Visão geral do Simple Certificate Enrollment Protocol

Contents

Introduction

Este documento descreve o Simple Certificate Enrollment Protocol (SCEP), um protocolo usado para a inscrição e outras operações de Public Key Infrastructure (PKI).

Informações de Apoio

O SCEP foi originalmente desenvolvido pela Cisco e está documentado em um rascunho da

Internet Engineering Task Force (IETF).

As suas principais características são:

- Modelo de solicitação/resposta baseado em HTTP (método GET; suporte opcional para método POST)
- Suporta apenas criptografia baseada em RSA
- Usa PKCS#10 como formato de solicitação de certificado
- Usa PKCS#7 para transmitir mensagens criptografadas/assinadas
- Suporta a concessão assíncrona pelo servidor, com pesquisa regular realizada pelo solicitante
- Tem suporte limitado para recuperação de lista de revogação de certificado (CRL) (o método preferido é por meio de uma consulta de ponto de distribuição de CRL (CDP), por motivos de escalabilidade)
- Não oferece suporte à revogação de certificado online (deve ser feito offline por outros meios)
- Requer o uso de um campo **senha de desafio** na Solicitação de assinatura de certificado (CSR), que deve ser compartilhado somente entre o servidor e o solicitante

A inscrição e o uso do SCEP geralmente seguem este fluxo de trabalho:

1. Obtenha uma cópia do certificado da autoridade de certificação (AC) e valide-o.
2. Gere um CSR e envie-o com segurança para a CA.
3. Pesquise o servidor SCEP para verificar se o certificado foi assinado.
4. Reinscreva-se conforme necessário para obter um novo certificado antes da expiração do certificado atual.
5. Recupere a CRL conforme necessário.

Autenticação CA

O SCEP usa o certificado CA para proteger a troca de mensagens para o CSR. Consequentemente, é necessário obter uma cópia do certificado CA. A operação **GetCACert** é usada.

Requisição

A solicitação é enviada como uma solicitação HTTP GET. Uma captura de pacote para a solicitação é semelhante a esta:

```
GET /cgi-bin/pkiclient.exe?operation=GetCACert
```

Resposta

A resposta é simplesmente o certificado CA codificado em binário (X.509). O cliente precisa validar se o certificado CA é confiável por meio de um exame da impressão digital/hash. Isso deve ser feito por meio de um método fora de banda (uma chamada telefônica para um administrador do sistema ou uma pré-configuração da impressão digital dentro do ponto de confiança).

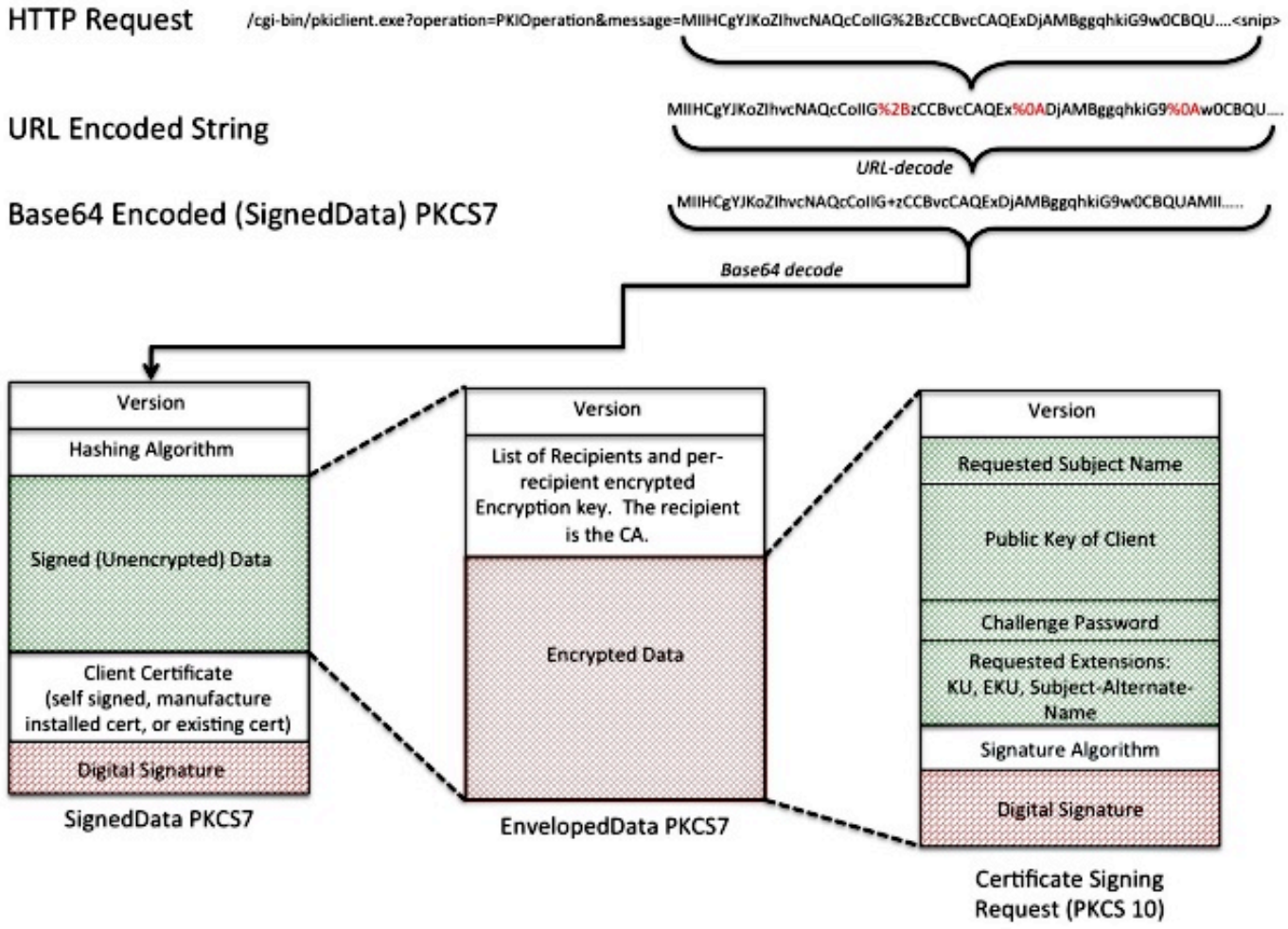
Inscrição de cliente

Requisição

A solicitação de inscrição é enviada como uma solicitação HTTP GET. Uma captura de pacote para a solicitação é semelhante a esta:

```
/cgi-bin/pkiclient.exe?operation=PKIOperation&message=
MIIHCgYJKoZIhvcNAQcCoIIG%2BzCCBvcCAQExDjA......<snip>
```

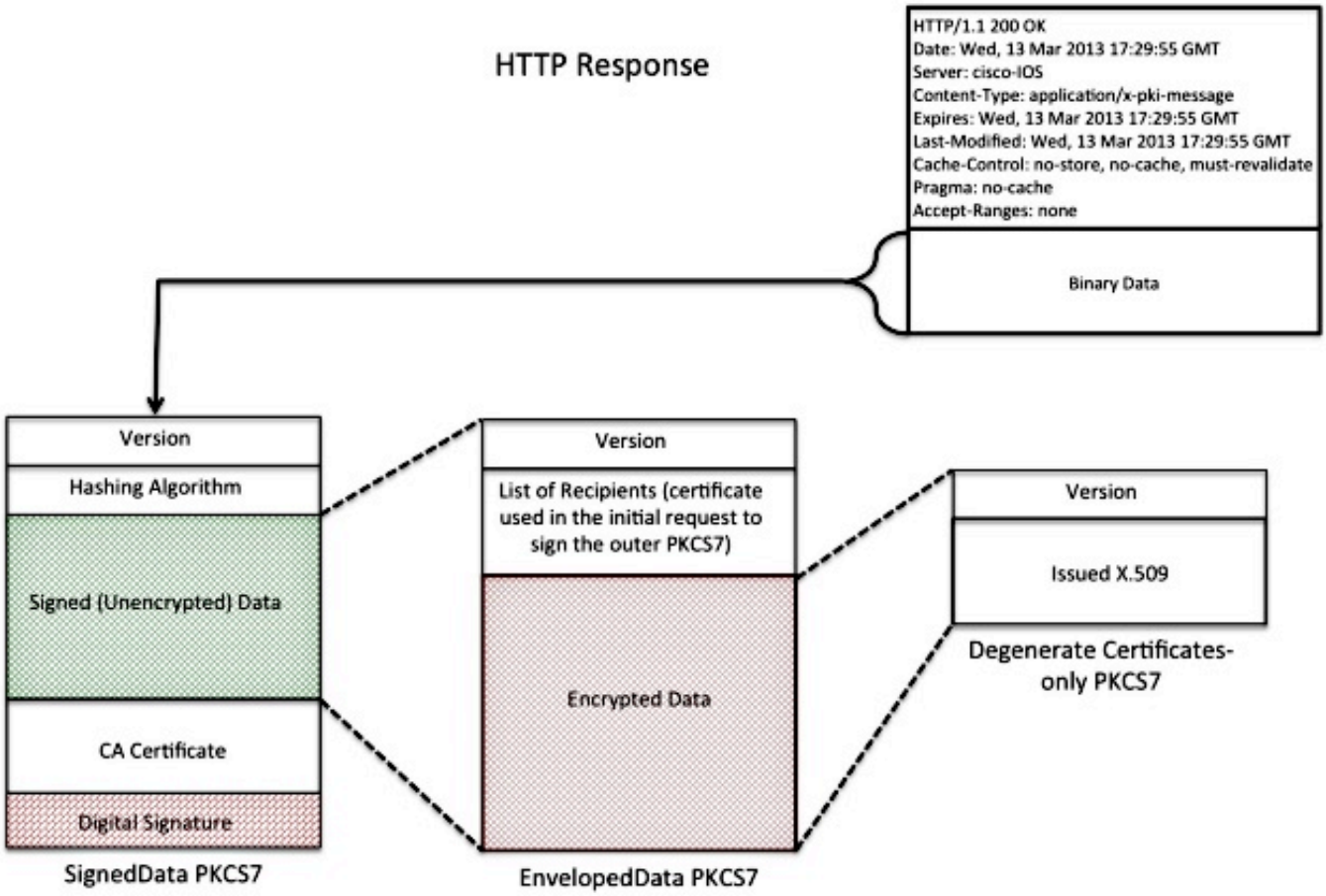
1. O texto após "message=" é uma string codificada por URL, que é extraída da string de solicitação GET.
2. O texto é então o URL decodificado em uma string de texto ASCII. Essa string de texto é um SignedData PKCS#7 codificado com base64.
3. O SignedData PKCS#7 é assinado pelo cliente com um destes certificados; é usado para provar que o cliente o enviou e que ele não foi alterado em trânsito:
   Um certificado autoassinado (usado na inscrição inicial)Um certificado instalado pelo fabricante (MIC)Uma certificação atual que expira em breve (reinscrição)
4. A parte "Dados assinados" do SignedData PKCS#7 é um EnvelopedData PKCS#7.
5. O EnvelopedData PKCS#7 é um contêiner que contém "Dados criptografados" e a "chave de descriptografia". A chave de descriptografia é criptografada com a chave pública do destinatário. Neste caso específico, o destinatário é a AC; como resultado. Somente a CA pode descriptografar "Dados criptografados".
6. A parte "Dados criptografados" do PKCS#7 envelope é o CSR (PKCS#10).



Resposta

A resposta à solicitação de inscrição do SCEP é um dos três tipos:

- **Rejeitar** - A solicitação é rejeitada pelo administrador por vários motivos, como: Tamanho da chave inválidoSenha de desafio inválidaA CA não pôde validar a solicitaçãoA solicitação solicitou atributos que a CA não autorizouA solicitação foi assinada por uma identidade na qual a CA não confia
- **Pendente** - O administrador da CA ainda não analisou a solicitação.
- **Êxito** - A solicitação é aceita e o certificado assinado é incluído. O certificado assinado é mantido dentro de um tipo especial de PKCS#7 chamado "Degenerate Certificate-Only PKCS#7", que é um contêiner especial que pode conter um ou mais X.509 ou CRLs, mas não contém um payload de dados assinados ou criptografados.



Reinscrição do cliente

Antes da expiração do certificado, o cliente precisa obter um novo certificado. Há uma pequena diferença comportamental entre renovação e rollover. A renovação ocorre quando o certificado de ID do cliente se aproxima da expiração e sua data de expiração não é a mesma (anterior) da data de expiração do certificado CA. A sobreposição ocorre quando o certificado de ID se aproxima da expiração e sua data de expiração é a mesma que a data de expiração do certificado da AC.

Renovação

Àmedida que a data de expiração de um certificado de ID se aproxima, um cliente SCEP pode

querer obter um novo certificado. O cliente gera um CSR e passa pelo processo de inscrição (como definido anteriormente). O certificado atual é usado para assinar o SignedData PKCS#7, que, por sua vez, prova a identidade para a CA. Ao receber o novo certificado, o cliente exclui imediatamente o certificado atual e o substitui pelo novo, cuja validade começa imediatamente.

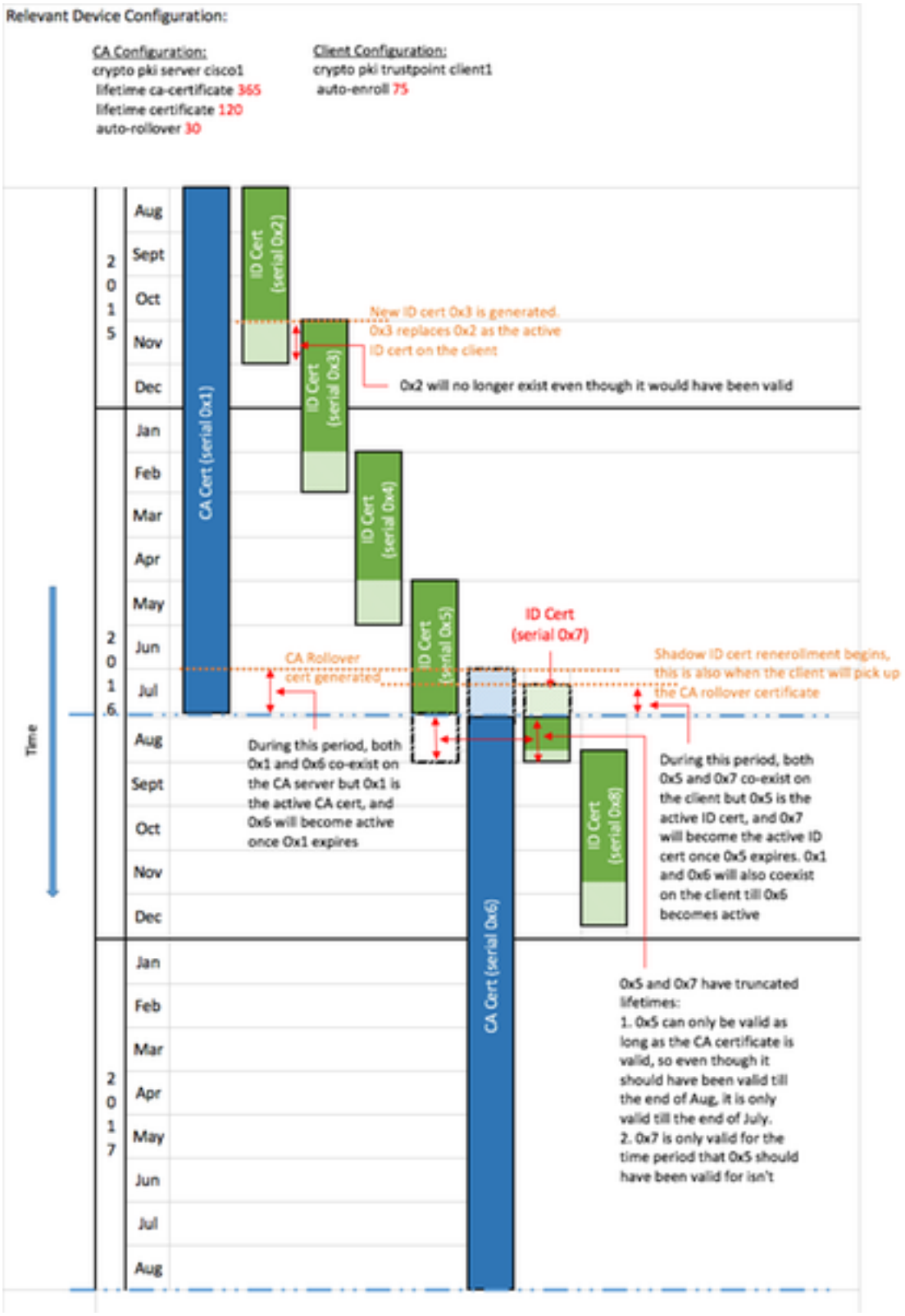
Sobreposição

Rollover é um caso especial em que o certificado CA expira e um novo certificado CA é gerado. A AC gera um novo certificado CA que se torna válido quando o certificado CA atual expira. Geralmente, a CA gera esse certificado "CA invisível" algum tempo antes do tempo de transferência, pois ele é necessário para gerar certificados "ID sombra" para os clientes.

Quando o certificado de ID do cliente SCEP se aproxima da expiração, o cliente SCEP consulta a CA para o certificado "CA sombra". Isso é feito com a operação **GetNextCACert**, como mostrado aqui:

```
GET /cgi-bin/pkiclient.exe?operation=GetNextCACert
```

Quando o cliente SCEP tiver o certificado "CA sombra", ele solicitará um certificado "ID da sombra" após o procedimento normal de inscrição. A AC assina o certificado "Shadow ID" com o certificado "Shadow CA". Ao contrário de uma solicitação de renovação normal, o certificado "Shadow ID" retornado se válido no momento da expiração do certificado CA (rollover). Como resultado, o cliente precisa manter uma cópia dos certificados anteriores e posteriores à substituição para a CA e o certificado de ID. No momento da expiração da CA (rollover), o cliente SCEP exclui o certificado CA e o certificado de ID atuais e os substitui pelas cópias "Sombra".



Blocos de construção

Essa estrutura é usada como os blocos de construção do SCEP.

Note: PKCS#7 e PKCS#10 não são específicas do SCEP.

PKCS#7

PKCS#7 é um formato de dados definido que permite que os dados sejam assinados ou criptografados. O formato de dados inclui os dados originais e os metadados associados necessários para executar a operação criptográfica.

Envelope assinado (SignedData)

O envelope assinado é um formato que transporta dados e confirma que os dados encapsulados não são alterados em trânsito através de assinaturas digitais. Inclui estas informações:

```
SignedData &colon;:= SEQUENCE {
version CMSVersion,
digestAlgorithms DigestAlgorithmIdentifiers,
encapContentInfo EncapsulatedContentInfo,
certificates [0] IMPLICIT CertificateSet OPTIONAL,
crls [1] IMPLICIT RevocationInfoChoices OPTIONAL,
signerInfos SignerInfos }
```

- Número da versão - Com o SCEP, a versão 1 é usada.
- Lista de algoritmos Digest Usados - Com o SCEP, há apenas um Assinante e, portanto, apenas um Algoritmo Hashing.
- Dados reais assinados - com o SCEP, este é um formato PKCS#7 Enveloped-data (Envelope criptografado).
- Lista de certificados dos assinantes - Com o SCEP, este é um certificado autoassinado na inscrição inicial ou o certificado atual se você se inscrever novamente.
- Lista dos assinantes e impressão digital gerada por cada signatário - Com o SCEP, há apenas um assinante.

Os dados encapsulados não estão criptografados ou ofuscados. Esse formato simplesmente fornece proteção contra a mensagem que é alterada.

Dados Envelopados (EnvelopedData)

O formato Dados Envelopados transporta dados criptografados e só podem ser descriptografados pelos destinatários especificados. Inclui estas informações:

```
EnvelopedData &colon;:= SEQUENCE {
version CMSVersion,
originatorInfo [0] IMPLICIT OriginatorInfo OPTIONAL,
recipientInfos RecipientInfos,
encryptedContentInfo EncryptedContentInfo,
unprotectedAttrs [1] IMPLICIT UnprotectedAttributes OPTIONAL }
```

- Número da versão - com SCEP, a versão 0 é usada.
- Lista de cada um dos destinatários e da chave de criptografia de dados criptografada relacionada - Com o SCEP, há apenas um destinatário (para solicitações: O servidor da AC; para respostas: o cliente).
- Os dados criptografados - são criptografados com uma chave gerada aleatoriamente (que foi criptografada com a chave pública do destinatário).

PKCS#10

PKCS#10 descreve o formato de um CSR. Um CSR contém as informações que os clientes solicitam para serem incluídos em seus certificados:

- Nome do assunto
- Uma cópia da chave pública
- Uma senha de desafio (opcional)
- Qualquer extensão de certificado solicitada, como:
  Uso de chave (KU)Uso de chave estendida (EKU)Nome alternativo do assunto (SAN)Nome Principal Universal (UPN)
- Uma impressão digital da solicitação

Aqui está um exemplo de CSR:

```
Certificate Request:
Data&colon;
Version: 0 (0x0)
Subject: CN=scepclient
Subject Public Key Info:

Public Key Algorithm: rsaEncryption Public-Key: (1024 bit)
Modulus:
00:cd:46:5b:e2:13:f9:bf:14:11:25:6d:ff:2f:43:
64:75:89:77:f6:8a:98:46:97:13:ca:50:83:bb:10:
cf:73:a4:bc:c1:b0:4b:5c:8b:58:25:38:d1:19:00:
a2:35:73:ef:9e:30:72:27:02:b1:64:41:f8:f6:94:
7b:90:c4:04:28:a1:02:c2:20:a2:14:da:b6:42:6f:
e6:cb:bb:33:c4:a3:64:de:4b:3a:7d:4c:a0:d4:e1:
b8:d8:71:cc:c7:59:89:88:43:24:f1:a4:56:66:3f:
10:25:41:69:af:e0:e2:b8:c8:a4:22:89:55:e1:cb:
00:95:31:3f:af:51:3f:53:ad
Exponent: 65537 (0x10001)
Attributes:
challengePassword :
Requested Extensions:
X509v3 Key Usage: critical
Digital Signature, Key Encipherment
X509v3 Subject Alternative Name:
DNS:webserver.example.com
Signature Algorithm: sha1WithRSAEncryption
8c:d6:4c:52:4e:c0:d0:28:ca:cf:dc:c1:67:93:aa:4a:93:d0:
d1:92:d9:66:d0:99:f5:ad:b4:79:a5:da:2d:6a:f0:39:63:8f:
e4:02:b9:bb:39:9d:a0:7a:6e:77:bf:d2:49:22:08:e2:dc:67:
ea:59:45:8f:77:45:60:62:67:64:1d:fe:c7:d6:a0:c3:06:85:
e8:f8:11:54:c5:94:9e:fd:42:69:be:e6:73:40:dc:11:a5:9a:
f5:18:a0:47:33:65:22:d3:45:9f:f0:fd:1d:f4:6f:38:75:c7:
a6:8b:3a:33:07:09:12:f3:f1:af:ba:b7:cf:a6:af:67:cf:47: 60:fc
```

Informações Relacionadas

- Rascunho IETF do SCEP
- SCEP legado usando o Guia de Configuração da CLI
- Configuração do suporte SCEP para BYOD

Appendix

Solicitações SCEP

Solicitar formato de mensagem

As solicitações são enviadas com um HTTP GET do formulário:

```
GET CGI-path/pkiclient.exe?operation=operation&message=message HTTP/version
```

Where:

- **CGI-path** depende do servidor e aponta para o programa Common Gateway Interface (CGI) que lida com solicitações SCEP: O Cisco IOS® CA usa uma string de caminho vazia.O Microsoft CA usa **/certsrv/mscep/mscep.dll**, que aponta para o serviço IIS MSCEP/ Network Device Enrollment Service (NDES).
- **Operação** identifica a operação que é executada.
- **A mensagem** transporta dados adicionais para essa operação (e pode estar vazia se não forem necessários dados reais).

Com o método GET, a parte da **mensagem** é de texto simples ou PKCS#7 codificado com Distinguished Encoding Rules (DER) convertido para Base64. Se o método POST for suportado, o conteúdo que seria enviado na codificação Base64 com GET pode ser enviado no formato binário com POST.

Visão esquemática

Valores possíveis para **operações** e seus valores **de mensagem** associados:

- **operação** = `PKIOperation`: **mensagem**é uma estrutura SCEP **pkiMessage**, baseada em PKCS#7 e codificada com DER e Base64.a estrutura **pkiMessage** pode ser destes tipos: **PKCSReq**: PKCS#10 CSR**GetCertInicial**: pesquisa para status de concessão de CSR**GetCert** ou **GetCRL**: recuperação de certificado ou CRL
- **operação** = **GetCACert**, **GetNextCACert** ou (opcional) **GetCACaps**: **a mensagem** pode ser omitida ou pode ser definida como um nome que identifica a CA.

Respostas do SCEP

Formato de mensagem de resposta

As respostas SCEP são retornadas como conteúdo HTTP padrão, com um **Content-Type** que depende da solicitação original e do tipo de dados retornados. O conteúdo DER é retornado como binário (não na base64 como a solicitação). O conteúdo do PKCS#7 pode ou não conter dados criptografados/com envelope assinado; se não o fizer (contém apenas um conjunto de certificados), será referido como **degenerate** PKCS#7.

Tipos de conteúdo

Valores possíveis para **Content-Type**:

application/x-pki-message:

- em resposta à operação **PKIOperation**, com **pkiMessage** do tipo: **PKCSReq, GetCertInitial, GetCert ou GetCRL**
- corpo da resposta é um **pkiMessage** do tipo: **CertRep**

application/x-x509-ca-cert:

- em resposta à operação **GetCACert**
- corpo da resposta é o certificado CA X.509 codificado por DER

application/x-x509-ca-ra-cert:

- em resposta à operação **GetCACert**
- corpo da resposta é um PKCS#7 degenerado codificado por DER que contém certificados CA e RA

application/x-x509-next-ca-cert:

- em resposta à operação **GetNextCACert**
- corpo da resposta é uma variação de um **pkiMessage** do tipo: **CertRep**

A estrutura de pkiMessage

OIDs SCEP

```
2.16.840.1.113733.1.9.2 scep-messageType
2.16.840.1.113733.1.9.3 scep-pkiStatus
2.16.840.1.113733.1.9.4 scep-failInfo
2.16.840.1.113733.1.9.5 scep-senderNonce
2.16.840.1.113733.1.9.6 scep-recipientNonce
2.16.840.1.113733.1.9.7 scep-transId
2.16.840.1.113733.1.9.8 scep-extensionReq
```

PkiMessage SCEP

- PKCS#7 **DadosAssinados**
- PKCS#7 EnvelopeData (chamado de **pkcsPKIEnvelope**; opcional, criptografado para o destinatário da mensagem)
  messageData (CSR, cert, CRL, ...)
- **SignerInfo** com **authenticatedAttributes**:
  IDdetransação, **messageType**, **senderNoncepkiStatus**, **recipientNonce** (somente resposta)**failInfo** (somente resposta + falha)

Tipo de mensagem SCEP

- solicitação:
  PKCSReq (19): PKCS#10 CSR**GetCertInicial** (20): pesquisa de inscrição de certificado**GetCert** (21): recuperação de certificado**GetCRL** (22): recuperação de CRL
- resposta:
  CertRep (3): resposta ao certificado ou solicitação de CRL

PkiStatus do SCEP

- **ÊXITO** (0): solicitação concedida (resposta em pkcsPKIEnvelope)
- **FALHA** (2): solicitação rejeitada (detalhes no atributo failInfo)
- **PENDENTE** (3): solicitação aguarda aprovação manual