

Appendix A: Technical References

Appendix A contains a level of technical detail useful to system administrators and developers.

In this Chapter

In this chapter you will find information on:

- CSV import file formats

Note: For information about internal processing when troubleshooting your Casper software, callback routines, etc. please contact CrypKey Technical Support at Support@CrypKey.com.

CSV File Formats

Casper requires import files to be in specific CSV data formats.

Files can be imported from external sources for processing in Casper. The file format used for imported data is the CSV (Comma Separated Values) format. CSV files are best prepared and edited using a tool such as Excel.

New CSV file formats have been defined (see *CSV File Formats for Casper Version 2.0* below). We have left intact the Casper Version 1.0 CSV file formats (see the section *CSV File Formats from Casper Version 1.0* below) for purposes of backward compatibility.

CSV FILE FORMATS FOR CASPER VERSION 2.0

There are two kinds of import files for Casper Version 2.0: Serial Numbers and Product Data. These file types have similar layouts:

- The first line contains column headings.
- The second line contains corresponding column types (char, integer).
- All subsequent lines contain data to be imported.

SERIAL NUMBERS FILE

We have included this file format in order to maintain backward compatibility with the import files that were used with the older import program, CasperDb.exe. This format is valid only for importing serial numbers for products that are already defined in the Casper database.

The serial numbers file must contain at least the columns shown in Table A-1 Serial Numbers – CSV File Format (Minimal Required Content):

Table A-1 Serial Numbers – CSV File Format (Minimal Required Content)

Name	Type	Width	Description
serial_num	char	20	A unique serial number that you have assigned to this customer.
product_code	char	20	Product code already set up in database.

If the product_code is not present, then the product, product_version, and license_type (see Table A-2 Serial Numbers – CSV File Format (Optional Content)) must be present.

Following is an example of the simplest possible Serial Number CSV file:

```
serial_num,product_code
char,char
10002,TESTING
10003,TESTING
```

This file will create 2 serial numbers (10002 and 10003) using the product code TESTING.

See Table A-2 for a list of the remaining columns that can be included in the Serial Numbers file:

Table A-2 Serial Numbers – CSV File Format (Optional Content)

Name	Type	Width	Description
product	char	20	Product name.

Name	Type	Width	Description
product_version	char	20	Product version.
cklevel	integer	n/a	License level.
ckrestriction_type	integer	n/a	License restriction type. (0=none, 1=days, 2=runs)
ckrestriction_value	integer	n/a	License restriction value (i.e., the number of days or runs)
cklicense_type	integer	n/a	License type. (0=fixed, 1=floating)
ckcopies	integer	n/a	Number of copies in the license.
company_number	integer	n/a	Company number (for internal use only).
time_bomb	date	n/a	Date on which the license quits.
add_on_license	integer	n/a	Flags that this is an add-on license. (0=false, 1=true)
ckoption1 through ckoption32	char	3	Flags that this option is on. ("off" or "on")
customer_num	char	6	Your customer number.
City	char	40	Customer's city.
State	char	15	Customer's state or province.
Phone	char	20	Customer's phone number.
Fax	char	20	Customer's fax number.
invoice_num	char	10	Your invoice number.
Email	char	40	Customer's email address.
country	char	40	The customer's country.
Zip	char	15	Customer's zip code.
company_name	char	30	Customer's company name.

Name	Type	Width	Description
address	char	50	Customer's address.
key_limit	integer	n/a	The number of times this serial number can be requested.
user1 through user10	char	50	Custom user information.
<u>customer_name</u>	<u>char</u>	<u>30</u>	Customer's name for specified serial number

Following is an example of a Serial Number file containing both required and optional data:

```

serial_num,product_code,invoice_num,
customer_name,company_name,city,state,country,zip,
phone,fax,cklevel,ckoption6,ckrestriction_type,
ckrestriction_value,cklicense_type,ckcopies,num_of_users
,key_limit,email
char,char,char,char,char,char,char,char,char,
char,char,integer,integer,integer,integer,integer,
integer
integer,integer,char
656,EXAMPLE_CODE,TD89,John,Crypkey,Calgary,Alberta,Canada,
T2X 2X9,(403)555-6666,(403)555-600,8,on,2,34,0,3,10,4,
test.data@data.com

```

PRODUCT DATA FILE

This file format allows import of any product data into the database. Files are best edited using a tool such as Excel.

The product data file must contain at least one line with the import_type column, which has the char format, and specifies the type of data contained in the line.

Table A-3 Product Data CSV File Format – Definition Line

import_type	Description
family	family of products

import_type	Description
family_prod	product
family_prodver	version of a product
family_link	linkage for a product and an executable, kind of like a default product code
executable	executable
executable_ver	version of an executable
executable_opt	option for an executable
executable_lev	level for an executable
product_code	product code describes a product, executable, options, level, and price that are linked together for Casper Server to generate licenses
product_codeopt	alternate way to specify the options related to a product code

While the `import_type` is the only required column in every line, other columns will be required in various lines, depending on the contents of the `import_type`. These requirements are described in Table A-3 Product Data CSV File Format – Definition Line above.

Table A-4 Product Data CSV File Format – Content Lines

import_type	Name	Req'd	Type	Width	Description
family	family	Y	char	20	The family name to create.

import_type	Name	Req'd	Type	Width	Description
	family_dosname	Y	char	8	The 8 character name for the family.
	description	N	char	250	Description.
family_prod	family	Y	char	20	The name of the family this product will be in.
	product	Y	char	20	The product name to create.
	description	N	char	250	Description.
family_prodver	family	Y	char	20	The name of the family this product version will be for.
	product	Y	char	20	The name of the product this product version will be for.
	prod_ver	Y	char	10	The product version to create.
	prod_ver_num	Y	integer	n/a	The product number for this product version.
	description	N	char	250	Description.
family_link	family	Y	char	20	The name of the family for the link.
	product	Y	char	20	The name of the product for the link.

import_type	Name	Req'd	Type	Width	Description
	prod_ver	Y	char	10	The name of the product version for the link.
	exec	Y	char	20	The name of the executable for the link.
	exec_ver	Y	char	20	The name of the executable version for the link.
executable	exec	Y	char	20	The executable name to create.
	exec_filename	Y	char	20	The file name for the executable.
	exec_password	Y	char	20	The CrypKey password for the executable.
	description	N	char	250	Description.
executable_ver	exec	Y	char	20	The name of the executable this version is for.
	exec_ver	Y	char	20	The name of the executable version to create.
	exec_ver_rttnum	Y	integer	n/a	The RTT number of the executable version.
	description	N	char	250	Description.
executable_opt	exec	Y	char	20	The name of

import_type	Name	Req'd	Type	Width	Description
					the executable this option is for.
	exec_ver	Y	char	20	The name of the executable version this option is for.
	exec_opt	Y	char	20	The executable option to create.
	exec_opt_bit	Y	integer	n/a	The option bit of this executable option.
	description	N	char	250	Description.
executable_lev	exec	Y	char	20	The name of the executable this level is for.
	exec_ver	Y	char	20	The name of the executable version this level is for.
	exec_lev	Y	char	20	The executable level to create.
	exec_lev_num	Y	integer	n/a	The number of this executable level.
	description	N	char	250	Description.
product_code	family	Y	char	20	The name of the family for the product code.
	product	Y	char	20	The name of the product for the product

import_type	Name	Req'd	Type	Width	Description
					code.
	prod_ver	Y	char	10	The name of the product version for the product code.
	exec	Y	char	20	The name of the executable for the product code.
	exec_ver	Y	char	20	The name of the executable version for the product code.
	prod_code	Y	char	20	The product code to create.
	prod_code_price	Y	char	20	The price of the product code, no "\$" and no commas.
	prod_code_rttnum	N	integer	n/a	The RTT number of the product code.
	prod_code_isaddon	N	integer	n/a	Flags that the product code is an add on license.
	prod_code_resttype	N	integer	n/a	The restriction type of the product code. (0=none, 1=days, 2=runs)
	prod_code_restval	N	integer	n/a	The restriction value of the product code (i.e., number of days or runs)

import_type	Name	Req'd	Type	Width	Description
	prod_code_isfixed	N	integer	n/a	Flags that the product code is fixed. (1=fixed, 0=floating)
	prod_code_copies	N	integer	n/a	The number of copies in the license of the product code.
	prod_code_optword	N	integer	n/a	The options of the product code expressed as a bit field (MSB = option 1).
	exec_lev	N	char	20	The name of the executable level of the product code.
	prod_code_conum	N	integer	n/a	The company number of the product code, for internal use only.
product_codeopt	prod_code	Y	char	20	The name of the product code to add the option to.
	exec_opt	Y	char	20	The name of the executable option to add to the product code.

In the above table, the “Req’d” column indicates whether the column is required for a given type of entry; Y=Yes, N=No (i.e., the column can be left blank).

Following is an example of a simple product data file:

```
import_type,exec,exec_filename,exec_password,description
```

```
char,char,char,char,char
```

```
executable,test_exec,test.exe,secret,test me executable
```

The above file will create an executable named “test_exec” with the file name “test.exe”, the CrypKey password “secret”, and the description “test me executable”.

You can combine all of the import types in a single file. Each line in the file will have a different import_type value. In the first two lines of the file, place all of the column headings and types that you need for all of the rows in the file. For each row of data in the file, enter data only in the columns that are needed for that particular import type. In any given line, data in columns not relevant to that import type will be ignored.

Following is an example of a complex product data file:

```
import_type,family,family_dosname,product,description,prod_
ver,prod_ver_num,exec,exec_filename,exec_password,exec_ver,
exec_ver_rttnum,exec_lev,exec_lev_num,exec_opt,exec_opt_bit
,prod_code,prod_code_price
```

```
char,char,char,char,char,char,integer,char,char,char,char,i
nTEGER,char,integer,char,integer,char,char
```

```
family,test_family,duhs,, "a testing kind of
""""family"""" ,,,,,,,,,,
```

```
family_prod,test_family,,test_product,a testing kind of
product,,,,,,,,,
```

```
family_prodver,test_family,,test_product,a testing kind of
version,test_ver,1,,,,,,,,,
```

```
executable,,,test me
executable,,,test_exec,test.exe,secret,,,,,,,,,
```

```
executable_ver,,,test me executable
version,,,test_exec,,,test_ever,3,,,,,
```

```
executable_lev,,,test me executable level
1,,,test_exec,,,test_ever,,test_lev1,1,,,,,
```

```
executable_lev,,,test me executable level
2,,,test_exec,,,test_ever,,test_lev2,2,,,,,
```

```

executable_opt,,,,test me executable option
1,,,test_exec,,,,test_ever,,,,test_opt1,1,,

executable_opt,,,,test me executable option
2,,,test_exec,,,,test_ever,,,,test_opt2,2,,

product_code,test_family,,test_product,,test_ver,,test_exec
,,,test_ever,,,,,test_code,15.99

```

As you can see from the above file, you can include in a single import file the complete information for a family, product, product version, executable, executable version, executable level, executable options and product code.

CSV FILE FORMATS FROM CASPER VERSION 1.0

The Casper Version 1.0 data import function converts a CSV (Comma Separated Values) data file into customer order information records. The CSV file format includes parameters for specifying a level and options and other licensing restrictions for use by the Site Key Generator. Such functionality enables you, the vendor, to easily convert data from your own sales or marketing databases into information ready to be processed as Casper customer order information records and associated licenses.

FILE STRUCTURE

Each Casper customer order CSV data file must contain the following information:

- a header line, specifying the sequence of data items contained in the file
- a data type line, specifying the data types (character, integer, etc.) of the data items in the header line
- customer data lines, each entered in the same sequence as the items shown in the header line.

Note: The header line items can be in any sequence and can include any combination of available data items. This can range from the three required items (serial number, product, product version) to all possible customer order and license restriction items. For a breakdown of a CSV file containing the complete set of header line data items, see Table A-5 .

Two types of database update can occur when you use the import function:

1. If the serial number contained in a customer data line does not exist in the Casper database, the data values in that line are inserted into Casper as a new customer order information record; and

2. If the serial number contained in a customer data line exists in the Casper database, the data values in that line overwrite the corresponding data values in the Casper customer order information record with the same serial number.

SAMPLE CSV FILES

The following examples illustrate how the CSV file structure operates.

Example 1

This CSV file contains six data values for three customers. Note the single header line, the single data type line, and the three customer data lines. Each line must end in a comma and carriage return.

```
serial_num,customer_name,product,product_version,cklevel,  
char,char,char,char,integer,  
123,jones,ABC,1.2,3,  
456,smith,DEF,2.0,4,  
789,brown,GHI,1.1,2,
```

In this example, the first four values of each customer data line appear in Casper's customer order information records. The fifth value (i.e., the cklevel integer 3, in the line beginning with 123) is used by the Site Key Generator the next time a license is issued for the applicable customer and product.

Example 2

This example contains six customer order information record data values, but no license-restriction values, for two customers:

```
product,serial_num,invoice_num,phone,fax,product_version,
char,char,char,char,char,char,
IC61CT970,90201,99E10,(403)-555-6666,(403)-555-6600,1.1,
IC61CT825,80264,99E11,(403)-555-6677,(403)-555-6611,1.2,
```

CSV FILE CONTAINING A COMPLETE SET OF DATA ITEMS

Table A-5 presents the complete set of data items that appear in the Casper interface and can be included in a CSV file. Each cell in the CSV Parameters column shows the contents of a single line in the CSV file. Although each set of parameters represents a continuous stream of comma-separated values on a single line, ending with a comma and carriage return, the table's dimensions force the parameters to wrap around several lines.

In the table, Customer Data Lines 1 and 2 contain sample values.

Table A-5 Customer Order CSV File Data Structure (Casper Version 1.0)

Line	CSV Parameters Corresponding to Casper Interface Fields
Header	serial_num,invoice_num,product,product_version, key_limit,customer_name,company_name,address,city, state, zip,cyountry,phone,fax,email,user1,user2,user8,cklevel,ckoption1, ckoption2,ckoption4,ckoption7,ckrestriction_type,ckrestriction_valu e, cklicense_type,ckcopies,
Data Type	char,char,char,char,integer,char,char,char,char,char, char,char,char,char,char,char,char,char,integer,char,char,char, char,integer,integer,char,integer,
Customer Data Line 1	abc,123,APP_1,1.2,5,jones,Ace Mfg. Inc.,167 Jones Ave.,New York, N.Y.,10019,U.S.A.,(212)432-1234,(212)432-1235, bjones@ace_sw.com,,,,,4,ON,ON,OFF,ON,1,60,0,10,
Customer Data Line 2	def,456,APP_2,1.1,4,smith,Zenith Distributors,75 Steeles Ave.,Toronto,Ont.,M4J 4T6,Canada,416-244-5678,416-244-5679, jsmith@zendist.com,,,,,3,OFF,OFF,ON,OFF,2,30,1,2,
Customer Data Line 3	. . . , . . .
Customer Data Line <i>n</i>	. . . ,

Notes:

- Header line: contains the actual system names for the data items. The names must be entered into the CSV file exactly as shown in the table, with commas (but no spaces) separating the items, and a comma and carriage return after the last item.

For a successful import, the CSV file must contain values for the following data items: serial_num, product, and product_version.

You can specify up to 10 User Field values. For illustration purposes, our example CSV table above shows only the names user1, user2, and user8. If you are importing User Field information, you must input these names (“user1”, “user2”, etc.) as literals in CSV Header lines, regardless of the names that you assign to the User Fields in the Customer Order Information window.

You can specify up to 32 options. For illustration purposes, our example CSV table shows only options 1, 2, 4, and 7 (ckoption1, ckoption2, etc.). The items and values shown in the Data Type and Customer Data lines parallel these.

- Data Type line: contains the data types — i.e., char (character) or integer — for each data item in the header line. These types must be entered exactly as shown, with commas (but no spaces) separating the items, and a comma and carriage return after the last item
- Customer Data Line 1: contains the first set of sample customer data values for the data items shown in the header line, with commas (but no spaces) separating the items, and a comma and carriage return after the last item. The data types must match, item for item, those in the Data Type line.

In each Customer Data line, the CSV values for product, and product_version must already have been defined in the Database. Otherwise, the data in that line will not be imported, and an error will be noted in the log file.

Note: If you are using Excel, rather than a text editor, you must precede a numeric “char” value with a single quotation mark ('). Otherwise, the value will be processed as an integer or decimal number, not as a string, and an error message will result.

- Customer Data Line 2: contains the second set of sample customer data values for the data items shown in the header line.
- Customer Data Line 3 . . . : additional customer data lines are symbolized here.

Table A-6 License Level, Option, and Restriction CSV File Data Structure (Casper Version 1.0) below describes the CSV data items that are included in Table A-5 Customer Order CSV File Data Structure (Casper Version 1.0) These are used only by the Site Key Generator, and are not inserted into customer order information records.

Table A-6 License Level, Option, and Restriction CSV File Data Structure
(Casper Version 1.0)

Parameter Name	Type	Range of Values
Cklevel	integer	0 to $2^{32} - 1$
ckoption1 . . .	char	ON,OFF
ckoption32		
ckrestriction_type	integer	0, 1, 2 Note: 0 = unlimited 1 = days 2 = runs
ckrestriction_value	integer	0 to $2^{15} - 1$
cklicense_type	integer	0, 1 Note: 0 = fixed 1 = floating
Ckcopies	integer	0 to $2^7 - 1$