

Entender as informações do certificado para criar uma cadeia de WLC 9800

Contents

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Geração de CSR](#)

[Certificado de Terceiros](#)

[CA raiz decodificada](#)

[CA Intermediário Decodificado](#)

[Certificado de dispositivo decodificado](#)

Introdução

Este documento descreve como decodificar um certificado com ferramentas online conhecidas e sua interpretação para criar uma cadeia de certificados na WLC 9800.

Pré-requisitos

Requisitos

A Cisco recomenda que você tenha conhecimento básico destes tópicos:

- Controlador de LAN sem fio (WLC) do Cisco Catalyst 9800
- Conceito de certificado digital, CSR (Certificate Signing Request).
- Software OpenSSL.

Componentes Utilizados

As informações neste documento são baseadas nestas versões de software e hardware:

- Software OpenSSL na versão 1.1.1w
- computador Windows

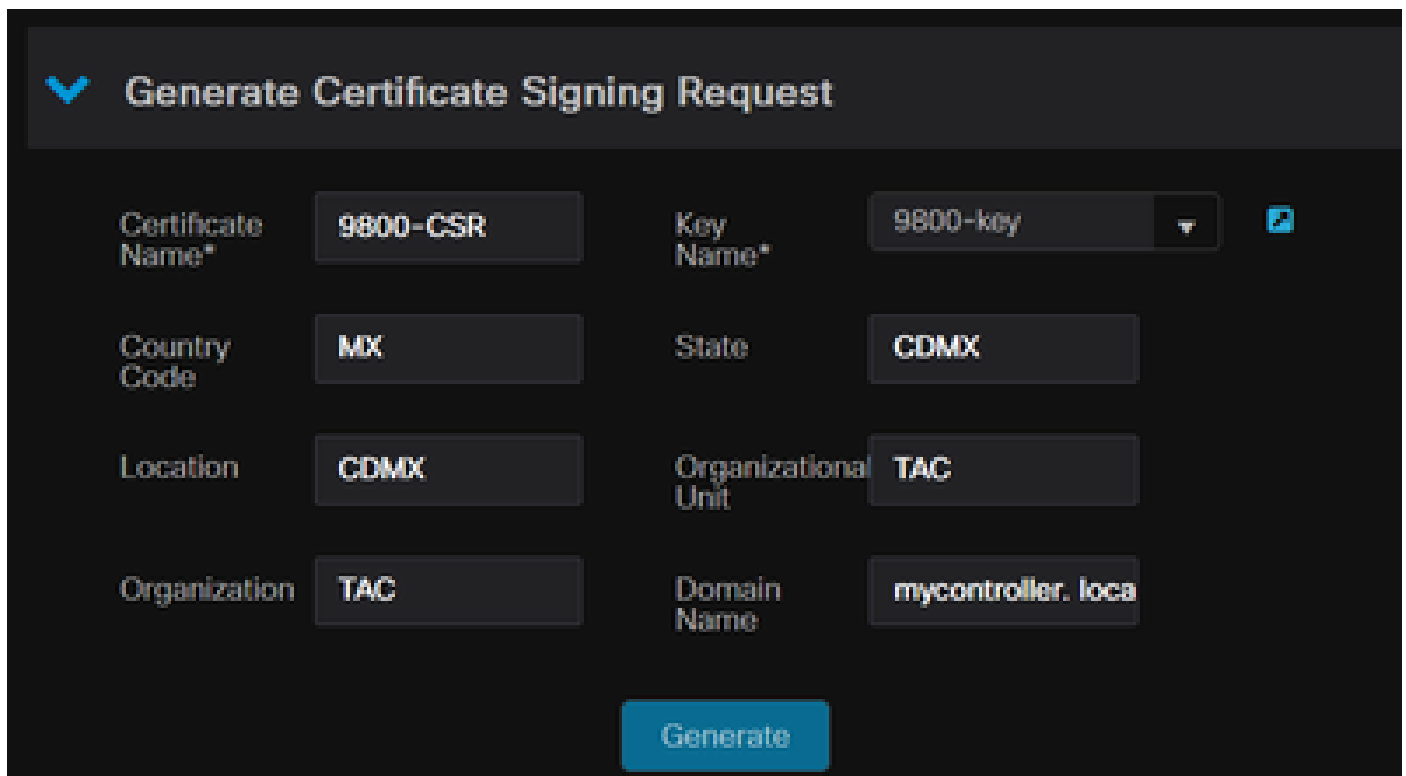
As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

Geração de CSR

O CSR pode ser gerado no controlador ou com o OpenSSL.

Para gerar um CSR no 9800 WLC, navegue para Configuration > Security > PKI Management > Add Certificate > Generate Certificate Signing Request.

Quando uma solicitação de assinatura de certificado é gerada, informações como chave privada, nome comum (CN), código do país, estado, local, organização e unidade organizacional são necessárias.



The screenshot shows a web interface for generating a Certificate Signing Request (CSR). The title is "Generate Certificate Signing Request". The form has the following fields and values:

Field	Value
Certificate Name*	9800-CSR
Key Name*	9800-key
Country Code	MX
State	CDMX
Location	CDMX
Organizational Unit	TAC
Organization	TAC
Domain Name	mycontroller.local

At the bottom of the form is a blue "Generate" button.

Geração de CSR no WLC

Todas as informações de CSR preenchidas na solicitação são exibidas na decodificação.

O software OpenSSL é a única fonte verdadeira quando um certificado é decodificado. Ele mostra todas as informações sobre ele.

Para decodificar um certificado em um computador Windows ou MacBook com OpenSSL instalado, abra o Prompt de Comando como Administrador e execute o comando `openssl x509 -in <certificate.crt> -text -noout`. A saída é mostrada como informações do console.



Observação: nem todas as versões de openssl são suportadas no 9800 WLC. As versões sugeridas são 0.9.8 e 1.1.1w

Há outras ferramentas online para decodificar certificados que mostram a saída de uma forma mais fácil de usar, como CertLogik e SSL Shopper, que não são apresentadas neste documento.

Lembre-se de que eles usam o mesmo comando OpenSSL já mencionado para decodificar os certificados.

Certificado de Terceiros

O CSR é enviado à Autoridade de Certificação (CA) para que seja assinado e devolvido. Faça o download de toda a cadeia de certificados para poder carregá-la na WLC.

Para entender a cadeia de um certificado, todos os arquivos recebidos pela CA podem ser decodificados. Verifique se eles estão no formato Base64.

Você pode receber vários arquivos da autoridade de certificação. Depende do número de arquivos CA intermediários.

Para identificar cada arquivo, você precisa decodificá-lo.

Quando um certificado assinado é decodificado, a seção Emissor é adicionada. Refere-se à CA que assinou o certificado.

Se você decodificar um arquivo CSR que não está assinado, a seção Emissor não existirá porque ele ainda não foi assinado.

Este é um exemplo de um cenário de autorização multinível ou certificado encadeado:

- CA raiz
- Certificado CA intermediário
- Certificado do dispositivo

CA raiz decodificada

Para uma CA raiz, como é a autoridade mais alta da cadeia, Emissor e Assunto devem ser iguais.

Certificate:

Data:

Version: 3 (0x2)

Serial Number:

4c:25:79:7e:57:f3:84:85:42:52:1f:c3:4b:f2:64:3f

Signature Algorithm: sha256WithRSAEncryption

Issuer: DC = com, DC = Root, CN = RootCA

Validity

Not Before: Apr 11 00:21:30 2024 GMT

Not After : Apr 11 00:31:30 2029 GMT

Subject: DC = com, DC = Root, CN = RootCA

Subject Public Key Info:

Public Key Algorithm: rsaEncryption

RSA Public-Key: (2048 bit)

Modulus:

00:a2:f5:8e:23:db:7b:09:e2:bf:c5:e0:31:a1:35:
7b:2f:f8:ed:fc:2f:4d:36:c6:b1:92:4e:80:52:6a:
1a:82:83:3f:77:06:34:ca:0f:2b:fc:ef:84:85:67:
40:de:a5:59:99:3d:d1:db:f8:ee:55:72:97:2a:bd:
7e:c5:05:c6:ec:6a:6d:00:ec:22:d5:ff:6a:cd:31:
49:a2:f0:8d:85:be:ba:e3:a0:db:31:07:e8:9c:3d:
d4:a9:ab:bc:73:90:b8:a2:ab:a2:87:0c:1d:ac:42:
f7:e4:26:49:28:18:93:a0:fd:1f:1a:7d:da:1b:e1:
60:87:dc:38:ce:b7:95:90:64:3d:2f:2b:bc:6e:d7:
2c:09:5a:54:11:dd:0e:58:63:b4:50:38:87:ea:28:
28:32:39:8c:e5:2b:b9:13:38:1f:3a:34:b9:32:33:
af:86:23:3a:40:38:fe:38:18:0c:67:a7:27:66:ab:
e3:11:66:25:f1:85:48:54:a8:05:0e:9f:02:64:09:
4f:63:be:a4:53:d5:d7:41:f0:cd:ad:b7:4c:8b:fd:
ab:a4:c7:fa:95:05:f9:ef:ed:54:ce:90:28:07:1d:
94:54:4f:bd:6c:7d:4e:a9:70:84:0b:dc:b3:73:3f:
af:d9:82:86:94:cf:29:35:53:8b:67:95:d3:00:5c:
ab:e1

CA raiz decodificada

CA Intermediário Decodificado

Para a CA intermediária, como é assinada pela CA raiz, o emissor deve corresponder ao CN da CA raiz.

Certificate:

Data:

Version: 3 (0x2)

Serial Number:

70:00:00:00:04:18:9f:53:1e:b0:cc:90:b7:00:00:00:00:00:04

Signature Algorithm: sha256WithRSAEncryption

Issuer: DC = com, DC = Root, CN = RootCA

Validity

Not Before: Apr 11 00:44:27 2024 GMT

Not After : Apr 11 00:54:27 2026 GMT

Subject: DC = com, DC = Root, CN = IntermediateCA

Subject Public Key Info:

Public Key Algorithm: rsaEncryption

RSA Public-Key: (2048 bit)

Modulus:

00:f1:c9:2b:1a:53:29:55:6d:bc:82:95:36:38:3a:
08:a4:9e:dd:81:c4:fc:0a:92:6c:2b:30:82:cd:62:
4c:91:38:ec:09:06:cc:fb:2b:f6:0f:09:43:d3:5a:
95:6a:3b:2b:4c:bc:d2:03:05:8e:0b:fd:0a:44:c2:
b8:c1:55:c0:4c:b5:d8:2d:cb:ab:4d:df:d5:d7:96:
87:21:ea:45:5b:32:db:bd:78:31:fa:5c:cb:1e:66:
62:8c:42:ff:3e:15:05:25:4e:bf:cd:5a:d7:3e:fb:
4a:2f:41:95:e0:37:f1:23:22:47:ee:7e:2e:9e:6f:
a0:24:fe:07:7d:7c:9b:cb:91:9d:05:b6:73:e4:c1:
c7:04:86:72:a4:6e:73:db:ca:1a:ee:9b:c1:0c:9a:
39:46:74:96:f8:6f:80:1e:5f:1a:cc:98:7c:91:be:
7c:98:8b:0d:08:4c:34:ab:30:9c:a0:02:0a:c4:65:
75:68:0b:f8:29:ea:92:6b:be:c6:83:19:79:fc:bd:
91:b9:f0:aa:1c:ed:fe:62:2c:27:d7:3e:8b:e3:db:
74:31:fe:a3:be:5d:8e:12:03:70:9f:f1:3c:0a:61:
e0:74:0b:08:00:1b:97:7d:01:dd:c7:24:04:7f:f6:
7e:18:e3:be:ef:a9:33:5d:47:0f:eb:52:6d:07:10:
f5:d5

CA Intermediário Decodificado

Certificado de dispositivo decodificado

Para o Certificado do Dispositivo, como é assinado pela CA Intermediária, o Emissor deve corresponder ao CN da CA Intermediária

```
Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number:
      76:00:00:00:03:65:c9:0f:4c:b8:29:d8:71:00:00:00:00:00:03
    Signature Algorithm: sha256WithRSAEncryption
    Issuer: DC = com, DC = Root, CN = IntermediateCA
    Validity
      Not Before: Apr 11 00:56:39 2024 GMT
      Not After : Apr 11 00:56:39 2025 GMT
    Subject: DC = com, DC = Root, CN = Users, CN = Administrator
    Subject Public Key Info:
      Public Key Algorithm: rsaEncryption
      RSA Public-Key: (2048 bit)
      Modulus:
        00:d6:24:8c:93:b4:44:13:48:35:94:98:1e:90:f8:
        1b:fc:18:63:df:0f:2a:05:95:38:22:7c:fc:75:69:
        8a:42:07:a8:f9:8b:5f:9f:f2:08:56:ed:d2:1a:b3:
        51:b8:d7:6b:6b:b1:13:aa:8a:ce:3f:c2:6d:cf:f1:
        98:9b:f5:45:1a:77:28:2f:63:d2:91:0c:8d:79:34:
        c2:02:f5:01:16:31:10:49:5c:51:5c:6d:2f:50:82:
        4c:b9:5a:b6:17:be:b6:1a:59:42:8c:97:3c:32:ef:
        cb:52:c7:28:f6:d0:d2:83:4b:ab:2c:5c:14:e1:6b:
        3e:a9:2c:c3:84:25:3b:24:23:d5:1a:7f:2f:42:08:
        45:ba:5b:c4:47:8d:04:52:12:1b:54:9f:9f:85:25:
        9c:ce:71:79:22:3a:19:99:1a:e4:25:9d:7f:91:f0:
        f2:4e:07:be:39:1f:9f:ed:6d:c1:28:33:66:25:54:
        91:62:0e:d3:03:19:69:cc:61:ac:a4:be:b3:ed:25:
        82:b9:77:85:71:30:f8:f7:53:a3:bd:22:a8:8f:0c:
        a7:97:d9:98:79:48:43:ed:5f:c5:c7:17:d0:cd:06:
        e8:da:d3:9b:0e:9e:04:a9:04:da:03:b3:86:96:0d:
        23:2c:3e:6d:81:04:99:38:15:c2:e9:76:da:79:41:
        db:51
```

Certificado de dispositivo decodificado

Em um cenário em que mais de 1 CA intermediário é usado, use o mesmo processo de decodificação.

Depois que a ordem de encadeamento é identificada, ela pode ser carregada no controlador.

A WLC 9800 precisa de toda a cadeia na ordem correta para que o certificado possa operar corretamente.

Para as etapas subsequentes de carregamento de um certificado para o controlador, consulte [Gerar e Download de Certificados CSR nas WLCs do Catalyst 9800](#).

Certifique-se de que você compreende o processo de decodificação antes de continuar. Em caso afirmativo, as próximas etapas precisam ser concluídas para que um certificado Web Auth, Web Admin ou Management seja carregado em uma WLC 9800.

Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.