Verification Code API

Contents

1. Service

LINK Mobility UK provides access to its SMS gateway service using the Verification Code API defined in this document to enable customer applications to send randomly generated codes to mobiles for the purpose of user authentication as two-factor authentication (2FA) and one-time passwords.

The service will generate the code and send it to the mobile ad can optionally return the code to the customer application or a reference that can later be used to verify a code.

In order to use the API in this document it is necessary to have an account with LINK Mobility UK.

1.1. Obtaining an account

To open an account to use the Verification code API please contact sales.uk@linkmobility.com.

2. Example Code

Example code to assist with verification codes can be found in the Examples section located at https://linkmobility.co.uk/developer/examples/.

A simple example of sending a verification code that has a length of 10 alpha (a-z) characters is shown below:

https://<server>:<port>/sendvcode/?clientid=vc*joe3920&destaddr=4478920111000&type=alpha&length=10&password=bce87eb9d0

3. Sending verification codes via SMS

This document includes the actions that can be used to send verification codes to mobiles on networks supported by the service. These actions create single SMS messages depending on the parameters.

A regular HTTP POST or GET can be used by a customer's application to perform an action to send a verification code using the Verification Code API. The parameters necessary for each action are shown in the following table (M – mandatory; O – optional):

Parameter	Action		
	sendvcode	sendvcode-ref	verifycode
clientid	M	M	M
password/key	M	M	M
destaddr	M	M	M
type	M	M	
length	M	M	
refid			M
vcode			M

The URL that is used has the following format:

https://<server>[:<port>]/<action>/ (secure HTTP)

Following the set-up of your account the actual server, port, clientid, password and secret for use in the actions will be provided by LINK Mobility UK.

3.1. Security and authentication

The encryption of communication between the application making the request and LINK Mobility UK's systems is achieved through HTTPS and it is recommended that HTTPS be used for all transactions.

There are two methods available to the application for authenticating to use this service: password or MD5. Either the password or the MD5 method MUST be used when submitting a HTTPS request and are passed in the password or key parameter respectively.

Password	A password is included in the parameters (the password parameter) submitted with the HTTPS request. The password is assigned when the account is provisioned by LINK Mobility UK. It should be noted that this password is passed

	"in the clear" (unencrypted) when using HTTP. If HTTPS is used the password will be encrypted.
MD5	An MD5 message digest is included in the parameters (the key parameter) submitted with the HTTP request. The secret and parameters are concatenated together to provide the input to the MD5 calculation. Note that the secret is **never** passed between the application and LINK Mobility UK "in the clear" – it is used as a seed in the MD5 calculation along with other parameters. The MD5 "message digest" is a 16-byte output calculated from a secret known to both the application sending the HTTP request and LINK Mobility UK's server, and other parameters associated with the action. The actual parameters and how the input to the MD5 calculation should be performed are included in the sections below describing the supported actions. The MD5 algorithm is included in various SDKs and programming languages (C/C++, Delphi, Java, JavaScript, PHP, Perl, VB). The MD5 algorithm is defined in http://www.ietf.org/rfc/rfc1321.txt.

The IP address of the application making the HTTP request can optionally be checked against the customer's account details. Requests coming from an application connecting from an IP address other than that setup in an account will be rejected. IP addresses for applications authorised to make use of a customer's account must be provided to LINK Mobility UK so that they can be entered into the account configuration.

3.2. Response from actions

After an action has been sent using HTTPS by the customer application, the action will be performed and a response given back to the customer's application. The response given will indicate the success or failure of the request and will be returned within the body (or data part) of the HTTPS response.

The "FAIL AUTH" response indicates that your clientid and password or clientid and key combination are incorrect, or that the IP address from where the HTTP request is being made is not permitted (if restriction of IP addresses is set-up for the account).

- *response* = [*success-response* | *failure-response*]
- *success-response* = SUCCESS <SP> *code* <SP> *msg-id*

- *success-response* = SUCCESS <SP> *ref-id* <SP> *msg-id*
- *msg-id = {string of up to 64 printable characters}*
- *ref-id = {string of 24 printable characters}*
- *failure-response* = FAIL [<SP> AUTH]

3.3. Actions

3.3.1. sendvcode – Send code to mobile user and return code sent

PURPOSE

Generates a code based on the type and length and sends the code to the mobile. The code generated is returned in the response.

PARAMETERS

Parameter	Description
destaddr	Mobile telephone number
type	numeric (0-9) alpha (a-z) alpha-caps (a-z,A-Z) alpha-caps-only (A-Z) alpha-caps-numeric (a-z,A-Z,0-9) alpha-caps-only-numeric (A-Z,0-9) alpha-numeric (a-z, 0-9) alpha-numeric-symb (a-z, 0-9, !#$%&*?{}) alpha-caps-numeric-symb (a-z,A-Z,0-9,!#$%&*?{}) alpha-caps-only-numeric-symb (A-Z,0-9,!#$%&*?{})
length	Length of the code to be generated (3-15 characters)
key	<secret><destaddr>

3.3.2. sendvcode-ref – Send code to mobile user and return reference to this code

PURPOSE

Generates a code based on the type and length and sends the code to the mobile. The reference to the code generated is returned in the response. The customer application is not provided with the actual generated code and uses the verifycode action to verify the correctness of the code subsequently provided by the user.

PARAMETERS

Parameter	Description
destaddr	Mobile telephone number
type	numeric (0-9) alpha (a-z) alpha-caps (a-z,A-Z) alpha-caps-only (A-Z) alpha-caps-numeric (a-z,A-Z,0-9) alpha-caps-only-numeric (A-Z,0-9) alpha-numeric (a-z, 0-9) alpha-numeric-symb (a-z, 0-9, !#$%&*?{}) alpha-caps-numeric-symb (a-z,A-Z,0-9,!#$%&*?{}) alpha-caps-only-numeric-symb (A-Z,0-9,!#$%&*?{})
length	Length of the code to be generated (3-15 characters)
key	<secret><destaddr>

3.3.3. verifycode – Verify a code is what was sent to mobile user

PURPOSE

Verifies the code provided by the user is the code that was generated for a given reference.

PARAMETERS

Parameter	Description
destaddr	Mobile telephone number(s)
refid	Reference (24 characters) to the code that was sent to the user and returned by sendvcode-ref
vcode	Code to be verified that was provided by user
key	<secret><destaddr>

3.4. Validity period

If a verification code is not received by a mobile within 5 minutes of the sendvcode or sendvcode-ref request then the mobile will most likely not receive the code. Non-delivery of the code after 5 minutes can however not be guaranteed.

Note that generated verification codes have no temporal association within the service. It is down to the customer application to associate a verification code with the time of a particular code generation request from their application.