文档背景

本文档通过分析Fabric中使用的crypto/x509包中的函数(方法)，为Fabric国密改造进一步抽象成接口做分析准备。

Crypto/x509

Fabric2.2.0代码是使用到crypto/x509包中的函数有如下几个

```
NewCertPool()
IsEncryptedPEMBlock()
DecryptPEMBlock()
EncryptPEMBlock()
ParsePKCS1PrivateKey()
MarshalPKCS1PrivateKey()
ParsePKCS8PrivateKey()
MarshalPKCS8PrivateKey()
ParseECPrivateKey()
MarshalECPrivateKey()
ParsePKIXPublicKey()
MarshalPKIXPublicKey()
ParseCertificate()
CreateCertificate()
ParseCRL()
```

cert_pool.go

```
NewCertPool()
SystemCertPool() //无
AddCert() //无
AppendCertsFromPEM() //无
Subjects() //无
```

NewCertPool()

```
./core/deliverservice/config.go:    certPool := x509.NewCertPool()
./core/peer/pkg_test.go:  certPool := x509.NewCertPool()
./core/operations/operations_suite_test.go: clientCertPool :=
x509.NewCertPool()
```

```
./core/operations/tls.go:   caCertPool := x509.NewCertPool()
./core/operations/tls_test.go:    clientCAPool := x509.NewCertPool()
./core/chaincode/accesscontrol/access_test.go:    ClientCAs:
x509.NewCertPool(),
./core/chaincode/accesscontrol/access_test.go:    RootCAs: x509.NewCertPool(),
./integration/e2e/e2e_test.go:  clientCertPool := x509.NewCertPool()
./discovery/client/client_test.go:    RootCAs:      x509.NewCertPool(),
./gossip/util/grpc.go:    RootCAs:      x509.NewCertPool(),
./internal/pkg/comm/creds_test.go:  certPool := x509.NewCertPool()
./internal/pkg/comm/creds_test.go:  certPool := x509.NewCertPool()
./internal/pkg/comm/creds_test.go:  expectedCertPool := x509.NewCertPool()
./internal/pkg/comm/creds_test.go:  certPool := x509.NewCertPool()
./internal/pkg/comm/creds_test.go:  certPool := x509.NewCertPool()
./internal/pkg/comm/server.go:          grpcServer.tls.config.ClientCAs =
x509.NewCertPool()
./internal/pkg/comm/server.go:  certPool := x509.NewCertPool()
./internal/pkg/comm/util_test.go:   RootCAs: x509.NewCertPool(),
./internal/pkg/comm/client.go:    client.tlsConfig.RootCAs =
x509.NewCertPool()
./internal/pkg/comm/client.go:  certPool := x509.NewCertPool()
./internal/pkg/comm/connection.go:  certPool := x509.NewCertPool()
./internal/pkg/comm/client_test.go: certPool := x509.NewCertPool()
./internal/pkg/comm/client_test.go:   tlsConfig.RootCAs = x509.NewCertPool()
./internal/pkg/comm/server_test.go: certPool := x509.NewCertPool()
./internal/pkg/comm/server_test.go: certPool := x509.NewCertPool()
./internal/pkg/comm/server_test.go: certPool := x509.NewCertPool()
./internal/pkg/comm/server_test.go:     RootCAs:      x509.NewCertPool(),
./internal/pkg/comm/server_test.go: certPoolCA := x509.NewCertPool()
./internal/pkg/comm/server_test.go: certPool := x509.NewCertPool()
./internal/pkg/peer/orderers/connection_test.go:    org1CertPool =
x509.NewCertPool()
./internal/pkg/peer/orderers/connection_test.go:    org2CertPool =
x509.NewCertPool()
./internal/pkg/peer/orderers/connection_test.go:    overrideCertPool =
x509.NewCertPool()
./internal/pkg/peer/orderers/connection_test.go:      newOrg1CertPool :=
x509.NewCertPool()
./internal/pkg/peer/orderers/connection_test.go:      globalCertPool =
x509.NewCertPool()
./internal/pkg/peer/orderers/connection.go: globalCertPool :=
x509.NewCertPool()
./internal/pkg/peer/orderers/connection.go:   certPool := x509.NewCertPool()
./orderer/consensus/kafka/config.go:    rootCAs := x509.NewCertPool()
./orderer/consensus/etcdraft/membership.go: tlsRoots := x509.NewCertPool()
./orderer/consensus/etcdraft/membership.go: tlsIntermediates :=
x509.NewCertPool()
./orderer/common/cluster/util.go:   tlsConfig.RootCAs = x509.NewCertPool()
./common/crypto/tlsgen/ca_test.go:    ClientCAs:    x509.NewCertPool(),
./common/crypto/tlsgen/ca_test.go:      RootCAs:      x509.NewCertPool(),
./common/grpclogging/grpclogging_suite_test.go: caCertPool =
x509.NewCertPool()
./msp/mspimplsetup_test.go:     opts: &x509.VerifyOptions{Roots:
x509.NewCertPool(), Intermediates: x509.NewCertPool()},
./msp/mspimplsetup_test.go:     opts: &x509.VerifyOptions{Roots:
x509.NewCertPool(), Intermediates: x509.NewCertPool()},
./msp/mspimplsetup_test.go:     opts: &x509.VerifyOptions{Roots:
x509.NewCertPool(), Intermediates: x509.NewCertPool()},
./msp/mspimplsetup_test.go:     opts: &x509.VerifyOptions{Roots:
x509.NewCertPool(), Intermediates: x509.NewCertPool()},
./msp/mspimplsetup_test.go:     opts: &x509.VerifyOptions{Roots:
x509.NewCertPool(), Intermediates: x509.NewCertPool()},
./msp/mspimplsetup_test.go:     opts: &x509.VerifyOptions{Roots:
x509.NewCertPool(), Intermediates: x509.NewCertPool()},
./msp/msp_test.go:  localMsp.(*bccspmsp).opts.Roots = x509.NewCertPool()
./msp/cert_test.go: msp.opts.Roots = x509.NewCertPool()
./msp/cert_test.go: msp.opts.Roots = x509.NewCertPool()
./msp/cert_test.go: msp.opts.Roots = x509.NewCertPool()
./msp/mspimplsetup.go:  msp.opts = &x509.VerifyOptions{Roots:
x509.NewCertPool(), Intermediates: x509.NewCertPool()}
./msp/mspimplsetup.go:  msp.opts = &x509.VerifyOptions{Roots:
x509.NewCertPool(), Intermediates: x509.NewCertPool()}
./msp/mspimplsetup.go:  opts := &x509.VerifyOptions{Roots: x509.NewCertPool(),
Intermediates: x509.NewCertPool()}
./vendor/google.golang.org/grpc/credentials/tls.go: cp := x509.NewCertPool()
./vendor/github.com/fsouza/go-dockerclient/client.go:   caPool :=
x509.NewCertPool()
./vendor/github.com/hyperledger/fabric-chaincode-go/shim/internal/config.go:
rootCertPool = x509.NewCertPool()
```

pem_decrypt.go

```
IsEncryptedPEMBlock()
DecryptPEMBlock()
EncryptPEMBlock()
```

IsEncryptedPEMBlock()

```
./bccsp/sw/keys.go: if x509.IsEncryptedPEMBlock(block) {
./bccsp/sw/keys.go: if x509.IsEncryptedPEMBlock(block) {
./bccsp/sw/keys.go: if x509.IsEncryptedPEMBlock(block) {
```

DecryptPEMBlock()

```
./bccsp/sw/keys.go:   decrypted, err := x509.DecryptPEMBlock(block, pwd)
./bccsp/sw/keys.go:   decrypted, err := x509.DecryptPEMBlock(block, pwd)
./bccsp/sw/keys.go:   decrypted, err := x509.DecryptPEMBlock(block, pwd)
```

EncryptPEMBlock()

```
/bccsp/sw/keys.go:    block, err := x509.EncryptPEMBlock(
./bccsp/sw/keys.go: block, err := x509.EncryptPEMBlock(
./bccsp/sw/keys.go:   block, err := x509.EncryptPEMBlock(
```

pkcs1.go

```
ParsePKCS1PrivateKey()
MarshalPKCS1PrivateKey()
ParsePKCS1PublicKey() //无
MarshalPKCS1PublicKey() 无
```

ParsePKCS1PrivateKey()

```
./bccsp/sw/keys.go: if key, err = x509.ParsePKCS1PrivateKey(der); err == nil {
```

MarshalPKCS1PrivateKey()

```
./internal/cryptogen/csp/csp_test.go: pkcs1Encoded :=
x509.MarshalPKCS1PrivateKey(rsaKey)

./orderer/mocks/util/util.go:     Bytes:
x509.MarshalPKCS1PrivateKey(privateKey),

./bccsp/sw/keyimport_test.go: raw := x509.MarshalPKCS1PrivateKey(k)
```

pkcs8.go

```
ParsePKCS8PrivateKey()
MarshalPKCS8PrivateKey()
```

ParsePKCS8PrivateKey()

```
./cmd/common/signer/signer.go:  if key, err := x509.ParsePKCS8PrivateKey(der);
err == nil {
./integration/configtx/configtx_test.go:  privateKey, err :=
x509.ParsePKCS8PrivateKey(pemBlock.Bytes)
./integration/raft/cft_test.go: caKeyWithoutType, err :=
x509.ParsePKCS8PrivateKey(keyAsDER.Bytes)
./discovery/test/integration_test.go: key, err :=
x509.ParsePKCS8PrivateKey(bl.Bytes)
./internal/cryptogen/csp/csp.go:  key, err :=
x509.ParsePKCS8PrivateKey(block.Bytes)
./bccsp/sw/keys.go: if key, err = x509.ParsePKCS8PrivateKey(der); err == nil {
./bccsp/sw/keys_test.go:  _, err = x509.ParsePKCS8PrivateKey(pemBlock.Bytes)
./vendor/github.com/hyperledger/fabric-config/configtx/msp.go:  privateKey,
err := x509.ParsePKCS8PrivateKey(pemBlock.Bytes)
```

MarshalPKCS8PrivateKey()

```
./cmd/common/signer/signer_test.go: pkcs8, err :=
x509.MarshalPKCS8PrivateKey(ecPK)
./internal/cryptogen/csp/csp_test.go: pkcs8Encoded, err :=
x509.MarshalPKCS8PrivateKey(rsaKey)
./internal/cryptogen/csp/csp.go:  pkcs8Encoded, err :=
x509.MarshalPKCS8PrivateKey(priv)
./common/crypto/tlsgen/key.go:  privBytes, err :=
x509.MarshalPKCS8PrivateKey(privateKey)
./vendor/github.com/hyperledger/fabric-config/configtx/msp.go:  privBytes, err
:= x509.MarshalPKCS8PrivateKey(priv)
```

sec1.go

```
ParseECPrivateKey()
MarshalECPrivateKey()
```

ParseECPrivateKey()

```
./cmd/common/signer/signer_test.go: ecPK, err :=
x509.ParseECPrivateKey(pemBlock.Bytes)
./cmd/common/signer/signer.go:  key, err := x509.ParseECPrivateKey(der)
./cmd/idemixgen/main.go:  key, err := x509.ParseECPrivateKey(block.Bytes)
./bccsp/sw/keys.go: if key, err = x509.ParseECPrivateKey(der); err == nil {
```

MarshalECPrivateKey

```
./cmd/common/signer/signer_test.go: ec1, err := x509.MarshalECPrivateKey(ecPK)
./cmd/idemixgen/main.go:    encodedRevocationSK, err := x509.MarshalECPrivateKey(revocationKey)
./gossip/comm/crypto_test.go: privBytes, err := x509.MarshalECPrivateKey(privateKey)
./internal/pkg/comm/testdata/certs/generate.go: keyBytes, err := x509.MarshalECPrivateKey(priv)
./common/grpclogging/grpclogging_suite_test.go: keyBytes, err := x509.MarshalECPrivateKey(key)
./bccsp/sw/keys.go: return x509.MarshalECPrivateKey(privateKey)
./bccsp/sw/keys.go:   raw, err := x509.MarshalECPrivateKey(k)
```

verify.go

```
Error() //无
Verify() //无
VerifyHostname() //无
```

x509

```
ParsePKIXPublicKey()
MarshalPKIXPublicKey()
String() //无
Error() //无
Equal() //无
CheckSignatureFrom() //无
CheckSignature() //无
CheckCRLSignature() //无
ParseCertificate() //
ParseCertificates() //无
CreateCertificate() //
ParseCRL() //
ParseDERCRL() //无
CreateCRL() //无
CreateCertificateRequest() //无
ParseCertificateRequest() //无
```

ParsePKIXPublicKey()

```
./bccsp/sw/keys.go: key, err := x509.ParsePKIXPublicKey(raw)
./bccsp/signer/signer.go: pk, err := x509.ParsePKIXPublicKey(raw)
./bccsp/idemix/handlers/revocation.go:  revocationPk, err :=
x509.ParsePKIXPublicKey(blockPub.Bytes)
./bccsp/pkcs11/impl_test.go:  pub, err := x509.ParsePKIXPublicKey(pkRaw)
./bccsp/pkcs11/impl_test.go:  pub, err := x509.ParsePKIXPublicKey(pkRaw)
```

MarshalPKIXPublicKey()

```
./cmd/idemixgen/main.go:    encodedRevocationPK, err :=
x509.MarshalPKIXPublicKey(revocationKey.Public())
./common/tools/idemixgen/idemixca/idemixca_test.go: encodedRevocationPK, err
:= x509.MarshalPKIXPublicKey(revocationkey.Public())
./bccsp/sw/impl_test.go:  pub, err :=
x509.MarshalPKIXPublicKey(&key.PublicKey)
./bccsp/sw/ecdsakey.go: raw, err = x509.MarshalPKIXPublicKey(k.pubKey)
./bccsp/sw/keys.go:   PubASN1, err := x509.MarshalPKIXPublicKey(k)
./bccsp/sw/keys.go:   raw, err := x509.MarshalPKIXPublicKey(k)
./bccsp/sw/keys_test.go:  der, err = x509.MarshalPKIXPublicKey(&key.PublicKey)
./bccsp/sw/keyimport_test.go: raw, err :=
x509.MarshalPKIXPublicKey(&k.PublicKey)
./bccsp/sw/ecdsa_test.go: bytes2, err := x509.MarshalPKIXPublicKey(k.pubKey)
./bccsp/signer/signer_test.go:  pkRaw, err :=
x509.MarshalPKIXPublicKey(&k.PublicKey)
./bccsp/idemix/handlers/revocation_test.go:       pkBytes, err =
x509.MarshalPKIXPublicKey(&idemixRevocationKey.PublicKey)
./bccsp/idemix/handlers/revocation_test.go:       raw, err =
x509.MarshalPKIXPublicKey(key.Public())
./bccsp/idemix/handlers/revocation.go:  raw, err =
x509.MarshalPKIXPublicKey(k.pubKey)
./bccsp/pkcs11/ecdsakey.go: raw, err = x509.MarshalPKIXPublicKey(k.pub)
```

ParseCertificate()

```
./integration/configtx/configtx_test.go:  cert, err :=
x509.ParseCertificate(pemBlock.Bytes)
./integration/raft/cft_test.go: caCert, err :=
x509.ParseCertificate(caCertAsDER.Bytes)
./integration/raft/cft_test.go: cert, err :=
x509.ParseCertificate(certAsDER.Bytes)
./gossip/api/crypto.go: cert, _ := x509.ParseCertificate(bl.Bytes)
./internal/cryptogen/ca/ca.go:  x509Cert, err :=
x509.ParseCertificate(certBytes)
```

```
./internal/cryptogen/ca/ca.go:      cert, err =
x509.ParseCertificate(block.Bytes)
./internal/pkg/comm/testdata/certs/generate.go: x509Cert, err :=
x509.ParseCertificate(certBytes)
./internal/pkg/comm/util.go:    cert, err :=
x509.ParseCertificate(block.Bytes)
./orderer/consensus/etcdraft/util.go: if _, err :=
x509.ParseCertificate(bl.Bytes); err != nil {
./orderer/consensus/etcdraft/membership.go: certificate, err :=
x509.ParseCertificate(pemBlock.Bytes)
./orderer/common/cluster/comm.go:   cert, err :=
x509.ParseCertificate(stub.ServerTLSCert)
./orderer/common/cluster/util_test.go:    cert, err :=
x509.ParseCertificate(bl.Bytes)
./orderer/common/cluster/util_test.go:      cert, err :=
x509.ParseCertificate(bl.Bytes)
./orderer/common/cluster/comm_test.go:  cert, err :=
x509.ParseCertificate(node2.nodeInfo.ServerTLSCert)
./orderer/common/cluster/util.go:     cert, err :=
x509.ParseCertificate(bl.Bytes)
./common/crypto/expiration.go:  cert, err := x509.ParseCertificate(bl.Bytes)
./common/crypto/tlsgen/key_test.go: cert, err :=
x509.ParseCertificate(block.Bytes)
./common/crypto/tlsgen/key.go:  cert, err :=
x509.ParseCertificate(block.Bytes)
./common/grpclogging/grpclogging_suite_test.go: ca, err :=
x509.ParseCertificate(tlsCert.Certificate[0])
./common/deliver/deliver_test.go:     cert, err =
x509.ParseCertificate(der.Bytes)
./common/policies/policy.go:  cert, err :=
x509.ParseCertificate(pemBlock.Bytes)
./msp/mspimpl.go: cert, err := x509.ParseCertificate(pemCert.Bytes)
./msp/mspimpl.go: cert, err := x509.ParseCertificate(bl.Bytes)
./msp/mspimpl.go: cert, err := x509.ParseCertificate(bl.Bytes)
./msp/cert.go:  return x509.ParseCertificate(newRaw)
./msp/msp_test.go:  cert, err := x509.ParseCertificate(bl.Bytes)
./msp/msp_test.go:  caCertFromFile, err := x509.ParseCertificate(bl.Bytes)
./msp/msp_test.go:  certFromFile, err := x509.ParseCertificate(bl.Bytes)
./msp/cert_test.go: cert, err := x509.ParseCertificate(certRaw)
./bccsp/sw/impl_test.go:  cert, err := x509.ParseCertificate(certRaw)
./bccsp/pkcs11/impl_test.go:  cert, err := x509.ParseCertificate(certRaw)
./vendor/github.com/hyperledger/fabric-config/configtx/msp.go:  certificate,
err := x509.ParseCertificate(pemBlock.Bytes)
./vendor/github.com/hyperledger/fabric-config/configtx/orderer.go:
clientTLSCert, err := x509.ParseCertificate(clientTLSCertBlock.Bytes)
./vendor/github.com/hyperledger/fabric-config/configtx/orderer.go:
serverTLSCert, err := x509.ParseCertificate(serverTLSCertBlock.Bytes)
```

CreateCertificate()

```
./integration/raft/cft_test.go: certBytes, err :=
x509.CreateCertificate(rand.Reader, cert, caCert, cert.PublicKey, caKey)
./integration/raft/cft_test.go: caCertBytes, err :=
x509.CreateCertificate(rand.Reader, caCert, caCert, caCert.PublicKey, caKey)
./gossip/comm/crypto_test.go: rawBytes, err :=
x509.CreateCertificate(rand.Reader, &template, &template,
&privateKey.PublicKey, privateKey)
./internal/cryptogen/ca/ca.go:  certBytes, err :=
x509.CreateCertificate(rand.Reader, template, parent, pub, priv)
./internal/pkg/comm/testdata/certs/generate.go: certBytes, err :=
x509.CreateCertificate(rand.Reader, template, parent, pub, priv)
./orderer/mocks/util/util.go: publicKeyCert, err :=
x509.CreateCertificate(rand.Reader, &template, &template, privateKey.Public(),
privateKey)
./common/crypto/tlsgen/key.go:  rawBytes, err :=
x509.CreateCertificate(rand.Reader, &template, parent, &privateKey.PublicKey,
certSigner)
./common/grpclogging/grpclogging_suite_test.go: derBytes, err :=
x509.CreateCertificate(rand.Reader, &template, &template, publicKey,
privateKey)
./common/grpclogging/grpclogging_suite_test.go: derBytes, err :=
x509.CreateCertificate(rand.Reader, &template, ca, publicKey,
tlsCert.PrivateKey)
./msp/cert_test.go: certRaw, err := x509.CreateCertificate(rand.Reader,
&template, &template, &k.PublicKey, k)
./bccsp/sw/impl_test.go:  certRaw, err := x509.CreateCertificate(rand.Reader,
&template, &template, pub, cryptoSigner)
./bccsp/pkcs11/impl_test.go:  certRaw, err :=
x509.CreateCertificate(rand.Reader, &template, &template, pub, cryptoSigner)
```

ParseCRL()

```
./msp/mspimplsetup.go:    crl, err := x509.ParseCRL(crlbytes)
./vendor/github.com/hyperledger/fabric-config/configtx/msp.go:  crl, err :=
x509.ParseCRL(crlBytes)
./vendor/github.com/hyperledger/fabric-config/configtx/msp.go:
certificateList, err := x509.ParseCRL(pemBlock.Bytes)
```