



Linkello API

version 2.0

Document Confidentiel

Ne peut être transmis ou copier partiellement ou totalement sans accord écrit préalable de Linkello



Introduction	3
Présentation	3
Principe	3
Mise en oeuvre	3
Prérequis (option sans toolkit)	4
Linkello REST endpoints	5
Créer un lien	5
Accéder aux propriétés d'un lien	6
Supprimer un lien	7
Récupérer les statistiques de création/consultation de liens sur une période	7
Récupérer le journal des appels sur une période	9
Accéder à la page de conférence Linkello - avec toolkit	10
Intégration du toolkit	10
Paramètres du toolkit	11
Personnalisation des couleurs de l'interface	12
Accéder à la page de conférence Linkello - sans toolkit	13
Accéder à la conférence en tant qu'organisateur	13
Accéder à la conférence en tant que visiteur	14
Signature des requêtes	14



Introduction

Le présent document a pour but de décrire le fonctionnement de l'API Linkello afin de permettre la création d'un lien Linkello depuis un service tiers autorisé.

Présentation

Principe

L'API Linkello permet de générer des liens de conférences Linkello depuis un service tiers. Cela autorise ainsi l'éditeur d'un service à créer des liens à la demande et de les partager via le média de son choix sans se soucier de la mise en œuvre de la partie conférence.

L'API n'est disponible qu'au travers de l'offre **Linkello API** et nécessite une étape de configuration pour que celle-ci soit disponible.

D'un point de vue technique, l'API Linkello est une API REST conçue pour une intégration de type serveur à serveur. A ce jour, elle expose 3 endpoints :

- link: créer, récupérer et supprimer un lien de conférence Linkello
- stats/calls: récupérer le journal d'appels sur une période donnée.
- stats/links: consulter les statistiques de création de liens sur une période donnée.

Mise en oeuvre

Pour éviter que l'API ne soit utilisée par des tiers non autorisés, les requêtes vers l'API Linkello sont authentifiées via une signature de type HMAC (lire la section "signature des requêtes").

Dans le cadre d'une phase de test de l'API nous pouvons désactiver le mécanisme d'authentification. Dans ce cas, la clé d'API suffit pour accéder aux endpoints. Cependant nous recommandons vivement d'utiliser notre mécanisme d'authentification pour les applications en production !

Pour la mise en oeuvre de la page de conférence 2 solutions vous sont proposées :

- intégrer le toolkit Linkello dans une page de votre service en ligne (recommandé)
- utiliser directement la page de conférence de Linkello (pas de toolkit) customisée à vos couleurs et qui apparaîtra dans un onglet à part. Dans ce cas, des pré-requis sont nécessaires.



Prérequis (uniquement dans le cas de l'option sans toolkit)

Dans le cas où vous ne souhaitez pas utiliser notre toolkit pour la mise en œuvre de la conférence, mais directement utiliser la page du service Linkello avec votre propre domaine, un certain nombre de prérequis vous seront demandés :

- **Définir un sous-domaine de votre domaine pour votre service de visio**
Il s'agit du sous-domaine qui sera utilisé pour les liens de conférences (ex: visio.mon-domaine.com). Ce sous-domaine sera communiqué au support pour être intégré à la configuration de votre compte.
- **Modifier la configuration DNS de votre domaine pour pointer sur notre serveur**
Contactez le support le connaître les informations techniques liées à votre compte
- **Fournir les éléments pour la customisation de la salle d'appels à vos couleurs**
 - **un logo principal**
format png/jpg et de taille 150×90
 - **un logo "pied de page"**
format png/jpg et de taille 65×65
 - **les codes couleur de votre charte graphique**
format hexadécimal (ex: #000000)
 - **une image de fond (optionnel)**
format png/jpg et de taille 2560x1600



Linkello REST endpoints

Créer un lien

```
POST https://demo-api.linkello.com/api/v1/link?app=[application id]
```

Paramètres de la requête :

- **app** : [String] identifiant de l'application (développement, intégration, production, ...)

Paramètres du corps de la requête :

- **label** : [String] titre donné à l'appel/conférence
- **media** : [String] "audio" ou "video", média utilisé pour l'appel/conférence

Exemple :

```
https://demo-api.linkello.com/api/v1/link/zKBEmOyqJ?app=9A8UThF5pqmzpy2cmLFVsO6ocHsawh5V
```

Corps JSON de la requête POST

```
{
  "label": "Hello World !",
  "media": "video"
}
```

Réponse JSON en cas de succès :

```
{
  "success": true,
  "data": {
    "key": "Irg5FPtYd" // identifiant public du lien.
  }
}
```

Réponse de l'api en cas d'erreur :

```
{
  "success": false,
  "error": "raison de l'erreur"
}
```

Document Confidentiel

Ne peut être transmis ou copier partiellement ou totalement sans accord écrit préalable de Linkello

Accéder aux propriétés d'un lien

```
GET https://demo-api.linkello.com/api/v1/link/[key]?app=[application id]
```

Paramètres de la requête :

- **key** : [String] identifiant public du lien
- **app** : [String] identifiant de l'application (développement, intégration, production, ...)

Exemple :

```
https://demo-api.linkello.com/api/v1/link/zKBEmOyqJ?app=9A8UThF5pqmzpy2cmLFVsO6ocHs  
awh5V
```

Réponse de l'api en cas de succès :

```
{  
  "success": true,  
  "data": {  
    "key": "Irg5FPtYd", // identifiant public du lien.  
    "label": "Hello World !", // titre de l'appel  
    "media": "video", // média utilisé pour l'appel  
    "room": "e80ZFww4pF16fbroMpc0", // id de la room pour l'appel  
    "deleteAt": "2019-10-17T13:45:31.718Z", // date d'expiration du lien  
    "isPrivate": false, // lien créé un compte privé  
  }  
}
```

Réponse de l'api en cas d'erreur :

```
{  
  "success": false,  
  "error": "raison de l'erreur"  
}
```



Supprimer un lien

```
DELETE https://demo-api.linkello.com/api/v1/link/[key]?app=[application id]
```

Remarque : par défaut un lien est automatiquement supprimé lorsqu'il arrive à expiration.

Paramètres de la requête :

- **key** : [String] identifiant public du lien
- **app** : [String] identifiant de l'application (développement, intégration, production, ...)

Exemple :

```
https://demo-api.linkello.com/api/v1/link/zKBEmOyqJ?app=9A8UThF5pqmzpy2cmLFVsO6ocHsawh5V
```

Réponse de l'api en cas de succès :

```
{
  succes: true
}
```

Réponse de l'api en cas d'erreur :

```
{
  "success": false,
  "error": "raison de l'erreur"
}
```

Récupérer les statistiques de création/consultation de liens sur une période

```
GET https://demo-api.linkello.com/api/v1/stats/links?app=[application id]&startDate=[iso start date]&endDate=[iso end date]&view=[interval]
```

Paramètres de la requête :

- **app** : [String] identifiant de l'application (développement, intégration, production, ...)
- **startDate** : [String] date de début de la plage de recherche au format iso
- **endDate** : [String] date de fin de la plage de recherche au format iso
- **view** : [String] mode d'agrégation des données, par jour (day) ou par mois (month)

Document Confidentiel

Ne peut être transmis ou copier partiellement ou totalement sans accord écrit préalable de Linkello



Exemple :

<https://demo-api.linkello.com/api/v1/stats/links?app=9A8UThF5pqmzpy2cmLFVsO6ocHsawh5V&endDate=2019-12-02T23:54:43.394Z&startDate=2019-10-27T16:54:43.394Z&period=month&view=day>

Réponse de l'api en cas de succès :

```
{
  "success": true,
  "data": [
    {
      "details": [
        {
          "new": {
            "total": 3,
            "audio": 0,
            "video": 3
          },
          "view": {
            "total": 1,
            "audio": 0,
            "video": 1
          },
          "date": {
            "month": 12,
            "year": 2019,
            "day": 2
          }
        }
      ],
      "sumNew": {
        "total": 3,
        "audio": 0,
        "video": 3
      },
      "sumView": {
        "total": 1,
        "audio": 0,
        "video": 1
      }
    }
  ]
}
```




Réponse de l'api en cas d'erreur :

```
{
  "success": false,
  "error": "raison de l'erreur"
}
```

Récupérer le journal des appels sur une période

```
GET https://demo-api.linkello.com/api/v1/stats/calls?app=[application
id]&startDate=[iso start date]&endDate=[iso end date]
```

Paramètres de la requête :

- **app** : [String] identifiant de l'application (développement, intégration, production, ...)
- **startDate** : [String] date de début de la plage de recherche au format iso.
- **endDate** : [String] date de fin de la plage de recherche au format iso.

Exemple :

```
https://demo-api.linkello.com/api/v1/stats/calls?app=9A8UThF5pqmzpy2cmLFVsO6ocHsawh5V
&endDate=2019-12-02T23:54:43.394Z&startDate=2019-10-27T16:54:43.394Z
```

Réponse de l'api en cas de succès :

```
{
  "success": true,
  "data": [
    {
      "key": "zKBE0yqJ",
      "media": "video",
      "label": "Hello World !",
      "log": [
        {
          "date": {
            "day": 2,
            "month": 12,
            "year": 2019,
            "hour": 11,
            "minutes": 17
          },
          "score": 5,
        }
      ]
    }
  ]
}
```

Document Confidentiel

Ne peut être transmis ou copier partiellement ou totalement sans accord écrit préalable de Linkello

```
    "participants": 2
  }
]
}
```

Réponse de l'api en cas d'erreur :

```
{
  "success": false,
  "error": "raison de l'erreur"
}
```



Accéder à la page de conférence Linkello - avec toolkit

Cette section ne concerne que les personnes qui ont choisi d'utiliser le toolkit Linkello pour intégrer l'écran de conférence au sein de leur service.

Intégration du toolkit

Le toolkit se présente sous la forme d'un fichier JS nommé **linkello.toolkit-vx.x.js** et doit être inséré dans la page qui servira à afficher l'écran de conférence.

L'écran de la conférence sera automatiquement construit à l'intérieur d'une balise **div** dont l'id devra impérativement être **root**. Les paramètres de la conférence seront passés au toolkit via des attributs html placés au sein de la balise DIV, comme ci-dessous.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    ...
  </head>
  <body>
    <div
      id="root"
      data-key="6XUxws2S5"
      data-language="fr"
      data-domain="api.linkello.com"
      data-domain="/static/sounds/join.mp3"
      data-logo-url="/static/images/logo.png",
      data-background-image-url="/static/images/wallpaper.jpg"
      data-background-repeat="false",
      data-exit-url="/"
      data-call-rating-on="true",
      data-signature="ts=1613577148000,v1=1d069a15...b543ad8601";
    ></div>
    <script src="/js/linkello.toolkit-v1.0.js"></script>
  </body>
</html>
```

Document Confidentiel

Ne peut être transmis ou copier partiellement ou totalement sans accord écrit préalable de Linkello



Paramètres du toolkit

data-key

clé publique de la conférence

data-language

langue de l'interface - "en" par défaut

data-domain

permet de spécifier un domaine - optionnel, "api.linkello.com" par défaut

data-sound-url

url du fichier audio joué lors de l'entrée d'un participant dans la conférence - optionnel, fichier audio Linkello par défaut

data-logo-url

url du logo - optionnel, logo Linkello par défaut

data-background-image-url

url de l'image de fond - optionnel, image de fond Linkello par défaut

data-background-repeat

true/false, permet de répéter l'image de fond - optionnel, false par défaut

data-exit-url

url qui doit être appelé lorsqu'on quitte la conférence - optionnel, "/" par défaut.

data-call-rating-on

true/false, désactive l'écran de notation de l'appel - optionnel, false par défaut

data-signature

signature permettant d'identifier l'utilisateur comme étant l'organisateur de l'appel, en l'absence de celle-ci l'utilisateur est considéré comme un visiteur - optionnel, vide par défaut.

Personnalisation des couleurs de l'interface

Au-delà des images il est possible de personnaliser l'aspect de l'interface en utilisant des règles CSS. Il existe également un certain nombre de couleurs prédéfinies qu'il est possible de modifier pour les adapter à votre identité visuelle.



```
<head>
...
<style>
  html:root {
    /* page background color */
    --linkello-page-bg-color: #f0f0f0;

    /* header background color */
    --linkello-header-bg-color: #ffffff;

    /* header font color */
    --linkello-header-text-color: #777777;

    /* page font color */
    --linkello-main-text-color: #777777;

    /* vu-meter color */
    --linkello-main-highlight: #1169b4;

    /* chat, active tab, toolbar buttons background color */
    --linkello-group-1-bg-color: #b3b6b9;

    /* chat, active tab, toolbar buttons font color */
    --linkello-group-1-text-color: #ffffff;

    /* badge, modal 'done' button, settings buttons, toolbar active buttons
background color */
    --linkello-group-2-bg-color: #1169b4;

    /* badge, modal 'done' button, settings buttons, toolbar active buttons font
color */
    --linkello-group-2-text-color: #ffffff;
  }
</style>
...
</head>
```

Document Confidentiel

Ne peut être transmis ou copier partiellement ou totalement sans accord écrit préalable de Linkello



Accéder à la page de conférence Linkello - sans toolkit

Cette section ne concerne que les personnes qui ont choisi de ne pas utiliser le toolkit Linkello pour intégrer la conférence.

Accéder à la conférence en tant qu'organisateur

Pour qu'un utilisateur puisse se connecter à la conférence en tant qu'organisateur, il faut **impérativement signer la requête** afin de vérifier que celui qui se présente comme l'organisation soit bien celui qu'il prétend être.

Cas n°1 : récupération du code HTML de la page de conférence depuis votre serveur.
La signature HMAC sera ajoutée dans le header X-Linkello-Signature de la requête.

```
GET https://demo-api.linkello.com/[key]?app=[application id]&name=[user name]
```

Cas n°2 : redirection vers la page de conférence.
La signature HMAC de la requête sera ajoutée via un paramètre sig

```
GET https://demo-api.linkello.com/[key]?app=[application id]&name=[user name]&sig=[signature]
```

Remarque : il n'est pas possible de modifier le header d'une requête de redirection, d'où la nécessité d'ajouter le paramètre "sig" à la requête pour passer la signature.

Paramètres de la requête :

- **key** : [String] identifiant public du lien
- **app** : [String] identifiant de l'application (développement, intégration, production, ...)
- **name** : [String] nom de l'organisateur de l'appel
- **sig** : [String] signature de la requête

Exemple :

```
https://demo-api.linkello.com/zKBEmOyqJ?app=9A8UThF5pqmzpy2cmLFVsO6ocHsawh5V&name=John%20Doe
```

Réponse en cas de succès :

Document Confidentiel

Ne peut être transmis ou copier partiellement ou totalement sans accord écrit préalable de Linkello



La page de conférence au format HTML

Attention : si vous souhaitez effectuer une redirection vers la page de conférence il faudra ajouter la signature dans le paramètre “sig” car il n’est pas possible de modifier le header HTTP d’une redirection

```
GET https://demo-api.linkello.com/[key]?app=[application id]&name=[user name]&sig=...
```

Accéder à la conférence en tant que visiteur

il suffit d’appeler l’url suivante :

```
GET https://demo-api.linkello.com/[key]
```

Signature des requêtes

Pour protéger efficacement les endpoints en production et prouver que l’émetteur d’une demande est bien celui qu’il prétend être, les requêtes vers l’API Linkello doivent être signées à l’aide d’une clé propre à l’intégrateur et qu’il doit garder secrète.

La signature de la requête se fait via un mécanisme de HMAC Authentication. Celle-ci est insérée dans la requête au niveau du header HTTP via le champ **X-Linkello-Signature**. Elle est calculée sur la base du payload du corps de la requête et d’un timestamp. Une fois générée, la signature a une durée de validité de 30s.

```
X-Linkello-Signature :  
ts=1571218706401,v1=cca31dbd7753b060ab9cf0380f915d54d2da1ed184ddf04bab7fbb2fa9aaf8a
```

La génération de la signature se fait avec la fonction :

```
LinkelloSignTool.getSignature ( jsonPayload )
```

Cette fonction prend en paramètre un objet javascript qui correspond au corps de la requête POST. Pour les requêtes GET et DELETE ce paramètre doit être omis.

Exemple de la signature d’une requête (NodeJS) :

Document Confidentiel

Ne peut être transmis ou copier partiellement ou totalement sans accord écrit préalable de Linkello



- *linkello-sign-tool.js*

```
const crypto = require ( 'crypto' );

function LinkelloSignTool ( secret )
{
  if ( !( this instanceof LinkelloSignTool ) )
  {
    return new LinkelloSignTool ( secret );
  }

  this.secret = secret;
}

LinkelloSignTool.prototype.getSignature = function ( jsonPayload )
{
  const ts = new Date ().getTime (),
        hmac = crypto.createHmac ( 'sha256', this.secret );

  hmac.update ( `${ts}.${JSON.stringify ( jsonPayload || {} )}` );

  return `ts=${ts},v1=${hmac.digest ( 'hex' )}`;
}

module.exports = LinkelloSignTool;
```

- *my-request.js*

```
const https = require ( 'https' ),
      signTool = require ( './linkello-sign-tool' ) ( 'PlqocS88Id' ); // clé secrète de l'api

function createLink ()
{
  const post_data = {
    media: 'video',
    label: 'Hello World'
  }
```

Document Confidentiel

Ne peut être transmis ou copier partiellement ou totalement sans accord écrit préalable de Linkello



```
    },
    post_options = {
      host: 'demo-api.linkello.com',
      port: '443',
      path: '/v1/api/link',
      method: 'POST',
      headers: {
        'Content-Type': 'application/json',
        'X-Linkello-Signature': signTool.getSignature ( post_data )
      }
    };

    // Set up the request
    const postReq = https.request ( post_options, res => { ... } );
    postReq.write ( JSON.stringify ( post_data ) );
    postReq.end ();
  }

function getLink ()
{
  const get_options = {
    host: 'demo.linkello.com',
    port: '443',
    path: '/v1/api/link/Irg5FPtYd',
    method: 'GET',
    headers: {
      'Content-Type': 'application/json',
      'X-Linkello-Signature': signTool.getSignature ()
    }
  };

  // Set up the request
  const getReq = https.request ( get_options, res => { ... } );
  getReq.end ();
}
```