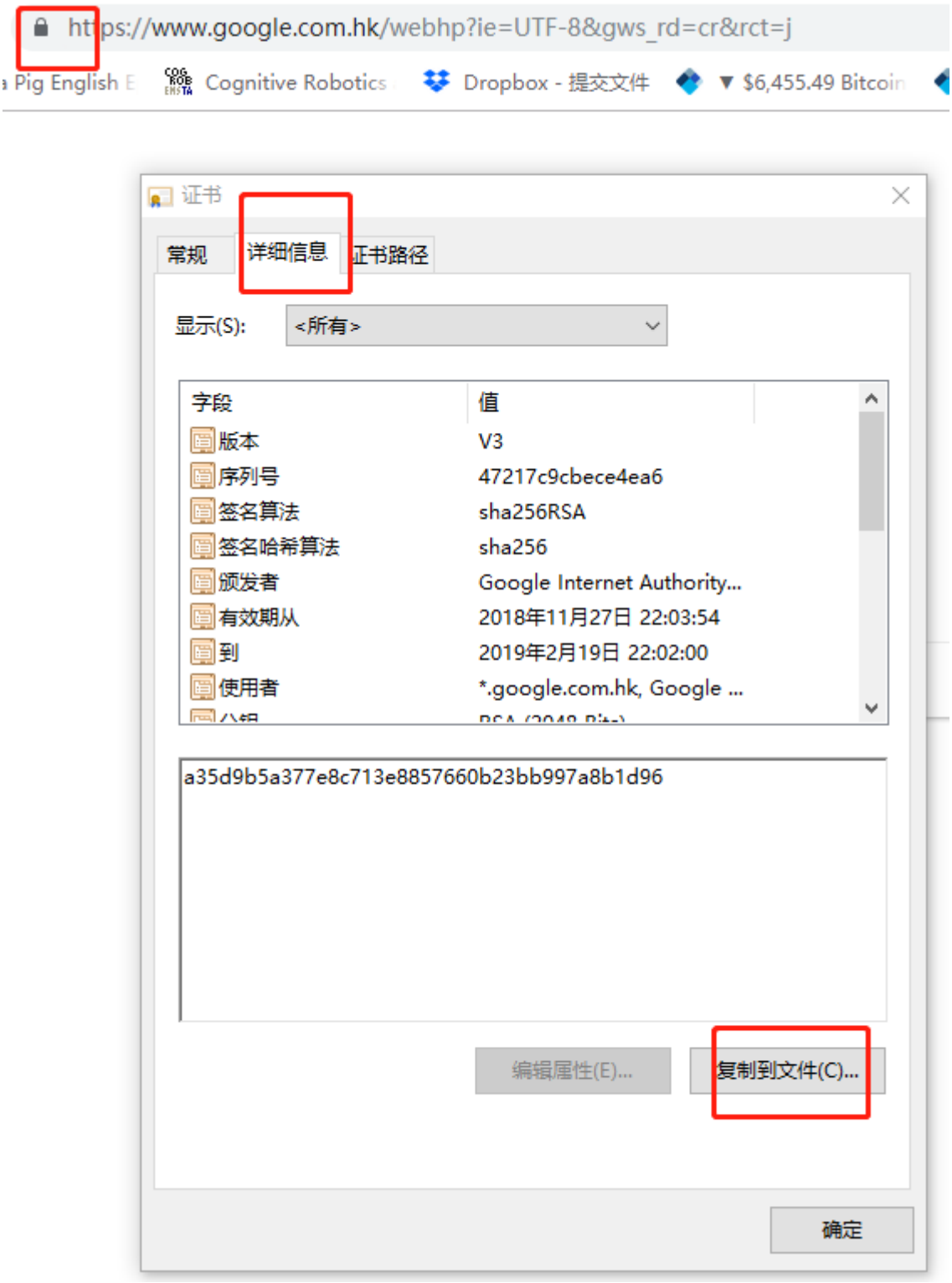
X.509 数字证书

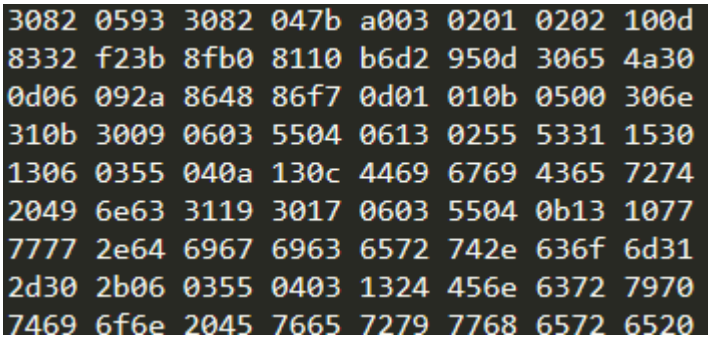
证书内容

版本、序列号、签名算法、签名哈希算法、颁发者、有效期、使用者、公钥、公钥参数、增强型密钥用法、使用者可选名称、授权信息访问、使用者密钥标识符、授权密钥标识符、证书策略、CRL分发点、基本约束、指纹

获取证书


https://www.google.com.hk/webhp?ie=UTF-8&gws_rd=cr&rct=j
Pig English
Cognitive Robotics
Dropbox - 提交文件
$6,455.49 Bitcoin
证书
常规
详细信息
证书路径
显示(S):
<所有>
字段
值
版本
V3
序列号
47217c9cbece4ea6
签名算法
sha256RSA
签名哈希算法
sha256
颁发者
Google Internet Authority...
有效期从
2018年11月27日 22:03:54
到
2019年2月19日 22:02:00
使用者
*.google.com.hk, Google ...
a35d9b5a377e8c713e8857660b23bb997a8b1d96
编辑属性(E)...
复制到文件(C)...
确定

证书的二进制文件解析



以上是文件的部分截图，文件的存储方式为：

> 变量类型 内容长度 存储内容

以上图的 `3082 0593` 为例，30 表示 sequence 类型， 30后面为存储长度，当长度大于 127 时，则使用‘8’，‘2’为需要使用2个字节表示内容长度，即 "0593"为内容长度，转化成 十进制为：1427，所以接下来的1427个字节均为该变量的内容。

一些常见的类型及其对应的16进制码：

```
sequence 30
version a0
integer 02
set     31
PrintableString us 13
true     01
octet string 04
bit string 03
```

数据结构：

```
struct signatureValue
{
  //使用sha1DSA算法时，签名值
  Dss-Sig-Value
    {
      int r;
      int s;
    }
    name
    {
        sequence    RDNSquence;
        set         RelativeDistinguishedName
        AttributeTypeAndValue             //属性类型及属性值
        {
            object     AttributeType;         //属性类型
            anytype    AttributeValue ;        //属性值
        }
    }
```

```
};

AlgorithmIdentifier signatureAlgorithm          //签名算法

struct tbsCertificate
{
  int version                          //版本号
  int serialNumber                     //序列号
  signature
  {
    algorithm                           //object
    parameters                          //defined by alogorithm
      {                                 //DSA(DSS)才有，RSA没有
        int p;
        int q;
        int g;
      }
  }
  validity                              //有效日期范围
    {
      notBefore:utcTime,generalTime;
      notAfter:utcTime,generalTime;
    }
  subject                               //主体名
  subjectPublicKeyInfo                   //subject 的公钥信息
  {
      AlgorithmIdentifier    algorithm;   //公钥算法
      BIT STRING subjectPublicKey;                  // 公钥值
  }
  bit string issuerUniqueID              // 发行者 ID
  bit string subjectUniqueID             // 使用者ID
  extensions                             //拓展部分
    {
      extnID                             //object 扩展元素的OID
      bool critical ;                    //标识这个扩展元素是否重要
      extnValue                          //8进制string
    }
};
struct Certificate
{
    tbsCertificate       TBSCertificate;      // 证书主体
    signatureAlgorithm   AlgorithmIdentifier; // 证书签名算法标识
};
```

代码实现：

本例使用Java实现

获取X.509实例、读入证书并创建证书实例

```
CertificateFactory cFactory = CertificateFactory.getInstance("X.509");
//读入cer文件
FileInputStream inputStream = new FileInputStream("G:\\githubCode\\java code\\x509\\src\\x509\\x509.cer")
Certificate certificate = cFactory.generateCertificate(inputStream);
//创建x509证书实例
X509Certificate x509Certificate = (X509Certificate)certificate;
```

获取并输出公钥编码：

```
//获取公钥编码：
byte[] pkBytes = x509Certificate.getPublicKey().getEncoded();
String pkey = x509Certificate.getPublicKey().toString();
System.out.println("Public Key:");
System.out.print(pkey.substring(0, 30));
System.out.print(pkey.substring(30,41));
for(int i = 42;i < pkBytes.length - 25;i++)
    System.out.print(pkBytes[i] + ",");
System.out.println(pkey.substring(pkey.length() - 25,pkey.length()));
```

输出：

```
Version : 3
Serial Number : 179611881458008052915774848837888850506
Signature Algorithm : SHA256withRSA
Issuer : CN=Encryption Everywhere DV TLS CA - G1, OU=www.digicert.com, O=DigiCert Inc, C=US
Validate:
Not before: Fri Mar 23 08:00:00 CST 2018
Not after: Sat Mar 23 20:00:00 CST 2019
Subject: CN=*.cnblogs.com
Public Key:
Sun RSA public key, 2048 bits
  modulus: -63,18,59,86,-76,99,66,-30,29,54,-30,-109,-12,117,-52,14,109,35,14,-57,44,-59,-76,-110,-88,-116,-56,-106,79,-60,15,38,110,-52,17,-17,58,-56,87,27,84,-38,122,22
  public exponent: 65537
Subject name: CN=*.cnblogs.com
Issuer: CN=Encryption Everywhere DV TLS CA - G1, OU=www.digicert.com, O=DigiCert Inc, C=US
Issuer Unique Identifier: null
Subject Unique Identifier: null
```