Cisco SocialMiner
Developers Guide — Release 8.5(3)

Contents:

5.1.3. Parameters for Proxy API

- **enabled** - true/false. Whether proxy use is enabled.
- **host** - The fully-qualified hostname or IP address of the proxy server. (Required if enabled parameter is true.)
- **port** - the HTTP port for the proxy server. (Required if enabled parameter is true.)
- **exclusions** - a list of hostnames to exclude from being used by the proxy. (Optional) Wildcards can be used. Examples:
    - localhost
    - *.cisco.com
    - 161.44.*
    - 192.168.1.1
    - **Note:** The exclusion list is limited to 255 total characters (not including the `<exclusion>` tags). There is an additional character per item in the list that acts as a separator.

6. Campaign API

The SocialMiner Campaign API allows you to create, update, delete, get, and list campaigns in the SocialMiner system.

Campaigns are collections of [feeds](#) and [filters](#) and generate lists of results matching the criteria defined in the campaign.

This API is represented on the SocialMiner User Interface by the Campaigns gadget.

The URL to access the API is **http://server:port/ccp-webapp/ccp/campaign**.

6.1. Campaign API Commands

- [create](#)
- [update](#)
- [delete](#)
- [get](#)
- [list](#)
- [suggestedtags](#)
- [Parameters for Campaign API](#)

6.1.1. create

Create a campaign.

URL:	http://server:port/ccp-webapp/ccp/campaign
HTTP Method:	POST
Parameters:	See [below.](#)
Example XML Request Payload:	``` <Campaign> <name>MyTestCampaign</name> <description>This is my test campaign</description> <publicId>MyTestCampaign</publicId> <feeds> <feed>http://192.168.0.1:80/ccp-webapp/ccp/feed/5000</feed> <feed>http://192.168.0.1:80/ccp-webapp/ccp/feed/5001</feed> </feeds> <filters> <filter>http://192.168.0.1:80/ccp-webapp/ccp/filter/6000</filter> <filter>http://192.168.0.1:80/ccp-webapp/ccp/filter/6001</filter> </filters> </Campaign> ```

| Example using cURL: | ```
curl -i -H "Content-type: application/xml" -X POST -d "<Campaign><name>MyTestCampaign</name><publicId>MyTestCampaign</publicId><description>This is my test campaign</description><include>Cisco Expert Advisor</include><exclude>ICM</exclude><feeds><feed>http://192.168.0.1/ccp-webapp/ccp/feed/100162</feed><feed>http://192.168.0.1/ccp-webapp/ccp/feed/100165</feed></feeds><filters><filter>http://192.168.0.1/ccp-webapp/ccp/feed/100167</filter><filter>http://192.168.0.1/ccp-webapp/ccp/feed/100168</filter></filters></Campaign>" http://user:password@192.168.0.1/ccp-webapp/ccp/campaign
``` |
|---|---|
| HTTP Response Headers: | The URI of the created resource is returned if the creation if the campaign succeeded. HTTP response:<br><br>```
HTTP/1.1 201 Created
Location: http://192.168.0.1/ccp-webapp/ccp/campaign/MyTestCampaign
Content-Type: text/plain
Content-Length: 0
Date: Tue, 12 Jan 2010 16:41:14 GMT
```<br><br>See also HTTP Responses. |

6.1.2. update

Update an existing campaign.

URL:	http://server:port/ccp-webapp/ccp/campaign/<publicId>
HTTP Method:	PUT
Parameters:	See below.
| Example XML Request Payload: | ```
<Campaign>
        <name>MyTestCampaign</name>
        <description>This is my test campaign</description>
                <feeds>
                <feed>http://192.168.0.1:80/ccp-webapp/ccp/feed/5000</feed>
                <feed>http://192.168.0.1:80/ccp-webapp/ccp/feed/5001</feed>
        </feeds>
        <filters>
                <filter>http://192.168.0.1:80/ccp-webapp/ccp/filter/6000</filter>
                <filter>http://192.168.0.1:80/ccp-webapp/ccp/filter/6001</filter>
        </filters>
        <changeStamp>8</changeStamp>
</Campaign>
``` |
| Example using cURL: | ```
curl -i -H "Content-type: application/xml" -X PUT -d "<Campaign><name>MyTestCampaign 5 Edited</name><description>Edited Description</description><include>Cisco Expert Advisor</include><exclude>ICM</exclude><feeds><feed>http://192.168.0.1/ccp-webapp/ccp/feed/103111</feed><feed>http://192.168.0.1/ccp-webapp/ccp/feed/103112</feed></feeds><filters><filter>http://192.168.0.1/ccp-webapp/ccp/feed/100167</filter><filter>http://192.168.0.1/ccp-webapp/ccp/feed/100168</filter></filters></Campaign>" http://user:password@192.168.0.1/ccp-webapp/ccp/campaign/103115
``` |

6.1.3. delete

Delete an existing campaign.

URL:	http://server:port/ccp-webapp/ccp/campaign/<publicId>
HTTP Method:	DELETE
| Example using cURL: | ```
curl -i -X DELETE http://user:password@192.168.0.1:80/ccp-webapp/ccp/campaign/MyTestCampaign
``` |
| HTTP Response Headers: | ```
HTTP/1.1 200 OK
Content-Type: text/plain
Content-Length: 0
Date: Tue, 12 Jan 2010 17:03:54 GMT
``` |

6.1.4. get

Return the data for a single campaign.

URL:	http://server:port/ccp-webapp/ccp/campaign/<publicId>
HTTP Method:	GET
Example XML Response:	Response includes a *metrics/socialContactCount* element that contains the number of results included in the campaign.

```
<Campaign>
        <changeStamp>1</changeStamp>
        <refURL>
                http://192.168.0.1:80/ccp-webapp/ccp/campaign/MyTestCampaign
        </refURL>
        <link>
                http://192.168.0.1:80/ccp-webapp/ccp/campaign/MyTestCampaign/results
        </link>
        <name>MyTestCampaign</name>
        <description>This is my test campaign</description>
        <publicId>MyTestCampaign</publicId>
        <includeExpr>Cisco Expert Advisor</includeExpr>
        <excludeExpr>ICM</excludeExpr>
        <feeds>
                <feed>http://192.168.0.1:80/ccp-webapp/ccp/feed/5000</feed>
                <feed>http://192.168.0.1:80/ccp-webapp/ccp/feed/5001</feed>
        </feeds>
        <filters>
                <filter>http://192.168.0.1:80/ccp-webapp/ccp/feed/6000</filter>
                <filter>http://192.168.0.1:80/ccp-webapp/ccp/feed/6001</filter>
        </filters>
        <metrics>
                <socialContactCount>12</socialContactCount>
        </metrics>
        <suggestedTagsURL>
                http://192.168.0.1/ccp-webapp/ccp/campaign/MyTestCampaign/suggestedtags
        </suggestedTagsURL>
</Campaign>
```

HTTP Response Headers:	

```
HTTP/1.1 200 OK
Content-Type: application/xml
Transfer-Encoding: chunked
Date: Tue, 12 Jan 2010 16:55:05 GMT
```

6.1.5. list

List all campaigns.

URL:	http://server:port/ccp-webapp/ccp/campaign?summary=<true/false> OR http://ServerIP:8080/ccp-webapp/ccp/campaign?metrics=<true/false>
HTTP Method:	GET
URL Parameters:	**summary** - true/false (optional, defaults to "false"). When "true", only a subset of the XML elements are returned. **metrics** Optional. The number of social contacts associated with this campaign. False by default.
Example XML Response:	Response includes a *metrics/socialContactCount* element that contains the number of results included in the campaign.

```
<Campaigns>
        <Campaign>
                <refURL>
                        http://192.168.0.1:80/ccp-webapp/ccp/campaign/100004
                </refURL>
                <name>MyTestCampaign</name>
                <description>This is my test campaign</description>
                <publicId>MyTestCampaign</publicId>
```

```
                            <changeStamp>0</changeStamp>
                            <feeds>
                                    <feed>http://192.168.0.1:80/ccp-webapp/ccp/feed/100000</feed>
                                    <feed>http://192.168.0.1:80/ccp-webapp/ccp/feed/100001</feed>
                            </feeds>
                            <metrics>
                                    <socialContactCount>12</socialContactCount>
                            </metrics>
                    </Campaign>
                    <Campaign>
                            <refURL>http://192.168.0.1:80/ccp-webapp/ccp/campaign/100005</refURL>
                            <name>MyTestCampaign2</name>
                            <description>This is my test campaign</description>
                            <publicId>MyTestCampaign2</publicId>
                            <changeStamp>0</changeStamp>
                            <feeds>
                                    <feed>http://192.168.0.1:80/ccp-webapp/ccp/feed/100000</feed>
                                    <feed>http://192.168.0.1:80/ccp-webapp/ccp/feed/100001</feed>
                            </feeds>
                            <metrics>
                                    <socialContactCount>345</socialContactCount>
                            </metrics>
                    </Campaign>
            </Campaigns>
```

| HTTP Response Headers: | ```
HTTP/1.1 200 OK
Content-Type: application/xml
Transfer-Encoding: chunked
Date: Tue, 12 Jan 2010 16:59:22 GMT
``` |

6.1.6. suggestedtags [contents]

Retrieve the suggested tags for social contacts in this campaign. Up to ten tags are returned based on how recent and how often a tag has been used in this campaign.

URL:	http://server:port/ccp-webapp/ccp/campaign/<publicId>/suggestedtags
HTTP Method:	get
Example XML Response:	The first 10 suggested tags are returned ``` <SuggestedTags> <tags> <tag>fresh</tag> <tag>cool</tag> <tag>slide</tag> </tags> </SuggestedTags> ```

6.1.7. Parameters for Campaign API [contents]

Create	Update	Get
name: Required. The name of the campaign.	**name:** Optional. The name of the campaign.	**socialContactCount** Optional. The number of social contacts associated with this campaign
publicid: Optional. URL-encoded version of name. Must be unique within object type.	**publicid:** Optional. URL-encoded version of name. Must be unique within object type. When a campaign is created, SocialMiner generates a unique publicId for the campaign, based on the name you provide when you use the **create** command. The publicId is then used with the other methods for campaign. You can generate your own publicId rather than have SocialMiner generate one based on the campaign name. Spaces, slashes, and backslashes are not allowed in the publicId, it cannot be blank, and the publicId must conform to RFC 3986 for URI syntax. When SocialMiner creates a publicId from the provided name element it replaces spaces with underscores and slashes or backslashes with hyphens. SocialMiner then formats the resulting string according to RFC 3986. The publicId is not editable after it is created and does not change if you change the name of the Campaign.	—

description: Optional. The description of the campaign.	**description:** Optional. The description of the campaign.	—
includeExpr: Optional. The searching expression to include	**includeExpr:** Optional. The searching expression to include	—
excludeExpr: Optional. The searching expression to exclude	**excludeExpr:** Optional. The searching expression to exclude	—
feeds: Optional. A list of feeds linked to the campaign	**feeds:** Optional. A list of feeds linked to the campaign	—
filters: Optional. A list of filters linked to the campaign	**filters:** Optional. A list of filters linked to the campaign	—
-	**changeStamp:** Required. The change stamp of the campaign record which is returned in GET **Important**: You must provide the current *changeStamp* of the feed when you perform an update. If you do not provide the current changestamp then the update fails. This mechanism is in place so that two clients cannot edit the feed at the same time. The changeStamp increments by 1 if the update is successful.	—

7. Campaign Results API

The SocialMiner Campaign Results API allows you to get the results for a campaign.

This API is represented on the SocialMiner User Interface in the Home tab.

The URL to access the API is **http://server:port/ccp-webapp/ccp/campaign/<publicId>/results**.

7.1. Commands

7.1.1. get

Get results for the specified campaign.

URL:	http://server:port/ccp-webapp/ccp/campaign/<publicId>/results
HTTP Method:	GET
URL Parameters:	• **postId**: String (optional) If provided, results are displayed started after the provided postId. Cannot be used if timestamp or firstResultIndex is specified. • **includePostWithPostId**: (optional) This option is ignored if postId is not specified. If includePostWithPostId is not specified or is specified and is set to false, the contact with ID equal to postId is not included in the campaign results. If includePostWithPostId is specified and is set to true, the contact with ID equal to postId is included in the campaign results. • **numberOfResults**: Integer (optional) The maximum number of results to be returned. Default is 50 and maximum is 200. • **timestamp**: Integer (optional) Display results older results than this timestamp. Defaults to the time of request if not provided. If firstResultIndex is not specified then timestamp assumes firstResultIndex = 0. • **firstResultIndex:** Integer (optional) The number of results to skip based on the timestamp. Used for pagination. Assuming numberOfResults is set at the default of 50, then you could create a "page 2" link by using the timestamp provided in the href of the `feed/link rel="self"` and a firstResultIndex of 50. Page 3 would use the same timestamp and a firstResultIndex of 100, etc.

- **filterStatus**: String (optional) . Display Social Contacts whose status matches a status within this field. Defaults to all if the parameter is not specified. You must specify a value for the parameter if the parameter is included or no social contacts are returned. Can be one or more of (case-insensitive):
  - RESERVED
  - HANDLED
  - UNREAD
  - DISCARDED
  - Multiple status example: http://192.168.0.1/ccp-webapp/ccp/campaign/Business_News/results**filterStatus=RESERVED&filterStatus=HANDLED**
- **filterTag**: String (optional). Display Social Contacts whose filter(s) matches one or more of the tags in this field. Defaults to all tags if not specified. Example: http://192.168.0.1/ccp-webapp/ccp/campaign/Business_News/results**filterTag=tag1&filterTag=tag2**
- **Notes:**
  - If timestamp is provided and firstResultIndex is not provided, then the results are displayed up to the"numberOfResults" with a creation date older than "timestamp", starting at index 0.
  - If timestamp is not provided and firstResultIndex is provided, then the results are displayed up to the "numberOfResults" with a creation date older than "now" starting at firstResultIndex.
  - If postId is provided, then the contact identified by the postId is used as the basis for the search. The social contact for the provided postId does not appear in the results.

Example Response:	Results are returned as an [ATOM 1.0](#) feed with the following elements: - **feed/title**: The name of the Campaign. - **feed/link rel = self & feed/id**: The URL of the results. - **feed/link rel = countsince:** The URL for the API call for the number of new social contacts since this result was retrieved. - **feed/link rel = next:** The URL to the next 50 results. Present only when there are more than 50 results left in the campaign. - **entry/title**: The title of the social contact. - **entry/link rel = alternate & entry/id**: The URL to the social contact. - **entry/link rel = socialcontact:** The URL for the API call of this social contact. - **entry/summary**: The content of the social contact. - **entry/published**: The date and time that the social contact was published in the form of *YYYY-MM-DD*T*HH:MM:SS*Z. If the social contact did not contain a published date then this is the date when the social contact was read by SocialMiner. - **entry/ccp:statustimestamp:** The timestamp of the last change to the status of the social contact. - **entry/ccp:status:** The current status of the social contact. - **entry/ccp:sctags/ccp:sctag** One or more tags associated with this social contact. - **entry/ccp:scstatususerid** The last user to change the status of this social contact. If blank and the status is unread then this social contact has never had a status change.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:ccp="http://www.cisco.com/ccbu/ccp/xml/socialcontact/1.0/" xmlns:dc="http://purl.org/dc/elements/1.1/">
        <title>Business News</title>
        <link rel="self" href="http://192.168.0.1/ccp-webapp/ccp/campaign/Business_News/results?timestamp=1278696169003" />
        <link rel="countsince" href="http://192.168.0.1/ccp-webapp/ccp/campaign/Business_News/count?postId=22E00F5310000129460A1EB40A568DDE" />
        <link rel="next" href="http://192.168.0.1/ccp-webapp/ccp/campaign/JingCampaign1/results?timestamp=1276525157775&amp;firstResultIndex=50" />
        <subtitle>This feed has been created by Cisco Customer Collaboration Platform</subtitle>
        <id>http://192.168.0.1/ccp-webapp/ccp/campaign/Business_News/results</id>
        <updated>2010-06-10T17:20:46Z</updated>
        <dc:date>2010-06-10T17:20:46Z</dc:date>
        <entry>
                <title>SEC OKs 'flash crash' fix</title>
                <link rel="alternate" href="http://rss.cnn.com/~r/rss/money_latest/~3/h0MxRXFew9I/index.htm" />
                <link rel="socialcontact" href="http://192.168.0.1/ccp-webapp/ccp/socialcontact/22E00F5310000129460A1EB40A568DDE" />
                <author>
                        <name />
                </author>
                <id>http://rss.cnn.com/~r/rss/money_latest/~3/h0MxRXFew9I/index.htm</id>
                <updated>2010-06-10T17:10:50Z</updated>
                <published>2010-06-10T17:10:50Z</published>
                <summary type="html">The Securities and Exchange Commission approved new rules Thursday that will halt trading uniformly across all U.S. markets for stocks experiencing wild price swings to prevent a repeat of last
month's "flash crash."&lt;img src="http://feeds.feedburner.com/~r/rss/money_latest/~4/h0MxRXFew9I" height="1" width="1"/&gt;</summary>
                <dc:creator />
                <dc:date>2010-06-10T17:10:50Z</dc:date>
                <ccp:scstatustimestamp>1283819058417</ccp:scstatustimestamp>
                <ccp:scstatus>reserved</ccp:scstatus>
                <ccp:scstatususerid>admin</ccp:scstatususerid>
                <ccp:sctags>
                        <ccp:sctag>sometag</ccp:sctag>
                        <ccp:sctag>anothertag</ccp:sctag>
                        <ccp:sctag>yetanothertag</ccp:sctag>
                </ccp:sctags>
        </entry>
        <entry> ... </entry>
        <entry> ... </entry>
        <entry> ... </entry>
</feed>
```

8. Campaign Results Count API

The SocialMiner Campaign Results Count API allows you to get a count of the results for a specified campaign. You can get a count of the results for the entire campaign or the count of results since a given post.

The URL to access the Campaign Results Count API is **http://server:port/ccp-webapp/ccp/campaign/<publicId>/count**.

8.1. Campaign Results Count API Commands

8.1.1. get

Get the number of results in an entire campaign or since a given postId.

URL:	http://server:port/ccp-webapp/ccp/campaign/<publicId>/count
HTTP Method:	GET
URL Parameter:	**postId**: (optional) The postId of the last post seen. This id is the unique id of the campaign results. A link to this API with the appropriate id in the url is included in the results atom feed. The count displays the number of results published after the referenced campaign result. If no postId is provided, the full number of results in the campaign is returned. The postId can be found in the campaign results, for example: `<link rel="countsince" href="http://192.168.0.100/ccp-webapp/ccp/campaign/MyTestCampaign/count?postId=92517B6610000128295CEBB40A568DDE" />`
Example XML Response:	`<count>44</count>`

9. Authentication API

The authentication API allows you to configure a connection to a Microsoft Active Directory (AD) server. You can specify all users who exist on an AD to have access to SocialMiner or you can specify a single group of AD users.

This API is represented on the SocialMiner User Interface in the System Administration gadget.

Note: Only the administrator created during install can use this API.

The URL to access the API is **http://server:port/ccp-webapp/ccp/authentication/**.

9.1. Authentication API Commands

- get
- update
- Parameters for Authentication API

9.1.1. get

Get the authentication information from CCP.

URL:	http://server:port/ccp-webapp/ccp/authentication/
HTTP Method:	GET

Example XML Response:	`<Authentication>`
	`    <enabled>true</enabled>`
	`    <managerDistinguishedName>CN=administrator,CN=users,DC=ccbu-doc-ad,DC=cisco,DC=com</managerDistinguishedName>`
	`    <managerPassword>********</managerPassword>`
	`    <primaryHost>10.86.132.145</primaryHost>`
	`    <primaryPort>3268</primaryPort>`

```
                <primaryUseSSL>false</primaryUseSSL>
                <refURL>http://192.168.0.1/ccp-webapp/ccp/authentication</refURL>
                <roleName></roleName>
        </Authentication>
```

Parameters:	See below.

9.1.2. update

update the authentication information from SocialMiner.

URL:	http://server:port/ccp-webapp/ccp/authentication/
HTTP Method:	PUT
Example XML Payload:	<pre><Authentication>\n <enabled>true</enabled>\n <primaryHost>ad.server</primaryHost>\n <primaryPort>3268</primaryPort>\n <primaryUseSSL>false</primaryUseSSL>\n <managerDistinguishedName>cn=admin,ou=users,dc=ad,dc=server</managerDistinguishedName>\n <managerPassword>password</managerPassword>\n <roleName>CCP_Users</roleName>\n</Authentication></pre>
Parameters:	See below.

9.1.3. Parameters for Authentication API

- **enabled**: true/false. Indicates if the authentication settings are used when trying to authenticate a user. If this is disabled then only the application administrator will have access to the system.
- **primaryHost**: Required if enabled. The host address of the AD server.
- **primaryPort**: Required if enabled. The host port.
- **primaryUseSSL**: true/false. Indicates if a secure connection should be established. This requires that a domain certificate be uploaded to the server and that the primaryPort allows secure connections. **If set to true, then you must follow the instructions in** Enabling SSL for Active Directory Authentication
- **managerDistinguishedName**: Required if enabled. The distinguished name of a user that has manager access to the AD server. For example CN=Administrator, CN=users, DC=MYSERVER, DC=COM.
- **managerPassword**: Required if enabled. The password of the user specified in the managerDistinguishedName field.
- **roleName**: Optional. The name of an AD role or group. All users in this AD role or group are provided access to CCP. Users on the AD who are not members of this role or group are not provided access to CCP. Blank or * indicates that all users in the AD are allowed to use the application.

9.2. Enabling SSL for Active Directory Authentication

You can enable secure authentication (SSL) against a Microsoft Active Directory server by exchanging the SocialMiner certificate with the AD server.

To enable SSL for the Active Directory connection:

On the Active Directory Server:

1. Verify that the Active Directory has the Certificate Services service installed.
2. Select **All Programs > Administrative Tools > Certificate Authority**.
3. Expand the domain node and select **Issued Certificates**.
4. Double click the certificate to open it
5. Open the Details tab and click **Copy to file**.
6. An Export wizard appears. In the wizard select DER encoded binary.
7. Using the wizard to select a location to save the file.

8. Click **Finish**.

On the SocialMiner Server:

1. Open the *Cisco Unified Operating System Administration* page. The link is available from the administration gadget.
2. Select **Security > Certificate Management**.
3. Click **Upload Certificate**.
4. For the Certificate Name, select **tomcat-trust**.
5. In the Upload File field, select the file to upload by clicking **Browse...** Select the certificate file you saved from the Active Directory server.
6. Click **Upload File**.
7. Restart the Cisco Tomcat service. Using the CLI, run the command `utils service restart Cisco Tomcat`.

10. Conversation API (Twitter)

The Conversation API allows you to view the conversational context of a specific Twitter social contact. If a Twitter social contact is in reply-to another tweet stored in SocialMiner, then this conversation can be retrieved. The API extracts all Twitter social contacts in a reply "chain".

To stay within Twitter's rate limits, Twitter conversations for tweets are constructed using the reply-to IDs stored in SocialMiner's datastore.

Please note that specifying an older tweet in the conversation will not show the newer tweets in the conversation. The conversation API will only list the replies to the selected contact. It will not list any replies that the selected contact was in reply to.

Also note that the conversation API is based on the reply-to IDs stored in SocialMiner. Reply-to IDs are only stored for contacts that are retrieved using a Twitter Account or Twitter Stream feed.

Note: Twitter Direct Messages (DMs) do are not linked to each other by Twitter, so the conversation aspect of DMs can not be retrieved.

10.1. Conversation API Commands

You can perform the following command:

10.1.1. list

Show the conversation of a given social contact.

URL:	http://server:port/ccp-webapp/ccp/conversation/<SocialContactID>/?replies=xxx
HTTP Method:	GET
URL Parameters:	• SocialContactID: string. The ID of the social contact whose conversation will be retrieved. Required. • replies: integer. Optional. The maximum number of replies to retrieve. Default to 32.
Example HTTP Request:	`https://192.168.0.1/ccp-webapp/ccp/conversation/25FD59151000012E001FFC6B0A568DF5?replies=3`

Example XML Response:	<pre><SocialContacts>
	  <SocialContact>
	    <author>ContactAuthor</author>
	    <description>Text of the tweet</description>
	    <id>F1217D711000012D000003AB0A568DF5</id>
	    <link>http://twitter.com/ContactAuthor/statuses/33460956554076160</link>
	    <publishedDate>1296827212000</publishedDate>
	    <refURL>http://scoialminer.cisco.com/ccp-webapp/ccp/socialcontact/F1217D711000012D000003AB0A568DF5</refURL>
	    <status>unread</status>
	    <statusTimestamp>1296830659988</statusTimestamp>
	    <statusUserId/>
	    <tags/>
	    <title>Tweet: ContactAuthor to Agent</title>
	  </SocialContact>
	  <SocialContact>
	    <author>Agent</author>
	    <description>Text of the tweet</description>
	    <id>F1217D711000012D000003AB0A568DF5</id>
	    <link>http://twitter.com/ContactAuthor/statuses/33460956554076160</link>
	    <publishedDate>1296827212000</publishedDate>
	    <refURL>http://scoialminer.cisco.com/ccp-webapp/ccp/socialcontact/F1217D711000012D000003AB0A568DF5</refURL>
	    <status>unread</status>
	    <statusTimestamp>1296830659988</statusTimestamp>
	    <statusUserId/>
	    <tags/></pre>

```
            <title>Tweet: Agent to ContactAuthor</title>
        </SocialContact>
        <SocialContact>
            <author>ContactAuthor</author>
            <description>Text of the tweet</description>
            <id>F1217D711000012D000003AB0A568DF5</id>
            <link>http://twitter.com/ContactAuthor/statuses/33460956554076160</link>
            <publishedDate>1296827212000</publishedDate>
            <refURL>http://scoialminer.cisco.com/ccp-webapp/ccp/socialcontact/F1217D711000012D000003AB0A568DF5</refURL>
            <status>handled</status>
            <statusTimestamp>1296830659988</statusTimestamp>
            <statusUserId>Agent</statusUserId>
            <tags/>
            <title>Tweet: ContactAuthor</title>
        </SocialContact>
    </SocialContacts>
```

HTTP Response Headers:	A *200 OK* HTTP header is returned on success.

11. Email API

The Email API allows you to configure a single SMTP server connection. An SMTP server connection is required to send email notifications.

This API is represented on the SocialMiner User Interface in the System Administration gadget.

The URL to access the API is **http://server:port/ccp-webapp/ccp/email/default**.

Note: Only the administrator created during install can use this API.

11.1. Email API Commands

- get
- update
- Parameters for Email API

11.1.1. get

Get the SMTP configuration.

URL:	http://server:port/ccp-webapp/ccp/email/default (requires administrator privileges)
HTTP Method:	GET
Example XML Response:	``` <Email> <smtpHost>10.86.141.293</smtpHost> <smtpPort>587</smtpPort> <smtpFromUser>FromUser@Here.net</smtpFromUser> <smtpHostUserName>userNameForEmailServer</smtpHostUserName> <smtpAuthenticationEnabled>true</smtpAuthenticationEnabled> <smtpEnabled>true</smtpEnabled> <smtpSslEnabled>true</smtpSslEnabled> <refURL>http://10.86.141.223/ccp-webapp/ccp/email/default</refURL> </Email> ```
Parameters:	See below.

11.1.2. update

Update the SMTP configuration.

URL:	http://server:port/ccp-webapp/ccp/email/default (requires administrator privileges)
HTTP Method:	PUT

Parameters:	See below.

11.1.3. Parameters for Email API

- **smtpHost**: The fully qualified host address of the SMTP server. (required when smtpEnabled = true)
- **smtpPort**: The SMTP port. Default is 587. (required when smtpEnabled = true)
- **smtpFromUser**: The email (reply-to) address of email sent by this SocialMiner server. (required when smtpEnabled = true)
- **smtpHostUserName**: The username used to log into the SMTP server.(required when smtpAuthenticationEnabled = true)
- **smtpHostUserPassword**: The password used to log into the SMTP server.(required when smtpAuthenticationEnabled = true)
- **smtpAuthenticationEnabled**: Whether SMTP authentication is required for the SMTP host. (optional, boolean)
- **smtpEnabled**: Whether this SMTP configuration is enabled. (optional, boolean, defaults to false)

12. Filter API

The SocialMiner Filter API allows you create, update, and delete filters.

This API is represented on the SocialMiner User Interface in the Filters gadget.

The URL to access the API is **http://server:port/ccp-webapp/ccp/filter**.

12.1. Filter API Commands

You can perform the following commands:

- create
- delete
- list
- get
- update
- Parameters for Filter API

12.1.1. create

Create a filter to be stored in the database.

URL:	http://server:port/ccp-webapp/ccp/filter
HTTP Method:	POST
Parameters:	See below.
Example XML Request Payload:	``` <Filter> <name>String</name> <description>String</description> <type>3</type> <rule>1</rule> <keywords> <keyword>Author to Exclude</keyword> </keywords> </Filter> ```
HTTP Response Headers:	The response contains the URL for the newly created filter. ``` HTTP/1.1 201 Created Location: http://192.168.0.1:80/ccp-webapp/ccp/filter/1266345862276 Content-Type: text/plain Content-Length: 0 Date: Tue, 16 Feb 2010 19:35:56 GMT ```