
i-SAS 3.0

Non-ActiveX

Development Guide



문서개정이력

[illegible]

목 차

1. i-SAS 3.0 개요	4
1.1 목적	4
1.2 i-SAS 3.0 지원 환경	4
1.3 i-SAS 3.0 업무 흐름도	4
1.4 모듈 흐름도	5
1.5 모듈구성	5
2. API	6
2.1 Import 라이브러리	6
2.2 인증코드 설정	6
2.3 버전정보 확인	6
2.4 모듈 강제종료	7
2.5 모듈 Update	7
2.6 스크래핑 환경변수 설정	7
2.7 인증서 목록 조회	8
2.8 인증서 식별자 검증	9
2.9 모듈 초기화	10
2.10 Open 및 Thread 할당	10
2.11 입력 데이터 설정 및 실행	11
2.12 입력 암호화	12
2.13 결과 복호화	13

1. i-SAS 3.0 개요

i-SAS 3.0 솔루션은 인터넷을 통해 제공되는 정보를 스크래핑하는 솔루션입니다.

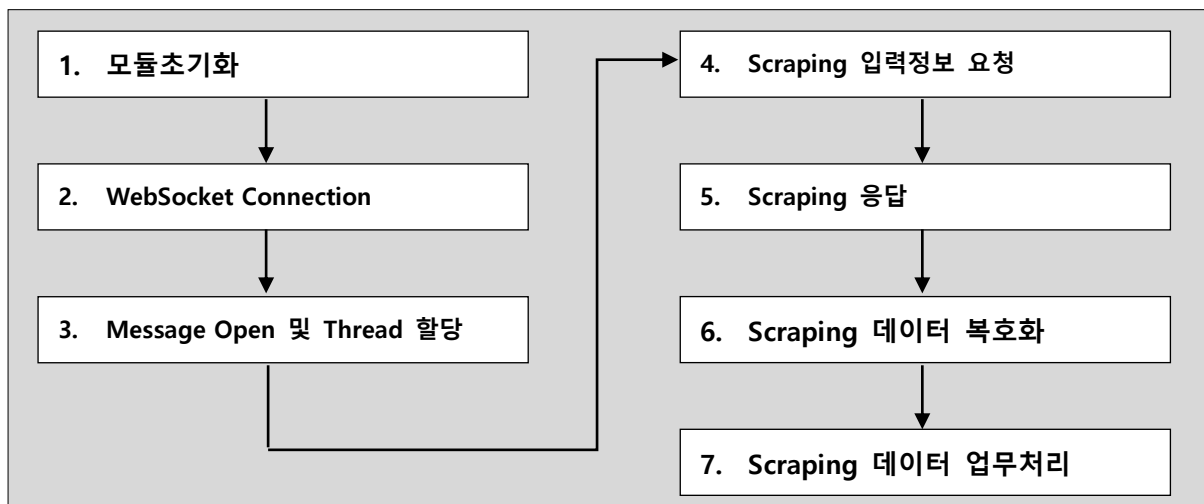
1.1 목적

본 문서는 i-SAS 3.0 개발자를 위한 Web 개발용 개발가이드 입니다.

1.2 i-SAS 3.0 지원 환경

데이터포맷	JSON	네트워크	공용망
사용언어	JAVA, JAVASCRIPT (jQuery 1.7 이상)		
지원 OS	Windows		
지원 브라우저	IE8 이상, Chrome, Firefox, Opera 등		
WebService Listen Port	14101	WebSocket Listen Port	14102
Domain	https://ibasews.coocon.co.kr (로컬 웹서비스)		
SSL	COMODO RSA Domain Validation Secure Server CA		

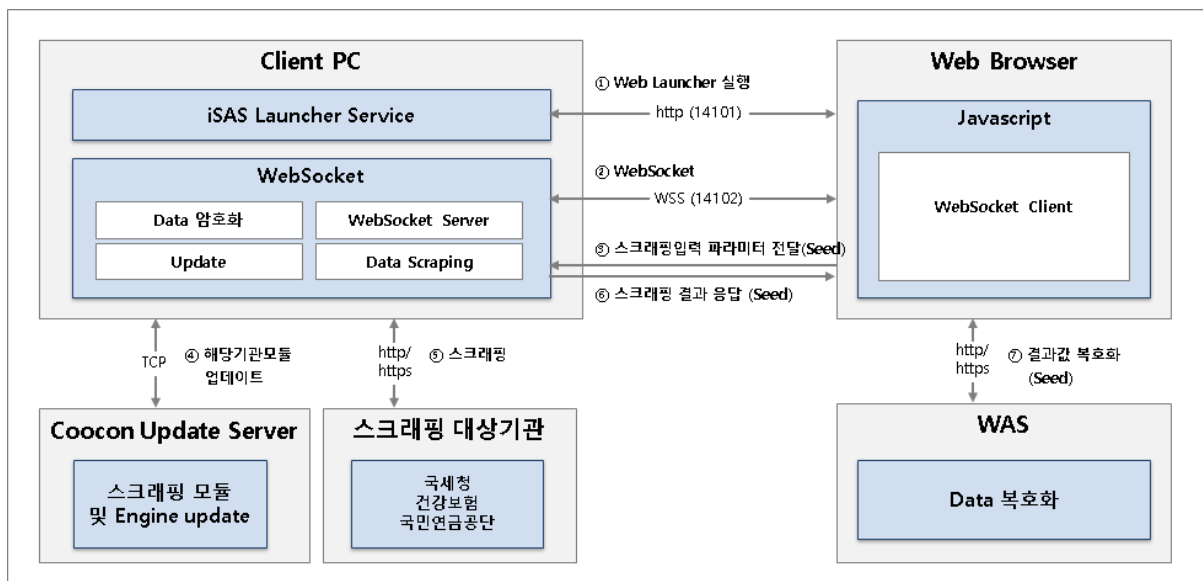
1.3 i-SAS 3.0 업무 흐름도



1. 모듈 초기화
: 모듈 초기화 하며, Local Sever를 실행시킵니다.
2. WebSocket Connection
: PC (Local Server) 와 Browser 간의 통신을 WebSocket 을 통해 연결합니다.
3. Message Open 및 Thread 할당

- : Local Server를 Open 시키며, 동시 작업진행 할 Thread 할당요청 합니다.
4. Scarping 입력정보 요청
: Local Server와 연결된 상태에서 Scarping 요청 데이터를 요청합니다.
 5. Scarping 응답
: Scarping 결과를 Callback 함수로 제공합니다.
 6. Scarping 데이터 복호화
: Scarping 된 결과를 서버로 전송하여 제공된 라이브러리를 통해 복호화 처리 합니다.
 7. Scarping 데이터 업무처리

1.4 모듈 흐름도



1.5 모듈구성

모듈명	설명
isas1.0.jar	i-SAS 3.0 스크래핑 결과 복호화 라이브러리 - 프로젝트 라이브러리(jar)에 포함시켜서 사용합니다.
NXiSAS.exe	i-SAS 3.0 설치파일 - 사용자 PC에 설치하는 스크래핑 설치 모듈입니다.

2. API

Web에서 스크래핑 요청 시 함수 기반으로 파라미터를 전달하고 CallBack 형태로 결과를 받는 방식을 사용합니다.

2.1 Import 라이브러리

Web 페이지에서 i-SAS 3.0호출시 필요로 한 자바스크립트 라이브러리 입니다.

[참고1] 샘플소스

```
<script src="/js/jquery-1.9.1.min.js"></script> <!-- jquery 1.7 이상 권고-->
<script src="/js/json2.js"></script>
<script src="/js/swfobject.js"></script> <!--IE 10이하 지원 -->
<script src="/js/web_socket.js"></script> <!--IE 10이하 지원 -->
<script src="/js/isasscaping.js"></script>
<link rel="stylesheet" href="/css/process_manager.css" /> <!--인증서 팝업용 -->
```

2.2 인증코드 설정

isasscaping.js 파일에 샘플과 같이 발급받은 정보로 설정합니다.

[참고1] 샘플소스

```
var CooconLicense = {
  "MstCompNo" : "1234567890",    // 고객업체 사업자번호
  "SubCompNo" : "1234567890",    // 고객업체 사업자번호
  "ProductNo" : "00000014"      // 쿨콘 발급일련번호
};
```

2.3 버전정보 확인

Web 페이지에서 URL정보를 호출하여, 현재설치 되어있는 버전정보를 확인할 수 있습니다..

<https://ibasews.coocon.co.kr:14101/VERSION>

[참고1] 응답결과

```
({"Result":{"Version":"2020.3.16.0"},"ErrorMessage":"","ErrorCode":"00000000"})
```

2.4 모듈 강제종료

Web에서 URL 호출하여, 현재실행중인 WebSocket 모듈을 강제로 종료 시킵니다.

<https://ibasews.coocon.co.kr:14101/KILL>

[참고1] 응답결과

```
{("ErrorMessage":"","ErrorCode":"00000000")};
```

2.5 모듈 Update

Web에서 URL 호출하여, 최신엔진 라이브러리로 업데이트 합니다. 업데이트 호출시 위 강제종료를 호출을 권장합니다.

<https://ibasews.coocon.co.kr:14101/update>

[참고1] 응답결과

```
{("ErrorMessage":"","ErrorCode":"00000000")};
```

2.6 스크래핑 환경변수 설정

Isasscraping.js 에 스크래핑 모듈실행 초기값 설정할 수 있습니다.

변수	설명
this.CurrentVersion	"2020.3.16.0" 버전 이상 일 경우만 허용 합니다.
this.RUNLEVEL	0: PC 사용자 권한 실행, 1: PC 관리자 권한으로 실행

[참고1] 샘플소스

```
this.CurrentVersion = "2020.3.16.0";
this.RUNLEVEL = 1 ;
```

2.7 인증서 목록 조회

해당 PC 설치된 전체 인증서 목록을 조회하여 Callback 으로 결과를 제공합니다.

속성	설명
RDN	인증서 식별명
Drive	인증서 저장소 Drive 경로
Type	인증서 타입
ExpiryDate	인증서 만료일자
User	인증서명
Issuer	발급기관명
SerialNo	인증서 일련번호

[참고1] 샘플소스

```
CooconiSASNX.getCertList(function(certList){
    console.log(certList);
});
```

[참고2] 샘플결과

```
{
  "ErrorCode": "00000000",
  "ErrorMessage": "",
  "Result": {
    "CertList": [
      {
        "SerialNo": "2812beb1",
        "Type": "은행개인",
        "ExpiryDate": "2019-04-24",
        "User": "홍길동()000002671",
        "Issuer": "yessignCA Class 2",
        "Drive": "C",
        "RDN": "CN=홍길동()000002671,OU=IBK,OU=persona,O=yessign,C=kr"
      }
    ]
  }
}
```

2.8 인증서 식별자 검증

인증서 식별자 검증 API는 인증서를 통해 인증서 사용자를 검증합니다.

속성	타입	설명
Module	String(필수)	"iSAS30/SASUtils"
Class	String(필수)	"iSASUtils"
Job	String(필수)	"verifyIDNum"
Input	Object	
주민사업자번호	String(필수)	
인증서	Object	
이름	String	
만료일자	String	
비밀번호	String	

[참고1] 요청 샘플

```
{
  "Module": "iSAS30/SASUtils",
  "Class": "iSASUtils",
  "Job": "verifyIDNum",
  "Input": {
    "주민사업자번호": "ENC=aaaaaa",
    "인증서": {
      "이름": "cn=홍길동(00003040201704221002671,ou=IBK,ou=personal4IB,o=yessign,c=kr",
      "만료일자": "",
      "비밀번호": "password!!"
    }
  }
}
```

속성	타입	설명
ErrorCode	String(필수)	결과 코드("00000000":정상, 그외 오류)
ErrorMessage	String	오류메세지
Result	Object	조회결과
UserName	String	식별자 명
VerifyResult	boolean	식별자 확인 결과

[참고2] 응답 샘플

```
{
  "Module": "iSAS30/SASUtils",
  "Class": "iSASUtils",
  "Job": "verifyIDNum",
  "로그인방식": "CERT",
  "Input": {
    "주민사업자번호": "*****",
    "인증서": {
      "이름": "cn=홍길동()000304820191120142636123,ou=IBK,ou=personal4IB,o=yessign,c=kr",
      "만료일자": "20201120",
      "비밀번호": "*****",
      "드라이브": "C"
    }
  },
  "Output": {
    "ErrorCode": "00000000",
    "ErrorMessage": "",
    "Result": {
      "UserName": "홍길동",
      "VerifyResult": true
    }
  }
}
```

2.9 모듈 초기화

WebSocket 엔진을 실행하며, 설치확인 실행결과를 Callback으로 결과를 제공합니다.

[참고1] 샘플소스

```
$(document).ready(function($){
  CooconiSASNX.init( function (o){
    if(!o){
      alert("설치 후 이용하시기 바랍니다.");
      return;
    }
  });
});
```

2.10 Open 및 Thread 할당

초기화된 엔진에 Thread 할당 및 라이선스 검증을 실행합니다.

[참고1] 샘플소스

```
var ThreadCount = 1 ; //한번에 실행할 Thread 길이설정
CooconiSASNX.open( ThreadCount , function (res){
    if(res.ErrorCode == "00000000" ){//Open 성공
        /**
         * 스레드 별 입력 설정 후 스크래핑실행
         **/
    }
});
```

2.11 입력 데이터 설정 및 실행

초기화된 엔진에 Thread 할당 및 라이선스 검증을 실행합니다.

[참고1] 샘플소스

```
var ThreadCount = 1 ; //한번에 실행할 Thread 길이설정
CooconiSASNX.open( ThreadCount , function (res){
    if(res.ErrorCode == "00000000" ){//Open 성공
        /**
         * 스레드 별 입력 설정 후 스크래핑실행
         **/

        for(var i =0 ; i< ThreadCount; i++){
            var inputList = [
                {"Module":"","Class":"","Job":"로그인","Input":{"로그인방식":"CERT","사용자아이디":"","사용자비밀번호":"","인증서":{"이름":"rdn","만료일자":"20180909","비밀번호":"cpw","Drive":"C"}}},
                {"Module":ModuleList[i],"Class":BankGubun,"Job":jobName,"Input":{}}
                {"Module":ModuleList[i],"Class":BankGubun,"Job":"로그아웃","Input":{}}
            ];

            /* Thread 별 Job List 설정 */
            CooconiSASNX.execute(inputList, i , function (res1, seq){

            });

        }
    }
});
```

2.12 입력 암호화

입력에 소스 누출되는 파라미터 값을 암호화하여 스크래핑 입력값으로 설정합니다.

샘플소스의 "function encrypt(data)"를 참고하시기 바랍니다.

[참고1] 샘플소스

```
var certParam = {};
var certname = "cn=홍길동(),ou=KDB,ou=personal,o=CrossCert,c=KR"
var encdata1 = encrypt( certname );
certParam["이름"] = encdata1;
certParam["만료일자"] = "20180510"; //YYYYMMDD
var encdata2 = encrypt( "1234qwer" );
certParam["비밀번호"] = encdata2;

/* 암호화 처리 */
function encrypt(data){
    var input = {} ;
    input['Data'] = data;
    input['Uid'] = CooconiSASNX.OpenUid;
    input['Action'] = CooconiSASNX.OpenAction;
    var param = "data="+ encodeURIComponent( JSON.stringify(input) );
    var rtn = {};
    $.ajax({
        url : "./encode.jsp?",
        type: "POST",
        data : param,
        async: false,
        success: function(data, textStatus, jqXHR)
        {
            rtn = data['Result'];
        },
        error: function (jqXHR, textStatus, errorThrown)
        {
            alert("error");
            rtn=null;
        }
    });
};
```

```

    return rtn;
}

```

2.13 결과 복호화

스크래핑 완료된 결과를 복호화합니다.

[참고1] 샘플소스(JavaScript)

```

...
CooconiSASNX.execute(inputList, i , function (res1, seq){
    /**
     * 복호화 요청
     */
    $.each(res1 , function (o){
        var r = res1[o];
        if( r['Result'] != null && r['Result'] != "" && typeof r['Result'] =="string"){
            /**
             * 스크래핑 결과 데이터 복호화
             */
            r['Result'] = JSON.parse( decrypt(r['Result']) );
        }
    });

    /**
     * 복호화 결과처리
     */
});

function decrypt(data){
    var input = {} ;
    input['Data'] = data;
    input['Uid'] = CooconiSASNX.OpenUid;
    input['Action'] = CooconiSASNX.OpenAction;
    var param = "data="+ encodeURIComponent( JSON.stringify(input) );

    var rtn = null;
    $.ajax({

```

```
url : "/decode.jsp?",
type: "POST",
data : param,
async: false,
success: function(data, textStatus, jqXHR)
{
    rtn = data['Result'];
},
error: function (jqXHR, textStatus, errorThrown)
{
    alert("error");
}
});
return rtn;
}
```

[참고2] 샘플소스(JSP)

```
<%@page import="java.util.Iterator"%>
<%@page import="com.coocon.securty.util.ISASSeedCBC"%>
<%@page import="org.json.simple.parser.JSONParser"%>
<%@page import="org.json.simple.JSONArray"%>
<%@page import="org.json.simple.JSONObject"%>
<%@ page language="java" contentType="application/json; charset=UTF-8" pageEncoding="UTF-8"%>
<%
    String in_data = (String) request.getParameter("data");

    if( in_data != null){

        /* JSON Parser */
        JSONParser jParser = new JSONParser();

        /* Input JSONArray */
        JSONObject inObj = (JSONObject)jParser.parse(in_data);
        String data = (String)inObj.get("Data");
        String Uid = (String)inObj.get("Uid");
        String Action = (String)inObj.get("Action");
        String decResults;

        if(Uid != null && Action != null && Uid.length() >= 30 && Action.length() >= 30){
            decResults = ISASSeedCBC.decrypt(data , Uid, Action );
        }else{
```

```
        decResults = ISASSeedCBC.decrypt(data);
    }

    JSONObject output = new JSONObject();
    output.put("Result", decResults);
    out.println(output.toJSONString());

} else {
    out.println("{}");
}
%>
```