BaswareONE HTTP interface

Index

REVISION HISTORY

Rev	Description of the change	Author	Date
1.0	Initial release	Niels Grootvriendt	16/04/2012
1.1	Basware styling	Thomas Dekeyser	15/10/2015

1 Introduction

Some companies want to connect to BaswareONE, but want to integrate their application instead of using Bizconnect. Those companies can connect to BaswareONE using the following scenarios.

2 Authentication

BaswareONE allows 2 types of authentication:

- Basic Authentication
- Certificate Logon

Which type is to be used depends on the configuration of the server you want to connect from. So, for each request, you should provide the correct authentication type.

3 Scenario’s with message flows

3.1 Synchronous upload of files to BaswareONE

3.1.1 Without signing the file

- Create a file on the back end system. This file can be encrypted, but this will have no added value when using a HTTPS connection. A HTTPS connection will be required by default.
- Push the file to BaswareONE Document Broker as described in 4.1.
- The BaswareONE HTTP server will receive the file and pass a part of it to the Document Router.
- This Document Router will decide what to do with it and give a response to the BaswareONE HTTP server. When the file was encrypted, the Document Router will have to figure out what to do with the file based on the context of the file, and not on the content. Decrypting the file will happen later on in the process.
- If the response was positive, the HTTP server will insert the document with the correct parameters into BaswareONE and returning the requester a response with audit information that can be used in the back end system.
- If the response was negative or the insert didn’t work for some reason, an error is returned to the requester with some information about the error.

3.1.2 With signing the file

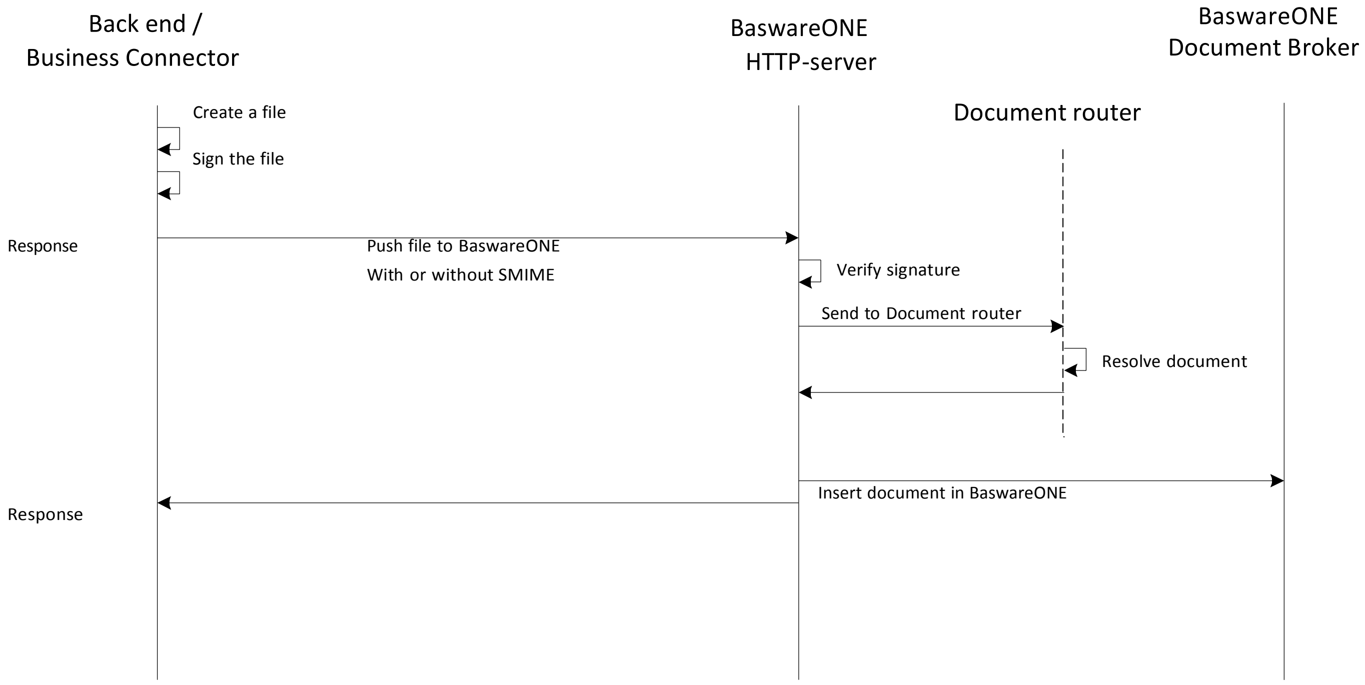
Create a file on the back end system (this can be encrypted, but see 3.1.1) and sign the file. Make sure the file is signed using S/Mime so the signature is part of the file!

Push the file to BaswareONE Document Broker as described in 4.2.

The BaswareONE HTTP server will first try to verify the signature based on the user profile and will pass the file without signature to the Document Router.

From now on, it will be processed as any other file as described above in 3.1.1.

3.1.3 Message flow



3.2 Asynchronous upload of files to BaswareONE

3.2.1 Server side setup

The asynchronous upload can be used for systems with a high load. The main part in having an asynchronous system is the usage of a message-queue between receiving the message and the BaswareONE back-end. In this way, the server remains running and accepting messages.

3.2.2 Client side response

For the client, the only difference with synchronous uploading is the response, which is received.

Sample result

```
<upload>
   <id>77460</id>
   <contentlength>4176</contentlength>
   <checksum type="md5">BBA80A67A0343A0168372731826CAC0</checksum>
   <time format="CCYYMMDDHHMMSS">20050202121516</time>
   <ref1>clientref1</ref1>
   <ref2>clientref2</ref2>
</upload>
```

3.3 Downloading files from BaswareONE

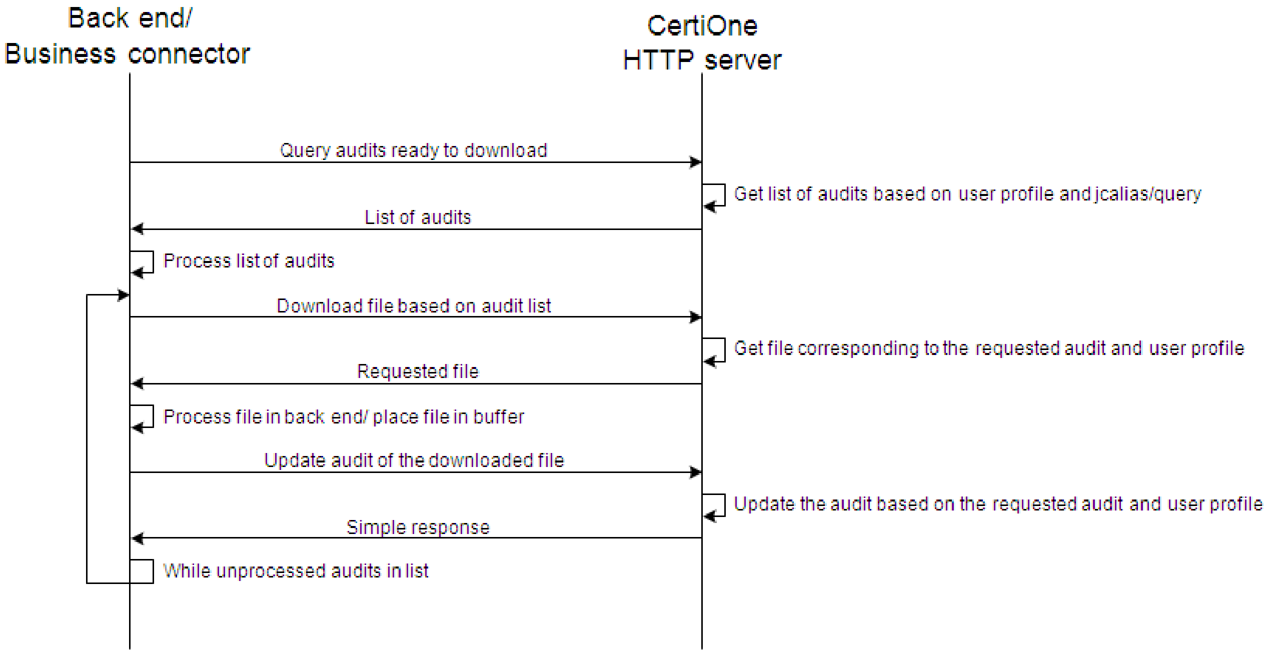
3.3.1 Downloading files

See which messages are ready to download by launching an URL as specified in 4.3. This will return a list of audit records in XML- format. This list will depend on the supplied user profile.

For each audit record in that list

- You can ask to download an audit file associated with that audit as specified in 4.4. This will return the file as-is.
- You can then process this file on the back end or put it ready to process.
- When done so, you can update the status of that audit on BaswareONE by requesting so as specified in 4.5.This way you won’t download any audit files twice.

3.3.2 Message flow


Back end/
Business connector
CertiOne
HTTP server
Query audits ready to download
Get list of audits based on user profile and jcalias/query
List of audits
Process list of audits
Download file based on audit list
Get file corresponding to the requested audit and user profile
Requested file
Process file in back end/ place file in buffer
Update audit of the downloaded file
Update the audit based on the requested audit and user profile
Simple response
While unprocessed audits in list

4 HTTP(s) Communication

Remarks

HTTPS is the standard!

The servlet will be username/password secured. This means that you first have to do a basic logon before you can start communicating.

All HTTP actions can be simulated in any browser by giving the correct URL.

4.1 URLs, SSL

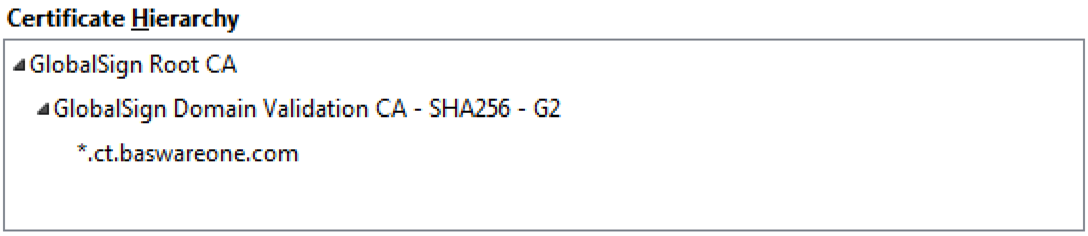
Environment	URL	Port
Test	https://www.ct.baswareone.com/httpserver	443
Production	https://www.baswareone.com/httpserver	443

Testing URL's

Environment	URL	Port
Test	https://www.ct.baswareone.com/httpserver?ai=blank	443
Production	https:// www.baswareone.com /httpserver?ai=blank	443

The SSL webserver Certificates used for the HTTPS connection can be wildcard certificates (like on "*.baswareone.com")

For example: Test certificates



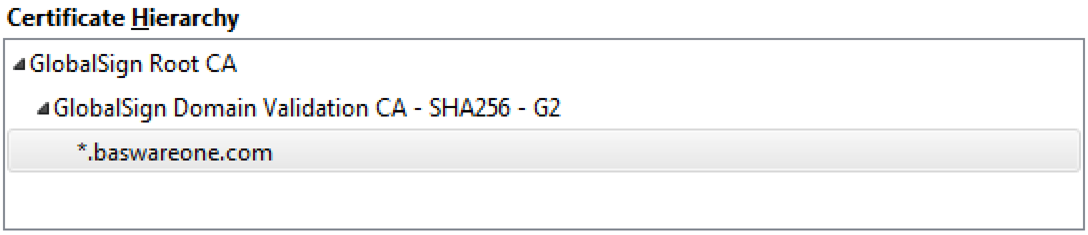
Certificate Hierarchy

◢GlobalSign Root CA

◢GlobalSign Domain Validation CA - SHA256 - G2

*.ct.baswareone.com

For example: production certificates



Certificate Hierarchy

◢GlobalSign Root CA

◢GlobalSign Domain Validation CA - SHA256 - G2

*.baswareone.com

The SSL Certificate chain can be easily retrieved by using any browser.
Hint: http://www.hosting.com/support/ssl/check-validity-of-ssl-certificate

4.2 File upload (HTTP POST)

Sample https post target: https://www.ct.baswareone.com/httpserver

Request headers

Parameter key	Parameter value	Mandatory
jcalias	bc.upload	Yes
Content-Type	application/mrt | application/edi | application/xml | application/obj	Yes
Authorization	When using basic authentication	Yes (unless certificate logon)
tid	<unique transaction id>	No
sender	<sendername>	No
receiver	<receivername>	No
messagetype	<messagetype>	No
route-network	<network to route the message to after processing on serverside>	No
route-address	<address to route the message to after processing on serverside>	No
<extrapropname>	<extrapropvalue>	No

Remarks

Authentication should be provided as explained in paragraph 2.

The non-mandatory parameters, mentioned in the table above, are recommended in order to use easy processing at server side.
The mime type should be correct e.g.: "multipart/form-data".
Pushing an encrypted file is possible but useless when using HTTPS.

Sample of TCP dump

```
POST /httpserver HTTP/1.1
Authorization: Basic U0RWOnN0ZXZlbg== Content-Type: application/mrt
jcalias: bc.upload
tid: sapidoc_democorp_shipnote.001.mrt
sender: mycompany
receiver: mytradingpartner messagetype: order
Host: test.certipost.be
Accept: text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2
Connection: keep-alive
……..
```

Sample result (OK)

```
<jcinsert>
   <auditno>77460</auditno>
   <icrf>17</icrf>
   <info>
      <dataset>dataset1</dataset>
      <sender>sendername</sender>
      <receiver>receivername</receiver>
      <intchggroup>intchggroupname</intchggroup>
      <intchgtype>intchgtypename</intchgtype>
      <intchgindividual>intchgindividualname</intchgindividual>
   </info>
   <docrouterrule><![CDATA[ 12:myrule ]]></docrouterrule>
</jcinsert>
```

Sample result (NOK)

```
<cs_error>
   <code>45</code>
   <msg>documentrouter error:docrouter1:returned no response:null</msg>
   <exception/>
   <location1>DocRouterAction.initAction</location1>
   <location2/>
   <location3/>
   <stacktrace/>
</cs_error>
```

4.3 File upload with SMIME (HTTP POST)

Sample https post target: https://www.ct.baswareone.com/httpserver

Request headers

Parameter key	Parameter value	Mandatory
jcalias	bc.upload	Yes
implicit	false	Yes

Content-Type	<complete signed contenttype>
	eg. multipart/signed; micalg=SHA-1; protocol="application/pkcs7-signature";
	boundary="----=_Part_11_29492753.1107787567805"
		Yes

datacontenttype	application/mrt | application/edi | application/xml | application/obj	No
tid	<unique transaction id>	No
sender	<sendername>	No
receiver	<receivername>	No
messagetype	<messagetype>	No
route-network	<network to route the message to after processing on serverside>	No
route-address	<address to route the message to after processing on serverside>	No

<extrapropname>	<extrapropvalue>	No

Remarks

- You should use the POST method to send the signed file.
- Authentication should be provided as explained in paragraph 2.
- The non-mandatory parameters, mentioned in the table above, are recommended in order to use easy processing on server side.
- The server can only verify a signature on the fly; not decrypt a file.

Sample of TCP dump

```
POST /httpserver HTTP/1.1
Authorization: Basic U0RWOnN0ZXZlbg==
Content-Type: multipart/signed; micalg=SHA-1; protocol="application/pkcs7-signature";
    boundary="----=_Part_11_29492753.1107787567805"
 implicit: false
tid: sapidoc_democorp_shipnote.001.mrt
sender: mycompany
receiver: mytradingpartner
messagetype: order datacontenttype: application/mrt
jcalias: bc.upload
Host: test.certipost.be
Accept: text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2
Connection: keep-alive
……..
```

Sample result (OK)

```
<jcinsert>
    <auditno>77460</auditno>
    <icrf>17</icrf>
    <docrouterrule><![CDATA[ 12:myrule ]]></docrouterrule>
</jcinsert>
```

Sample result (NOK)

```
<cs_error>
    <code>1</code>
    <msg>Exception during veryfing message:user certificate is not trusted by root certificates in rootcerts</msg>
    <exception>java.lang.SecurityException</exception>
    <location1>SMIMEAction.writeResponse</location1>
    <location2/>
    <location3/>
    <stacktrace>java.lang.SecurityException:java.lang.SecurityException: user certificate is not trusted by root certificates in rootcerts at 7) at … java.lang.Thread.run(Unknown Source)</stacktrace>
    <param1>abcdefg</param1>
</cs_error>
```

4.4 Getting list of audit records (HTTP GET)

Sample https post target: https://www.ct.baswareone.com/httpserver

URL parameters

Parameter key	Parameter value	Mandatory
jcalias	bc.querylist	Yes

Remarks

The returned list of audit records depends on the user profile (user name & password), but also depends on the parameter jcalias. For each user multiple queries can be defined. Each such query can be called by its name; the value of the parameter jcalias.

Sample of TCP dump

```
GET /httpserver HTTP/1.1
Authorization: Basic Y29vbGRyaW5rczI6Y29vbGRyaW5rczI=
jcalias: bc.querylist
Host: test.certipost.be
Accept: text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2
Connection: keep-alive
```

Sample result (OK)

```
<queryresult>
  <be.netbase.databinding.Wrapper>
    <data_vector>
      <be.netbase.jc.core.AuditRecord>
        <enabled>true</enabled>
        <contenttype>28005</contenttype>
        <auditno>123</auditno>
        <intchgindividual>XML_INVOVERVIEW_CERTIPOST_DUMMY</intchgindividual>
        <receiver>DUMMY</receiver>
        <sender>CERTIPOST</sender>
        <intchggroup>INVOVERVIEW</intchggroup>
        <intchgtype>XML_INVOVERVIEW_CERTIPOST</intchgtype>
        <creationdate>20091019191347</creationdate>
        <network>DISK</network>
        <status>102</status>
        <customstatus>0</customstatus>
        <modificationdate>20091028141203</modificationdate>
        <sentreceiveddate>20091028141203</sentreceiveddate>
        <constransdate>20091028141104</constransdate>
        <importexportdate>20091019191347</importexportdate>
        <preprocesspostprocessdate>0</preprocesspostprocessdate>
        <businessprocesseddate>20091019191522</businessprocesseddate>
        <tpreceiveddate>0</tpreceiveddate>
        <tpprocesseddate>0</tpprocesseddate>
        <customdate>0</customdate>
        <edireference>1</edireference>
        <networkreference>/usr/local/volume1/jcommercec1/spool/in/dsk.cs.in.1254985089987.15
984.dat</networkreference>
        <businessreference1>Dummy Organization</businessreference1>
        <businessreference2>Dummy Organization</businessreference2>
        <businessreference3>HTTP</businessreference3>
        <tpreference1></tpreference1>
        <tpreference2>;;;;;;NOBILL</tpreference2>
        <tpreference3>SAXMAPPING</tpreference3>
        <customreference1></customreference1>
        <customreference2></customreference2>
        <networksessionid>conn.-810141359701619413.-
1038857329813289025.txt</networksessionid>
        <traceid></traceid>
        <traceid2>dom.-8670883796824885677.-5866130800325163409.txt</traceid2>
        <traceid3>conn.-810141359701619413.-1038857329813289025.txt</traceid3>
        <traceid4>374943</traceid4>
        <customkeyvalue1></customkeyvalue1>
        <customkeyvalue2>1656</customkeyvalue2>
        <customkeyvalue3>xml=538</customkeyvalue3>
        <customkeyvalue4>161</customkeyvalue4>
        <misc1>xxxyyy</misc1>
        <misc2></misc2>
        <misc3></misc3>
        <misc4></misc4>
```

```
          <misc5></misc5>
          <timestamp>1256735523283</timestamp>
          <timestampstring></timestampstring>
          <name></name>
          <persistid>5</persistid>
          <checksumalgorithm>STORED</checksumalgorithm>
          <checksumvalue>Ca/TokdHRTF5/hNQIEegjw==</checksumvalue>
          <customstatus2>0</customstatus2>
          <customstatus3>104</customstatus3>
          <customstatus4>0</customstatus4>
          <customstatus5>2</customstatus5>
          <misc6>outbound;5;sax.2.pdf</misc6>
          <misc7>sax.2.pdf</misc7>
          <misc8></misc8>
          <actor></actor>
          <occasion></occasion>
        </be.netbase.jc.core.AuditRecord>
        <be.netbase.jc.core.AuditRecord>
          <enabled>true</enabled>
          <contenttype>28005</contenttype>
          <auditno>124</auditno>
          <intchgindividual>XML_INVOVERVIEW_CERTIPOST_DUMMY</intchgindividual>
          <receiver>DUMMY</receiver>
          <sender>CERTIPOST</sender>
          <intchggroup>INVOVERVIEW</intchggroup>
          <intchgtype>XML_INVOVERVIEW_CERTIPOST</intchgtype>
          <creationdate>20091109154200</creationdate>
          <network>DISK</network>
          <status>102</status>
          <customstatus>0</customstatus>
          <modificationdate>20091109154304</modificationdate>
          <sentreceiveddate>20091109154304</sentreceiveddate>
          <constransdate>20091109154204</constransdate>
          <importexportdate>20091109154200</importexportdate>
          <preprocesspostprocessdate>0</preprocesspostprocessdate>
          <businessprocesseddate>20091109154304</businessprocesseddate>
          <tpreceiveddate>0</tpreceiveddate>
          <tpprocesseddate>0</tpprocesseddate>
          <customdate>0</customdate>
          <edireference>2</edireference>
          <networkreference>/usr/local/volume1/jcommercec1/spool/in/dsk.cs.in.1256916399113.49262.dat</networkreference>
          <businessreference1>Dummy Organization</businessreference1>
          <businessreference2>Dummy Organization</businessreference2>
          <businessreference3>HTTP</businessreference3>
          <tpreference1></tpreference1>
          <tpreference2>;;;;;;NOBILL</tpreference2>
          <tpreference3>SAXMAPPING</tpreference3>
          <customreference1></customreference1>
          <customreference2></customreference2>
          <networksessionid>conn.3077170350403106806.5103396823618351589.txt</networksessionid>
          <traceid></traceid>
          <traceid2>dom.-4490622564681732025.-124446042427746222.txt</traceid2>
          <traceid3>conn.3077170350403106806.5103396823618351589.txt</traceid3>
          <traceid4>375927</traceid4>
          <customkeyvalue1></customkeyvalue1>
          <customkeyvalue2>1710</customkeyvalue2>
          <customkeyvalue3>xml=538</customkeyvalue3>
          <customkeyvalue4>161</customkeyvalue4>
          <misc1>xxxyyyzzz</misc1>
          <misc2></misc2>
          <misc3></misc3>
          <misc4></misc4>
          <misc5></misc5>
          <timestamp>1257777784040</timestamp>
          <timestampstring></timestampstring>
          <name></name>
          <persistid>5</persistid>
          <checksumalgorithm>STORED</checksumalgorithm>
        <checksumvalue>tfj6lT3fOnxm8Pqj3wOp8A==</checksumvalue>
        <customstatus2>0</customstatus2>
        <customstatus3>104</customstatus3>
        <customstatus4>0</customstatus4>
        <customstatus5>2</customstatus5>
        <misc6>outbound;5;sax.2.pdf</misc6>
        <misc7>sax.2.pdf</misc7>
        <misc8></misc8>
        <actor></actor>
        <occasion></occasion>
      </be.netbase.jc.core.AuditRecord>
    </data_vector>
  </be.netbase.databinding.Wrapper>
</queryresult>
```

Sample result (NOK)

```
<cs_error>
  <code>60</code>
  <msg>ERROR:cannot downloadjcquery, no mgmntsession,no user setting,no download setting for alias or parameter missing:null</msg>
  <exception/>
  <location1>ActionJCWebQuery.writeResponse</location1>
  <location2/>
  <location3/>
  <stacktrace/>
  <param1>azerty</param1>
</cs_error>
```

Sample result (OK, but no data available)

```
<webquery>
  <msg>No Data Available</msg>
  <param1/>
</webquery>
```

4.5 Getting an audit file (HTTP GET)

Sample https post target: https://www.ct.baswareone.com/httpserver?jcalias=bc.download&auditno=123

URL parameters

Parameter key	Parameter value	Mandatory
jcalias	bc.download	Yes
auditno	<audit number>	Yes

Remarks

The audit number specified doesn't have to be in the list that was returned by querying BaswareONE Document Broker specifying jcalias. This parameter here is only used to complete the user profile. When the user requests a file that is in the list linked with jcalias, then this file will be supplied as long as the user has the right to download this file.

Sample of TCP dump

```
GET /httpserver HTTP/1.1
Authorization: Basic Y29vbGRyaW5rczI6Y29vbGRyaW5rczI=
jcalias: bc.download
auditno: 123
Host: test.certipost.be
Accept: text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2
Connection: keep-alive
```

Sample result (OK)

The file will be returned as-is.

Sample result (NOK)

Analogous to all error results.

4.6 Status update of audit records (HTTP GET)

Sample https post target: https://www.ct.baswareone.com/httpserver?jcalias=bc.update&auditno=123

URL parameters

Parameter key	Parameter value	Mandatory
jcalias	bc.update	Yes
auditno	<audit number to update >	Yes

Remarks

Other audit records than those linked with the jcalias can be updated as long as the user has enough rights. The parameter jcalias is only used to complete the user profile.

Sample of TCP dump

```
GET /httpserver HTTP/1.1
Authorization: Basic Y29vbGRyaW5rczI6Y29vbGRyaW5rczI=
jcalias: bc.update auditno: 123
Host: test.certipost.be
Accept: text/html, image/gif, image/jpeg, *; q=.2, */*; q=.2
Connection: keep-alive
```

Sample result (OK)

```
<updateresult>
  <be.netbase.databinding.Wrapper>
    <data_vector>
      <be.netbase.jc.core.AuditRecord>
        <enabled>true</enabled>
        <contenttype>28005</contenttype>
        <auditno>123</auditno>
        <intchgindividual>XML_INVOVERVIEW_CERTIPOST_DUMMY</intchgindividual>
        <receiver>THULE</receiver>
        <sender>CERTIPOST</sender>
        <intchggroup>INVOVERVIEW</intchggroup>
        <intchgtype>XML_INVOVERVIEW_CERTIPOST</intchgtype>
        <creationdate>20120302102702</creationdate>
        <network>DISK</network>
        <status>405</status>
        <customstatus>0</customstatus>
        <modificationdate>20120305101400</modificationdate>
        <sentreceiveddate>20120302102834</sentreceiveddate>
        <constransdate>20120302102831</constransdate>
        <importexportdate>20120302102702</importexportdate>
        <preprocesspostprocessdate>0</preprocesspostprocessdate>
        <businessprocesseddate>20120302102834</businessprocesseddate>
        <tpreceiveddate>20120305101400</tpreceiveddate>
        <tpprocesseddate>0</tpprocesseddate>
        <customdate>0</customdate>
        <edireference>57</edireference>
        <networkreference>/usr/local/volume1/jcommercec1/spool/in/dsk.cs.in.1330601450773.28
54014197069808773.dat</networkreference>
        <businessreference1>Dummy Organization</businessreference1>
        <businessreference2>Dummy Organization</businessreference2>
        <businessreference3>HTTP</businessreference3>
        <tpreference1></tpreference1>
        <tpreference2>;;;;;;NOBILL</tpreference2>
        <tpreference3>SAXMAPPING</tpreference3>
        <customreference1></customreference1>
        <customreference2></customreference2>
        <networksessionid>20120302.10h/conn.7719639379889399127.-
3137481226860013536.txt</networksessionid>
        <traceid>
          <prms>
            <prm>
              <nm><![CDATA[filedate]]></nm>
              <val><![CDATA[20120302102702]]></val>
            </prm>
            <prm>
              <nm><![CDATA[myiiname]]></nm>
              <val><![CDATA[XML_INVOVERVIEW_CERTIPOST_DUMMY]]></val>
            </prm>
            <prm>
              <nm><![CDATA[ai]]></nm>
              <val><![CDATA[JCINSERTACTION]]></val>
            </prm>
            <prm>
              <nm><![CDATA[rule_id]]></nm>
              <val><![CDATA[161]]></val>
            </prm>
            <prm>
```

```
              <nm><![CDATA[central-auditing-app-id]]></nm>
              <val><![CDATA[c1.stage]]></val>
            </prm>
            <prm>
              <nm><![CDATA[contenttype]]></nm>
              <val><![CDATA[28005]]></val>
            </prm>
            <prm>
              <nm><![CDATA[rule_id_name]]></nm>
              <val><![CDATA[161:DYNAMIC FILENAME]]></val>
            </prm>
            <prm>
              <nm><![CDATA[filename]]></nm>
              <val><![CDATA[/usr/local/volume1/jcdatasets/retail/tmp/invoice_metafile_30856.
20120302102702.xml]]></val>
            </prm>
            <prm>
              <nm><![CDATA[protocol]]></nm>
              <val><![CDATA[DISK]]></val>
            </prm>
            <prm>
              <nm><![CDATA[deletewhenok]]></nm>
              <val><![CDATA[true]]></val>
            </prm>
            <prm>
              <nm><![CDATA[diskdaemonurl]]></nm>
              <val><![CDATA[local://be.netbase.comms.server.disk.DISKDaemon]]></val>
            </prm>
            <prm>
              <nm><![CDATA[original-auditno]]></nm>
              <val><![CDATA[]]></val>
            </prm>
            <prm>
              <nm><![CDATA[md5checksum]]></nm>
              <val><![CDATA[r2hH6nYSzR96yVoNkyACIg==]]></val>
            </prm>
            <prm>
              <nm><![CDATA[file]]></nm>
              <val><![CDATA[/usr/local/volume1/jcdatasets/retail/tmp/invoice_metafile_30856.
20120302102702.xml]]></val>
            </prm>
            <prm>
              <nm><![CDATA[rule_name]]></nm>
              <val><![CDATA[DYNAMIC FILENAME]]></val>
            </prm>
            <prm>
              <nm><![CDATA[function]]></nm>
              <val><![CDATA[insertdocrouter]]></val>
            </prm>
          </prms>
        </traceid>
        <traceid2>20120302.10h/dom.-2311959573301311889.-
7691110237161227993.txt</traceid2>
        <traceid3>20120302.10h/conn.7719639379889399127.-
3137481226860013536.txt</traceid3>
        <traceid4>825163</traceid4>
        <customkeyvalue1></customkeyvalue1>
        <customkeyvalue2>1710</customkeyvalue2>
        <customkeyvalue3>xml=541</customkeyvalue3>
        <customkeyvalue4>161</customkeyvalue4>
        <misc1></misc1>
        <misc2></misc2>
        <misc3></misc3>
        <misc4></misc4>
        <misc5></misc5>
        <timestamp>1330938840859</timestamp>
        <timestampstring></timestampstring>
        <name></name>
        <persistid>5</persistid>
        <checksumalgorithm>STORED</checksumalgorithm>
        <checksumvalue>r2hH6nYSzR96yVoNkyACIg==</checksumvalue>
        <customstatus2>1</customstatus2>
        <customstatus3>104</customstatus3>
        <customstatus4>0</customstatus4>
        <customstatus5>2</customstatus5>
        <misc6>outbound;5;sax.2.pdf</misc6>
        <misc7>sax.2.pdf</misc7>
        <misc8></misc8>
        <actor></actor>
        <occasion></occasion>
      </be.netbase.jc.core.AuditRecord>
    </data_vector>
  </be.netbase.databinding.Wrapper>
</updateresult>
```

Sample result (NOK)

```
<cs_error>
  <code>0</code>
  <msg>Can't execute query: Can't find auditrecord #123 in database</msg>
  <exception></exception>
  <location1></location1>
  <location2></location2>
  <location3></location3>
  <stacktrace><![CDATA[]]></stacktrace>
</cs_error>
```