverflow.com/>.Acesso em: 11 de jul. de 2022.

Wiki.OpenSSL. Disponível em: <https://wiki.openssl.org/index.php/Libcrypto_API> . Acesso em: 11 de jul. de 2022.