

SANGFOR_SSL_v7.6.8_CONF

HTTP(S)短信认证配置指导



深信服智安全
SANGFOR SECURITY

2019 年 11 月

修订历史					
编号	修订内容简述	修订日期	修订前版本号	修订后版本号	修订人
1	编写	20191104		V1.0	qjj

■ 版权声明

本文中出现的任何文字叙述、文档格式、插图、照片、方法、过程等内容，除另有特别注明，版权均属深信服所有，受到有关产权及版权法保护。任何个人、机构未经深信服的书面授权许可，不得以任何方式复制或引用本文的任何片断。

目录

第 1 章 概述.....	2
第 2 章 准备工作.....	2
第 3 章 配置步骤.....	2
3.1 认证配置.....	3
3.2 服务器配置.....	3
3.2.1 模式.....	4
3.2.2 多步请求.....	10
3.2.3 请求配置.....	10
3.2.4 响应配置.....	13
3.2.5 授权策略.....	错误！未定义书签。
第 4 章 案例分享.....	16
4.1 案例一：HTTP(S)普通模式下与统一认证平台对接.....	16
4.2 案例二：HTTP(S)高级模式下与统一认证平台对接.....	错误！未定义书签。

第 1 章 概述

SSL 与认证服务器对接，认证使用 HTTP/HTTPS 的协议完成。从而实现 SSL 上使用短信验证码的辅助认证完成双因子认证。

第 2 章 准备工作

获取第三方认证服务器的接口文档

第 3 章 配置步骤

在 SSL 设备控制台【SSL VPN 设置】-【认证设置】-【辅助认证】-【第三方辅助认证】中点击 **新建** 按钮 选择 **HTTP(S)验证码认证**。



3.1 认证配置

认证配置

[查看详细配置文档](#)

认证名称: *
描述:
重新发送间隔: 30 * 秒钟 (10-3600之间)
验证码有效期: 1 * 分钟 (1-1440之间)
国家代码: 86 * 例如, 中国的为86
发送验证码方式: ☒ 通过手机号码标识用户, 并发送验证码 ☐ 通过用户其他信息标识用户, 并发送验证码
短信认证提示: 验证码已发送到您的手机: {{user.mobile_phone}}, 请查收! * 变量: 手机号码: {{user.mobile_phone}}; 用户名{{user.user_name}}
短信内容: 您好, {{user.user_name}}, 本次登录验证码为: {{config.sms_code}}, 有效期{{config.sms_interval}}分钟
[恢复默认短信内容模板](#)
参数变量说明:
登录用户名: {{user.user_name}}
登录IP: {{user.login_ip}}
验证码: {{config.sms_code}}
验证码的有效期: {{config.sms_interval}}

认证名称: 自定义认证的名称, 如 第三方 HTTP 认证对接

描述: 可以选填认证的描述

重新发送间隔: 自定义短信重新发送的间隔

验证码有效期: 自定义验证码有效时间

国家代码: 定义手机号的国际国家代码

发送验证码方式: 发送验证码的方式有两种, 有以下区别

1、通过手机号码标识用户; 要求用户必须配置手机号码 (如未配置, 登录时无法进行验证码认证)

2、通过其他信息标识用户, 不强制要求用户配置手机号码; 进行验证码认证时, 可通过用户名、用户描述等信息识别用户, 并发送验证码

短信认证提示: 弹出短信认证提示框时的提示内容

短信内容: 自定义短信内容

3.2 服务器配置

按照统一认证服务器提供的 HTTP(S) 协议内容进行构造配置。

服务器配置

01 第一次请求 +

请求配置 响应配置

自定义请求变量

请求地址: GET https:// *

请求超时: 15 * 秒 (5-60之间)

请求URL参数

添加 删除 编辑

请求头部

请求Cookie

字段名	值
-----	---

3.2.1 模式

服务器配置分为普通模式和高级模式。普通模式和高级模式下都能使用模板变量、模板函数，但是使用的方式存在差异。

普通模式：在普通的配置页面即可使用普通模式下的模板变量。

高级模式：在“自定义请求变量”和“数据响应预处理”中可使用高级模式中的模板变量。

注：通过模板函数可以对模板变量或者固定值进行相应的加解密、编解码操作

3.2.1.1 普通模式

普通模式下可以使用的模板变量包括：user、config、env。（详见表 1）

模板变量名称	普通模式下使用方式	说明
user	{{user.data}}	用户在线信息体中的变量和客户端提交的信息 说明：data 支持的扩展参数见表 1.1

config	{{config.data}}	各认证插件中，自有的全局配置变量。如：HTTP(S)验证码认证中的全局配置项《短信内容》 说明：data 支持的扩展参数见表 1.2
env	{{env.data}}	高级模式中脚本定义或响应解析配置中自定义的 说明：data 支持的参数为自定义的变量

表 1 普通模式下模板变量

字段	使用方式	说明	备注
user_name	{{user.user_name}}	用户名	在主认证时输入了用户名，主认证及后续的认证都能使用。
user_psw	{{user.user_psw}}	用户密码	在主认证时输入了用户密码，主认证及后续的认证都能使用。
user_note	{{user.user_note}}	用户描述	如果是本地用户，在新建用户时配置了描述信息，使用这个用户的登录时就可以使用；不论是本地用户还是外部用户只要主认证结果有返回描述信息，则后续的认证也可以使用。
login_ip	{{user.login_ip}}	登录 IP	在整个认证过程中都可以使用。
mobile_phone	{{user.mobile_phone}}	手机号码	如果是本地用户，

			在新建用户时配置了手机号码，使用这个用户的登录时就可以使用；不论是本地用户还是外部用户只要主认证结果有返回手机号码，则后续的认证也可以使用。
client_type	{{user.client_type}}	客户端类型	在整个认证过程中都可以使用。
lang_type	{{user.lang_type}}	语言（中文、英文）	在整个认证过程中都可以使用。
hid	{{user.hid}}	硬件特征码	在使用短信审批硬件特征码，才能获取到。
svpn_inputtoken	{{user.svpn_inputtoken}}	令牌码	在使用令牌辅助认证时，输入令牌码，才能获取到。
macaddress	{{user.macaddress}}	mac 地址	硬件特征码后面有短信认证或者令牌认证，才能获取到。
hostname	{{user.hostname}}	主机名	硬件特征码后面有短信认证或者令牌认证，才能获取到。

表 1.1 user 模板变量支持的扩展参数

字段名	使用方式	说明
native_code	{{config.native_code}}	国家码

sms_content	{{config.sms_content}}	短信内容
sms_tips	{{config.sms_tips}}	提示语
sms_interval	{{config.sms_interval}}	发送间隔
sms_period	{{config.sms_period}}	短信有效期
sms_code	{{config.sms_code}}	短信验证码

表 1.2 config 模板变量支持的扩展参数

普通模式下可以使用的模板函数包括：base64Encode、base64Decode、md5、sha1、encodeURIComponent、decodeURIComponent、encodeURIComponent、decodeURI。（详见表 2）

模板函数名称	普通模式下使用方式	说明
base64Encode	{{base64Encode:data}}	对模板变量（参考表 1）进行 base64 编码。 如：{{base64Encode:user.user_name}}，对用户名进行 base64 编码。
base64Decode	{{base64Decode:data}}	对模板变量（参考表 1）进行 base64 解码。 如：{{base64Decode:user.user_name}} 对用户名 base64 解码。
md5	{{md5:data}}	对模板变量（参考表 1）计算 md5 值。 如：{{md5:user.user_pwd}} 计算用户密码的 md5 值。 注：默认小写 32 位
sha1	{{sha1:data}}	对模板变量（参考表 1）计算 sha1 值。 如：{{sha1:user.user_pwd}} 计算用户密码的 sha1 值。
encodeURIComponent	{{encodeURIComponent:data}}	对模板变量（参考表 1）进行 url 编码。 不会被此方法编码的字符：- _ . ! ~ * ' () 。

		如： <pre>{{encodeURIComponent:user.user_note}}</pre> ，对用户描述信息 url 编码。
decodeURIComponent	<pre>{{decodeURIComponent:data}}</pre>	对模板变量（参考表 1）进行 url 解码。 如： <pre>{{decodeURIComponent:user.user_note}}</pre> ，对用户描述信息 url 解码。
encodeURIComponent	<pre>{{encodeURIComponent:data}}</pre>	该方法不会对 ASCII 字母和数字进行编码，也不会对这些 ASCII 标点符号进行编码： - _ . ! ~ * ' () 。 该方法的目的是对 URI 进行完整的编码，因此对以下在 URI 中具有特殊含义的 ASCII 标点符号，encodeURIComponent() 函数是不会进行转义的： ; / ? : @ & = + \$, # 如： {{encodeURIComponent:user.user_note}} ，对用户描述信息 url 编码。
decodeURI	<pre>{{decodeURI:data}}</pre>	对模板变量（参考表 1）进行 url 解码。 如： {{decodeURI:user.user_note}}， 对用户描述信息 url 解码。

表 2 模板参数支持的算法

3.2.1.2 高级模式

高级模式下可以使用的模板变量包括：user、config、env、response。（详见表 3）

模板变量名称	高级模式下使用方式	说明
user	user.data	用户在线信息体中的变量和客户端提交的信息

		说明：data 支持的扩展参数见表 1.1
config	config.data	各认证插件中，自有的全局配置变量。如：HTTP(S) 验证码认证中的全局配置项《短信内容》 说明：data 支持的扩展参数见表 1.2
env	env.data	高级模式中脚本定义或响应解析配置中自定义的 说明：data 支持的参数为自定义的变量
response	response.data	统一认证服务器返回的响应数据。 支持： response.body 包体数据 response.header 头部数据

表 3 高级模式模板变量

高级模式下可以使用的模板函数包括：

模板函数名称	高级模式使用方式	说明
crypto	Crypto.function()	Nodejs 的加解密库，function() 的支持见： https://github.com/brix/crypto-js
iconv-lite	IconvLite.function() ()	Nodejs 的编解码库，function() 的支持见： https://github.com/ashtuchkin/iconv-lite
xml2js	Xml2js.function()	Nodejs 的 Xml 转 json 库，function() 的支持见： https://github.com/Leonidas-from-XIV/node-xml2js 注意：此模板函数只能在响应解析配置的高级模式中使用
cheerio	解析复杂的 xml 格式	具体使用请参考库文档： https://cheerio.js.org
url	解析 URL	具体使用请参考库的文档： http://nodejs.cn/api/url.html

表 4 高级模式使用的库

3.2.2 多步请求

点击  按钮可以添加多次认证交互，最多支持 5 次。一般应用于有多次认证相互的情况下，下一次的请求依赖于上一步的认证结果。



点击  按钮可以删除。



在多步请求中，每步请求都会默认把前面返回的 cookie 带上。如某个请求有 3 步，第一次返回了 cookie: aa=bb，则第二次请求时会把这个 cookie 带上，第三步时也会带上它。

3.2.3 请求配置

3.2.3.1 GET 方式

请求配置

响应配置

自定义请求变量

请求地址:

GET

https://

*

请求超时:

15

* 秒 (5-60之间)

请求URL参数

添加

删除

编辑

请求头部	字段名	值
请求Cookie		

请求地址：GET，以 GET 方法向配置的地址发送 HTTP(S) 请求。后面填写请求的地址，支持 http 和 https 协议

请求超时：设置超时时间，默认 15 秒。

请求 URL 参数：配置 HTTP(S) 协议的请求 URL 参数，其中字段名的值可以填写固定值或者模板变量（见表 1）；对添加的数据可以进行删除、编辑操作。

请求头部：配置 HTTP(S) 协议的请求头部，其中字段名的值可以填写固定值或者模板变量（见表 1）；对添加的数据可以进行删除、编辑操作。

请求 Cookie：配置 HTTP(S) 协议的请求 cookie，其中字段名的值可以填写固定值或者模板变量（见表 1）；对添加的数据可以进行删除、编辑操作。

注：默认编码格式为 UTF-8，且无需配置。

3.2.3.2 POST 方式

The screenshot shows the 'Request Configuration' (请求配置) tab in the Sangfor Security Center. The 'Request Address' (请求地址) is set to POST and https://. The 'Request Timeout' (请求超时) is set to 15 seconds. The 'Request URL Parameters' (请求URL参数) section is expanded, showing a table with columns for 'Field Name' (字段名) and 'Value' (值). There are also expandable sections for 'Request Headers' (请求头部), 'Request Cookies' (请求Cookie), and 'Request Body' (请求体).

请求地址：POST，以 POST 方法向配置的地址发送 HTTP(S) 请求。

请求超时：设置超时时间，默认 15 秒。

请求 URL 参数：配置 HTTP 协议的请求 URL 参数，其中字段名的值可以填写固定值或者模板变量（见表 1）；对添加的数据可以进行删除、编辑操作。

请求头部：配置 HTTP 协议的请求头部，其中字段名的值可以填写固定值或者模板变量

（见表 1）；对添加的数据可以进行删除、编辑操作。

请求 Cookie：配置 HTTP 协议的请求 cookie，其中字段名的值可填写固定值或者模板变量（见表 1）；对添加的数据可以进行删除、编辑操作。

请求体：配置 HTTP 协议 POST 方法的请求体，其内容可以填写固定值或者模板变量（见表 1）。

注：默认编码格式为 UTF-8，且无需配置。

服务器配置

01 第一次请求 +

请求配置 响应配置

自定义请求变量

请求地址: POST https:// *

请求超时: 15 * 秒 (5-60之间)

请求URL参数 例如: username={{user.user_name}}&password={{user.user_pwd}}

请求头部

请求Cookie

请求体

自定义变量：若预定义的变量不满足需求时，可以使用自定义请求变量，可通过 node.js 脚本来自定义新的模板变量，并能在请求配置中被使用。

3.2.4 响应配置

格式：将统一认证服务器返回的结果数据，按照 XML 或者 JSON 的格式进行解析。

添加：配置响应解析规则。即将统一认证服务器返回的结果数据，按照 XML 或者 JSON 格式解析字段名的值，并将值保存至对应配置的变量名（支持的变量见下表）中。

可存储的变量	说明
user.mobilephone	手机号码
user.user_note	用户描述
user.grp_id	所属用户组
user.role_ids	角色
env.data	env环境变量

表 5 响应解析存储变量名

删除：删除选择的响应解析规则数据

编辑：编辑选择的响应解析规则数据

注意：在响应解析存储变量名（表 5）表中，第三方返回的手机号、用户描述、所属用户组、角色、环境变量，只在当前整个认证周期内有效，第二次认证时会重置。

认证成功条件：配置认证成功条件表达式，即只有当表达式成立时，才认为当次认证

请求通过。表达式中可以填写固定值或者模板变量，

注意：应当谨慎填写此表达式，避免原本认证失败的却认证成功，建议是多种情况的账号测试，如认证成功的账号、认证失败的账号。

服务器配置

01 第一次请求 +

请求配置 响应配置

格式: ☒ JSON ☐ XML [响应数据预处理](#)

+ 添加 - 删除 编辑

响应解析字段名	本地变量

认证成功条件: 等于 * [i](#)

☒ 启用Cookie代理下发 [i](#)

子域名: * 填写子域名地址

启用 Cookie 代理下发：启用后，VPN 将会把认证过程中统一认证服务器返回结果中的 cookie，下发至 VPN 客户端。此功能能可以用来实现单点登录。

子域名：VPN 代理下发 cookie 的域名，需要保证和 VPN 的域名为同根域。



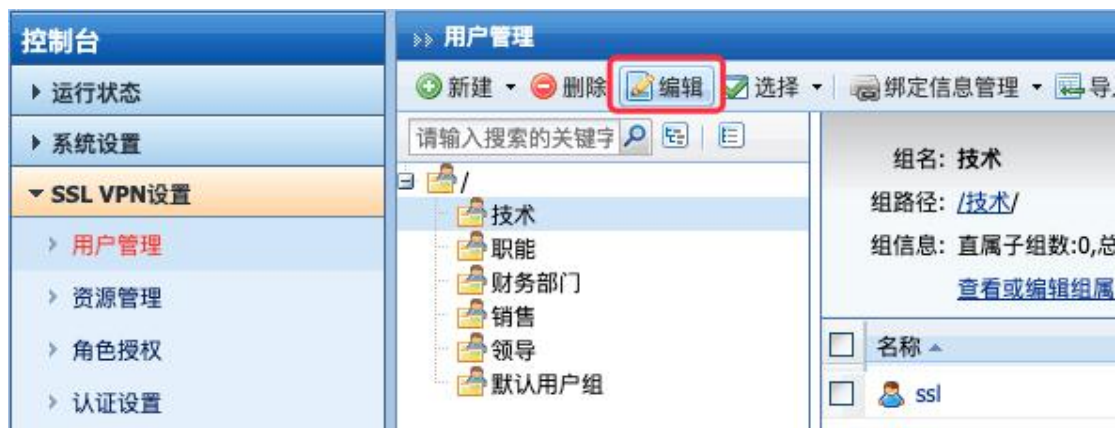
响应数据预处理：启用后，可通过 nodejs 脚本来对统一认证服务器返回的结果数据进行预处理。高级模式中预处理后的数据，将再被配置的响应解析规则进行解析。

注意：结果是保存在 `result` 这个字段中，并且 `result` 默认为 json 格式，我们在处理数据时也是从这个字段中提取数据，使用方法 `result.code = 0` 或者 `result["code"] = 0`，当存在多级时 `result.xxx.xxx = "xxx"`。只有当认证成功后才会设置用户属性字段。

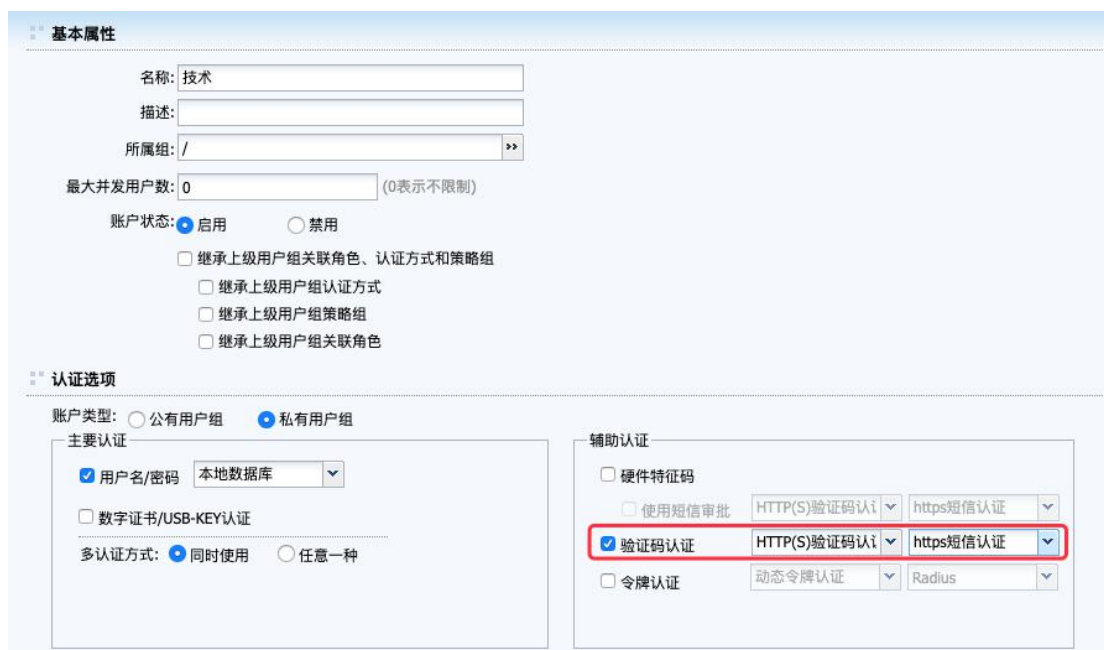
3.3 关联用户/用户组

配置完成后，需要在控制台【SSL VPN 设置】-【用户管理】中将认证系统与用户/用户组关联起来。

选择需要关联的用户/用户组，点击 **编辑** 按钮



在辅助认证中选择“HTTP(S)验证码认证”然后选择对接好的认证系统。



注意：若认证选项是灰色不可配置状态，这是勾选了“继承上级用户组认证方式”取消勾选即可配置。

第4章 案例分享

4.1 案例一：HTTP(S)高级模式下与短信平台对接

某公司在主认证后还需要使用短信功能作为辅助认证，该短信认证接口如下：

4.1 接口定义

接口名称	短信发送接口
接口描述	处理短信发送（支持普通短信、长短信）
请求内容	INDEX: http://sms.深信服.com/yktsms/send BODY: appid=QP2Szdmbwby6z3qLli9lAg0WdpACNNuO5&mobile=13800138000&msg=urlencode(短信内容, utf8)&sign=md5(appid+mobile+msg+appkey) 完整URL实例，如： http://sms.深信服.com/yktsms/send?appid=QP2Szdmbwby6z3qLli9lAg0WdpACNNuO5&mobile=13800138000&msg=%E3%80%90%E5%90%88%E4%BC%97%E7%A7%91%E6%8A%80%E3%80%91%E6%82%A8%E5%A5%BD%2C%E4%BD%A0%E7%9A%84%E9%AA%8C%E8%AF%81%E7%A0%81%E6%98%AF123456&sign=844794001e535231edf4a979c615e1ce
请求方式	POST
应答格式	字符串（逗号分隔）
接口协议	HTTP

4.2 输入参数

字段名称	是否必填	说明	数据类型	长度
appid	必填	用户接入编号，接口信息唯一标识，由深信服分配	String	32
mobile	必填	需要发送的手机号(多个号码以英文逗号“,”分隔) 一次性最多 100 个号码	String	<=400
msg	必填	短信内容为utf-8编码格式，并且用urlencode的编码 示例：java: URLEncoder.encode(content, "utf-8")	String	<=500
sign	必填	由三个参数的值和appkey的值，用字符串方式直接拼接，然后用md5 算法加密（ MD5、32小写 ） md5(appid+mobile+msg+appkey) 中间无需空格、无需“+”号 appid、appkey 由深信服分配	String	32

4.3 输出参数

字段名称	说明	数据类型	长度
code	逗号分隔第一位，状态码。0表示提交成功，非 0 表示提交失败	String	<=4
taskid	逗号分隔第二位，任务 ID，接口返回的 taskid	String	<=11
msg	逗号分隔第三位，状态描述	String	<=50

参数示例：

成功返回示例：

0,11906141154338793,成功

失败返回示例：

-2,0,appId对应的商户信息不存

配置如下所示：

1. 认证配置：

认证配置

[查看详细配置文档](#)

认证名称:

描述:

重新发送间隔: 30 * 秒钟 (10-3600之间)

验证码有效期: 1 * 分钟 (1-1440之间)

国家代码: 86 * 例如，中国的为86

发送验证码方式: ☒ 通过手机号码标识用户，并发送验证码
☐ 通过用户其他信息标识用户，并发送验证码

短信认证提示: 验证码已发送到您的手机: {{user.mobile_phone}}, 请查收!

* 变量: 手机号码: {{user.mobile_phone}}; 用户名{{user.user_name}}

短信内容: 您好, {{user.user_name}}, 本次登录验证码为: {{config.sms_code}}, 有效期{{config.sms_interval}}分钟

[恢复默认短信内容模板](#)

参数变量说明:

登录用户名: {{user.user_name}}

登录IP: {{user.login_ip}}

验证码: {{config.sms_code}}

验证码的有效期: {{config.sms_interval}}

2. 配置请求方法、接口地址和请求头部

服务器配置

01 第一次请求

+

请求配置

响应配置

自定义请求变量

请求地址:

POST

http://sms. .com/yktsms/send

 *

请求超时:

15

 * 秒 (5-60之间)

> 请求URL参数

添加

删除

编辑

字段名	值
Content-Type	application/x-www-form-urlencoded

> 请求头部

> 请求Cookie

> 请求体

发送测试信息

3. 按照接口文档配置请求体，由于 sgin 的必填字段需要进行 MD5 加密，使用自定义请求变量。



4. 响应配置：该服务器返回的是一个非 json 和 xml 格式的字段串，我们需要对返回的数据做预处理操作。在此处我们检测其返回的字符串是否带有“成功”这个关键字。



5. 判断结果

服务器配置

01 第一次请求 

请求配置

响应配置

响应数据预处理

格式: ☒ JSON ☐ XML

 添加  删除  编辑

响应解析字段名	本地变量
---------	------

认证成功条件: 等于 * 

☐ 启用Cookie代理下发 

发送测试信息