

Readme RSA SecurID Software Token Converter 2.4.1



August 4, 2009

Introduction

RSA SecurID Software Token Converter is a command line utility that converts a software token record that has been exported as an SDTID file from XML format to a Compressed Token Format consisting of an 81-digit numeric string. The converted token can then be provisioned to an RSA SecurID application running on a mobile device. The provisioning process varies, depending on the functionality of the device. In some applications, the user may need to import the converted token by using the device's copy/paste function or by manually entering a numeric string. In other applications, the user may only need to click a hyperlink contained in an e-mail message or SMS text message to deliver the token to the device.

You can use the Token Converter with software token applications that support provisioning tokens using Compressed Token Format. Version 2.4.1 of the Token Converter supports the following applications:

- RSA SecurID Software Token for Windows Mobile
- RSA SecurID Token for the Java ME Platform
- RSA SecurID Software Token for iPhone Devices
- Verizon Software Token for RSA SecurID

For more information, see the documentation for your software token application.

The Token Converter supports 128-bit (AES) software tokens. It does not support 64-bit (SID) tokens.

The Token Converter runs on Windows operating systems and on Red Hat Linux. The Linux version is built for use on Red Hat Enterprise Server 4 (RH ES 4), using Linux kernel version 2.6.9-78.

Command Line Syntax

Important: You can convert only one SDTID file at a time from XML format to Compressed Token Format.

The Token Converter command syntax is as follows. The token filename (*filename.sdtid*) is required.

On Windows:

```
C:\path_name\TokenConverter filename.sdtid options
```

On Linux:

```
/path_name/TokenConverter filename.sdtid options
```

Options

The supported command options for this utility are:

- | | |
|----|--|
| -f | Use this option with the Windows Mobile and Java ME applications. This option specifies that the numeric string will be separated by delimiters. The string consists of blocks of five digits separated by hyphens, for example, nnnnn-nnnnn-nnnnn, and so on. Hyphens are included for readability to assist users whose device functionality requires that they enter the numeric string into the device manually. If you do not use the -f option, the Token Converter outputs the numeric string without delimiters. |
|----|--|

Note: Do not use the -f option with the iPhone or Verizon software token applications.

-o filename Outputs a text file (ASCII) containing the numeric string. If you enter a filename that already exists, the Token Converter overwrites the original file.
If you do not use this option, the numeric string is written to the screen.

-iphone Use this option only with the iPhone application. This option specifies that the output of the Token Converter will be a specially formatted hyperlink (URL) containing the converted token. The data is output in the format:

```
com.rsa.securid.iphone://ctf?ctfData=NNNNNNNNNN
```

where N represents a digit in the numeric string.

The hyperlink can be delivered to the user's iPhone device in an e-mail message. The message format must be set to HTML.

Note: When you use the **-iphone** option, the Token Converter ignores the **-f** option. Also, when you use the **-iphone** option, the Token Converter ignores the **-v** option and always formats the converted token file as **-v 2**.

-p password Specifies the password required for a password-protected token record. You must use this option if you set a password on the token record when you issued it in RSA Authentication Manager. The converted token retains the password, and the user is prompted to enter it to complete the token installation.

-v value Use this option only with the Windows Mobile and Java ME applications. This option specifies the format of a converted token file. Valid values are 1 and 2 (default). Choose one of the following:

- Use the **-v 1** option when converting token files for version 2.2 of the Windows Mobile or Java ME applications. The **-v 1** option does not support device binding.
- Use the **-v 2** option when converting token files for version 2.3 of the Windows Mobile or Java ME applications. You must use the **-v 2** option if you bound the token to a specific device in Authentication Manager.

-version Displays the Token Converter version on the screen. This option is used to assist RSA Customer Support in troubleshooting if you encounter problems converting tokens using Compressed Token Format. For example, the token may not be converted properly if you use an earlier version of the Token Converter that does not support your software token application.

The following command displays the version:

```
TokenConverter -version  
2.4.1
```

Using the Token Converter on Windows

To use the Token Converter on Windows:

1. Save the Token Converter kit to a directory on your computer. Extract the Token Converter executable (**TokenConverter.exe**) and the DLL file (**sdti2tsf.dll**) to the same directory.
2. Place the token records (SDTID files) that you want to convert into the same directory where you saved the executable and DLL.
3. Open a command line, and change to the directory where you saved the executable and DLL.
4. Enter a command using syntax similar to the following:

```
C:\path_name\TokenConverter filename.sdtid options
```

Windows Command Examples

The following command uses the `-iphone` option to convert a password-protected token to a URL containing the 81-character numeric string, and outputs the URL to a text file. Use the `-iphone` option when converting a token intended for the iPhone application. After converting the token, send an HTML-formatted e-mail message with the URL containing the token to a user's device.

```
C:\path_name\TokenConverter user2-passwordtoken.sdtid -p tokenpw1 -iphone -o tokenfile.txt
```

The text file created by this command consists of a specially formatted hyperlink containing token data similar to the following:

```
com.rsa.securid.iphone://ctf?ctfData=200002068164720663136011170432774461076477164632456201026172115044046062716712650
```

The following command converts the token to an 81-digit numeric string with no delimiters, and creates a text file containing the numeric string. This command is appropriate if the application requires the user to import the token manually, and the device has a copy/paste function. The user can copy the number from an e-mail and paste it into the application.

```
C:\path_name\TokenConverter user2-50.sdtid -o tokenfile.txt
```

The following command converts the token to a delimited 81-digit numeric string and creates a text file containing the numeric string. Use the `-f` option with the Windows Mobile or Java ME application to delimit the string if the application requires the user to enter the string and the device does not support copy and paste. Instruct users not to enter the hyphens.

```
C:\path_name\TokenConverter user2-50.sdtid -f -o tokenfile.txt
```

The following command converts a password-protected token to an 81-digit numeric string with no delimiters, and creates a text file containing the numeric string.

```
C:\path_name\TokenConverter user2-passwordtoken.sdtid -p tokenpw1 -o tokenfile.txt
```

The following command converts a password-protected token to a delimited 81-digit numeric string, and creates a text file containing the numeric string. The numeric string format is version 1, which does not support device binding and is required with version 2.2 of the Windows Mobile or Java ME applications.

```
C:\path_name\TokenConverter user2-passwordtoken.sdtid -v 1 -p tokenpw1 -f -o tokenfile.txt
```

The following command displays the version of the Token Converter.

```
C:\path_name\TokenConverter -version
```

Using the Token Converter on Red Hat Linux

To use the Token Converter on Linux:

1. Save the Token Converter executable (**TokenConverter**) to a directory.
2. Open a command line, and change to the directory where you saved the executable.
3. Enter a command using syntax similar to the following:

```
/path_name/TokenConverter filename.sdtid options
```

Linux Command Examples

The following command uses the `-iphone` option to convert a password-protected token to a URL containing the 81-character numeric string, and outputs the URL to a text file. Use the `-iphone` option when converting a token intended for the iPhone application. After converting the token, send an HTML-formatted e-mail message with the URL containing the token to a user's device.

```
/path_name/TokenConverter user2-passwordtoken.sdtid -p t0kenpw1 -iphone -o tokenfile
```

The text file created by this command consists of a specially formatted hyperlink containing token data similar to the following:

```
com.rsa.securid.iphone://ctf?ctfData=200002068164720663136011170432774461076477164632456201026172115044046062716712650
```

The following command converts the token to an 81-digit numeric string with no delimiters, and creates a text file containing the numeric string. This command is appropriate if the application requires the user to import the token manually, and the device has a copy/paste function. The user can copy the number from an e-mail and paste it into the application.

```
/path_name/TokenConverter user2_50.sdtid -o tokenfile
```

The following command converts the token to a delimited 81-digit numeric string and creates a text file containing the numeric string. Use the `-f` option with the Windows Mobile or Java ME application to delimit the string if the application requires the user to enter the string and the device does not support copy and paste. Instruct users not to enter the hyphens.

```
/path_name/TokenConverter user2_50.sdtid -f -o tokenfile
```

The following command converts a password-protected token to an 81-digit numeric string with no delimiters, and creates a text file containing the numeric string.

```
/path_name/TokenConverter user2_passwordtoken.sdtid -p t0kenpw1 -o tokenfile
```

The following command converts a password-protected token to a delimited 81-digit numeric string, and creates a text file containing the numeric string. The numeric string format is version 1, which does not support device binding and is required with version 2.2 of the Windows Mobile or Java ME applications.

```
/path_name/TokenConverter user2_passwordtoken.sdtid -v 1 -p t0kenpw1 -f -o tokenfile
```

The following command displays the version of the Token Converter.

```
/path_name/TokenConverter -version
```

Exit Codes

You can use RSA SecurID Software Token Converter 2.4.1 within your own scripts to do additional processing, such as bulk conversion of tokens or, for the iPhone application, inserting the generated URL into an e-mail template. The following table lists the exit codes that the Token Converter returns to support scripting.

Exit Code	Meaning
0	Token converted properly
1	Invalid command line parameter - output file
2	Invalid command line parameter - password
3	Invalid command line parameter - version
4	Invalid command line parameter - insufficient arguments
5	Invalid command line parameter - Invalid arguments
6	Record is not a software token
7	Correct password needed
8	Parse failed
9	General ctf parse failure
10	Error opening output file

Known Issues

Token cannot be converted if SDTID file contains double-byte characters in UserFirstName, UserLastName, or UserLogin fields

Tracking Number: 119824

Problem: The Token Converter cannot convert SDTID files that contain double-byte characters in the token record's **UserFirstName**, **UserLastName**, and **UserLogin** fields. A double-byte character set contains characters encoded in two bytes. This typically occurs with Asian languages. If any of these fields contain double-byte characters, the token is not converted, and the user receives the following message: "Error parsing XML for the following reason: Failed to parse token record." However, SDTID files that contain double-byte characters in the **Nickname** field are properly converted.

Solution: This is a limitation in RSA Authentication Manager 7.1. To work around the problem, edit the user's first name, last name, and User ID in Authentication Manager to remove double-byte characters, and then issue the token as an SDTID file. The new SDTID file should then convert properly.

Getting Support and Service

RSA SecurCare Online	https://knowledge.rsasecurity.com
Customer Support Information	www.rsa.com/support
RSA Secured Partner Solutions Directory	www.rsasecured.com

© 2009 RSA Security Inc. All rights reserved.

Trademarks

RSA and the RSA logo are registered trademarks of RSA Security Inc. in the United States and/or other countries. For the most up-to-date listing of RSA trademarks, go to www.rsa.com/legal/trademarks_list.pdf. EMC is a registered trademark of EMC Corporation. All other goods and/or services mentioned are trademarks of their respective companies.