TeleSign

Migration Guide

Facebook Account Kit to TeleSign SMS Verify

Table of Contents

TeleSign

Your Guide to Successfully Migrating From Facebook Account Kit

If you are a developer or the business owner of a web property, you have long been aware of the importance of verifying your users using multifactor authentication (2FA) to keep both your platform and your users safe. A popular way to do this has been to utilize Facebook's popular Account Kit... until now. Account Kit is in the midst of a shut down. We're going to tell you what you need to know and what to do next to keep your platform safe and secure.

What Is The Timeline?

September 9th, 2019

As of September 9th, 2019 no new apps were able to integrate with Account Kit. Existing integrations continued to function as normal.

December 9th, 2019

On December 9th, 2019 the daily SMS limit will be reduced to 1,000 messages per app ID per day. Looking at a general verification use case, this means 1,000 one-time passcode SMS messages per day in a standard 2FA sequence.

March 9th, 2020

On March 9th, 2020 Account Kit will be deprecated and no longer function. Business owners will be able to retrieve their app's user data for an additional 90 days until June 9th, 2020. After that Account Kit will be completely dead. Failure to migrate from this time will cause your verification system to fail.

TeleSign

Migrate to TeleSign SMS

Enjoy the Upgrade

TeleSign is the pioneer in phone-based verification and we have been supporting 21 of the top 25 global web properties for years. TeleSign's flagship verification solution provides a safe, GDPR and CCPA compliant alternative to Account Kit that can scale effectively worldwide. Our team of experts has decades of experience protecting customers from the start-up level all the way to enterprise.

TeleSign's SMS Verify API delivers patented phone based verification and two-factor authentication using a time-based one time passcode sent over SMS.

High Quality, Direct-to-Carrier Routes

Connects directly to hundreds of carriers around the world and ensure the highest delivery and conversion rate across multiple routes.

Smart Message Splitting

Prevent the breaking of critical pieces of information such as URLs, email addresses and more to ensure messages are delivered and viewed as intended.

Long Message Support

Separates SMS message over 160 characters based on device type and operator requirements and sends it to the user's handset with instructions to reassemble the message back in the correct order as one SMS.

Message Encoding

Utilize broad character sets and deliver different message encodings to handsets for global operations (GSM7, Latin1, and various Unicode encodings).

SMPP Interface

Send messaging requests via Short Message Peer-to-Peer (SMPP) protocol to ensure high throughput and interoperability with your SMPP Short Message Entities (SME/ESME).

Secure, Time-Based, One-Time Passcode

Verifies users in real time by sending a one-time code via SMS that can be used to authenticate a known user, verify a transaction or block fraudsters from opening illegitimate accounts.

Reliable Global Coverage

Reach and ensure message delivery to end-users in over 200 countries and territories and in 87 different languages.

Inbound SMS for Interactive Two-Way

Send an SMS with an OTP along with transaction details, and allow the user to reply with the code to approve or deny the request.

Advanced Phone Number Cleansing

Properly formats phone number entered by end-user to ensure greater global delivery.

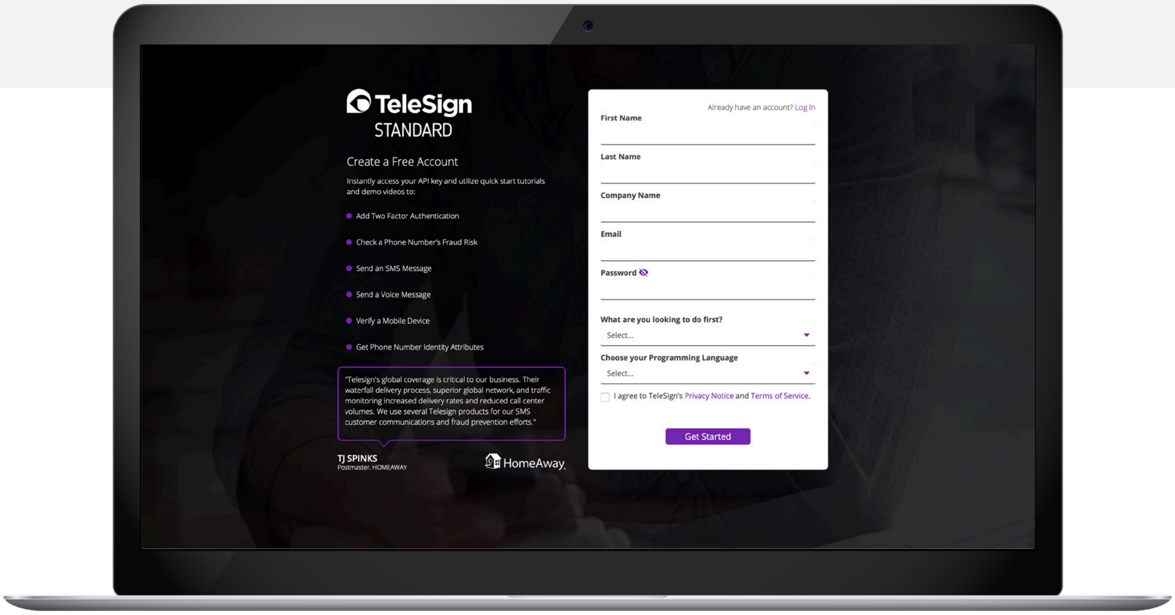
Instant Messenger App Integrations

Send OTPs directly through to Facebook Messenger users. Instantly authenticate users in one of their preferred communication channels to reach more users.

Not only is TeleSign's SMS Verify a wholistic replacement for Account Kit, it also provides 24/7 support, traffic quality monitoring, and troubleshooting expertise.

How to Get Started

1. Create a TS account (here)
2. Get a Customer ID & API Key



TeleSign STANDARD

Create a Free Account

Instantly access your API key and utilize quick start tutorials and demo videos to:

- Add Two Factor Authentication
- Check a Phone Number's Fraud Risk
- Send an SMS Message
- Send a Voice Message
- Verify a Mobile Device
- Get Phone Number Identity Attributes

"Telesign's global coverage is critical to our business. Their waterfall delivery process, superior global network, and traffic monitoring increased delivery rates and reduced call center volumes. We use several Telesign products for our SMS customer communications and fraud prevention efforts."

TJ SPINKS
Postmaster, HOMEAWAY

HomeAway.

Already have an account? Log In

First Name

Last Name

Company Name

Email

Password

What are you looking to do first?
Select...

Choose your Programming Language
Select...

□ I agree to TeleSign's Privacy Notice and Terms of Service.

Implementation: TeleSign's SMS Verify API is simple to use and easy to implement

In these code samples, we provide an overview of how to send an SMS message with a verification code included and cover the work needed on both the client and server side. Furthermore, we show how you would validate the code entered by the end user in an instance in which TeleSign is generating the verification code on your behalf. Note that TeleSign also supports the option for the verification code to be generated by you.

Web Interface Example (HTML/JavaScript)

```
<!DOCTYPE html>
<html lang=”en”>
<head>
  <title>Login with TeleSign Verify</title>
  <meta name=”viewport” content=”width=device-width, initial-scale=1”>
  <link rel=”stylesheet”
        href=”https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css”
        crossorigin=”anonymous”>
  <script src=”https://code.jquery.com/jquery-3.4.1.min.js”
          crossorigin=”anonymous”></script>
  <script src=”https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js”
          crossorigin=”anonymous”></script>
</head>
<body>
  <div class=”container”>
    <h2>Migrate from FaceBook Account Kit to TeleSign Verify</h2>
      <div class=”input-group”>
        <input type=”text” id=”phone_number” class=”form-control” placeholder=”phone number”/>
        <span class=”input-group-btn”>
          <button onclick=”telesignVerify();” class=”btn btn-default” type=”submit”>SMS Login</button>
        </span>
      </div>
    <div id=”step-two” style=”display: none”>
      <div class=”result-message”>
      </div>
      <div class=”input-group”>
        <input type=”hidden” id=”reference_id”>
        <input type=”text” id=”verify_code” class=”form-control” placeholder=”Verification Code”>
        <span class=”input-group-btn”>
          <button onclick=”telesignGetstatus();” class=”btn btn-primary” type=”submit”>Verify</button>
        </span>
      </div>
    </div>
  </div>
</body>
<script>
  function telesignVerify() {
    const phoneNumber = $(“#phone_number”).val().replace(/\W/g, ‘’);
    const data = new URLSearchParams();
    data.append(“phone_number”, phoneNumber);
    // Send SMS using TeleSign Verify
    fetch(“/telesign-verify”, {
      method: ‘POST’,
      body: data
    })
      .then(response => response.json())
      .then((json) => {
        console.log(json);
        if (json.success) {
          $(“.result-message”).empty().append(“Enter Verification Code you received on the device”);
          $(“#verify_code”).val(‘’);
          $(“#reference_id”).val(json.reference_id);
          $(“#step-two”).show();
        } else {
          console.log(“Error sending token.”)
        }
      })
      .catch((err) => {
        console.log(err);
      });
  }

  function telesignGetstatus() {
    const verify_code = $(“#verify_code”).val();
    const reference_id = $(“#reference_id”).val();
    const data = new URLSearchParams();
    data.append(“verify_code”, verify_code);
    data.append(“reference_id”, reference_id);
    fetch(“/telesign-getstatus”, {
      method: ‘POST’,
      body: data
    })
      .then(response => response.json())
      .then((json) => {
        if (json.success) {
          alert(“Success! Redirect browser now”)
        } else {
          $(“#verify_code”).val(“”);
          alert(“Wrong Code!”)
        }
      })
      .catch((err) => {
        console.log(err);
      });
  }
</script>
</html>
```

Server Side Example (Python)

```
from telesignenterprise.verify import VerifyClient
import json

customer_id = “FFFFFFFF-EEEE-DDDD-1234-AB1234567890”
api_key = “EXAMPLE----TE8sTgg45yusumoN6BYsBVkh+yRJ5czgsnCehZaOYldPJdmFh6NeX8kunZ2zU1YWaUw/0wV6xfw==”


# handles requests to /telesign-verify
def telesign_verify(request):
    phone_number = request.POST.get(‘phone_number’, ‘’)
    verify = VerifyClient(customer_id, api_key)
    sms_response = verify.sms(phone_number)
    data = {
        ‘success’: True,
        ‘reference_id’: sms_response.json[‘reference_id’],
    }
    return json.dumps(data)


# handles requests to /telesign-getstatus
def telesign_getstatus(request):
    user_entered_verify_code = request.POST.get(‘verify_code’, ‘’)
    reference_id = request.POST.get(‘reference_id’, ‘’)
    verify = VerifyClient(customer_id, api_key)
    status_response = verify.status(reference_id,
                                    verify_code=user_entered_verify_code)
    data = {
        ‘success’: status_response.json[‘verify’][‘code_state’] == ‘VALID’,
    }
    return json.dumps(data)
```

Need more help? Contact us at contact@telesign.com with any questions.

Helpful Resources

SMS Verify Product Page

SMS Verify Developer Documentation

Using 2FA to Prevent Account Takeover

TeleSign GitHub

Our platform connects and protects online experiences with sophisticated customer identity and engagement solutions. Through APIs that deliver user verification, data insights, and communications, we solve today’s unique customer challenges by bridging your business to the complex world of global telecommunications.

For more information, please visit **www.telesign.com** | **@TeleSign**