**NHN 웹플랫폼개발팀 / 2010.06.24**

# API Gateway HMAC 인증 가이드 (for .NET)

## 목적

API Gateway는 NHN 콘텐츠 데이터를 단일화 된 채널로 외부에 제공하기 위하여 인증, 쿼터제한, 통계, 포맷변환 등의 기능을 제공하는 플랫폼이다.

본 문서에서는 API Gateway의 인증 방법 중 하나인 HMAC 인증을 통하여 API를 호출하는 방법에 대한 가이드를 제공한다.

## 인증 방법

HMAC 인증을 통해 NHN 서비스 API를 호출하기 위해서는 호출할 URL에 대해 Encrypt된 URL 값을 얻어야 한다.

EncryptedUrl은 각 업체에 발급된 제휴업체 Key, API URL, 호출 시간의 조합으로 생성된다.

* 제휴업체 Key : NHN에서 발급한 업체별로 고유한 Key, 업체별로 파일로 제공된다.
* API URL : API 수행을 위하여 호출할 HTTP URL
* 호출 시간 : API를 호출하는 시간의 Time Serial

API Gateway는 수신한 URL의 Hash 값을 확인하여 해당 요청이 유효한지 여부를 검사한다.

## 설치

* 전달받은 파일들을 원하는 위치에 저장한다. (이를 제공되는 파일 목록은 **6.**  항목 참고)
* 작성 중인 Application에서 ApiGateway-MAC.dll을 참조에 추가한다.
* **NHNAPIGatewayKey.properties를 윈도우의 시스템의 폴더에 저장한다.** (별도의 설치 폴더를 원할 경우 getEncryptUrl 함수의 2번째 인자로 전달한다.)

## 사용 방법

* 호출할 URL을 설정한다.

String url = "http://dev.apis.naver.com/제휴 업체 ID/hmac/hmactest.xml";

* 호출할 URL의 Encrypt된 값을 얻기 위해 MACManager의 getEncryptUrl 함수를 실행한다.

- 서비스 API를 호출할 때 마다 새로운 encryptedUrl을 얻어야 한다.

String encryptedUrl = MACManager.getEncryptUrl(url);

* 생성된 encryptedUrl을 이용하여 데이터를 요청한다.
* 오류가 발생했을 경우 exception을.발생시킨다. (try-catch문의 사용을 권장)

## 호출 결과

API 호출 후 HMAC 인증 성공 여부에 따라 아래 형태의 값이 반환된다.

* 인증 실패 : 인증 실패에 대한 error code와 message가 반환된다

예) 인증 실패에 대해 결과 값을 XML 형태로 반환.

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<result>

<message>HMAC Authentication failed</message>

<error\_code>022</error\_code>

</result>

* 인증 성공 : API 호출에 대한 결과 값이 반환된다.

예) HMAC 인증 테스트 성공 시 결과

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<result>

<authentication>true</authentication>

</result>

## 제공 파일

|  |  |
| --- | --- |
| 파일명 | 비고 |
| ApiGateway-MAC.dll | 인증을 위해 NHN에서 제공하는 library 파일 |
| NHNAPIGatewayKey.properties | 제휴사별로 발급되는 Key가 저장된 파일 |

## 예제 console 프로젝트

아래는 HMAC 인증이 적용되었는지 확인하는 예제이다. 표시된 부분을 원하는 URL로 변경 후 테스트하면 된다. 예제에서는 C# .NET Console Project를 생성하여 결과 HTML 페이지를 콘솔창에 출력하는 예제이다.

/\*

\* sample project for ApiGate-Mac.dll

\* 2010.06.24

\* Copyright 2010 NHN Corp. All rights Reserved.

\* NHN PROPRIETARY. Use is subject to license terms.

\*/

using System;

using System.Collections.Generic;

using System.Text;

using System.Net;

using System.IO;

using NHNSecurityClient;

namespace HMACConsole

{

class Program

{

static void Main(string[] args)

{

string url = string.Empty;

try

{

// Make the request url

url = MACManager.getEncryptUrl("http://dev.apis.naver.com/제휴 업체 ID로 대체/hmac/hmactest.xml?param=paramValue");

}

catch (Exception e)

{

e.ToString();

}

// Create a request for the URL.

WebRequest request = WebRequest.Create(url);

// If required by the server, set the credentials.

request.Credentials = CredentialCache.DefaultCredentials;

// Get the response.

WebResponse response = request.GetResponse();

// Display the status.

Console.WriteLine(((HttpWebResponse)response).StatusDescription);

// Get the stream containing content returned by the server.

Stream dataStream = response.GetResponseStream();

// Open the stream using a StreamReader for easy access.

StreamReader reader = new StreamReader(dataStream);

// Read the content.

string responseFromServer = reader.ReadToEnd();

// Display the content.

Console.WriteLine(responseFromServer);

// Clean up the streams and the response.

reader.Close();

response.Close();

}

}

}

## 참고 URL

* MSDN : <http://msdn.microsoft.com/ko-kr/library/456dfw4f(v=VS.80).aspx>  
   <http://msdn.microsoft.com/ko-kr/library/system.security.cryptography.hmac(VS.80).aspx>