

IPscreener API v2.1

Table of Contents

- [Table of Contents](#)
- [General](#)
 - [API updates / change log](#)
 - [Service Management and Support](#)
 - [Terms of Use](#)
 - [Disclaimer](#)
- [Introduction to IPscreener](#)
 - [System Security Frame Work](#)
 - [API Authentication Procedure](#)
 - [API Pricing Model](#)
 - [The IPscreener Search Procedure](#)
 - [IPscreener optimization and index selection structure](#)
 - [Boost mode \(beta\)](#)
- [API Endpoints](#)
 - [The index API](#)
 - [The IPscreener search API](#)
 - [Request result](#)
 - [Receive result](#)
 - [The PDF API for document retrieval](#)



IPscreener API Specification Version 2.1

General

This is a technical documentation that describes the IPscreener API's interaction with the AutoMatch search engine. The specification is intended for those who are experienced in programming software applications and intend to implement the IPscreener API service. This documentation is subject to continuous change, however normally backward compatibility is maintained.

API updates / change log

This document will be updated with the latest information regarding the API structure and functionality. Notifications of changes will be sent to registered users via email.

Service Management and Support

The support is managed by the IPscreener team and contact is encouraged for any inquiries, as well for technical matters as for user specific questions. For any questions, please contact our support team by sending an email to support@ipscreener.com.

- Support is available Monday-Friday 8:30-17:00 CEST
- The target uptime for the IPscreener service is 99%, excluding scheduled service break points (normally 4 occasions per year scheduled on weekends).
- A ticket response will normally be provided within 2 hours during office hours, where we address the following topics:
 - Notification that the support team has received your request.
 - Possible request from the support team for further information.
 - Information about how the issue is going to be handled.
- Solutions and assistance will be provided in-line with timescales dependent on the priority of the support request:
 - Within 8 hours (during business hours) for issues classified as High priority.
 - Within 48 hours for issues classified as Medium priority.
 - Within 10 working days for issues classified as Low priority.

Terms of Use

The terms of use for the APIs are regulated by agreements and the general terms for web services. We apply a strict fair usage principle and in endeavoring to make the IPscreener API stable for everyone to use, we place restrictions on the number of requests you can make to the service. If you exceed the limits set out we may throttle, suspend or terminate your access. The set limit may be adjusted by us when updating the IPscreener API. For more information please refer to the general terms, which sent upon request from support@ipscreener.com.

Disclaimer

- The sample method calls, success response and error response in API details are provided for general information purposes only.
- The terminology used in this document may differ from other specifications of API's for fetching of data.

Introduction to IPscreeener

IPscreeener is a decision support tool based on semantic technology used for automatic prior art retrieval. When feeding a raw text to the IPscreeener API, the AutoMatch search engine converts this text to a fingerprint representation. This representation is used to fetch the most similar documents reflecting the context of the query text. This is done from a specially tailored global reference database of patent documents. Thus, relevant prior art is identified and presented to a user in a client platform without any manual interaction needed. Therefore, each review of an idea along the innovation process is given a support for faster decisions, better priority of resources and at a higher quality.

The IPscreeener API is implemented as a standalone-module that is easily integrated with any existing commercial or internal patent administration system. The API service is built using REST principles which ensures predictable URLs that supports easy connectivity and communication. For a descriptive overview of the data flow see Fig. 1. The IPscreeener REST API follows standard HTTP rules, enabling a wide range of HTTP clients to be able to interact with the service.

The IPscreeener API system architecture

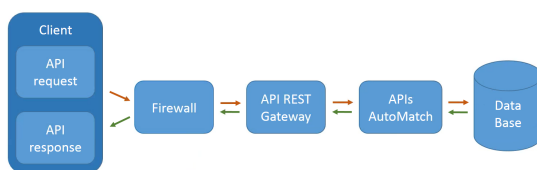


Fig. 1 The IPscreeener API data flow architecture showing data traffic over Internet for a query and response.

System Security Frame Work

The API is hosted in a protected and secure environment including the protection of the user information and data.

- We have our own dedicated self-hosted servers for hosting the API, located in a 24/7 guarded vault in Stockholm, Sweden.
- A security surveillance system protects and prevents unauthorized traffic from accessing the network. It also ensures the implementation and compliance of the organization's regulatory policy.
- A hardware based firewall is implemented to secure the network traffic.
- When the clients access the API, no client information is stored/cached in the system except for the session ticket and case reference number. This data is used to track the number of performed searches.
- The API server has an EV SSL certificate implemented which provides an assured encryption level.

API Authentication Procedure

The API authentication is implemented as token authentication over the secure HTTPS protocol. By default there is no restriction on the IP-address from which the requests can be sent or from which domain the request must come from (CORS). If the client requires additional security they can contact the support team to have a IP-address restriction and CORS enabled for their API authorization key. The client will then have to provide a list of IP numbers or an IP range that will be white listed, i.e. requests to API service is only possible from the specified IP addresses. Likewise, the same principle applies for registering whitelisted domains. You may retrieve API credentials and register associated IP restrictions by contacting the IPscreeener support team, support@ipscreener.com.

There are two levels of user identification employed by the IPscreeener API service. If the user is an end customer, e.g. direct end user, only the authorization key is needed when accessing the service, where said key also includes the client identification. When the client is a platform provider which in turn have their respective clients being the platform provider's end users, also the respective end user client-id has to be submitted in the header in the API request. In order for a platform provider to register and obtain a client-id for a new platform client, contact the IPscreeener support team, support@ipscreener.com. A description of the hierarchy is revealed in Fig. 2.

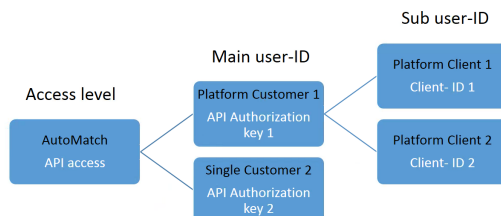


Fig. 2 The access level hierarchy used to identify search requests for platform providers and end users.

API Pricing Model

The IPscreeener API uses normally a transaction-fee based pricing model. Hence, you will need to purchase credits (the number of searches to be performed) in advance to use the API service. For more information contact the IPscreeener team at support@ipscreener.com. A notification e-mail will be sent when there are few credits remaining or when the contract period is about to expire.

The IPscreeener Search Procedure

The procedure for performing a search request via the API services is done according to the following steps. A general overview and introduction to this procedure of these steps is presented in Fig. 3 below.

- An API authorization key is needed for performing any access request.
- The request is initiated by the user specifying the search query text and how many records to be fetched in the result set. Optionally the user may also manually select what indexes to search with the query text. A maximum of three indexes may be searched in parallel. Index values is received by using the Index API.
- As a response to a query a session ticket is received. This ticket is used to fetch result data when ready for delivery. Thus, note that polling from the requesting part needed.
- The response is a result list with a number of hits representing the top most relevant documents corresponding to the content of the search query text. The hits are sorted in order of relevance. The result list includes for each hit the patent number, the associated relevance score and the additional requested document information.
- To fetch a PDF document for hit provided in the search response, insert each patent number a URL parameter request in the PDF API. The PDF document is encoded in base64 format.
- The maximum number of requests allowed are limited to 10 fetches per minute.

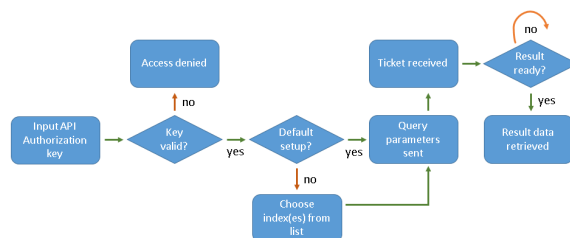


Fig. 3 IPscreeener search procedure and identification steps.

IPscreeener optimization and index selection structure

The AutoMatch search engine comprises advanced logic to optimize the search query according to the input text. In the default setup it automatically select the most appropriate settings for best recall and precision. This includes the possibility to also retrieve the paragraph considered by the algorithm to be most relevant for a specific retrieved document. The user may also manually select specific technically optimized domain indexes to be used during the query process. This includes the selection of customer specific indexes such as e.g. the archive of internal invention disclosures or adapted customizations. A summary overview of the index setup is shown in Fig. 4. For more information on special indexes and how to order and create them, contact the IPScreener support team, support@ipscreener.com.

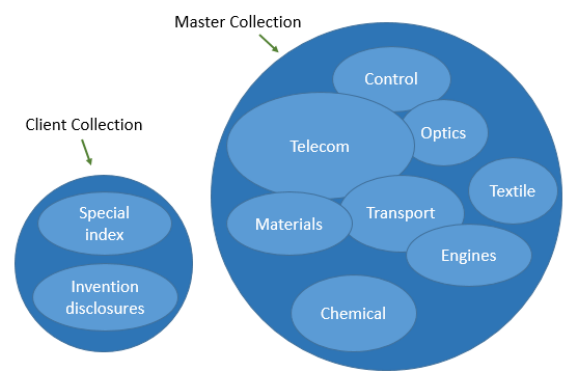


Fig 4. Showing the layout structure for technical optimizations versus the IPScreener master collection.

Boost mode (beta)

The next generation of IPScreener has an improved algorithm and extra layers of semantic analysis. This means that the engine identifies the technical domain that is optimized for the search query as such and tunes the settings accordingly. Currently the pilot running with the new analytic features is in a separate option named Boost. For trying out this new feature, use the switch "boost" with the optional index parameter. This feature is still in beta mode until further notice.

API Endpoints

The index API

The Index API is used to retrieve the indexes accessible by a customer by the end client, whereby both the technical description of an index as well as the access number is provided. If no index is specified in the search request, the master index will be used as default. A maximum of three parallel indexes may be accessed in one single IPScreener query this includes customer specific indexes as well. If you are a platform provider, by adding the client-id header you will also see the technical index list available for your respective clients in your platform.

URL	https://api.auto-match.se/v2.1/index
Method	GET
Headers	Required: authorization=[UTF-8 encoded max 100 char] Optional: client-id=[positive integer]

**Success
Response**

Body content format: JSON

HTTP Status Code: 200 OK

```
{
  "status": "success",
  "message": "Resource fetched successfully",
  "data": [
    {
      "index-id": 1,
      "name": "Master"
    },
    {
      "index-id": 27,
      "name": "Telecommunications"
    },
    {
      "index-id": 29,
      "name": "Mechanical elements"
    }
  ]
}
```

Error Response	<p>HTTP Status Code: 401 Unauthorized (No API authorization key found)</p> <pre>{ "message": "No API Key is found in request" }</pre>
	<p>HTTP Status Code: 403 Forbidden (Using wrong API authorization key)</p> <pre>{ "message": "Invalid authentication credentials" }</pre>
	<p>HTTP Status Code: 403 Forbidden (When the IP restriction is activated and source IP is not whitelisted)</p> <pre>{ "message": "Your IP address is not allowed" }</pre>
	<p>HTTP Status Code: 429 Too many requests (API rate limit exceeded)</p> <pre>{ "message": "API rate limit exceeded" }</pre>
	<p>HTTP Status Code: 422 Unprocessable entity (When using invalid client-id)</p> <pre>{ "status": "error", "message": "No permission granted for this client-id.", "data": null }</pre>
	<p>Sample Call (cURL)</p> <pre>curl --request GET \ --url https://api.auto-match.se/v2/index \ --header 'authorization: xxx' \ --header 'client-id: xxx'</pre>
<p>Notes</p> <ul style="list-style-type: none"> • authorization: An API authorization key must be sent with all requests. You need to contact the IPscreeener support team to get the API authorization key. • client-id (Optional): This value is only refers to platform providers and identifies their respective clients. To retrieve your respective list of client-ids, contact the IPscreeener support. <p><u>Success Response</u></p> <ul style="list-style-type: none"> • data: Retrieval of a list of indexes accessible for a specific API authorization key. 	

The IPscreeener search API

This API receives a text query input and then uses the AutoMatch semantic search engine to retrieve similar results from the worldwide patent database collection. The search procedure includes a first response to a query by returning a session ticket, which then is used to retrieve the results when ready for retrieval. Processing an IPscreeener query typically takes 15-30 seconds, and the patent data is delivered in a JSON format. The number of records delivered depends on the figure specified by the requested-hits parameter and are presented in order of ranked relevance. If the requested-hits parameter is null or absent, 25 records are returned by default. The request procedure is optimized for returning up to 100 records.

If no switches are provided IPscreeener will select the most appropriate general settings for carrying out the semantic matching procedure, and returns a single result list. If the query specifies several indexes to be queried in the request, the number of results set by the requested-hits parameter applies to all. Thus, this is the number retrieved for each and one of the queried indexes, where the results for each index are presented sequentially according to the request order in the query string.

To note; we have a pilot running on a new feature where IPscreeener performs an extra layer of analysis. For trying this new feature out use the switch "boost" as the optional index parameter. This feature is still in beta mode until further notice.

Request result

URL	https://api.auto-match.se/v2.1/search
Meth od	POST
Hea ders	<p>Required: authorization=[UTF-8 encoded]</p> <p>reference-number=[UTF-8 encoded max 100 char]</p> <p>Optional: client-id=[positive integer]</p>
Body	<p>Required: query=[UTF-8 encoded]</p> <p>Optional: requested-hits=[positive integer]</p> <p>view=[bibliographic, claim, description, image, pdf, passage]</p> <p>index=[positive integer(s) with comma separator or value 'boost' (beta)]</p>
Succ ess Res ponse	<p>Body content format: JSON</p> <p>The below response will be returned while the request is still being processed.</p> <p>HTTP Status Code: 200 OK</p> <pre>{ "status": "success", "message": "Ticket received", "data": "<ticket value>" }</pre> <p>The below response will be returned once the search request has been processed. To view the result please use the GET search API with the ticket value.</p> <p><i>Do note; the response below is an example where we have truncated the data for several fields to make the view more condensed and comprehensive.</i></p>
	HTTP Status Code: 401 Unauthorized (No API authorization key found)

**Error
Response**

```
{
  "message": "No API Key is found in request"
}
```

HTTP Status Code: 403 Forbidden (Using wrong API authorization key)

```
{
  "message": "Invalid authentication credentials"
}
```

HTTP Status Code: 403 Forbidden (When the IP restriction is activated and source IP is not whitelisted)

```
{
  "message": "Your IP address is not allowed"
}
```

HTTP Status Code: 422 Unprocessable entity (When using invalid client-id)

```
{
  "status": "error",
  "message": "No permission granted for this client-id.",
  "data": null
}
```

HTTP Status Code: 422 Unprocessable entity (When null value passed in query parameter)

```
{
  "status": "error",
  "message": "Query parameter is required and should not be empty.",
  "data": null
}
```

HTTP Status Code: 422 Unprocessable entity (When null value passed in reference-number parameter)

```
{
  "status": "error",
  "message": "Reference-number parameter is required and should not be empty or more than 100 characters.",
  "data": null
}
```

HTTP Status Code: 422 Unprocessable entity (When wrong value passed in requested-hits parameter)


```
{
  "status": "error",
  "message": "The requested hits must be an integer. The
requested hits must be at least 1.",
  "data": null
}
```

HTTP Status Code: 422 Unprocessable entity (When greater than 100 value passed in requested-hits parameter)

```
{
  "status": "error",
  "message": "The requested hits may not be greater than
100.",
  "data": null
}
```

HTTP Status Code: 422 Unprocessable entity (When wrong value passed in view parameter)

```
{
  "status": "error",
  "message": "The accepted values for view are:
bibliographic, claim, description, image, pdf, passage. The same
value may occur once.",
  "data": null
}
```

HTTP Status Code: 429 Too many requests (API rate limit exceeded)

```
{
  "message": "API rate limit exceeded"
}
```

HTTP Status Code: 422 Unprocessable entity (When wrong value passed into index parameter)

```
{
  "status": "error",
  "message": "You do not have permission to this index or
index is offline.",
  "data": null
}
```

HTTP Status Code: 422 Unprocessable entity (When null value passed into index parameter)

```
{
  "status": "error",
  "message": "You must at least select 1 index.",
  "data": null
}
```

	<p>HTTP Status Code: 408 Request Timeout (When one or many indexes are offline or having other troubles)</p> <pre> { "status": "error", "message": "One or more indexes seems to be offline.", "data": null } </pre>
Sample Call (cURL)	<pre> curl --request POST \ --url https://api.auto-match.se/v2.1/search \ --header 'authorization: xxx' \ --header 'client-id: xxx' \ --header 'reference-number: SE-123' \ --data 'query=car&index=1&requested-hits=2&view=bibliographic' </pre>
Notes	<p>Header</p> <p>client-id: This value relates to the identification of an end client for a platform provider. To retrieve a client-id for a new or existing client, contact the IPScreener support team.</p> <p>authorization: An API authorization key must be sent with all requests. You need to contact the IPScreener support team to get the API authorization key.</p> <p>reference-number: This number is used for tracking the API calls for each client. The same number which is sent as input is returned in the header without any modifications.</p> <p>Body</p> <p>query: The text on which the search is performed.</p> <p>requested-hits: This is the number of relevant hit records which you want to fetch in the result list. If this parameter is not used in the search request, by default 25 records are returned.</p> <p>view: This input parameter is used to specify the data fields that are required to be returned for each hit in the response. You can specify a combination of fields using comma to separate the values. If this parameter is not used is all data fields except the description field are fetched.</p> <p>The available field parameters are:</p> <p>bibliographic (includes Applicant, Inventor, Class, Title and Abstract fields, all sections can return multiple data)</p> <p>claim (Covers all the patent claims)</p> <p>description (This is the full description, tables and charts are excluded)</p> <p>image (Presented in base64 format)</p> <p>pdf (An URL link to the PDF API is provided)</p> <p>passage (The paragraph within a document that the AutoMatch engine considers to be most relevant)</p> <p>Header response</p> <p>x-ratelimit-limit-minute: The figure shows how many request of the API endpoint that is possible to do per minute. Accessing the API service at a higher rate will result in a block access until following subsequent one minute period has begun.</p> <p>x-ratelimit-remaining-minute: This indicates how many requests for the current minute that are left within the current minute period.</p> <p>Success Response</p> <p>data: the session ticket value associated with a search request used to GET the search results.</p>

Receive result

U R L	https://api.auto-match.se/v2.1/search
M e t h o d	GET
H e a d e r	Required: authorization=[UTF-8 encoded max 100 char]
B o d y	Required: ticket=[ticket value, string]
S u c c e s s R e s p o n s e	Body content format: JSON The below response will be returned while the request is still being processed. HTTP Status Code: 202 Accepted <div><pre>{ "status": "success", "message": "Processing", "data": null }</pre></div> <p>When checking if data is ready for retrieval you only use the parameter ticket value. The response below will be returned once the search request has been processed and is ready for delivery.</p> <p><i>Do note; the response below is an example where we have truncated the data for several fields to make the view more condensed and comprehensive.</i></p> HTTP Status Code: 200 OK <div><pre>{ "status": "success", "message": "Resource Fetched Successfully", "data": { "automatch-results": { "<index-x value>": [{ "relevance": { "score": 514.81525, "position": 1, "country": "US", "number": "4830558", "kind-code": "A", "family-number": 1339803, "publication-date": "1989-05-16" }, "bibliographic": { "applicant": [{ "country": "US", "name": "THEODORE J SWEENEY & CO" }], } }] } } }</pre></div>

```

"inventor": [
  {
    "country": "US",
    "name": "SWEENEY THEODORE J"
  }
],
"class": [
  {
    "group": "F16B",
    "main": "F16B21",
    "sub": "F16B21/07",
    "type": "IPC"
  },
  {
    "group": "F16B",
    "main": "F16B5",
    "sub": "F16B5/00",
    "type": "IPC"
  },
  {
    "group": "F16B",
    "main": "F16B47",
    "sub": "F16B47/00",
    "type": "CPC"
  },
  {
    "group": "F16B",
    "main": "F16B47",
    "sub": "F16B47/003",
    "type": "CPC"
  }
],
"title": [
  {
    "language": "en",
    "text": "Adhesively securable
fastener"
  }
],
"abstract": [
  {
    "language": "en",
    "text": "Described is an adhesively
securable fastener comprising a body having an opening; an adhesive
positioned within said body and registering with the opening of the
chamber, a plunger means for applying pressure to said adhesive such
as against an anvil means thereby adhesively securing the fastener to
a substrate. Also described is a method for securing a substrate
using the fastener described herein."
  }
],
"claim": {
  "language": "en",
  "text": "What is claimed is:\n 1.\n 1. a self-
contained adhesively securable fastener for securing an article\n to
a substrate wherein there is a body member containing an adhesive\n
reservoir with plunger means for expelling adhesive therefrom and
wherein\n the body member also has a surrounding surface portion
adapted to\n substantially mate with a substrate when the body member
is urged there\n against, the invention characterized by:\n channel
means in said surface portion radiating from adjacent said\n
reservoir toward the periphery of the surface portion for
distributing\n adhesive in the reservoir uniformly into the interface
between such\n surface portion and substrate when the body member is
urged thereagainst.\n 2.\n 2. The fastener of claim 1 there is
adhesive in the reservoir and the\n adhesive is a quicksetting
adhesive.\n 3.\n 3. The fastener of claim 1 there is adhesive in the
reservoir and the\n adhesive is a micro-encapsulated adhesive.\n 4.\n 4. The fastener of claim 1 there is adhesive in the reservoir and
the\n adhesive is an anaerobic adhesive.\n 5.\n 5. The fastener of

```

claim 4 wherein the adhesive is a micro-encapsulated\n anaerobic adhesive.\n 6.\n 6. The fastener of claim 1 wherein the plunger means contains a plurality\n of recessed areas for movement of the adhesive therethrough during the\n application of pressure to the adhesive by the plunger means.\n 7.\n 7. The fastener of claim 1 wherein the body is adapted to permit other\n instruments to be attached or to be hung therefrom.\n 8.\n 8. The fastener of claim 1 wherein the body has a threaded portion for\n application of a nut or bolt thereto.\n 9.\n 9. The invention defined by claim 1 wherein said adhesive reservoir is\n formed as a separate member from the body member whereby adhesive\n contained within the reservoir does not contact the body member until\n expelled from the reservoir.\n 10.\n 10. The invention defined by claim 9 wherein said reservoir is hermetically\n sealed.\n 11.\n 11. The invention defined by claim 10 wherein said reservoir is removably\n receivable in the body member.\n 12.\n 12. The invention defined by claim 1 wherein said surface portion\n terminates in a flexible peripheral lip for preventing entry of air into\n the interface between the surface portion and a substrate when the body\n member is urged thereagainst.\n 13.\n 13. A method of securing a fastener to a substrate comprising:\n a. holding the fastener of claim 12 to the substrate to be fastened;\n b. applying pressure on the plunger means;\n c. forcing the adhesive from the reservoir and into the interface between\n said surface portion and the substrate and substantially uniformly\n distributing it throughout the interface; and\n d. curing the adhesive thereby joining the fastener to the substrate\n 14.\n 14. The method of claim 13 wherein the adhesive is a quicksetting adhesive.\n 15.\n 15. The method of claim 14 wherein the quicksetting adhesive is an\n anaerobic adhesive.\n 16.\n 16. The method of claim 13 wherein the adhesive is a micro-encapsulated\n adhesive.\n 17.\n 17. The method of claim 13 wherein the adhesive is a micro-encapsulated\n anaerobic adhesive.\n 18.\n 18. The method of claim 13 wherein the adhesive from the body has the\n ability to extend to the periphery of the lip and throughout the\n circumference of the lip attached to the substrate."

},

"description": {
 "language": "en",
 "text": "BACKGROUND OF THE INVENTION\n The

present case is concerned with the fastener art as well as with the art\n of quicksetting adhesives.\n The utilization of quicksetting adhesive has been sharply diminished\n because of the handling problem associated with said adhesive. Mechanical\n fasteners have been considered undesirable because of the potential for\n the fastener to loosen or disengage, thereby losing its effectiveness.\n SUMMARY OF THE INVENTION\n It is an object of the present invention to provide a fastener which can be\n safe means for handling quickset adhesives. It is also an object of the\n present invention to mechanically and adhesively secure a fastener to a\n substrate. It is also an object of the present invention to describe a\n lightweight adhesively securable fastener.\n All of these objects are accomplished by an adhesively securable fastener\n comprising:\n a body having an opening;\n an adhesive positioned within said body; and\n registering with the opening of the body, a plunger means for applying\n pressure to said adhesive thereby adhesively securing the fastener to a\n desired substrate.\n preferably the adhesive is a quicksetting adhesive such as an anaerobic\n adhesive or a cyanoacrylate type.\n BRIEF DESCRIPTION OF THE DRAWINGS\n FIG. 1 is a schematic view of the adhesively securable fastener of the\n present invention in a side sectional view;\n FIG. 2 is the fastener of FIG. 1 in an assembled fashion immediately\n adjacent a substrate;\n FIG. 3 is an alternative embodiment of the fastener of FIG. 2 in an engaged\n condition;\n FIG. 4 is a bottom view of the fastener of FIG. 1;\n FIG. 5 is a side perspective view of the plunger means useful in the\n fastener of the present invention;\n FIG. 6 is a bottom view of the fastener of FIG. 3 taken along the line\n 6--6;\n FIG. 7 is an alternative embodiment of the fastener of the present\n invention inserted within a hole of a substrate to which the fastener is\n to be applied, said fastener showing deflectable tabs;\n FIG. 8 shows the fastener of FIG. 7 in an engaged condition;\n FIG. 9 shows an alternative embodiment of the plunger means useful in the\n fastener of the present invention;\n FIG. 10 shows the plunger means of FIG. 9

in engagement in the fastener of the present invention; FIG. 11 shows an alternative embodiment of the body of the fastener of the present invention; FIG. 12 is an alternative embodiment of the present invention showing a diaphragm holding the adhesive mass in the body of the fastener of the present invention; FIG. 13 is an alternative embodiment of the fastener of the present invention having a frangible stem on a plunger means; FIG. 14 shows an alternative means of holding the adhesive mass useful in the fastener of the present invention; FIG. 15 is an alternative means for a plunger useful in the fastener of the present invention; FIG. 16 is an alternative means for the body useful in the fastener of the present invention; FIG. 17 shows the combination of the fastener, and adhesive and the body of FIGS. 14, 15 and 16 assembled; FIG. 18 shows the fastener of FIG. 17 in an engaged position; FIG. 19 is an alternative embodiment of the fastener of the present invention in the unengaged condition; FIG. 20 shows the fastener of FIG. 19 in an engaged condition. [...]"

```
    },
    "passage": {
      "section": "description",
      "language": "en",
      "text": "means for handling quickset
```

adhesives. It is also an object of the present invention to mechanically and adhesively secure a fastener to a substrate. It is also an object of the present invention to describe a lightweight adhesively securable fastener. All of these objects are accomplished by an adhesively securable fastener comprising: a body having an opening; an adhesive positioned within said body; and registering with the opening of the body, a plunger means for applying pressure to said adhesive thereby adhesively securing the fastener to a desired substrate. preferably the adhesive is a quicksetting adhesive such as an anaerobic adhesive or a cyanoacrylate type. BRIEF DESCRIPTION OF THE DRAWINGS FIG. 1 is a schematic view of the adhesively securable fastener of the present invention in a side sectional view; FIG. 2 is the fastener of FIG. 1 in an assembled fashion immediately adjacent a substrate; FIG. 3 is an alternative embodiment of the fastener of FIG. 2 in

```
    },
    "image": {
      "data": "<base64 encoded data>"
    },
    "pdf": "https://api.auto-match.se/v2/pdf/US-
```

4830558-A"

```
    }
  ]
}
}
```

E r r o r R e s p o n s e	HTTP Status Code: 204 No Content (When no result could be found) <pre><blank></pre>
	HTTP Status Code: 404 Not Found (When wrong value passed into ticket parameter) <pre>{ "status": "error", "message": "Ticket not found", "data": null }</pre>
	HTTP Status Code: 404 Not Found (When empty value passed into ticket parameter) <pre>{ "status": "error", "message": "Parameter ticket can not be empty", "data": null }</pre>
	HTTP Status Code: 429 Too many requests (API rate limit exceeded) <pre>{ "message": "API rate limit exceeded" }</pre>
S a m p l e C a l (c U R L)	<pre>curl --request GET \ --url https://api.auto-match.se/v2.1/search?ticket=<value> \ --header 'authorization: xxx'</pre>
N o t e s	<p>Header</p> <p>authorization: An API authorization key must be sent with all requests. You need to contact the IPscreeener support team to get the API authorization key.</p> <p>Body</p> <p>ticket: The ticket value used to retrieve the results is the ticket session identification data string received by the POST search request.</p> <p>Success Response</p> <p>index value: After the parameter automatch-result there is an integer value, e.g. index-1. This value indicates the specific index and associated settings used for performing the matching procedure. If several indexes are targeted with a search (maximum three in parallel), each result list will be presented separately, one after the other.</p> <p>score: The score is the AutoMatch engine indication of the estimated relevance of a document.</p> <p>position: The position refers to the record number of the hit in the search results, where the search results are sorted on the relevance score value in descending order.</p> <p>country: This returns a two-letter country code describing the origin of the patent publication.</p> <p>number: It is the publication number for the application or the granted patent retrieved.</p>

kind-code: The kind codes are used to identify the type of patent publication. More information on this syntax is available at: www.wipo.org. Some of the most common kind codes are:

- A1 - Publ. of Application with search report
- A2 - Publ. of Application without search report
- B1 - Patent publication
- B2 - Patent after modification

family-number: Patent family is a set of interrelated patent applications filed in one or more countries to protect the same invention by inventor(s). Each patent family has a unique number and this value returns that number. Family numbers are provided by EPO according to the Inpadoc specification.

publication-date: The publication date is the date on which a patent application/grant is first published. It is the date on which the document is made available to the public.

bibliographic: The bibliographic data section includes reference data on who is behind the patent application as well as summary data on the content and the technical field.

applicant (array): This field provides information about the patent owner(s) or applicant(s).

- **country:** It is the two-letter country code of the patent owner(s) or applicant(s).
- **name:** This field returns the name of the patent owner(s) or applicant(s).

inventor (array): This field provides information about the inventor(s).

- **country:** It is the two-letter country code of the inventor(s).
- **name:** This field returns the name of the inventor(s).

class (array): The classification scheme is a system of codes that groups inventions according to technical area, where IPC and CPC is the most common. The class information is divided into the following hierarchy, including four sections:

- **sub-class:** This is the class information on group level, e.g. H04M.
- **group:** This is the class information on main level e.g. H04M15.
- **sub-group:** This is the complete class information e.g. H04M15/03.
- **type:** This declares the classification system referred to e.g. IPC, CPC

title (array): This section includes the full title of the patent.

- **language:** The language tag in which the title of patent is written.
- **text:** It is the title text of the patent.

abstract (array): This is the summary describing the essence of the scope of a patent.

- **language:** The language in which the abstract of patent is given.
- **text:** It is the text content of the abstract of the patent.

claim (array): A claim defines exactly what is claimed by the invention and therefore what is sought to be protected. It clearly lays down what the patent does and does not cover.

- **language:** The language in which the claim of patent is written.
- **text:** It is the text content of the claim of the patent.

description (array): The detailed description describes in detail what the invention is and how it is made and used. It reflects the complete picture of the invention.

- **language:** The language in which the description of patent is written.
- **text:** It is the text content of the description of the patent.

image: It contains the first image of the patent document.

- **data:** Image encrypted in Base64 format. This will always be the first image of the patent (if first image exists).

pdf: It provides a PDF API link. This link enables access to download the full PDF document.

passage: Shows the paragraph within a document that the AutoMatch engine considered to be most relevant to the query.

- **section:** The section where the relevant paragraph is located
- **data:** The paragraph within a document considered to be most relevant by the engine.
- **language:** The language in which the passage of patent is written.

The PDF API for document retrieval

The PDF API is used to retrieve the original patent document(s) corresponding to a search query. From the requested patent number input via an URL query, the API request returns a PDF file in base64 format.

URL	https://api.auto-match.se/v2/pdf/patent-number
Method	GET
Headers	Required: authorization=[UTF-8 encoded]
URL Parameters	Required: patent-number=[UTF-8 encoded]
Success Response	HTTP Status Code: 200 OK <div><pre>{ "status": "success", "message": "Resource Fetched Successfully.", "data": "<Complete PDF document encoded in Base64 format>" }</pre></div>

Error Response	HTTP Status Code: 401 Unauthorized (No API authorization key found) <pre>{ "message": "No API Key is found in request" }</pre>
	HTTP Status Code: 403 Forbidden (Using wrong API authorization key) <pre>{ "message": "Invalid authentication credentials." }</pre>
	HTTP Status Code: 403 Forbidden (When the IP restriction is activated and source IP is not whitelisted) <pre>{ "message": "Your IP address is not allowed." }</pre>
	HTTP Status Code: 404 Not found (PDF is not available) <pre>{ "status": "error", "message": "The requested url or its parameters are wrong.", "data": null }</pre>
	HTTP Status Code: 422 Unprocessable entity (When null value passed in patent-number parameter) <pre>{ "status": "error", "message": "Patent-number parameter should not be empty.", "data": null }</pre>
	HTTP Status Code: 429 Too many requests (API rate limit exceeded) <pre>{ "message": "API rate limit exceeded" }</pre>
Sample Call (cURL)	<pre>curl --request GET \ --url https://api.auto-match.se/v2/pdf/US-2006258329-A1 \ --header 'authorization: xxx'</pre>

Notes	<p>authorization: An API authorization key must be sent with all requests. You need to contact the IPscreeener support team to get the API authorization key.</p> <p>patent-number: This descriptor represents uniquely a patent publications. The full patent number is based on a country code, a number and a kind code, separated with hyphens, e.g. US-2006258329-A1. The format of the patent number must be DocDB.</p> <p><u>Success Response</u></p> <p>data: Retrieval of a complete PDF document encoded in Base64 format.</p>
--------------	--