

GrippinTower Authentication SDK

文書番号	V2. 1-SDK1
------	------------

目次

1. GRIPPINTOWER AUTHENTICATION SDK 概要	3
1.1. システムの構成	3
1.2. GrippinTower Authentication SDK のインストール	4
2. GRIPPINTOWER AUTHENTICATION SDK の関数	5
2.1. 関数一覧	5
2.2. auth	5
2.3. info	6
2.4. 応答コード	6
3. サンプルコード	8
3.1. ASP.NET (C#) サンプルコード (GptwrAuthService.dll)	8
3.2. ASP サンプルコード (GptwrAuthServiceCom.dll)	9
3.3. JSP サンプルコード (GptwrAuthService.jar)	10
3.4. PHP サンプルコード (GptwrAuthService.class.php)	11
3.5. VC++ (MFC) サンプルコード (GptwrAuthService.dll)	12

1. GrippinTower Authentication SDK 概要

1.1. システムの構成

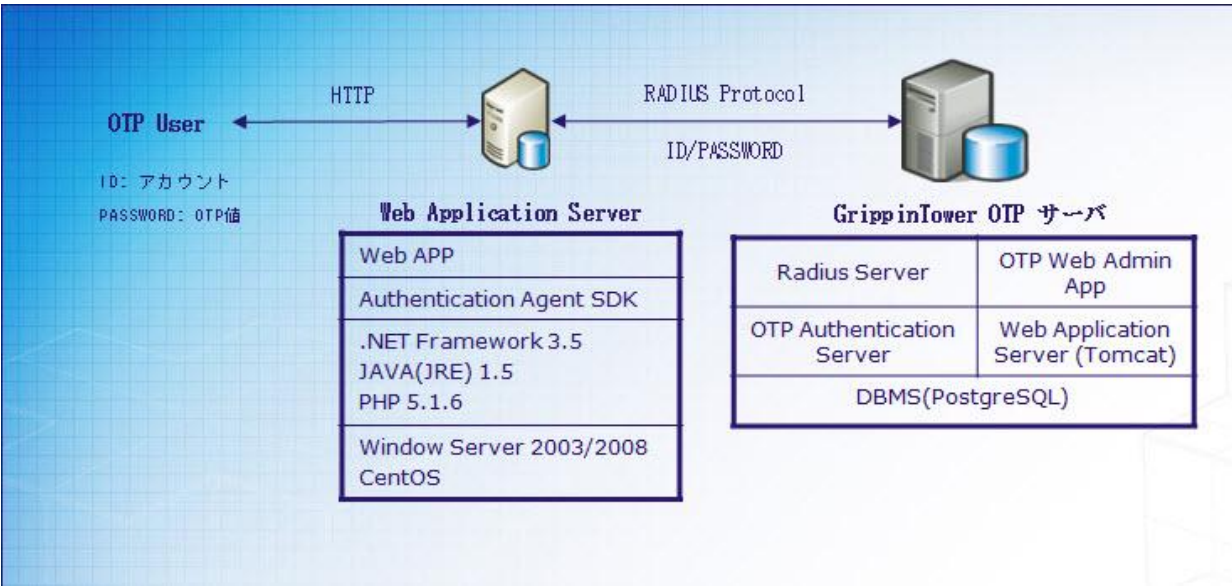
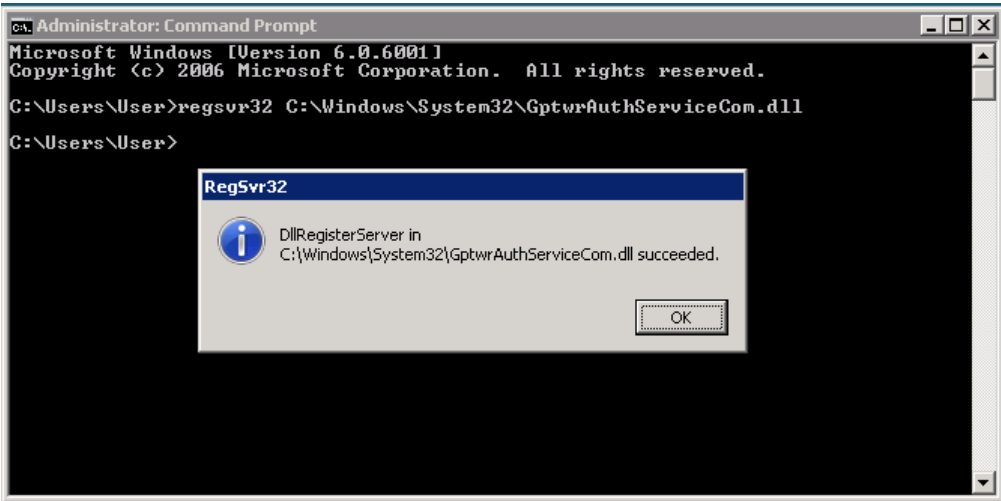


図 1-1 ASP.NET の場合

項目	ライブラリのファイル名		備考
ASP.NET	lib	lib/asp.net/GptwrAuthService.dll	Window Server 2003/2008 IIS 6.0 .NET Framework 3.5
	sample	sample/asp.net	
ASP	lib	lib/asp/GptwrAuthService.dll	Window Server 2003/2008 IIS 6.0
		lib/asp/GptwrAuthServiceCom.dll	
	sample	sample/asp	
JAVA	lib	lib/jsp/GptwrCommon.jar	JAVA(JRE) 1.5
		lib/jsp/GptwrAuthService.jar	
	sample	sample/jsp	
PHP	lib	lib/php/GptwrAuthService.class.php	PHP 5.1.6
		lib/php/radius.class.php	
	sample	sample/php	

1.2. GrippinTower Authentication SDK のインストール

項目	説明
ASP.NET	32bit OS : %SYSTEMROOT%\system32\GptwrAuthService.dll 64bit OS : %SYSTEMROOT%\SysWOW64\GptwrAuthService.dll
ASP	1. 下記のパ스에該当する dll をインストール 1) 32bit OS %SYSTEMROOT%\system32\GptwrAuthService.dll %SYSTEMROOT%\system32\GptwrAuthServiceCom.dll 2) 64bit OS %SYSTEMROOT%\SysWOW64\GptwrAuthService.dll %SYSTEMROOT%\SysWOW64\GptwrAuthServiceCom.dll 2. regsvr32 コマンドで登録 1) 32bit OS regsvr32 %SYSTEMROOT%\system32\GptwrAuthServiceCom.dll 2) 64bit OS regsvr32 %SYSTEMROOT%\SysWOW64\GptwrAuthServiceCom.dll 
JAVA	\$CLASSPATH のパスに GptwrCommon.jar, GptwrAuthService.jar をコピー (ex: WEB-INF/lib)
PHP	-

2. GrippinTower Authentication SDK の関数

2.1. 関数一覧

機能名	関数名	説明
認証要請	auth	認証サーバに OTP 値の有効性を要請
バージョン情報	info	SDK のバージョン情報をリターン

2.2. auth

項目	説明
機能	ユーザが入力した OTP 値の有効性を検証します。
プロトタイプ	<pre># .NET (C#) IntPtr auth(string ip, string port, string secret, string id, string pwd, string timeout) # ASP BSTR auth(BSTR ip, BSTR port, BSTR secret, BSTR id, BSTR pwd, BSTR timeout) # JSP String GptwrAuthService.auth(String ip, String port, String secret, String id, String pwd, String timeout) # PHP string GptwrAuthService::auth(string ip, string port, string secret, string id, string pwd, string timeout)</pre>
パラメータ	[I] ip: 認証サーバの IP アドレス (中継サーバを利用する場合は、中継サーバの IP アドレス) [I] port: 認証サーバの Radius ポート (中継サーバを利用する場合は、中継サーバの Radius ポート)。デフォルト : UDP 1812。 [I] secret: 共有秘密鍵 (管理コンソールで、クライアント登録時に設定した値) [I] id: ユーザの ID (アカウント) [I] pwd: OTP 値 [I] timeout: 認証要請時の応答タイムアウト (単位 : 秒)
応答値	処理結果は以下のフォーマット (文字列) で応答します。 “応答コード#連続認証失敗件数#応答メッセージ”。区切り文字は(#)です。 (.NET の場合、Marshal.PtrToStringAnsi 関数を使用してください。詳細は、サンプルを参考にしてください。)
その他	エラー内容は、管理コンソールの Log/Monitoring>履歴管理>認証履歴照会 画面から確認できます。

2.3. info

項目	説明
機能	SDK のバージョン番号を返します。
プロトタイプ	<pre># .NET (C#) IntPtr info() # ASP BSTR info() # JSP String GptwrAuthService.info() # PHP string GptwrAuthService::info();</pre>
パラメータ	なし (Not Required)
応答値	SDK バージョン番号を文字列で返します。 (.NET の場合、 Marshal.PtrToStringAnsi 関数を使用してください。詳細は、サンプルを参考にしてください。)
その他	-

2.4. 応答コード

応答コード	内容	備考
0	OTP 認証成功	OTP 認証に成功した場合
6000	OTP 認証失敗	OTP 認証に失敗した場合
6001	OTP 認証失敗(OTP 再使用)	既に使用した OTP 値を再使用した場合
6002	OTP 認証失敗(時間の補正が必要)	OTP 値に時間差が発生し、管理コンソールから補正が必要な場合
6007	入力値エラー	SDK の入力値エラー
6008	DB エラー	DB への接続ができなかった場合など
6010	OTP 入力形式エラー	OTP 値を文字、または 6 桁未満の数字を入力した場合
6011	長期間未使用エラー	長期間、使用していないユーザの場合
6012	アカウントが無効(disabled)状態	該当ユーザのアカウント状態が無効 (disabled) の場合
6020	Radius エラー	Radius 認証要請が正しくない場合
6023	ユーザが存在しない、またはアクセス権限エラー	該当ユーザが存在しない、またはアクセス権限がない場合
6024	使用可能なトークンがないエラー	該当ユーザが使用可能なトークン (OTP 機器/AnyOTP) が登録されてない場合

6025	連続認証エラー回数の超過エラー	連続認証エラー回数を超えた場合
6026	認証ポリシー未設定エラー	管理コンソールで、認証ポリシーが設定されていない場合
6027	S/W PIN 未設定エラー	S/W PIN 使用時、S/W PIN が設定されていない場合
6028	S/W PIN エラー	S/W PIN が一致しない場合
6029	EMIL OTP 期間満了エラー	EMAIL OTP の使用期間が満了された場合
6040	サーバ未応答	認証サーバの IP アドレス、ポート番号が間違っているか、管理コンソールに登録されていないクライアントから認証要求があった場合
6041	Radius 応答エラー	認証サーバとクライアントの共有秘密鍵が一致しない場合
6042	SDK 内部エラー	SDK 内部でエラーが発生した場合。例) ソケットエラー

3. サンプルコード

3.1. ASP.NET (C#) サンプルコード (GptwrAuthService.dll)

```
<%@ Page Language="C#" %>
<%@ Import Namespace = "System.Runtime.InteropServices" %>
<script runat=server>

////////////////////////////////////
[DllImport("GptwrAuthService.dll", CharSet = CharSet.Ansi)]
    public static extern IntPtr auth(string ip, string port, string secret, string id, string pwd, string
timeout);
[DllImport("GptwrAuthService.dll", CharSet = CharSet.Ansi)]
    public static extern IntPtr info();
////////////////////////////////////
string serverIp = "127.0.0.1"; // 認証サーバ (中継サーバ) の IP アドレス
string serverPort = "1812"; // Radius ポート番号
string secret = "sharedsecret"; // 共有秘密鍵
string timeout = "3"; // タイムアウト (秒)
string rCode = ""; // 応答コード
string authFailCnt = ""; // 連続認証失敗件数
string rMsg = ""; // 応答メッセージ

public void Page_Load(Object s, EventArgs e){
    string userName = Request.Form["name"]; // ユーザ ID
    string pwd = Request.Form["password"]; // OTP 値
    //////////////////////////////////////
    //API 呼出。OTP 認証要請、SDK バージョン番号
    string result = Marshal.PtrToStringAnsi( auth(serverIp, serverPort, secret, userName, pwd, timeout) );
    string sdkinfo = Marshal.PtrToStringAnsi( info() );
    //////////////////////////////////////
    // 処理結果は、"応答コード#連続認証失敗件数#応答メッセージ" 形式でリターン
    string[] tmp = result.Split('#');
    if (tmp.Length >= 3){
        rCode = tmp[0];
        authFailCnt = tmp[1];
        rMsg = tmp[2];
    }
    if ( "0" == rCode){
        // TODO: OTP 認証成功処理
    } else {
        // TODO: OTP 認証失敗処理
    }
    //////////////////////////////////////
}
</script>
```


3.2. ASP サンプルコード (GptwrAuthServiceCom.dll)

```
<%
Dim serverIp          ' 認証サーバ（中継サーバ）の IP アドレス
Dim port              ' Radius ポート番号
Dim secret            ' 共有秘密鍵
Dim timeout           ' タイムアウト（秒）
Dim result            ' SDK 認証結果
Dim info              ' SDK バージョン番号
Dim rCode             ' 応答コード
Dim authFailCnt       ' 連続認証失敗件数
Dim rMsg              ' 応答メッセージ
Dim tmp

serverIp = "127.0.0.1"
port = "1812"
secret = "sharedsecret"
timeout = "3"
' //////////////////////////////////////
' // API 呼出。OTP 認証要請、SDK バージョン番号
Set obj = Server.CreateObject( "GptwrAuthServiceCom.GptwrAuthService" )
result = obj.auth(serverIp, port, secret, Request("name"), Request("password"), timeout)
info = obj.info()
' //////////////////////////////////////
' //////////////////////////////////////
' //処理結果は、"応答コード#連続認証失敗件数#応答メッセージ" 形式でリターン
tmp = Split(result, "#")
If ( UBound(tmp) >= 2) Then
    rCode = tmp(0)
    authFailCnt = tmp(1)
    rMsg = tmp(2)
End If

If ( "0" = rCode) Then
    ' // TODO: OTP 認証成功処理
Else
    ' // TODO: OTP 認証失敗処理
End If
' //////////////////////////////////////

Set obj = Nothing
%>
```

3.3. JSP サンプルコード (GptwrAuthService.jar)

```
<%@ page import="com.dreammirae.gt.radius.client.*" %>
<%
    //////////////////////////////////////
    String serverIp = "127.0.0.1";           //認証サーバ（中継サーバ）の IP アドレス
    String port = "1812";                   // Radius ポート番号
    String sharedSecret = "sharedsecret"; //共有秘密鍵
    String timeout = "3";                   //タイムアウト（秒）
    String result = null;                   // SDK 認証結果
    String info = null;                     // SDK バージョン番号
    String rCode = null;                    //応答コード
    String authFailCnt = null;              //連続認証失敗件数
    String rMsg = null;                     //応答メッセージ
    String name = request.getParameter("name"); // ユーザ ID
    String password = request.getParameter("password"); // OTP 値
    //////////////////////////////////////

    //////////////////////////////////////
    // API 呼出。OTP 認証要請、SDK バージョン番号
    result = GptwrAuthService.auth(serverIp, port, sharedSecret, name, password, timeout);
    info = GptwrAuthService.info();
    //////////////////////////////////////

    //////////////////////////////////////
    //処理結果は、“応答コード#連続認証失敗件数#応答メッセージ” 形式でリターン
    String[] tmp = result.split("#");
    if ( tmp.length >= 3 ){
        rCode = tmp[0];
        authFailCnt = tmp[1];
        rMsg = tmp[2];
    }
    if ( "0".equals(rCode) ){
        // TODO: OTP 認証成功処理
    }else{
        // TODO: OTP 認証失敗処理
    }
    //////////////////////////////////////
%>
```

3.4. PHP サンプルコード (GptwrAuthService.class.php)

```
<?php

////////////////////////////////////
require_once('GptwrAuthService.class.php' );
////////////////////////////////////

////////////////////////////////////
$serverIp = '127.0.0.1';           //認証サーバ（中継サーバ）の IP アドレス
$port = '1812';                   // Radius ポート番号
$secret = 'sharedsecret';         //共有秘密鍵
$timeout = '3';                   //タイムアウト（秒）
$username = $_POST['name'];        // ユーザ ID
$password = $_POST['password'];    // OTP 値
////////////////////////////////////

////////////////////////////////////
// API 呼出。OTP 認証要請、SDK バージョン番号
$result = GptwrAuthService::auth( $serverIp, $port, $secret, $username, $password, $timeout);
$info = GptwrAuthService::info();
////////////////////////////////////

////////////////////////////////////
//処理結果は、“応答コード#連続認証失敗件数#応答メッセージ” 形式でリターン
list($rCode, $authFailCnt, $rMsg) = split('#', $result);
if ( "0" == $rCode){
    // TODO: OTP 認証成功処理
}else{
    // TODO: OTP 認証失敗処理
}
////////////////////////////////////

?>
```

3.5. VC++ (MFC) サンプルコード (GptwrAuthService.dll)

```
////////////////////////////////////
typedef char*(*auth)(
    const char *serverIp, const char *port, const char *secret, const char *username,
    const char *password, const char *timeout );
typedef char*(*info)(void);
info lpInfoFunc;
auth lpAuthFunc;
HINSTANCE hDll;
////////////////////////////////////

////////////////////////////////////
hDll = LoadLibrary("GptwrAuthService.dll");
if (!hDll) {
    AfxMessageBox("該当する dll ファイルがありません。"); return;
}
lpInfoFunc=(info)GetProcAddress(hDll,"info");
lpAuthFunc=(auth)GetProcAddress(hDll,"auth");
if (lpInfoFunc==NULL || lpAuthFunc ==NULL) {
    AfxMessageBox("該当する関数がありません。!"); return;
}
////////////////////////////////////

////////////////////////////////////
// API 呼出。OTP 認証要請、SDK バージョン番号
CString foo;
foo.Format("%s", lpAuthFunc("127.0.0.1", "1812", "secret", "user", "123456", "3"));
// TODO: 処理結果は、"応答コード#連続認証失敗件数#応答メッセージ" 形式でリターン
CString rCode;
CString authFailCnt;
CString rMsg;
AfxExtractSubString( rCode, foo, 0, '#' );
AfxExtractSubString( authFailCnt, foo, 1, '#' );
AfxExtractSubString( rMsg, foo, 2, '#' );
if ( "0" == rCode ){
    AfxMessageBox("success"); // TODO: OTP 認証成功処理
}else{
    AfxMessageBox("failed"); // TODO: OTP 認証失敗処理
}
////////////////////////////////////
////////////////////////////////////
FreeLibrary(hDll);
////////////////////////////////////
```