PKI数据格式

目录

[简介](#)
[先决条件](#)
[要求](#)
[使用的组件](#)
[规则](#)
[ASN.1表示法](#)
[BER/CER/DER编码](#)
[DER十六进制转储](#)
[Base64编码](#)
[PEM编码](#)
[X.509证书和CRL](#)
[PKCS标准](#)
[相关信息](#)

简介

本文档介绍最常见的公钥基础设施(PKI)数据格式和编码。

先决条件

要求

Cisco 建议您了解以下主题：

- 公钥加密（基本概念）。
- 公钥基础设施（基本概念）。

使用的组件

本文档不限于特定的软件和硬件版本。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您使用的是真实网络，请确保您已经了解所有命令的潜在影响。

规则

有关文档规则的信息，请参阅 Cisco 技术提示规则。

ASN.1表示法

抽象语法表示法1(ASN.1)是定义数据类型和值以及这些数据类型和值如何在各种数据结构中使用和组合的正式语言。该标准的目标是定义信息的抽象语法，而不限制信息的编码方式以便传输。

以下是摘自X.509 *RFC的示例*:

```
   Version  ::=  INTEGER  {  v1(0), v2(1), v3(2)  }
CertificateSerialNumber ::= INTEGER
Validity ::= SEQUENCE {
notBefore Time,
notAfter Time }
Time ::= CHOICE {
utcTime UTCTime,
generalTime GeneralizedTime }
```

请参阅来自国际电信联盟(ITU-T)标准站点的以下文档：

- X.680 ASN.1:基本记法的规范
- X.681 ASN.1:信息对象规范
- X.682 ASN.1:约束规范
- X.683 ASN.1:ASN.1规范的参数化

ITU-T建议搜索 — 在Rec**中搜索**X.**509。或标准**,**语言**设置为**ASN.1**。

BER/CER/DER编码

ITU-T定义了一种将ASN.1中描述的数据结构编码为二进制数据的标准方法。X.690定义基本编码规则(BER)及其两个子集，规范编码规则(CER)和可分辨编码规则(DER)。 这三个字段都基**于封装在分层结构中的类型长度值数据字段**，该分层结构由**SEQUENCEs**、**SETs**和**CHOICEs**构建，具有以下不同：

- BER提供多种对同一数据进行编码的方法，这不适合加密操作。
- CER提供明确的编码，并使用不确定长度的数据，在特定情况下使用数据结束标记。
- DER提供明确的编码，并在特定情况下使用显式长度标记。
- 其中，DER是处理PKI和加密负载时通常遇到的DER。

示例：在DER中，20位值1010 1011 1100 1101 1110编码为：

- **标记**:0x03（位字符串）
- **篇幅：**0x04（字节）
- **值：**0x**04** ABCDE**0**
- **完整DER编**码：0x030404ABCDE0

前导**04**表示必须丢弃编码值的最后4位(等于后**0**位)，因为编码值不以字节边界结束。

请参阅TU-T标准站点的以下文档：

- X.690 ASN.1编码规则：基本编码规则(BER)、规范编码规则(CER)和可分辨编码规则(DER)的

规范

在Wikipedia网站中，请参阅以下文档：

- 基本编码规则
- 规范编码规则
- 可分辨编码规则

DER十六进制转储

Cisco IOS、自适应安全设备(ASA)和其他设备使用show running-config命令将DER内容显示为十六进制转储内容。 以下是输出：

```
crypto pki certificate chain root
certificate ca 01
30820213 3082017C A0030201 02020101 300D0609 2A864886 F70D0101 04050030
1D310C30 0A060355 040B1303 54414331 0D300B06 03550403 1304726F 6F74301E
170D3039 30373235 31313436 33325A17 0D313230 37323431 31343633 325A301D
...
```

这种十六进制转储可以通过各种方式转换回DER。例如，可以删除空格字符并将其输入xxd程序：

```
$ cat ca.hex | tr -d ' ' | xxd -r -p -c 32 | openssl x509 -inform der -text -noout
```

另一种简单的方法是使用此Perl脚本：

```
#!/usr/bin/perl
foreach (<>) {
s/[^a-fA-F0-9]//g;
print join("", pack("H*", $_));
}

$ perl hex2der.pl < hex-file.txt > der-file.der
```

此外，转换证书转储的精简方法，每个证书转储之前都从bash命令行手动复制到扩展名为.hex的文件，如下所示：

```
for hex in *.hex; do
b="${hex%.hex}"
hex2der.pl < "$hex" > "$b".der
openssl x509 -inform der -in "$b".der > "$b".pem
openssl x509 -in "$b".pem -text -noout > "$b".txt
done
```

每个文件都会产生：

- file.hex — 原始文件（必须仅包含十六进制数字）。
- file.der - DER（二进制）格式的证书。
- file.pem - PEM（Base64 +页眉/页脚）格式的证书。
- file.txt -用户友好、可读的证书版本。

Base64编码

Base64编码表示二进制数据，仅包含64个可打印字(A-Za-z0-9+/)uuencode编码。在从二进制到

Base64的转换中，原始数据的每6位块都编码为一个带转换表的8位可打印ASCII字符。因此，编码后的数据大小增加了33%（数据乘以8除以6位，等于1.333）。

24位缓冲区用于将三(3)组八(8)位转换为四(4)组六(6)位。因此，在输入数据流的末尾可能需要一(1)或两(2)个字节的填充。填充位在Base64编码数据的末尾表示，在编码期间，每组八(8)个填充位个等于(=)号表示。

请参阅Wikipedia中的此示例。

请参阅RFC 4648中的最新信息：Base16、Base32和Base64数据编码。

PEM编码

隐私增强邮件(PEM)是用于交换安全邮件的完整互联网工程任务组(IETF)PKI标准。它不再被广泛使用，但其封装语法被广泛借用，以格式化和交换Base64编码的PKI相关数据。

PEM RFC 1421,第4.4节：封装机制，定义PEM消息，其格式为：封装边界(EB)（基于RFC 934），由封装边界(EB)分隔。

```
-----BEGIN PRIVACY-ENHANCED MESSAGE-----
Header: value
Header: value
...

Base64-encoded data
...
-----END PRIVACY-ENHANCED MESSAGE-----
```

在现实中，当分发PEM格式的数据时，会使用以下边界格式：

```
-----BEGIN type-----
...
-----END type-----
```

类型可与其他密钥或证书一起使用，例如：

- RSA
-
-
-
- X509 CRL

注意：虽然RFC并未强制要求执行此操作，但EB中前导破折号和尾随破折号(-)的数量非常重要，应始终为五(5)。 否则，某些应用（如OpenSSL）会阻塞输入。另一方面，Cisco IOS等其他应用根本不需要EB。

有关详细信息，请参阅以下最新的RFC:

- RFC 1421：PEM第I部分：消息加密和身份验证过程
- RFC 1422：PEM第II部分：基于证书的密钥管理
- RFC 1423：PEM第III部分：算法、模式和标识符
- RFC 1424：PEM第IV部分：关键认证和相关服务

X.509证书和CRL

X.509是X.500的子集，是关于开放系统互连的扩展ITU规范。它专门处理证书和公钥，并已被IETF改为Internet标准。X.509提供了结构和语法，在RFC中使用ASN.1表示法表示，以存储证书信息和证书撤销列表。

在X.509 PKI中，CA颁发绑定公钥的证书，例如：特定可分辨名称(DN)的Rivest-Shamir-Adleman(RSA)或数字签名算法(DSA)密钥，或替代名称(如电子邮件地址或完全限定域名(FQDN))的密钥。 DN遵循X.500标准中的结构。示例如下：

```
CN=common-nameOU=organizational-unitO=organizationL=locationC=country
```

由于ASN.1定义，X.509数据可以编码为DER以便以二进制形式交换，或者，可以转换为Base64/PEM以用于基于文本的通信方式，例如在终端上复制粘贴。

- 请参阅本ITU-T标准文档X.509开放系统互连 — 目录：公钥和属性证书框架。
- 参阅 RFC 5280：X.509证书和证书撤销列表(CRL)配置文件(了解详细信息)。

PKCS标准

公钥加密标准(PKCS)是RSA实验室的规格，已部分发展为行业标准。那些最常遇到的问题，处理这些话题；但是，并非所有数据都处理数据格式。

PKCS#1(RFC 3347) — 涵盖基于RSA的加密（加密原语、加密/签名方案、ASN.1语法）的实施方面。

PKCS#5(RFC 2898) — 涵盖基于密码的密钥派生。

PKCS#7(RFC 2315)和S/MIME RFC 3852 **-** 定义用于传输签名和加密数据及相关证书的消息语法。通常仅用作X.509证书的容器。

PKCS#8 — 定义用于传输明文或加密RSA密钥对的消息语法。

PKCS#9(RFC 2985) — 定义其他对象类和身份属性。

PKCS#10(RFC 2986) — 定义证书签名请求(CSR)的消息语法。CSR由实体发送到CA，并包含由CA签名的信息，如公钥信息、身份和其他属性。

PKCS#12 — 定义一个容器，用于将相关PKI数据(通常为实体密钥对 + **实体证书** + **根和中间CA证书**)打包到单个文件中。这是微软个人信息交换(PFX)格式的发展。

请参阅以下资源：
- PKCS上的维基百科条目
- PKCS上的“RSA实验”页

相关信息

- **技术支持和文档 - Cisco Systems**