Readme
RSA SecurID
Software Token Converter 2.3.1

The Security Division of EMC

October 13, 2008

Introduction

RSA SecurID Software Token Converter converts an RSA SecurID software token (SDTID file) to a numeric string so that it can be imported to a mobile device running an RSA SecurID software token application. The process for importing the converted token depends on the application. Some applications provide an option that requires the user to import the converted token by using the device's copy/paste function or by manually entering the numeric string. With other applications, an automated process may be used to deliver the converted token to the user's device.

You can use the Token Converter with RSA SecurID Token 2.2 for Windows Mobile, RSA SecurID Token for the Java ME Platform, versions 2.2 and 2.3, and other RSA SecurID software token applications that support importing tokens through numeric input.

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 software token at a time to a numeric string.

The Token Converter is run from a command line, using the following command syntax, where the token filename (*filename*.sdtid) is required.

On Windows:

```
C:\path_name\TokenConverter231 filename.sdtid -options
```

On Linux:

```
/path_name/TokenConverter231 filename.sdtid -options
```

The supported options are:

-f	Outputs an 81-digit numeric string in blocks of five digits separated by hyphens, for example, nnnnn-nnnnn-nnnnn, and so on. The hyphens are for readability only. If you do not use the -f option, the converter outputs the token as 81 digits with no delimiters.
-o *filename*	Outputs a text file (ASCII) that contains the numeric string. If the SecurID application requires the user to import the token manually, you can e-mail the text file to the user. If the device has a copy/paste function, the user can copy and paste the numeric string into the application. Otherwise, the user must enter the string manually. If you do not use this option, the numeric string is written only to the screen. If the same filename already exists, the converter overwrites the old file.
-v *version*	Specifies the numeric input format version. Use the -v option when generating tokens for the Windows Mobile or Java ME applications. Valid values are 1 and 2 (default). Version 2 supports binding a token to a specific device. Version 1 does not support device binding. Use -v 1 when converting token files for version 2.2 of the Windows Mobile or Java ME applications. Use -v 2 when converting token files for version 2.3 of the Java ME application.

-p *password*	Specifies the password required for a password-protected token record. You must use this option if the token is password protected. The numeric string retains the password, and the user must enter it when importing the token.

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 (**TokenConverter231.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\TokenConverter231 filename.sdtid -options
```

Windows Examples

The following command converts the token to an 81-digit numeric string with no delimiters, and creates a text file containing the numeric string in the same directory as the Token Converter. 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\TokenConverter231 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 in the same directory as the Token Converter. Use the -f option 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\TokenConverter231 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 in the same directory as the Token Converter. The -p *password* option is required if you have set a password on the token file in RSA Authentication Manager.

```
C:\path_name\TokenConverter231 user2-passwordtoken.sdtid -p t0kenpw1 -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 in the same directory as the Token Converter. The numeric string format is version 1, which indicates that the converted tokens are intended for version 2.2 of the Windows Mobile or Java ME applications.

```
C:\path_name\TokenConverter231 user2-passwordtoken.sdtid -v 1 -p t0kenpw1 -f -o
tokenfile.txt
```

Using the Token Converter on Red Hat Linux

To use the Token Converter on Linux:

1. Save the Token Converter executable (**TokenConverter231**) 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/TokenConverter231 filename.sdtid -options
```

Linux Examples

The following command converts the token to an 81-digit numeric string with no delimiters, and creates a text file containing the numeric string in the same directory as the Token Converter. 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/TokenConverter231 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 in the same directory as the Token Converter. Use the -f option 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/TokenConverter231 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 in the same directory as the Token Converter. The -p *password* option is required if you have set a password on the token file in RSA Authentication Manager.

```
/path_name/TokenConverter231 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 in the same directory as the Token Converter. The numeric string format is version 1, which indicates that the converted tokens are intended for version 2.2 of the Windows Mobile or Java ME applications.

```
/path_name/TokenConverter231 user2_passwordtoken.sdtid -v 1 -p t0kenpw1 -f -o
tokenfile
```

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**

Trademarks