

04. gRPC 실용사례 php

본 문서에서는 php 에서 gRPC 를 사용하여 Client 프로그램을 작성 하는 방법을 소개 한다. php 에서 grpc 설치 [여기](#) 의 5. PHP 에서 gRPC 사용하기 를 참조 한다.

1. Interface 를 위한 .proto 파일

실용사례 에서 공통으로 사용할 .proto 파일로 이름은 `xdb_grpc.proto` 로 작성 하였다.

01. gRPC 실용사례 CPP 에 있는 내용과 동일 하다.

```
syntax = "proto3";

// Option for Java.
option java_multiple_files = true;
option java_package = "com.hancomsechre.xecuredb.grpc";
option java_outer_classname = "XecuredbProto";

// Option for Objective-C.
option objc_class_prefix = "XDB";

package xdbproto;

// The Xecuredb service definition.
service Xecuredb {
    rpc Encrypt (EncRequest) returns (EncReply) {}
    rpc Decrypt (DecRequest) returns (DecReply) {}
    rpc CryptoHash (HashRequest) returns (HashReply) {}
}

// The request message containing the alias and plain text.
message EncRequest {
    string alias = 1;
    string plain = 2;
}

// The response message containing the err code and cipher text.
message EncReply {
    int32 err = 1;
    string msg = 2;
    string cipher = 3;
}

// The request message containing the alias and cipher text.
message DecRequest {
    string alias = 1;
    string cipher = 2;
}

// The response message containing the err code and plain text.
message DecReply {
    int32 err = 1;
    string msg = 2;
}
```

```

    string plain = 3;
}

// The request message containing the alogorithm id and plain text.
message HashRequest {
    int32 id = 1;
    string plain = 2;
}

// The response message containing the err code and hash string.
message HashReply {
    int32 err = 1;
    string msg = 2;
    string cipher = 3;
}

```

2. protoc 를 이용한 Interface 파일 생성

php 에서 gRPC 를 사용하기 위해서는 protoc 를 이용하여 interface 파일을 생성 하여야 한다. 다음은 interface 파일을 생성 하기 위한 명령 이다. `create_python_src.sh`

```

# php
protoc -I ../proto --php_out=./ --grpc_out=./ --plugin=protoc-gen-grpc=`which
grpc_php_plugin` xdb_grpc.proto

```

위 명령을 실행 하면 작업 directory 에 `GPBMetadata`, `xdbproto` 두개의 directory 가 생성 된다.

`GPBMetadata` 의 파일들

`xdbGrpc.php`

`xdbproto` 의 파일들

`DecReply.php`
`DecRequest.php`
`EncReply.php`
`EncRequest.php`
`HashReply.php`
`HashRequest.php`
`xecuredbClient.php`

3. 종속성 파일 설치

php 에서 gRPC 를 사용하기 위해서는 관련 Package 를 설치 하여야 한다. Package 설치 는 composer 를 이용 하는데 `composer.json` 파일을 만들고 `composer install` 명령을 실행 하여 설치 할 수 있다. 다음은 composer.json 파일 이다.

```
{
  "name": "php_xdb_client",
  "description": "gRPC encrypt for PHP",
  "require": {
    "grpc/grpc": "^v1.3.0",
    "google/protobuf": "^v3.3.0"
  },
  "autoload": {
    "psr-4": {
      "": "route_guide/"
    }
  }
}
```

위 파일을 작성 한후 다음 명령으로 설치를 실행 한다.

```
composer install
```

실행이 완료 되면 작업 directory 에 `vendor` directory 가 추가 된다.

3. Client 코드 작성

php 를 이용한 client 는 php_xdb_client.py 로 작성 하였으며 source code 는 다음과 같다.

```
<?php

require dirname(__FILE__).'/vendor/autoload.php';

@include_once dirname(__FILE__).'/Xdbproto/XecuredbClient.php';
@include_once dirname(__FILE__).'/Xdbproto/EncReply.php';
@include_once dirname(__FILE__).'/Xdbproto/EncRequest.php';
@include_once dirname(__FILE__).'/Xdbproto/DecReply.php';
@include_once dirname(__FILE__).'/Xdbproto/DecRequest.php';
@include_once dirname(__FILE__).'/Xdbproto/HashReply.php';
@include_once dirname(__FILE__).'/Xdbproto/HashRequest.php';

@include_once dirname(__FILE__).'/GPBMetadata/XdbGrpc.php';

function getEncrypt($alias,$plain)
{
    $client = new Xdbproto\XecuredbClient('localhost:50052', [
        'credentials' => Grpc\ChannelCredentials::createInsecure(),
    ]);
    $request = new Xdbproto\EncRequest();
    $request->setAlias($alias);
    $request->setPlain($plain);

    list($reply, $status) = $client->Encrypt($request)->wait();
}
```

```

    $cipher = $reply->getCipher();

    return $cipher;
}

function getDecrypt($alias,$cipher)
{
    $client = new xdbproto\XecuredbClient('localhost:50052', [
        'credentials' => Grpc\ChannelCredentials::createInsecure(),
    ]);
    $request = new xdbproto\DecRequest();
    $request->setAlias($alias);
    $request->setCipher($cipher);

    list($reply, $status) = $client->Decrypt($request)->wait();
    $plain = $reply->getPlain();

    return $plain;
}

function getHash($algo,$plain)
{
    $client = new xdbproto\XecuredbClient('localhost:50052', [
        'credentials' => Grpc\ChannelCredentials::createInsecure(),
    ]);
    $request = new xdbproto\HashRequest();
    $request->setId($algo);
    $request->setPlain($plain);

    list($reply, $status) = $client->Hash($request)->wait();
    $strhash = $reply->getHash();

    return $strhash;
}

$alias = 'normal';
$plain = '1234567890123';
$strEnc = '';
$strDec = '';
$strHash = '';
$algo = 6;

$strEnc = getEncrypt($alias,$plain);
$strDec = getDecrypt($alias,$strEnc);
$strHash = getHash($algo, $plain);

echo "Encrypt client received: ".$strEnc."\n";
echo "Decrypt client received: ".$strDec."\n";
echo "Hash    client received: ".$strHash."\n";

```

protoc 에 의해 xdb_grpc_pb2_grpc.py 에 xecuredbStub stub 코드가 생성 된다. XecuredbStub 는 xdb_grpc.proto 에서 정의한 service Xecuredb 를 이용하여 생성 된다. 주의 해서 볼 부분은 stub.Encrypt, stub.Decrypt, stub.Hash 의 parameter로 xdb_grpc_pb2.EncRequest, xdb_grpc_pb2.DecRequest, xdb_grpc_pb2.HashRequest 함수를 사용 하였고 그 함수들의 parameter 로 서버로 전달할 값 들을 넣어서 호출하는 과정 이다. python 에서도 비교적 쉽게 gRPC 를 사용 할 수 있다.

4. php client 프로그램 실행

프로그램의 실행은 다음 명령을 이용한다.

```
php php_xdb_client.php

or

php -d extension=grpc.so php_xdb_client.php
```

위의 명령 중 `php -d extension=grpc.so php_xdb_client.php` 를 실행 하면 `PHP warning: Module 'grpc' already loaded in Unknown on line 0` message 가 나올 수 있는데 `php.ini` 파일에 `extension=grpc.so` 가 등록 되어 있는 경우는 해당 Warning message 가 Display 된다.

실행 결과

```
Encrypt client received: AAGo8vet8aHoqsGorhx1CsXL
Decrypt client received: 1234567890123
Hash    client received: vKK0Gis14TfIP+40ave9Hg9SvVYFg8oHobQvmUTFxQs=
```

5. Remark

위 프로그램은 gRPC examples 에 있는 Helloworld 를 참조 하여 작성 하였다.

- 끝 -