# RSA BSAFE®

## Crypto-J Cryptographic Module

## Security Policy (jsafe)

**Version 3.6**
**August 1, 2007**

**Cryptographic components for Java**

## Contact Information

See our Web sites for regional Customer Support telephone and fax numbers.

| RSA Security Inc. | RSA Security Ireland Limited |
|---|---|
| www.rsasecurity.com | www.rsasecurity.ie |

## Trademarks

ACE/Agent, ACE/Server, Because Knowledge is Security, BSAFE, ClearTrust, Confidence Inspired, e-Titlement, IntelliAccess, Keon, RC2, RC4, RC5, RSA, the RSA logo, RSA Secured, the RSA Secured logo, RSA Security, SecurCare, SecurID, SecurWorld, Smart Rules, The Most Trusted Name in e-Security, Transaction Authority , and Virtual Business Units are either registered trademarks or trademarks of RSA Security Inc. in the United States and/or other countries. EMC is a registered trademark of EMC Corporation. All other goods and/or services mentioned are trademarks of their respective companies.

## License Agreement

This software and the associated documentation are proprietary and confidential to RSA Security Inc., are furnished under license and may be used and copied only in accordance with the terms of such license and with the inclusion of the copyright below. This software and any copies thereof may not be provided or otherwise made available to any other person.

Neither this software nor any copies thereof may be provided to or otherwise made available to any third party. No title to or ownership of the software or any intellectual property rights thereto is hereby transferred. Any unauthorized use or reproduction of this software may be subject to civil and/or criminal liability.

This software is subject to change without notice and should not be construed as a commitment by RSA Security Inc.

## Note on Encryption Technologies

This product may contain encryption technology. Many countries prohibit or restrict the use, import or export of encryption technologies and current use, import and export regulations should be followed when exporting this product.

## Distribution

This document may be freely reproduced and distributed whole and intact including this Copyright Notice.

## RSA Security Inc. Notice

The RC5® Block Encryption Algorithm With Data-Dependent Rotations is protected by U.S. Patent #5,724,428 and #5,835,600.

Compaq MultiPrime™ technology is protected by U.S. Patent #5,848,159 and is the subject of patent applications in other countries.

This product includes patented technology licensed from Entrust Technologies Inc. (US Patent# 5,699,431).

Published August 1, 2007

# Table of Contents

# 1 Introduction

This is a non-proprietary cryptographic module security policy for the RSA BSAFE® Crypto-J Cryptographic Module version 3.6 (Crypto-J Cryptographic Module). This security policy describes how the Crypto-J Cryptographic Module meets the security requirements of FIPS 140-2, and how to securely operate the Crypto-J Cryptographic Module. This policy is prepared as part of the Level 1 FIPS 140-2 validation of the Crypto-J Cryptographic Module.

The Crypto-J distribution includes two API interfaces:

| | |
|---|---|
| • jsafeFIPS.jar | "JSAFE" application programmer interface to the Crypto-J Cryptographic Module |
| • jsafeJCEFIPS.jar | "JCE" application programmer interface to the Crypto-J Cryptographic Module. |

This security policy deals only with the JSAFE interface to the Crypto-J Cryptographic Module. For details of how the FIPS 140-2 evaluation applies to the JCE module, see *RSA BSAFE Crypto-J Security Policy (jsafeJCE)*.

FIPS 140-2 (Federal Information Processing Standards Publication 140-2 – *Security Requirements for Cryptographic Modules*) details the U.S. Government requirements for cryptographic modules. More information about the FIPS 140-2 standard and validation program is available on the NIST website http://csrc.nist.gov/cryptval/.

## 1.1 References

This document deals only with operations and capabilities of the Crypto-J Cryptographic Module in the technical terms of a FIPS 140-2 cryptographic module security policy. More information is available on the Crypto-J Cryptographic Module and the entire RSA BSAFE product line:

- The RSA website contains information on the full line of products and services at http://www.rsa.com.
- An overview of the Crypto-J Cryptographic Module is located at http://www.rsa.com/node.asp?id=1204.
- The RSA BSAFE product overview is provided at http://www.rsa.com/node.asp?id=1202.
- For answers to technical or sales related questions, see the contact details in section 6 Contacting RSA on page 19.

## 1.2 Terminology

The Crypto-J Cryptographic Module is also referred to as the Cryptographic Module. There are two application programmer interfaces to the Crypto-J Cryptographic Module. All references to the Crypto-J Cryptographic Module apply to both interfaces unless explicitly noted.

## 1.3 Document Organization

This Security Policy document is one document in the complete FIPS 140-2 Certification Submission package. In addition to this document, the complete submission package contains:

- Executive Summary document
- Vendor Evidence document
- Finite State Machine document
- Module software listing
- Other supporting documentation as additional references.

This document explains the Crypto-J Cryptographic Module features and functionality relevant to FIPS 140-2. This section, Introduction, provides an overview and introduction to the Security Policy. The Crypto-J Cryptographic Module section, on page 6, describes the Cryptographic Module and how it meets the FIPS 140-2 requirements. Secure Operation of Crypto-J, on page 14, addresses the required configuration for the FIPS140-mode of operation. Services, on page 16, lists all of the functions provided by the Cryptographic Module. Acronyms, on page 17, lists the definitions for the acronyms used in this document.

With the exception of this non-proprietary security policy, the FIPS 140-2 Certification Submission Documentation is RSA-proprietary, and releasable only under appropriate non-disclosure agreements. For access to these documents, please contact RSA.

# 2 Crypto-J Cryptographic Module

This section provides an overview of the Crypto-J Cryptographic Module, through the following topics:

- Introduction
- Cryptographic Module
- Module Interfaces
- Roles and Services
- Cryptographic Key Management
- Cryptographic Algorithms
- Self-Test.

## 2.1 Introduction

More than a billion copies of the RSA BSAFE technology are embedded in today's most popular software applications and hardware devices. Encompassing the most widely-used and rich set of cryptographic algorithms as well as secure communications protocols, RSA BSAFE software is a set of complementary security products relied on by developers and manufacturers worldwide.

The Crypto-J software library is the world's most trusted Java-language cryptography component, and is at the heart of the RSA BSAFE product line. It includes a wide range of data encryption and signing algorithms, including AES, Triple-DES, the high-performing RC5, the RSA Public Key Cryptosystem, the DSA government signature algorithm, and the MD5 and SHA1 message digest routines. Its software libraries, sample code and complete standards-based implementation enable near-universal interoperability for your networked and e-business applications. Any programmer using the RSA BSAFE Crypto-J tools can easily create secure applications without a background in cryptography, mathematics or number theory.

## 2.2 Cryptographic Module

This Cryptographic Module is classified as a multi-chip standalone module for FIPS 140-2 purposes. As such, the module is tested on a particular operating system and computer platform. The cryptographic boundary includes the Cryptographic Module running on selected platforms that are running selected operating systems, while configured in "single user" mode.

The Crypto-J Cryptographic Module is validated for all FIPS 140-2 Level 1 security requirements. The Cryptographic Module is packaged in a Java Archive (JAR) file containing all the code for the module. In addition, the Cryptographic Module relies on the physical security provided by the host in which it runs.

The JSAFE application programmer interface to the Cryptographic Module is provided in the `jsafeFIPS.jar` file.

The Crypto-J Cryptographic Module was tested on the following platforms:

- Microsoft® Windows® XP SP2 (32-bit) and Sun™ JDK™ 1.5
- Microsoft® Windows® XP Professional SP2 (64-bit) and Sun™ JDK™ 1.5
- Solaris 10, UltraSparc v8+ (32-bit), Sun JDK 1.5
- Solaris 10, UltraSparc v9 (64-bit), Sun JDK 1.5 (64-bit)
- HP-UX 11.23, Itanium2 (32-bit), HP JDK 5.0
- HP-UX 11.23, Itanium2 (64-bit), HP JDK 5.0 (64-bit)
- Red Hat Enterprise Linux AS 4.0, x86 (32-bit), Sun JDK 1.5
- Red Hat Enterprise Linux AS 4.0, x86-64 (64-bit), Sun JDK 1.5 (64-bit)
- Novell® SUSE® Linux Enterprise Server 9, x86 (32-bit), Sun JDK 1.5
- Novell SUSE Linux Enterprise Server 9, x86-64 (64-bit), Sun JDK 1.5 (64-bit)
- AIX 5L™ v5.3, Power PC® (32-bit), IBM JDK 1.5
- AIX 5L v5.3, Power PC (64-bit), IBM JDK 1.5 (64-bit)

Compliance is maintained on platforms for which the binary executable remains unchanged. This includes (but is not limited to):

- Microsoft
  - Windows 2000, Service Pack 4 Sun JDK 1.1.8/1.3.1/1.4.2/1.5, IBM JDK 1.4.2
  - Windows XP (SP1 and SP2), Sun JDK 1.1.8/1.3.1/1.4.2/1.5, IBM JDK 1.4.2
  - Windows 2003 Server, Sun JDK 1.1.8/1.3.1/1.4.2/1.5, IBM JDK 1.4.2
- Sun
  - Solaris™ 8, UltraSparc® v9 (32-bit), Sun JDK 1.3.1/1.4.2/1.5
  - Solaris 8, UltraSparc v9 (64-bit), Sun JDK 1.5 (64-bit)
  - Solaris 9, UltraSparc v9 (32-bit), Sun JDK 1.3.1/1.4.2/1.5
  - Solaris 9, UltraSparc v9 (64-bit), Sun JDK 1.5 (64-bit)
  - Solaris 10, UltraSparc v9 (32-bit), Sun JDK 1.3.1/1.4.2/1.5
- Linux®
  - Red Hat® Linux 7.2, x86 (32-bit), Sun JDK 1.3.1/1.4.2/1.5
  - Red Hat Enterprise Linux AS 3.0, x86 (32-bit), Sun JDK 1.3.1/1.4.2/1.5
  - Red Hat Enterprise Linux AS 4.0, x86 (32-bit), Sun JDK 1.3.1/1.4.2
  - Novell® SUSE® Linux Enterprise Server 9, x86 (32-bit), Sun JDK 1.4.2
- HP
  - HP-UX 11.11, PA-RISC 2.0 (32-bit), HP JDK 1.4.2/5.0
  - HP-UX 11.23, Itanium2 (32-bit), HP JDK 1.4.2
- IBM®
  - AIX 5L™ v5.3, Power PC® (32-bit), IBM JDK 1.5

For a resolution on the issue of "Multi user" modes, see the NIST document *Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program* at the government website http://csrc.nist.gov/cryptval/140-1/FIPS1402IG.pdf.

## 2.3 Module Interfaces

As a multi-chip standalone module, the Crypto-J Cryptographic Module's physical interfaces consist of a keyboard, mouse, monitor, serial ports, network adapters, and so on.

The underlying logical interface to the module is the Application Program Interface (API), documented in the *RSA BSAFE Crypto-J Developer's Guide*. The module provides for Control Input through the API calls. Data Input and Output are provided in the variables passed with API calls, and Status Output is provided in the returns and error codes documented for each call.

## 2.4 Roles and Services

The Crypto-J Cryptographic Module meets all FIPS140-2 Level 1 requirements for Roles and Services, implementing both a Crypto Officer role and a Crypto User role. As allowed by FIPS 140-2. the module does not require user identification or authentication for these roles. Only one role can be active at a time, and the module does not allow concurrent operators.

The API for control of the module is through the `CryptoJ` class. The JAR file provides the path to the `CryptoJ` class, as shown in the following table.

**Table 1. Path to the CryptoJ Class**

| Interface | JAR File | Prefix |
|---|---|---|
| JSAFE | jsafeFIPS.jar | com.rsa.jsafe |
| JCE | jsafeJCEFIPS.jar | com.rsa.jsafe.crypto |

### 2.4.1 Crypto Officer Role

An operator can assume the Crypto Officer role by invoking the method `<PREFIX>.CryptoJ.setRole()` with the argument `CRYPTO_OFFICER_ROLE`.

Once in the Crypto Officer role, the operator can start the power-up self-tests on demand by calling the method `<PREFIX>.CryptoJ.runSelfTests()`.

The Crypto Officer can perform this operation manually at the command prompt by navigating to the directory containing the appropriate jar file, and typing:

```
java -cp <JARFILE> <PREFIX>.CryptoJ -testAll
```

Alternatively, the Crypto Officer can call the operation programmatically:

```
<PREFIX>.CryptoJ.runSelfTests();
```

When the Crypto-J Cryptographic Module is loaded, the power-up self-tests run automatically. After passing the integrity check, the self-tests are not run again unless the module is unloaded and reloaded. So, calling the self-tests on demand only results in the power-up known-answer tests (KATs) and pairwise consistency checks being performed.

### 2.4.2 Crypto User Role

The Crypto User role is the default operating role. However, an operator can explicitly assume the Crypto User role by invoking the method `<PREFIX>.CryptoJ.setRole()` with the argument `USER_ROLE`.

The Crypto-J Cryptographic Module API, its functions, and capabilities are documented in the *RSA BSAFE Crypto-J Developer's Guide*. A full list of services is also provided in section 4 Services on page 16.

# 2.5 Cryptographic Key Management

## 2.5.1 Key Generation

The Crypto-J Cryptographic Module supports generation of the DSA, RSA, and Diffie-Hellman (DH) public and private keys. The module also employs a FIPS 186-2 compliant random number generator for generating asymmetric and symmetric keys used in algorithms such as AES, TDES, RSA, DSA or Diffie-Hellman.

## 2.5.2 Key Storage

The Crypto-J Cryptographic Module does not provide long-term cryptographic key storage. Storage of keys is the responsibility of the user of the Cryptographic Module.

Volatile (that is, short-term) memory storage of cryptographic keys and CSPs employed by the cryptographic module is handled, as shown in the following table. The Crypto User and Crypto Officer roles have equal and complete access to all keys and CSPs.

**Table 2. Key and CSP storage**

| Item | Storage |
|---|---|
| AES keys | In volatile memory only (plaintext) |
| Triple DES keys | In volatile memory only (plaintext) |
| HMAC with SHA1 and SHA2 keys | In volatile memory only (plaintext) |
| Diffie-Hellman public key | In volatile memory only (plaintext) |
| Diffie-Hellman private key | In volatile memory only (plaintext) |
| RSA public key | In volatile memory only (plaintext) |
| RSA private key | In volatile memory only (plaintext) |
| DSA public key | In volatile memory only (plaintext) |
| DSA private key | In volatile memory only (plaintext) |
| PRNG seeds (FIPS 186-2) | In volatile memory only (plaintext) |

## 2.5.3 Key Protection

All key data resides in internally allocated data structures and can only be output using the module's defined API. The operating system and Java Runtime Environment (JRE) protects memory and process space from unauthorized access.

## 2.5.4 Key Zeroization

All key data resides in internally allocated data structures that are "cleaned up" by the Java Virtual Machine's (JVM) garbage collector. Java often handles memory in ways that are unpredictable and transparent to the user, and a user can ensure sensitive data is properly zeroized by making use of the `clearSensitiveData()` method for clearing sensitive data. For more information about clearing sensitive data, see Clearing Sensitive Data in the *RSA BSAFE Crypto-J Developer's Guide*.

## 2.6 Cryptographic Algorithms

The Crypto-J Cryptographic Module supports a wide variety of cryptographic algorithms. The FIPS140-2 requirements specify that when the module is in a FIPS140-mode of operation, only FIPS140-approved algorithms be used.

The Cryptographic Module implements algorithm enforcement, such that when the module is in FIPS140_MODE, only FIPS140- approved algorithms are available for use.

The following table lists the FIPS140-approved algorithms provided by the Cryptographic Module, when operating in FIPS140_MODE.

**Table 3. Crypto-J FIPS140-approved algorithms**

| Algorithm | Certificate Number |
|---|---|
| AES – ECB, CBC, CFB (128), OFB (128) – [128, 192, 256 bit key sizes] | 487 |
| AES – CTR | 487 |
| Diffie-Hellman Key Agreement | Non-Approved (allowed in FIPS140_MODE) |
| Digital Signature Algorithm (DSA) | 197 |
| FIPS 186-2 General Purpose [(x-Change Notice]; (SHA-1)] | 264 |
| FIPS 186-2 [(x-Change Notice); (SHA-1)] | 264 |
| HMAC-SHAx (where x is 1, 224, 256, 384, or 512) | 240 |
| RSASSA-PSS (sign, verify) (SHA-1) | 199 |
| RSASSA-PSS (sign, verify) (SHA-224, SHA-256, SHA-384, SHA-512) | 199 |
| RSA PKCS#1 v1.5 (sign, verify) (SHA-1,SHA-224,SHA-256,SHA-384,SHA-512) | 199 |
| Secure Hash Standard (SHA-1, SHA-224, SHA-256, SHA-384, SHA-512) | 553 |
| Triple DES - ECB, CBC, CFB (64 bit), and OFB (64 bit) | 497 |
| RSA X9.31 (keygen, sign, verify) | 199 |

The following table lists the non-FIPS140-approved algorithms provided by the Cryptographic Module.

**Table 4. Crypto-J non-FIPS140-approved algorithms**

| Algorithm |
|---|
| DESX |
| DES |
| MD2 |
| MD5. See Note below. |
| Random Number Generators (ANSI X9.31, MD5Random, SHA1Random) |
| The RC2® block cipher |
| The RC4® stream ciphe |
| The RC5® block cipher |
| PBEWithSHA1And3DES |
| RSA OAEP for key transport |
| Raw RSA encryption and decryption |
| RSA Keypair Generation MultiPrime (2 or 3 primes) |
| RIPEMD160 |
| HMAC-MD5 |

On systems running a Java Runtime Environment (JRE) version 1.3.1 or earlier, the MD5 algorithm is enabled for use in FIPS140_MODE because the JCE framework requires that MD5 be available for JCE validation. It is the responsibility of the user of the Cryptographic Module to ensure that this non-FIPS140 algorithm is not used in other contexts.

For more information on using Crypto-J in a FIPS140-compliant manner, and the modes available in Crypto-J, see section 3 Secure Operation of Crypto-J on page 14.

# 2.7 Self-Tests

The Crypto-J Cryptographic Module performs a number of power-up and conditional self-tests to ensure proper operation. If any of these tests fails, the module throws a `SecurityException`, which provides a status output, and aborts the operation that caused the conditional self-tests to fail.

## 2.7.1 Power-Up Self-Tests

The power-up self-tests implemented are as follows:

- PRNG KATs
- AES KATs
- DES KATs
- TDES KATs
- SHA-1 KATs
- SHA-224 KATs
- SHA-256 KATs
- SHA-384 KATs
- SHA-512 KATs
- HMAC SHA-1 KATs
- HMAC SHA-224 KATs
- HMAC SHA-256 KATs
- HMAC SHA-384 KATs
- HMAC SHA-512 KATs
- Pairwise consistency checks for DSA and RSA
- Software/firmware integrity check.

Power-up self-tests are executed automatically when the module is loaded by the JRE.

## 2.7.2 Conditional Self-Tests

The Crypto-J Cryptographic Module performs two conditional self-tests: a pair-wise consistency tests each time the module generates a DSA or RSA public/private key pair, and a continuous random number generator test each time the module produces random data per the FIPS 186-2 standard.

## 2.7.3 Mitigation of Other Attacks

RSA key operations implement blinding by default, providing a defense against timing attacks. Blinding is implemented through blinding modes, for which the following options are available:

- Blinding mode off.
- Blinding mode with no update, where the blinding value is squared for each operation.
- Blinding mode with full update, where a new blinding value is used for each operation.

# 3 Secure Operation of Crypto-J

The Crypto-J Cryptographic Module does not require any special configuration to operate in conformance with FIPS 140-2 requirements. The following guidance must be followed, however, to achieve a FIPS140-mode of operation.

## 3.1 Crypto Officer Guidance

The Crypto Officer is responsible for installing the module. Installation instructions are provided in the *RSA BSAFE Crypto-J Installation Guide*.

The module's default state is `FIPS140_MODE`.

## 3.2 Crypto User Guidance

The Crypto User must only use algorithms approved for use in a FIPS140-mode of operation, as listed in Table 3 on page 11. The FIPS140-approved bit-length for a DSA key pair must be 1024 bits in length, and the FIPS140-approved RNGs must be seeded with values of at least 160 bits in length. The FIPS140-approved bit lengths for an RSA[1] key pair must be between 1024 and 4096 bits in multiples of 512. The FIPS140-approved bit lengths for the Diffie-Hellman[2] key agreement must be between 1024 and 2048 bits. The FIPS140-approved bit lengths for an HMAC key must be between 80 and 4096 bits.

If RSA key generation is requested in FIPS140 mode, the module always uses the FIPS140-approved RSA X9.31 key-generation procedure.

Crypto Users should take care to zeroize CSPs when they are no longer needed. For more information on clearing sensitive data, see Clearing Sensitive Data in the *RSA BSAFE Crypto-J Developer's Guide*.

The module's default state is `FIPS140_MODE`.

## 3.3 Role Changes

If a user of the Crypto-J Cryptographic Module needs to operate the module in different roles, then the user must ensure that all instantiated cryptographic objects are destroyed before changing from the Crypto User role to the Crypto Officer role, or unexpected results could occur.

[1] When used for transporting keys and using the minimum allowed modulus size, the minimum strength of encryption provided is 80 bits.

[2] Using the minimum allowed modulus size, the minimum strength of encryption provided is 80 bits.

## 3.4 Modes of Operation

There are three modes of operation:

- `FIPS140_MODE`
- `FIPS140_SSL_MODE`
- `NON_FIPS140_MODE`.

The following table lists the values that can be used in the `setMode()` method to change the mode of operation, and the algorithms available in that mode.

**Table 5. Values in setMode to Change the Mode of Operation**

| Value in setMode() | Algorithms Available |
|---|---|
| `CryptoJ.FIPS140_MODE` | Only FIPS140-approved algorithms are available. |
| `CryptoJ.FIPS140_SSL_MODE` | All FIPS140-approved algorithms, plus MD5, are available. |
| `CryptoJ.NON_FIPS140_MODE` | All Crypto-J algorithms are available. |

If a user of the Crypto-J Cryptographic Module needs to operate the module in different modes then the user must ensure that all instantiated cryptographic objects are destroyed before changing modes, or unexpected results could occur.

The cryptographic module does not enforce checking for serialization or deserialization of objects. When operating in FIPS140_MODE is it the responsibility of the user to ensure that handling of serialized objects is performed in a manner such that the module mode is preserved.

# 4 Services

The Crypto-J Cryptographic Module meets all FIPS 140-2 Level 1 requirements for Roles and Services, implementing both a Crypto Officer role and a Crypto User role. As allowed by FIPS 140-2, the module does not require user identification or authentication for these roles. Only one role can be active at a time, and the module does not allow concurrent operators.

The following table lists the services provided by the  Cryptographic Module in terms of the module's interface. For more information on each function, see the *RSA BSAFE Crypto-J Developer's Guide*.

**Table 6. Services for Crypto-J (jsafeFIPS.jar)**

| Service | Service | Service |
|---|---|---|
| CryptoJ.runSelfTests* | JSAFE_KeyAgree | JSAFE_PublicKey |
| CryptoJ.setRole | JSAFE_KeyAttributes | JSAFE_Recode |
| CryptoJ.getRole | JSAFE_KeyPair | JSAFE_SecretKey |
| CryptoJ.setMode | JSAFE_MAC | JSAFE_SecureRandom |
| CryptoJ.getMode | JSAFE_MessageDigest | JSAFE_Session |
| CryptoJ.getState | JSAFE_Obfuscator | JSAFE_SessionSpec |
| CryptoJ.selfTestPassed | JSAFE_Object | JSAFE_Signature |
| JSAFE_AsymmetricCipher | JSAFE_Parameters | JSAFE_SymmetricCipher |
| JSAFE_Key | JSAFE_PrivateKey | JSAFE_VerifyPQG |

*Only available to the Crypto Officer role.

# 5 Acronyms

| Acronym | Definition |
|---|---|
| AES | Advanced Encryption Standard. A fast block cipher with a 128-bit block, and keys of lengths 128, 192 and 256 bits. This will replace DES as the US symmetric encryption standard. |
| API | Application Programming Interface. |
| Attack | Either a successful or unsuccessful attempt at breaking part or all of a cryptosystem. Various attack types include an algebraic attack, birthday attack, brute force attack, chosen ciphertext attack, chosen plaintext attack, differential cryptanalysis, known plaintext attack, linear cryptanalysis, and middleperson attack. |
| CBC | Cipher Block Chaining. A mode of encryption in which each ciphertext depends upon all previous ciphertexts. Changing an IV alters the ciphertext produced by successive encryptions of an identical plaintext. |
| CFB | Cipher Feedback. A mode of encryption that produces a stream of ciphertext bits rather than a succession of blocks. In other respects, it has similar properties to the CBC mode of operation. |
| CSP | Cryptographic Service Provider. |
| DES | Data Encryption Standard. A symmetric encryption algorithm with a 56-bit key. See also Triple DES. |
| Diffie-Hellman | The Diffie-Hellman asymmetric key exchange algorithm. There are many variants, but typically two entities exchange some public information (for example, public keys or random values) and combines them with their own private keys to generate a shared session key. As private keys are not transmitted, eavesdroppers are not privy to all of the information that composes the session key. |
| DSA | Digital Signature Algorithm. An asymmetric algorithm for creating digital signatures. |
| ECB | Electronic Code Book. A mode of encryption in which identical plaintexts are encrypted to identical ciphertexts, given the same key. |
| Encryption | The transformation of plaintext into an apparently less readable form (called ciphertext) through a mathematical process. The ciphertext may be read by anyone who has the key that decrypts (undoes the encryption) the ciphertext. |
| FIPS | Federal Information Processing Standards. |
| HMAC | Keyed-Hashing for Message Authentication Code. |
| KAT | Known Answer Test. |
| Key | A string of bits used in cryptography, allowing people to encrypt and decrypt data. Can be used to perform other mathematical operations as well. Given a cipher, a key determines the mapping of the plaintext to the ciphertext. Various types of keys include: distributed key, private key, public key, secret key, session key, shared key, subkey, symmetric key, and weak key. |
| MD5 | A secure hash algorithm created by Ron Rivest. MD5 hashes an arbitrary-length input into a 16-byte digest. |
| NIST | National Institute of Standards and Technology. A division of the US Department of Commerce (formerly known as the NBS) which produces security and cryptography-related standards. |
| OFB | Output Feedback. A mode of encryption in which the cipher is decoupled from its ciphertext. |
| OS | Operating System. |
| PC | Personal Computer. |
| private key | The secret key in public key cryptography. Primarily used for decryption but also used for encryption with digital signatures. |
| PRNG | Pseudo Random Number Generator. |
| RC2 | Block cipher developed by Ron Rivest as an alternative to the DES. It has a block size of 64 bits and a variable key size. It is a legacy cipher and RC5 should be used in preference. |

| Acronym | Definition |
|---|---|
| RC4 | Symmetric algorithm designed by Ron Rivest using variable length keys (usually 40 bit or 128 bit). |
| RC5 | Block cipher designed by Ron Rivest. It is parameterizable in its word size, key length and number of rounds. Typical use involves a block size of 64 bits, a key size of 128 bits and either 16 or 20 iterations of its round function. |
| RNG | Random Number Generator. |
| RSA | Public key (asymmetric) algorithm providing the ability to encrypt data and create and verify digital signatures. RSA stands for Rivest, Shamir, and Adleman, the developers of the RSA public key cryptosystem. |
| SHA | Secure Hash Algorithm. An algorithm which creates a unique hash value for each possible input. SHA takes an arbitrary input which is hashed into a 160-bit digest. |
| SHA-1 | A revision to SHA to correct a weakness. It produces 160-bit digests. SHA-1 takes an arbitrary input which is hashed into a 20-byte digest. |
| SHA-2 | The NIST-mandated successor to SHA-1, to complement the Advanced Encryption Standard. It is a family of hash algorithms (SHA-256, SHA-384 and SHA-512) which produce digests of 256, 384 and 512 bits respectively. |
| TDES | Triple-DES. |

# 6 Contacting RSA

The RSA Web site contains the latest news, security bulletins and information about coming events.

The RSA BSAFE Web site contains product information.

The RSA Laboratories Web site contains frequently asked questions.

## 6.1 Support and Service

If you have any questions or require additional information, see RSA Support or RSA SecurCare Online.

## 6.2 Feedback

We welcome your feedback on RSA documentation. Please email userdocs@rsa.com.