DNS对接说明

公钥文件内容 public.key

```
-----BEGIN PUBLIC KEY-----
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCMFpufLquk70Sxz/ivAt3rZlp0
MIzrwyESd7pDC0frT2jgr5qMDXVvBhm1glI08o9SGCF7xIgiPgaula+k00C3jJx+
fgAoPrp+Pvwv207JNKq1ThCp5z8XSUiDKHWHXww0Ad4vR4R2GNdXXjrRzcnefho8
0dPQwfr0MAwDNF+PPwIDAQAB
-----END PUBLIC KEY-----
```

私钥文件private.pem PKCS1版本内容

```
-----BEGIN RSA PRIVATE KEY-----
MIICXAIBAAKBgQCMFpufLquk70Sxz/ivAt3rZlp0MIzrwyESd7pDC0frT2jgr5qM
DXVvBhm1glI08o9SGCF7xIgiPgaula+k00C3jJx+fgAoPrp+Pvwv207JNKq1ThCp
5z8XSUiDKHWHXww0Ad4vR4R2GNdXXjrRzcnefho80dPQwfr0MAwDNF+PPwIDAQAB
AoGABG4YmMWYjvw0D210v1AXZYDUnnUEl9YRXyyjGsiqCxJ1ThmviPtJ+etW98r5
v4vmJWH2+RuDdJXCj1XTeqJlmvxYx+ixiIszW2BX4Ld5oyzhYAv+610kedRIvpDG
L1AucZr/TiQU/37l8HP/nhODT1ik/fHSbI3gdOIdo3uYCQUCQQDLag4Hy/1f1B9d
80Y66mz5DQfk3uayZKij7ISpGETzKTVLAJ0kcJdKwWgdH26hkXmQNQW9zMUC6rUq
ljOV50yTAkEAsE2k9j3le+i1QheK5zA3HLqnAPIapxHGEU8K+JIN8ON0ISrg9Uuh
jP9ZLrGFAHIYuL0u00GcWpOkTPoXW1VKJQJBAKOIj98U0I7KKq2Nd1jGPvW61C3c
RfwFkM64x65qJISZDI9P3wX8vqUK+HjxXC/olKKq/gKpLXo614t1VeOeIKECQASv
q6jX8Feg0eCV251VjMF4vKOry8WRFWC0ZyhASZqTkSyohR5ACmVDDE3Pbiea4MlY
TfxTjCJkNkPNif1lgPkCQG9iUaZvmeiCrWHEBaVAx1tb00FTpLZ1NG28HayXugiw
43YvidGk0nZkQOxE78bpkJ1FmA8xMd5n69t5eB479nc=
-----END RSA PRIVATE KEY-----
```

私钥文件pkcs8.pem PKCS8版本内容

```
-----BEGIN PRIVATE KEY-----
MIICdgIBADANBgkqhkiG9w0BAQEFAASCAmAwggJcAgEAAoGBAIwWm58uq6TvRLHP
+K8C3etmWnQwjOvDIRJ3ukMLR+tPaOCvmowNdW8GGbWCUjTyj1IYIXvEiCI+Bq6V
r6TTQLeMnH5+ACg+un4+/C/bTsk0qrVOEKnnPxdJSIModYdfDDQB3i9HhHYY11de
```

```
OtHNyd5+GjzR09DB+vQwDAM0X48/AgMBAAECgYAEbhiYxZiO/DQPbXS/UBdlgNSe
dQSX1hFfLKMayKoLEnVOGa+I+0n561b3yvm/i+YlYfb5G4N0lcKPVdN6omWa/FjH
6LGIizNbYFfgt3mjLOFgC/7rXSR51Ei+kMYvUC5xmv9OJBT/fuXwc/+eE4NPWKT9
8dJsjeB04h2je5gJBQJBAMtqDgfL/V/UH13zRjrqbPkNB+Te5rJkqKPshKkYRPMp
NUsAnSRwl0rBaB0fbqGReZA1Bb3MxQLqtSqWM5XnTJMCQQCwTaT2PeV76LVCF4rn
MDccuqcA8hqnEcYRTwr4kg3w43QhKuD1S6GM/1kusYUAchi4vS7TQZxak6RM+hdb
VUolAkEAo4iP3xTQjsoqrY13WMY+9brULdxF/AWQzrjHrmokhJkMj0/fBfy+pQr4
ePFcL+iUoqr+AqktejrXi3VV454goQJABK+rqNfwV6DR4JXbnVWMwXi8o6vLxZEV
YLRnKEBJmpORLKiFHkAKZUMMTc9uJ5rgyVhN/FOMImQ2Q82J/WWA+QJAb2JRpm+Z
6IKtYcQFpUDHW1vTQVOktnU0bbwdrJe6CLDjdi+J0aTSdmRA7ETvxumQnUWYDzEx
3mfr23l4Hjv2dw==
-----END PRIVATE KEY-----
```

此处使用PKCS8 版本密钥的原因，是因为提供的PKCS1版本私钥，Java程序无法进行加载和处理，需要转换成PKCS8版本才可使用。转换的命令为 `openssl pkcs8 -topk8 -inform PEM -in private.pem -outform pem -nocrypt -out pkcs8.pem` ，从PKCS8转回PKCS1版本的命令为 `openssl rsa -in pkcs8.pem -out pkcs1.pem` ，参考地址：https://www.cnblogs.com/cocoajin/p/10510574.html

Java版本私钥加密公钥解密实例程序

程序使用中，依赖于hutool组件提供的加解密程序。

```
package com.sschen;

import cn.hutool.core.util.CharsetUtil;
import cn.hutool.core.util.HexUtil;
import cn.hutool.core.util.StrUtil;
import cn.hutool.crypto.SecureUtil;
import cn.hutool.crypto.asymmetric.KeyType;
import cn.hutool.crypto.asymmetric.RSA;

/**
 * Hello world!
 *
 */
public class App
{
    private static final String publicKey =
"MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCMFpufLquk70Sxz/ivAt3rZlp0" +
            "MIzrwyESd7pDC0frT2jgr5qMDXVvBhm1glI08o9SGCF7xIgiPgaula+k00C3jJx+" +
            "fgAoPrp+Pvwv207JNKq1ThCp5z8XSUiDKHWHXww0Ad4vR4R2GNdXXjrRzcnefho8" +
            "0dPQwfr0MAwDNF+PPwIDAQAB";
```

```
    private static final String privateKey =
"MIICdgIBADANBgkqhkiG9w0BAQEFAASCAmAwggJcAgEAAoGBAIwWm58uq6TvRLHP" +
            "+K8C3etmWnQwjOvDIRJ3ukMLR+tPaOCvmowNdW8GGbWCUjTyj1IYIXvEiCI+Bq6V" +
            "r6TTQLeMnH5+ACg+un4+/C/bTsk0qrVOEKnnPxdJSIModYdfDDQB3i9HhHYY11de" +
            "OtHNyd5+GjzR09DB+vQwDAM0X48/AgMBAAECgYAEbhiYxZiO/DQPbXS/UBdlgNSe" +
            "dQSX1hFfLKMayKoLEnVOGa+I+0n561b3yvm/i+YlYfb5G4N0lcKPVdN6omWa/FjH" +
            "6LGIizNbYFfgt3mjLOFgC/7rXSR51Ei+kMYvUC5xmv9OJBT/fuXwc/+eE4NPWKT9" +
            "8dJsjeB04h2je5gJBQJBAMtqDgfL/V/UH13zRjrqbPkNB+Te5rJkqKPshKkYRPMp" +
            "NUsAnSRwl0rBaB0fbqGReZA1Bb3MxQLqtSqWM5XnTJMCQQCwTaT2PeV76LVCF4rn" +
            "MDccuqcA8hqnEcYRTwr4kg3w43QhKuD1S6GM/1kusYUAchi4vS7TQZxak6RM+hdb" +
            "VUolAkEAo4iP3xTQjsoqrY13WMY+9brULdxF/AWQzrjHrmokhJkMj0/fBfy+pQr4" +
            "ePFcL+iUoqr+AqktejrXi3VV454goQJABK+rqNfwV6DR4JXbnVWMwXi8o6vLxZEV" +
            "YLRnKEBJmpORLKiFHkAKZUMMTc9uJ5rgyVhN/FOMImQ2Q82J/WWA+QJAb2JRpm+Z" +
            "6IKtYcQFpUDHW1vTQVOktnU0bbwdrJe6CLDjdi+J0aTSdmRA7ETvxumQnUWYDzEx" +
            "3mfr23l4Hjv2dw==";
    public static void main( String[] args )
    {
        RSA rsa = new RSA(privateKey, publicKey);
        String md5Str = SecureUtil.md5("{\"code\":\"0\",\"domainList\":
[{\"name\":\"dev-api-im.raymannet.com\",\"ips\":
[\"10.3.0.231\",\"10.3.0.243\"],\"timeout\":600},{\"name\":\"dev-
dns.leimans.com\",\"ips\":[\"10.3.0.231\",\"10.3.0.243\"],\"timeout\":600},
{\"name\":\"dev-file-im.raymannet.com\",\"ips\":
[\"10.3.0.231\",\"10.3.0.243\"],\"timeout\":600},{\"name\":\"dev-longlink-
im.raymannet.com\",\"ips\":[\"10.3.0.231\",\"10.3.0.243\"],\"timeout\":600},
{\"name\":\"dev-manage-im.raymannet.com\",\"ips\":
[\"10.3.0.231\",\"10.3.0.243\"],\"timeout\":600},{\"name\":\"dev-
manager.leimans.com\",\"ips\":
[\"10.3.0.231\",\"10.3.0.243\"],\"timeout\":600}],\"builtinIpList\":
[\"10.3.0.231\",\"10.3.0.243\"],\"computeTime\":1611048777052}");
        System.out.println(md5Str);
        byte[] encrypt = rsa.encrypt(StrUtil.bytes(md5Str,
CharsetUtil.CHARSET_UTF_8), KeyType.PrivateKey);
        System.out.println(HexUtil.encodeHex(encrypt));

        byte[] decrypt = rsa.decrypt(encrypt, KeyType.PublicKey);
        System.out.println(StrUtil.str(decrypt, CharsetUtil.CHARSET_UTF_8));
    }
}
```

DNS加密过程说明

DNS最终返回结果


```
{
    "code": 0,
    "codeMsg": "success",
    "data": {
        "domainCheckStr": "{\"code\":\"0\",\"domainList\":[{\"name\":\"dev-api-
im.raymannet.com\",\"ips\":[\"10.3.0.231\",\"10.3.0.243\"],\"timeout\":600},
{\"name\":\"dev-dns.leimans.com\",\"ips\":
[\"10.3.0.231\",\"10.3.0.243\"],\"timeout\":600},{\"name\":\"dev-file-
im.raymannet.com\",\"ips\":[\"10.3.0.231\",\"10.3.0.243\"],\"timeout\":600},
{\"name\":\"dev-longlink-im.raymannet.com\",\"ips\":
[\"10.3.0.231\",\"10.3.0.243\"],\"timeout\":600},{\"name\":\"dev-manage-
im.raymannet.com\",\"ips\":[\"10.3.0.231\",\"10.3.0.243\"],\"timeout\":600},
{\"name\":\"dev-manager.leimans.com\",\"ips\":
[\"10.3.0.231\",\"10.3.0.243\"],\"timeout\":600}],\"builtinIpList\":
[\"10.3.0.231\",\"10.3.0.243\"],\"computeTime\":1611048777052}",
        "clientIp": "0:0:0:0:0:0:0:1",
        "timestamp": 1611048777510,
        "effectTime": 1611135177510,
        "signature":
"c43034ab2626baaa6e5e7068ec1cc16f5acae82c38277de057e1ab6cf02c264f1bfe1c769955ecbd1
d0233ce7259448f56f6f2fad0b021351d67486e41e43eac8f5dddd50d9bfd313a86063932cfa082637
e2e8803ce9de96a538f3b33d1a6fd4573371fba7601733ab1226085d349f5fd63f75ecdec6523d7cf7
38f2e45d69ec4bb3eba13f2ef0bb6c6e4f9bce4d4f8400b6e1878b9df95c2bb4ad46b615c330003c0d
7aa419769c517777f0cf5cea2ba1a17ec8425dc440864b240b85b31d80c4f3f1812abb9e8bbf5fec45
4f1aba238bffefa87c3febd606a47911286b9d304eac820aae2a0728ef12039c8329c54448f175ae48
c876b18478edb9f3eecfb"
    }
}
```


其中

- `domainlist` 部分为除了DNS当前域名之外的其他域名及IP地址的对应关系
- `builtiniplist` 为DNS当前域名对应的IP地址，
- `computeTime` 为XML计算的时间，毫秒级
- `clientip` 本次请求的客户端地址
- `timestamp` 本次请求的时间戳，毫秒级
- `effecttime` 本次请求的数据过期时间，毫秒级，默认为24小时有效
- `signature` 计算结果私钥加密摘要结果

signature 校验字符串计算过程

使用 `domainCheckStr` 内容进行MD5摘要计算，加密内容如下：

```
{"code":"0","domainList":[{"name":"dev-api-im.raymannet.com","ips":
["10.3.0.231","10.3.0.243"],"timeout":600},{"name":"dev-dns.leimans.com","ips":
["10.3.0.231","10.3.0.243"],"timeout":600},{"name":"dev-file-
im.raymannet.com","ips":["10.3.0.231","10.3.0.243"],"timeout":600},{"name":"dev-
longlink-im.raymannet.com","ips":["10.3.0.231","10.3.0.243"],"timeout":600},
{"name":"dev-manage-im.raymannet.com","ips":
["10.3.0.231","10.3.0.243"],"timeout":600},{"name":"dev-
manager.leimans.com","ips":
["10.3.0.231","10.3.0.243"],"timeout":600}],"builtinIpList":
["10.3.0.231","10.3.0.243"],"computeTime":1611048777052}
```

转小写后得到MD5值，此处为：`429a61b388d38a81b0df7835d91903eb` 。

将计算得到的MD5字符串进行私钥加密，并将加密完毕后的字节数组转为16进制字符串，得到加密结果：

```
c43034ab2626baaa6e5e7068ec1cc16f5acae82c38277de057e1ab6cf02c264f1bfe1c769955ecbd1d
0233ce7259448f56f6f2fad0b021351d67486e41e43eac8f5dddd50d9bfd313a86063932cfa082637e
2e8803ce9de96a538f3b33d1a6fd4573371fba7601733ab1226085d349f5fd63f75ecdec6523d7cf73
8f2e45d69ec4bb3eba13f2ef0bb6c6e4f9bce4d4f8400b6e1878b9df95c2bb4ad46b615c330003c0d7
aa419769c517777f0cf5cea2ba1a17ec8425dc440864b240b85b31d80c4f3f1812abb9e8bbf5fec454
f1aba238bffefa87c3febd606a47911286b9d304eac820aae2a0728ef12039c8329c54448f175ae48c
876b18478edb9f3eecfb
```

是否被篡改的验证思路

使用 `domainCheckStr` 内容进行MD5摘要计算，计算前去除换行符及多余空格，得到的MD5值，转小写。

使用公钥解密 `signature` 节点值，将解密得到的结果同计算出的MD5进行比较，如果值相同，则未修改，否则IP地址部分被篡改。