State of West Virginia
Department of Administration
Purchasing Division
2019 Washington Street East
Post Office Box 50130
Charleston, WV 25305-0130

Solicitation

NUMBER	PAGE
BPH14178	1

ADDRESS CORRESPONDENCE TO ATTENTION OF:

BOB KILPATRICK
304-558-0067

VENDOR

RFQ COPY
TYPE NAME/ADDRESS HERE

SHIP TO

HEALTH AND HUMAN RESOURCES
BPH - THREAT PREPAREDNESS

505 CAPITOL STREET, SUITE 200
CHARLESTON, WV
25301 304-558-1218

DATE PRINTED
04/29/2014

BID OPENING DATE: 05/21/2014 BID OPENING TIME 1:30PM

LINE	QUANTITY	UOP	CAT. NO.	ITEM NUMBER	UNIT PRICE	AMOUNT
	THE WEST VIRGINIA PURCHASING DIVISION FOR THE AGENCY, THE WEST VIRGINIA DEPARTMENT OF HEALTH AND HUMAN RESOURCES, BUREAU FOR PUBLIC HEALTH, CENTER FOR THREAT PREPAREDNESS, IS SOLICITING BIDS TO ESTABLISH A CONTRACT FOR THE ONE-TIME PURCHASE OF TWELVE (12) DIGITAL DUAL BAND MOBILE BASE RADIO SYSTEMS, PER THE ATTACHED DOCUMENTATION					
	ATTACHMENTS INCLUDE: 1. INSTRUCTIONS TO VENDORS SUBMITTING BIDS 2. GENERAL TERMS AND CONDITIONS 3. BPH14178 SPECIFICATIONS (INCLUDING PRICING PAGE) 4. CERTIFICATION AND SIGNATURE PAGE 5. PURCHASING AFFIDAVIT 6. VENDOR PREFERENCE CERTIFICATE					
	** SUCCESSFUL VENDOR MUST DELIVER ALL CONTRACT ITEMS SO THAT AGENCY WILL RECEIVE BY JUNE 30, 2014 **					
0001	12	EA		725-27		
	DIGITAL DUAL BAND MOBILE RADIO BASE SYSTEM PER SPECS					

SIGNATURE

TELEPHONE

DATE

TITLE

FEIN

ADDRESS CHANGES TO BE NOTED ABOVE

WHEN RESPONDING TO SOLICITATION, INSERT NAME AND ADDRESS IN SPACE ABOVE LABELED 'VENDOR'

INSTRUCTIONS TO VENDORS SUBMITTING BIDS

1. **REVIEW DOCUMENTS THOROUGHLY:** The attached documents contain a solicitation for bids. Please read these instructions and all documents attached in their entirety. These instructions provide critical information about requirements that if overlooked could lead to disqualification of a Vendor's bid. All bids must be submitted in accordance with the provisions contained in these instructions and the Solicitation. Failure to do so may result in disqualification of Vendor's bid.

2. **MANDATORY TERMS:** The Solicitation may contain mandatory provisions identified by the use of the words "must," "will," and "shall." Failure to comply with a mandatory term in the Solicitation will result in bid disqualification.

3. **PREBID MEETING:** The item identified below shall apply to this Solicitation.

☑ A pre-bid meeting will not be held prior to bid opening.

□ A **NON-MANDATORY PRE-BID** meeting will be held at the following place and time:

□ A **MANDATORY PRE-BID** meeting will be held at the following place and time:

All Vendors submitting a bid must attend the mandatory pre-bid meeting. Failure to attend the mandatory pre-bid meeting shall result in disqualification of the Vendor's bid. No one person attending the pre-bid meeting may represent more than one Vendor.

An attendance sheet provided at the pre-bid meeting shall serve as the official document verifying attendance. The State will not accept any other form of proof or documentation to verify attendance. Any person attending the pre-bid meeting on behalf of a Vendor must list on the attendance sheet his or her name and the name of the Vendor he or she is representing. Additionally, the person attending the pre-bid meeting should include the Vendor's E-Mail address, phone number, and Fax number on the attendance sheet. It is the Vendor's responsibility to locate the attendance sheet and provide the required information. Failure to complete the attendance sheet as required may result in disqualification of Vendor's bid.

All Vendors should arrive prior to the starting time for the pre-bid. Vendors who arrive after the starting time but prior to the end of the pre-bid will be permitted to sign in, but are charged with knowing all matters discussed at the pre-bid.

Questions submitted at least five business days prior to a scheduled pre-bid will be discussed at the pre-bid meeting if possible. Any discussions or answers to questions at the pre-bid meeting are preliminary in nature and are non-binding. Official and binding answers to questions will be published in a written addendum to the Solicitation prior to bid opening.

4. **VENDOR QUESTION DEADLINE:** Vendors may submit questions relating to this Solicitation to the Purchasing Division. Questions must be submitted in writing. All questions must be submitted on or before the date listed below and to the address listed below in order to be considered. A written response will be published in a Solicitation addendum if a response is possible and appropriate. Non-written discussions, conversations, or questions and answers regarding this Solicitation are preliminary in nature and are non-binding.

Question Submission Deadline: Tuesday, May 13, 2014 by 5:00pm EST

Submit Questions to: Robert P Kilpatrick, Senior Buyer
2019 Washington Street, East
Charleston, WV 25305
Fax: (304) 558-4115
(Vendors should not use this fax number for bid submission)
Email: robert

5. **VERBAL COMMUNICATION:** Any verbal communication between the Vendor and any State personnel is not binding, including that made at the mandatory pre-bid conference. Only information issued in writing and added to the Solicitation by an official written addendum by the Purchasing Division is binding.

6. **BID SUBMISSION:** All bids must be signed and delivered by the Vendor to the Purchasing Division at the address listed below on or before the date and time of the bid opening. Any bid received by the Purchasing Division staff is considered to be in the possession of the Purchasing Division and will not be returned for any reason. The Purchasing Division will not accept bids, modification of bids, or addendum acknowledgment forms via e-mail. Acceptable delivery methods include hand delivery, delivery by courier, or facsimile. The bid delivery address is:

Department of Administration, Purchasing Division
2019 Washington Street East
Charleston, WV 25305-0130

The bid should contain the information listed below on the face of the envelope or the bid may not be considered:

SEALED BID: ____________________
BUYER: Robert P Kilpatrick, File 22
SOLICITATION NO.: BPH14178
BID OPENING DATE: May 21, 2014
BID OPENING TIME: 1:30pm EST
FAX NUMBER: 304-558-3970

In the event that Vendor is responding to a request for proposal, the Vendor shall submit one original technical and one original cost proposal plus NA convenience copies of each to the Purchasing Division at the address shown above. Additionally, the Vendor should identify the bid type as either a technical or cost proposal on the face of each bid envelope submitted in response to a request for proposal as follows:

BID TYPE: □ Technical
☑ Cost

7. **BID OPENING:** Bids submitted in response to this Solicitation will be opened at the location identified below on the date and time listed below. Delivery of a bid after the bid opening date and time will result in bid disqualification. For purposes of this Solicitation, a bid is considered delivered when time stamped by the official Purchasing Division time clock.

Bid Opening Date and Time: Wednesday, May 21, 2014 at 1:30pm EST

Bid Opening Location:
Department of Administration, Purchasing Division
2019 Washington Street East
Charleston, WV 25305-0130

8. **ADDENDUM ACKNOWLEDGEMENT:** Changes or revisions to this Solicitation will be made by an official written addendum issued by the Purchasing Division. Vendor should acknowledge receipt of all addenda issued with this Solicitation by completing an Addendum Acknowledgment Form, a copy of which is included herewith. Failure to acknowledge addenda may result in bid disqualification. The addendum acknowledgement should be submitted with the bid to expedite document processing.

9. **BID FORMATTING:** Vendor should type or electronically enter the information onto its bid to prevent errors in the evaluation. Failure to type or electronically enter the information may result in bid disqualification.

GENERAL TERMS AND CONDITIONS:

1. **CONTRACTUAL AGREEMENT:** Issuance of a Purchase Order signed by the Purchasing Division Director, or his designee, and approved as to form by the Attorney General's office constitutes acceptance of this Contract made by and between the State of West Virginia and the Vendor. Vendor's signature on its bid signifies Vendor's agreement to be bound by and accept the terms and conditions contained in this Contract.

2. **DEFINITIONS:** As used in this Solicitation/Contract, the following terms shall have the meanings attributed to them below. Additional definitions may be found in the specifications included with this Solicitation/Contract.

   2.1 **"Agency"** or **"Agencies"** means the agency, board, commission, or other entity of the State of West Virginia that is identified on the first page of the Solicitation or any other public entity seeking to procure goods or services under this Contract.

   2.2 **"Contract"** means the binding agreement that is entered into between the State and the Vendor to provide the goods and services requested in the Solicitation.

   2.3 **"Director"** means the Director of the West Virginia Department of Administration, Purchasing Division.

   2.4 **"Purchasing Division"** means the West Virginia Department of Administration, Purchasing Division.

   2.5 **"Purchase Order"** means the document signed by the Agency and the Purchasing Division, and approved as to form by the Attorney General, that identifies the Vendor as the successful bidder and Contract holder.

   2.6 **"Solicitation"** means the official solicitation published by the Purchasing Division and identified by number on the first page thereof.

   2.7 **"State"** means the State of West Virginia and/or any of its agencies, commissions, boards, etc. as context requires.

   2.8 **"Vendor"** or **"Vendors"** means any entity submitting a bid in response to the Solicitation, the entity that has been selected as the lowest responsible bidder, or the entity that has been awarded the Contract as context requires.

3. **CONTRACT TERM; RENEWAL; EXTENSION:** The term of this Contract shall be determined in accordance with the category that has been identified as applicable to this Contract below:

□ **Term Contract**

Initial Contract Term: This Contract becomes effective on

and extends for a period of year(s).

Renewal Term: This Contract may be renewed upon the mutual written consent of the Agency, and the Vendor, with approval of the Purchasing Division and the Attorney General's office (Attorney General approval is as to form only). Any request for renewal must be submitted to the Purchasing Division Director thirty (30) days prior to the expiration date of the initial contract term or appropriate renewal term. A Contract renewal shall be in accordance with the terms and conditions of the original contract. Renewal of this Contract is limited to successive one (1) year periods. Automatic renewal of this Contract is prohibited. Notwithstanding the foregoing, Purchasing Division approval is not required on agency delegated or exempt purchases. Attorney General approval may be required for vendor terms and conditions.

Reasonable Time Extension: At the sole discretion of the Purchasing Division Director, and with approval from the Attorney General's office (Attorney General approval is as to form only), this Contract may be extended for a reasonable time after the initial Contract term or after any renewal term as may be necessary to obtain a new contract or renew this Contract. Any reasonable time extension shall not exceed twelve (12) months. Vendor may avoid a reasonable time extension by providing the Purchasing Division Director with written notice of Vendor's desire to terminate this Contract 30 days prior to the expiration of the then current term. During any reasonable time extension period, the Vendor may terminate this Contract for any reason upon giving the Purchasing Division Director 30 days written notice. Automatic extension of this Contract is prohibited. Notwithstanding the foregoing, Purchasing Division approval is not required on agency delegated or exempt purchases, but Attorney General approval may be required.

Release Order Limitations: In the event that this contract permits release orders, a release order may only be issued during the time this Contract is in effect. Any release order issued within one year of the expiration of this Contract shall be effective for one year from the date the release order is issued. No release order may be extended beyond one year after this Contract has expired.

□ **Fixed Period Contract:** This Contract becomes effective upon Vendor's receipt of the notice to proceed and must be completed within days.

☑ **One Time Purchase:** The term of this Contract shall run from the issuance of the Purchase Order until all of the goods contracted for have been delivered, but in no event shall this Contract extend for more than one fiscal year.

□ **Other:** See attached.

4. **NOTICE TO PROCEED:** Vendor shall begin performance of this Contract immediately upon receiving notice to proceed unless otherwise instructed by the Agency. Unless otherwise specified, the fully executed Purchase Order will be considered notice to proceed

5. **QUANTITIES:** The quantities required under this Contract shall be determined in accordance with the category that has been identified as applicable to this Contract below.

□ **Open End Contract:** Quantities listed in this Solicitation are approximations only, based on estimates supplied by the Agency. It is understood and agreed that the Contract shall cover the quantities actually ordered for delivery during the term of the Contract, whether more or less than the quantities shown.

□ **Service:** The scope of the service to be provided will be more clearly defined in the specifications included herewith.

□ **Combined Service and Goods:** The scope of the service and deliverable goods to be provided will be more clearly defined in the specifications included herewith.

☑ **One Time Purchase:** This Contract is for the purchase of a set quantity of goods that are identified in the specifications included herewith. Once those items have been delivered, no additional goods may be procured under this Contract without an appropriate change order approved by the Vendor, Agency, Purchasing Division, and Attorney General's office.

6. **PRICING:** The pricing set forth herein is firm for the life of the Contract, unless specified elsewhere within this Solicitation/Contract by the State. A Vendor's inclusion of price adjustment provisions in its bid, without an express authorization from the State in the Solicitation to do so, may result in bid disqualification.

7. **EMERGENCY PURCHASES:** The Purchasing Division Director may authorize the Agency to purchase goods or services in the open market that Vendor would otherwise provide under this Contract if those goods or services are for immediate or expedited delivery in an emergency. Emergencies shall include, but are not limited to, delays in transportation or an unanticipated increase in the volume of work. An emergency purchase in the open market, approved by the Purchasing Division Director, shall not constitute of breach of this Contract and shall not entitle the Vendor to any form of compensation or damages. This provision does not excuse the State from fulfilling its obligations under a One Time Purchase contract.

8. **REQUIRED DOCUMENTS:** All of the items checked below must be provided to the Purchasing Division by the Vendor as specified below.

□ **BID BOND:** All Vendors shall furnish a bid bond in the amount of five percent (5%) of the total amount of the bid protecting the State of West Virginia. The bid bond must be submitted with the bid.

□ **PERFORMANCE BOND:** The apparent successful Vendor shall provide a performance bond in the amount of . The performance bond must be issued and received by the Purchasing Division prior to Contract award. On construction contracts, the performance bond must be 100% of the Contract value.

□ **LABOR/MATERIAL PAYMENT BOND:** The apparent successful Vendor shall provide a labor/material payment bond in the amount of 100% of the Contract value. The labor/material payment bond must be issued and delivered to the Purchasing Division prior to Contract award.

In lieu of the Bid Bond, Performance Bond, and Labor/Material Payment Bond, the Vendor may provide certified checks, cashier's checks, or irrevocable letters of credit. Any certified check, cashier's check, or irrevocable letter of credit provided in lieu of a bond must be of the same amount and delivered on the same schedule as the bond it replaces. A letter of credit submitted in lieu of a performance and labor/material payment bond will only be allowed for projects under $100,000. Personal or business checks are not acceptable.

□ **MAINTENANCE BOND:** The apparent successful Vendor shall provide a two (2) year maintenance bond covering the roofing system. The maintenance bond must be issued and delivered to the Purchasing Division prior to Contract award.

□ **WORKERS' COMPENSATION INSURANCE:** The apparent successful Vendor shall have appropriate workers' compensation insurance and shall provide proof thereof upon request.

□ **INSURANCE:** The apparent successful Vendor shall furnish proof of the following insurance prior to Contract award and shall list the state as a certificate holder:

□ **Commercial General Liability Insurance:**
or more.

□ **Builders Risk Insurance:** builders risk – all risk insurance in an amount equal to 100% of the amount of the Contract.

□

□

□

□

□

The apparent successful Vendor shall also furnish proof of any additional insurance requirements contained in the specifications prior to Contract award regardless of whether or not that insurance requirement is listed above.

□ **LICENSE(S) / CERTIFICATIONS / PERMITS:** In addition to anything required under the Section entitled Licensing, of the General Terms and Conditions, the apparent successful Vendor shall furnish proof of the following licenses, certifications, and/or permits prior to Contract award, in a form acceptable to the Purchasing Division.

□

□

□

□

The apparent successful Vendor shall also furnish proof of any additional licenses or certifications contained in the specifications prior to Contract award regardless of whether or not that requirement is listed above.

9. **LITIGATION BOND:** The Director reserves the right to require any Vendor that files a protest of an award to submit a litigation bond in the amount equal to one percent of the lowest bid submitted or $5,000, whichever is greater. The entire amount of the bond shall be forfeited if the hearing officer determines that the protest was filed for frivolous or improper purpose, including but not limited to, the purpose of harassing, causing unnecessary delay, or needless expense for the Agency. All litigation bonds shall be made payable to the Purchasing Division. In lieu of a bond, the protester may submit a cashier's check or certified check payable to the Purchasing Division. Cashier's or certified checks will be deposited with and held by the State Treasurer's office. If it is determined that the protest has not been filed for frivolous or improper purpose, the bond or deposit shall be returned in its entirety.

10. **ALTERNATES:** Any model, brand, or specification listed herein establishes the acceptable level of quality only and is not intended to reflect a preference for, or in any way favor, a particular brand or vendor. Vendors may bid alternates to a listed model or brand provided that the alternate is at least equal to the model or brand and complies with the required specifications. The equality of any alternate being bid shall be determined by the State at its sole discretion. Any Vendor bidding an alternate model or brand should clearly identify the alternate items in its bid and should include manufacturer's specifications, industry literature, and/or any other relevant documentation demonstrating the equality of the alternate items. Failure to provide information for alternate items may be grounds for rejection of a Vendor's bid.

11. **EXCEPTIONS AND CLARIFICATIONS:** The Solicitation contains the specifications that shall form the basis of a contractual agreement. Vendor shall clearly mark any exceptions, clarifications, or

other proposed modifications in its bid. Exceptions to, clarifications of, or modifications of a requirement or term and condition of the Solicitation may result in bid disqualification.

12. **LIQUIDATED DAMAGES:** Vendor shall pay liquidated damages in the amount NA for NA

This clause shall in no way be considered exclusive and shall not limit the State or Agency's right to pursue any other available remedy.

13. **ACCEPTANCE/REJECTION:** The State may accept or reject any bid in whole, or in part. Vendor's signature on its bid signifies acceptance of the terms and conditions contained in the Solicitation and Vendor agrees to be bound by the terms of the Contract, as reflected in the Purchase Order, upon receipt.

14. **REGISTRATION:** Prior to Contract award, the apparent successful Vendor must be properly registered with the West Virginia Purchasing Division and must have paid the $125 fee if applicable.

15. **COMMUNICATION LIMITATIONS:** In accordance with West Virginia Code of State Rules §148-1-6.6, communication with the State of West Virginia or any of its employees regarding this Solicitation during the solicitation, bid, evaluation or award periods, except through the Purchasing Division, is strictly prohibited without prior Purchasing Division approval. Purchasing Division approval for such communication is implied for all agency delegated and exempt purchases.

16. **FUNDING:** This Contract shall continue for the term stated herein, contingent upon funds being appropriated by the Legislature or otherwise being made available. In the event funds are not appropriated or otherwise made available, this Contract becomes void and of no effect beginning on July 1 of the fiscal year for which funding has not been appropriated or otherwise made available.

17. **PAYMENT:** Payment in advance is prohibited under this Contract. Payment may only be made after the delivery and acceptance of goods or services. The Vendor shall submit invoices, in arrears, to the Agency at the address on the face of the purchase order labeled "Invoice To."

18. **UNIT PRICE:** Unit prices shall prevail in cases of a discrepancy in the Vendor's bid.

19. **DELIVERY:** All quotations are considered freight on board destination ("F.O.B. destination") unless alternate shipping terms are clearly identified in the bid. Vendor's listing of shipping terms that contradict the shipping terms expressly required by this Solicitation may result in bid disqualification.

20. **INTEREST:** Interest attributable to late payment will only be permitted if authorized by the West Virginia Code. Presently, there is no provision in the law for interest on late payments.

21. **PREFERENCE:** Vendor Preference may only be granted upon written request and only in accordance with the West Virginia Code § 5A-3-37 and the West Virginia Code of State Rules. A Resident Vendor Certification form has been attached hereto to allow Vendor to apply for the preference. Vendor's

failure to submit the Resident Vendor Certification form with its bid will result in denial of Vendor Preference. Vendor Preference does not applyto construction projects.

22. **SMALL, WOMEN-OWNED, OR MINORITY-OWNED BUSINESSES:** For any solicitations publicly advertised for bid on or after July 1, 2012, in accordance with West Virginia Code §5A-3-37(a)(7) and W. Va. CSR § 148-22-9, any non-resident vendor certified as a small, women-owned, or minority-owned business under W. Va. CSR § 148-22-9 shall be provided the same preference made available to any resident vendor. Any non-resident small, women-owned, or minority-owned business must identify itself as such in writing, must submit that writing to the Purchasing Division with its bid, and must be properly certified under W. Va. CSR § 148-22-9 prior to submission of its bid to receive the preferences made available to resident vendors. Preference for a non-resident small, women-owned, or minorityowned business shall be applied in accordance with W. Va. CSR § 148-22-9.

23. **TAXES:** The Vendor shall pay any applicable sales, use, personal property or any other taxes arising out of this Contract and the transactions contemplated thereby. The State of West Virginia is exempt from federal and state taxes and will not pay or reimburse such taxes.

24. **CANCELLATION:** The Purchasing Division Director reserves the right to cancel this Contract immediately upon written notice to the vendor if the materials or workmanship supplied do not conform to the specifications contained in the Contract. The Purchasing Division Director may cancel any purchase or Contract upon 30 days written notice to the Vendor in accordance with West Virginia Code of State Rules § 148-1-7.16.2.

25. **WAIVER OF MINOR IRREGULARITIES:** The Director reserves the right to waive minor irregularities in bids or specifications in accordance with West Virginia Code of State Rules § 148-1-4.6.

26. **TIME:** Time is of the essence with regard to all matters of time and performance in this Contract.

27. **APPLICABLE LAW**: This Contract is governed by and interpreted under West Virginia law without giving effect to its choice of law principles. Any information provided in specification manuals, or any other source, verbal or written, which contradicts or violates the West Virginia Constitution, West Virginia Code or West Virginia Code of State Rules is void and of no effect.

28. **COMPLIANCE**: Vendor shall comply with all applicable federal, state, and local laws, regulations and ordinances. By submitting a bid, Vendors acknowledge that they have reviewed, understand, and will comply with all applicable law.

29. **PREVAILING WAGE:** On any contract for the construction of a public improvement, Vendor and any subcontractors utilized by Vendor shall pay a rate or rates of wages which shall not be less than the fair minimum rate or rates of wages (prevailing wage), as established by the West Virginia Division of Labor under West Virginia Code §§ 21-5A-1 et seq. and available at http://www.sos.wv.gov/administrative-law/wagerates/Pages/default.aspx. Vendor shall be responsible for ensuring compliance with prevailing wage requirements and determining when prevailing wage

requirements are applicable. The required contract provisions contained in West Virginia Code of State Rules § 42-7-3 are specifically incorporated herein by reference.

30. **ARBITRATION:** Any references made to arbitration contained in this Contract, Vendor's bid, or in any American Institute of Architects documents pertaining to this Contract are hereby deleted, void, and of no effect.

31. **MODIFICATIONS**: This writing is the parties' final expression of intent. Notwithstanding anything contained in this Contract to the contrary, no modification of this Contract shall be binding without mutual written consent of the Agency, and the Vendor, with approval of the Purchasing Division and the Attorney General's office (Attorney General approval is as to form only). **No Change shall be implemented by the Vendor until such time as the Vendor receives an approved written change order from the Purchasing Division.**

32. **WAIVER:** The failure of either party to insist upon a strict performance of any of the terms or provision of this Contract, or to exercise any option, right, or remedy herein contained, shall not be construed as a waiver or a relinquishment for the future of such term, provision, option, right, or remedy, but the same shall continue in full force and effect. Any waiver must be expressly stated in writing and signed by the waiving party.

33. **SUBSEQUENT FORMS:** The terms and conditions contained in this Contract shall supersede any and all subsequent terms and conditions which may appear on any form documents submitted by Vendor to the Agency or Purchasing Division such as price lists, order forms, invoices, sales agreements, or maintenance agreements, and includes internet websites or other electronic documents. Acceptance or use of Vendor's forms does not constitute acceptance of the terms and conditions contained thereon.

34. **ASSIGNMENT**: Neither this Contract nor any monies due, or to become due hereunder, may be assigned by the Vendor without the express written consent of the Agency, the Purchasing Division, the Attorney General's office (as to form only), and any other government agency or office that may be required to approve such assignments. Notwithstanding the foregoing, Purchasing Division approval may or may not be required on certain agency delegated or exempt purchases.

35. **WARRANTY**: The Vendor expressly warrants that the goods and/or services covered by this Contract will: (a) conform to the specifications, drawings, samples, or other description furnished or specified by the Agency; (b) be merchantable and fit for the purpose intended; and (c) be free from defect in material and workmanship.

36. **STATE EMPLOYEES:** State employees are not permitted to utilize this Contract for personal use and the Vendor is prohibited from permitting or facilitating the same.

37. **BANKRUPTCY**: In the event the Vendor files for bankruptcy protection, the State of West Virginia may deem this Contract null and void, and terminate this Contract without notice.

38. [RESERVED]

39. CONFIDENTIALITY: The Vendor agrees that it will not disclose to anyone, directly or indirectly, any such personally identifiable information or other confidential information gained from the Agency, unless the individual who is the subject of the information consents to the disclosure in writing or the disclosure is made pursuant to the Agency's policies, procedures, and rules. Vendor further agrees to comply with the Confidentiality Policies and Information Security Accountability Requirements, set forth in http://www.state.wv.us/admin/purchase/privacy/default.html.

40. DISCLOSURE: Vendor's response to the Solicitation and the resulting Contract are considered public documents and will be disclosed to the public in accordance with the laws, rules, and policies governing the West Virginia Purchasing Division. Those laws include, but are not limited to, the Freedom of Information Act found in West Virginia Code § 29B-1-1 et seq.

If a Vendor considers any part of its bid to be exempt from public disclosure, Vendor must so indicate by specifically identifying the exempt information, identifying the exemption that applies, providing a detailed justification for the exemption, segregating the exempt information from the general bid information, and submitting the exempt information as part of its bid but in a segregated and clearly identifiable format. Failure to comply with the foregoing requirements will result in public disclosure of the Vendor's bid without further notice. A Vendor's act of marking all or nearly all of its bid as exempt is not sufficient to avoid disclosure and WILL NOT BE HONORED. Vendor's act of marking a bid or any part thereof as "confidential" or "proprietary" is not sufficient to avoid disclosure and WILL NOT BE HONORED. In addition, a legend or other statement indicating that all or substantially all of the bid is exempt from disclosure is not sufficient to avoid disclosure and WILL NOT BE HONORED. Vendor will be required to defend any claimed exemption for nondisclosure in the event of an administrative or judicial challenge to the State's nondisclosure. Vendor must indemnify the State for any costs incurred related to any exemptions claimed by Vendor. Any questions regarding the applicability of the various public records laws should be addressed to your own legal counsel prior to bid submission.

41. LICENSING: In accordance with West Virginia Code of State Rules §148-1-6.1.7, Vendor must be licensed and in good standing in accordance with any and all state and local laws and requirements by any state or local agency of West Virginia, including, but not limited to, the West Virginia Secretary of State's Office, the West Virginia Tax Department, West Virginia Insurance Commission, or any other state agency or political subdivision. Upon request, the Vendor must provide all necessary releases to obtain information to enable the Purchasing Division Director or the Agency to verify that the Vendor is licensed and in good standing with the above entities.

42. ANTITRUST: In submitting a bid to, signing a contract with, or accepting a Purchase Order from any agency of the State of West Virginia, the Vendor agrees to convey, sell, assign, or transfer to the State of West Virginia all rights, title, and interest in and to all causes of action it may now or hereafter acquire under the antitrust laws of the United States and the State of West Virginia for price fixing and/or unreasonable restraints of trade relating to the particular commodities or services purchased or acquired by the State of West Virginia. Such assignment shall be made and become effective at the time the purchasing agency tenders the initial payment to Vendor.

43. **VENDOR CERTIFICATIONS:** By signing its bid or entering into this Contract, Vendor certifies (1) that its bid was made without prior understanding, agreement, or connection with any corporation, firm, limited liability company, partnership, person or entity submitting a bid for the same material, supplies, equipment or services; (2) that its bid is in all respects fair and without collusion or fraud; (3) that this Contract is accepted or entered into without any prior understanding, agreement, or connection to any other entity that could be considered a violation of law; and (4) that it has reviewed this RFQ in its entirety; understands the requirements, terms and conditions, and other information contained herein. Vendor's signature on its bid also affirms that neither it nor its representatives have any interest, nor shall acquire any interest, direct or indirect, which would compromise the performance of its services hereunder. Any such interests shall be promptly presented in detail to the Agency.

The individual signing this bid on behalf of Vendor certifies that he or she is authorized by the Vendor to execute this bid or any documents related thereto on Vendor's behalf; that he or she is authorized to bind the Vendor in a contractual relationship; and that, to the best of his or her knowledge, the Vendor has properly registered with any State agency that may require registration.

44. **PURCHASING CARD ACCEPTANCE:** The State of West Virginia currently utilizes a Purchasing Card program, administered under contract by a banking institution, to process payment for goods and services. The Vendor must accept the State of West Virginia's Purchasing Card for payment of all orders under this Contract unless the box below is checked.

□ Vendor is not required to accept the State of West Virginia's Purchasing Card as payment for all goods and services.

45. **VENDOR RELATIONSHIP:** The relationship of the Vendor to the State shall be that of an independent contractor and no principal-agent relationship or employer-employee relationship is contemplated or created by this Contract. The Vendor as an independent contractor is solely liable for the acts and omissions of its employees and agents. Vendor shall be responsible for selecting, supervising, and compensating any and all individuals employed pursuant to the terms of this Solicitation and resulting contract. Neither the Vendor, nor any employees or subcontractors of the Vendor, shall be deemed to be employees of the State for any purpose whatsoever. Vendor shall be exclusively responsible for payment of employees and contractors for all wages and salaries, taxes, withholding payments, penalties, fees, fringe benefits, professional liability insurance premiums, contributions to insurance and pension, or other deferred compensation plans, including but not limited to, Workers' Compensation and Social Security obligations, licensing fees, *etc.* and the filing of all necessary documents, forms and returns pertinent to all of the foregoing. Vendor shall hold harmless the State, and shall provide the State and Agency with a defense against any and all claims including, but not limited to, the foregoing payments, withholdings, contributions, taxes, Social Security taxes, and employer income tax returns.

46. **INDEMNIFICATION:** The Vendor agrees to indemnify, defend, and hold harmless the State and the Agency, their officers, and employees from and against: (1) Any claims or losses for services rendered

by any subcontractor, person, or firm performing or supplying services, materials, or supplies in connection with the performance of the Contract; (2) Any claims or losses resulting to any person or entity injured or damaged by the Vendor, its officers, employees, or subcontractors by the publication, translation, reproduction, delivery, performance, use, or disposition of any data used under the Contract in a manner not authorized by the Contract, or by Federal or State statutes or regulations; and (3) Any failure of the Vendor, its officers, employees, or subcontractors to observe State and Federal laws including, but not limited to, labor and wage and hour laws.

47. **PURCHASING AFFIDAVIT:** In accordance with West Virginia Code § 5A-3-10a, all Vendors are required to sign, notarize, and submit the Purchasing Affidavit stating that neither the Vendor nor a related party owe a debt to the State in excess of $1,000. The affidavit must be submitted prior to award, but should be submitted with the Vendor's bid. A copy of the Purchasing Affidavit is included herewith.

48. **ADDITIONAL AGENCY AND LOCAL GOVERNMENT USE:** This Contract may be utilized by and extends to other agencies, spending units, and political subdivisions of the State of West Virginia; county, municipal, and other local government bodies; and school districts ("Other Government Entities"). This Contract shall be extended to the aforementioned Other Government Entities on the same prices, terms, and conditions as those offered and agreed to in this Contract. If the Vendor does not wish to extend the prices, terms, and conditions of its bid and subsequent contract to the Other Government Entities, the Vendor must clearly indicate such refusal in its bid. A refusal to extend this Contract to the Other Government Entities shall not impact or influence the award of this Contract in any manner.

49. **CONFLICT OF INTEREST:** Vendor, its officers or members or employees, shall not presently have or acquire any interest, direct or indirect, which would conflict with or compromise the performance of its obligations hereunder. Vendor shall periodically inquire of its officers, members and employees to ensure that a conflict of interest does not arise. Any conflict of interest discovered shall be promptly presented in detail to the Agency.

50. **REPORTS:** Vendor shall provide the Agency and/or the Purchasing Division with the following reports identified by a checked box below:

☑ Such reports as the Agency and/or the Purchasing Division may request. Requested reports may include, but are not limited to, quantities purchased, agencies utilizing the contract, total contract expenditures by agency, etc.

□ Quarterly reports detailing the total quantity of purchases in units and dollars, along with a listing of purchases by agency. Quarterly reports should be delivered to the Purchasing Division via email at purchasing.requisitions@wv.gov.

51. **BACKGROUND CHECK:** In accordance with W. Va. Code § 15-2D-3, the Director of the Division of Protective Services shall require any service provider whose employees are regularly employed on the grounds or in the buildings of the Capitol complex or who have access to sensitive or critical information

to submit to a fingerprint-based state and federal background inquiry through the state repository. The service provider is responsible for any costs associated with the fingerprint-based state and federal background inquiry.

After the contract for such services has been approved, but before any such employees are permitted to be on the grounds or in the buildings of the Capitol complex or have access to sensitive or critical information, the service provider shall submit a list of all persons who will be physically present and working at the Capitol complex to the Director of the Division of Protective Services for purposes of verifying compliance with this provision.

The State reserves the right to prohibit a service provider's employees from accessing sensitive or critical information or to be present at the Capitol complex based upon results addressed from a criminal background check.

Service providers should contact the West Virginia Division of Protective Services by phone at (304)558-9911 for more information.

52. **PREFERENCE FOR USE OF DOMESTIC STEEL PRODUCTS:** Except when authorized by the Director of the Purchasing Division pursuant to W. Va. Code § 5A-3-56, no contractor may use or supply steel products for a State Contract Project other than those steel products made in the United States. A contractor who uses steel products in violation of this section may be subject to civil penalties pursuant to W. Va. Code § 5A-3-56. As used in this section:

a. "State Contract Project" means any erection or construction of, or any addition to, alteration of or other improvement to any building or structure, including, but not limited to, roads or highways, or the installation of any heating or cooling or ventilating plants or other equipment, or the supply of and materials for such projects, pursuant to a contract with the State of West Virginia for which bids were solicited on or after June 6, 2001.

b. "Steel Products" means products rolled, formed, shaped, drawn, extruded, forged, cast, fabricated or otherwise similarly processed, or processed by a combination of two or more or such operations, from steel made by the open heath, basic oxygen, electric furnace, Bessemer or other steel making process.

The Purchasing Division Director may, in writing, authorize the use of foreign steel products if:

a. The cost for each contract item used does not exceed one tenth of one percent (.1%) of the total contract cost or two thousand five hundred dollars ($2,500.00), whichever is greater. For the purposes of this section, the cost is the value of the steel product as delivered to the project; or

b. The Director of the Purchasing Division determines that specified steel materials are not produced in the United States in sufficient quantity or otherwise are not reasonably available to meet contract requirements.

53. PREFERENCE FOR USE OF DOMESTIC ALUMINUM, GLASS, AND STEEL: In Accordance with W. Va. Code § 5-19-1 et seq., and W. Va. CSR § 148-10-1 et seq., for every contract or subcontract, subject to the limitations contained herein, for the construction, reconstruction, alteration, repair, improvement or maintenance of public works or for the purchase of any item of machinery or equipment to be used at sites of public works, only domestic aluminum, glass or steel products shall be supplied unless the spending officer determines, in writing, after the receipt of offers or bids, (1) that the cost of domestic aluminum, glass or steel products is unreasonable or inconsistent with the public interest of the State of West Virginia, (2) that domestic aluminum, glass or steel products are not produced in sufficient quantities to meet the contract requirements, or (3) the available domestic aluminum, glass, or steel do not meet the contract specifications. This provision only applies to public works contracts awarded in an amount more than fifty thousand dollars ($50,000) or public works contracts that require more than ten thousand pounds of steel products.

The cost of domestic aluminum, glass, or steel products may be unreasonable if the cost is more than twenty percent (20%) of the bid or offered price for foreign made aluminum, glass, or steel products. If the domestic aluminum, glass or steel products to be supplied or produced in a "substantial labor surplus area", as defined by the United States Department of Labor, the cost of domestic aluminum, glass, or steel products may be unreasonable if the cost is more than thirty percent (30%) of the bid or offered price for foreign made aluminum, glass, or steel products.

This preference shall be applied to an item of machinery or equipment, as indicated above, when the item is a single unit of equipment or machinery manufactured primarily of aluminum, glass or steel, is part of a public works contract and has the sole purpose or of being a permanent part of a single public works project. This provision does not apply to equipment or machinery purchased by a spending unit for use by that spending unit and not as part of a single public works project.

All bids and offers including domestic aluminum, glass or steel products that exceed bid or offer prices including foreign aluminum, glass or steel products after application of the preferences provided in this provision may be reduced to a price equal to or lower than the lowest bid or offer price for foreign aluminum, glass or steel products plus the applicable preference. If the reduced bid or offer prices are made in writing and supersede the prior bid or offer prices, all bids or offers, including the reduced bid or offer prices, will be reevaluated in accordance with this rule.

SPECIFICATIONS

1. **PURPOSE AND SCOPE:** The West Virginia Purchasing Division is soliciting bids on behalf of the West Virginia Department of Health and Human Resources, Bureau for Public Health, Center for Threat Preparedness (Agency) to establish a contract for the one time purchase of twelve (12) Digital Dual Band Mobile Radio Base Systems. Upon receipt of a purchase order, Vendor must be able to deliver all Contract Items so that Agency receives by June 30, 2014.

2. **DEFINITIONS:** The terms listed below shall have the meanings assigned to them below. Additional definitions can be found in section 2 of the General Terms and Conditions.

   2.1 "Contract Item" means the Digital Dual Band Mobile Radio Base System described in Section 3.1.1 herein.

   2.2 "Pricing Page" means the page upon which Vendor should list its proposed price for the Contract Item in the manner requested. The Pricing Page is included in this RFQ and attached hereto as Exhibit A.

   2.3 "RFQ" means the official request for quotation published by the Purchasing Division and identified as BPH14178.

   2.4 "MHz" means megahertz.

   2.5 "kHz" means kilohertz.

   2.6 "dB" means decibel.

   2.7 "GPS" means global positioning system.

   2.8 "dBm" means decibel milliwatts.

   2.9 "kbps" means kilobits per second.

3. **GENERAL REQUIREMENTS:**

   3.1 Mandatory Contract Item Requirements: Contract Item must meet or exceed the mandatory requirements listed below.

3.1.1 Contract Item #1 – Quantity: 12 - Digital Dual Band Mobile Radio Base System, Motorola APX 7500 Multi-Band Mobile Radio Base System, or Equal.

3.1.1.1 Digital Dual Band Mobile Radio Base System shall be a Motorola, APX 7500 Multi-Band Mobile Radio Base System, or Equal. If alternate digital dual band mobile radio base system is bid, Vendor shall provide product specifications or literature to confirm item meets the following mandatory requirements.

3.1.1.2 Digital Dual Band Mobile Radio Base System shall consist of a Radio Frequency (RF) base station configured as a consolette that includes a signal aggregate device configured as a junction box allowing a desk top controller to be located up to five hundred (500) feet from the RF base station; desk top controller; digital dual band mobile radio; power supply; and a digital remote board.

3.1.1.3 Digital Dual Band Mobile Radio Base System shall be capable of a minimum of one thousand two hundred fifty (1,250) channel operation and shall be P-25 Phase 2 compliant.

3.1.1.4 Digital Dual Band Mobile Radio Base System desk top controller shall be a Motorola MC5000 Deskset Controller, or Equal, be compatible to the Digital Dual Band Mobile Radio Base System described in Section 3.1.1.1 herein, and meet the following mandatory requirements. If alternate desk top controller is bid, Vendor shall provide product specifications or literature to confirm item meets the following mandatory requirements.

3.1.1.4.1 Desk top controller shall be similar in appearance to an office desk telephone and shall be configured to allow full control of the base system through a display and preprogrammed function buttons.

3.1.1.4.2 Desk top controller shall support speaker telephone and handset operations, foot switch, handset jacks, and logging recorder jack.

3.1.1.4.3 Desk top controller shall remotely control all radio functions and frequencies of the consolette.

3.1.1.4.4 Desk top controller buttons shall be duplicates of the digital dual band mobile radio, with customized labels.

3.1.1.4.5 Desk top controller shall have automatic line leveling to compensate for variations in telephone lines.

3.1.1.4.6 Desk top controller shall have a 2-line, 20-character backlit display capable of showing all text messages generated by the digital dual band mobile radio control head.

3.1.1.4.7 Desk top controller shall have a reversible housing mount to allow the unit to be operated on a desk top or wall mounted without the need for additional accessories or equipment.

3.1.1.4.8 Desk top controller shall come standard with a 110/220 volt power supply.

3.1.1.4.9 Desk top controller shall accept line impedance of 600Ω nominal to 10KΩ.

3.1.1.5 Digital Dual Band Mobile Radio Base System digital dual band mobile radio shall be a Motorola APX 7500 Multi-Band Mobile Radio, or Equal, be compatible to the Digital Dual Band Mobile Radio Base System described in Section 3.1.1.1 herein, and meet the following mandatory requirements. If alternate digital dual band mobile radio is bid, Vendor shall provide product specifications or literature to confirm item meets the following mandatory requirements.

3.1.1.5.1 Digital dual band mobile radio shall be capable of operating in all of the following frequency bands:
3.1.1.5.1.1 VHF: 136-174 MHz
3.1.1.5.1.2 UHF R1: 380-470 MHz

3.1.1.5.1.3 700 MHz: 764-776 MHz and 794-806 MHz

3.1.1.5.1.4 800 MHz: 806-824 MHz and 851-870 MHz

3.1.1.5.2 Digital dual band mobile radio shall be capable of utilizing 6.25 kHz; 12.5 kHz; and 25 kHz channel spacing.

3.1.1.5.3 Digital dual band mobile radio shall be capable of using AES/DES-OFB encryption algorithms.

3.1.1.5.4 Digital dual band mobile radio shall have an Encryption Algorithm Capacity of eight (8).

3.1.1.5.5 Digital dual band mobile radio encryption keys module shall be capable of storing a minimum of 1,024 keys; and programmable for 128 common key reference (CKR) or 16 physical identifier (PID).

3.1.1.5.6 Digital dual band mobile radio's encryption frame re-sync interval shall be P25 CAI 300 milliseconds.

3.1.1.5.7 Digital dual band mobile radio encryption keying shall be loaded by a key variable loader.

3.1.1.5.8 Digital dual band mobile radio synchronization shall be XL-counter addressing; OFB-Output feedback.

3.1.1.5.9 Digital dual band mobile radio vector generator shall be a National Institute of Standards and Technology's approved random number generator.

3.1.1.5.10 Digital dual band mobile radio encryption shall be digital.

3.1.1.5.11 Digital dual band mobile radio key erasure shall be accomplished by keyboard command and tamper detection.

3.1.1.5.12 Digital dual band mobile radio encryption shall meet the Federal Information Processing Standards FIPS 140-2 Level 3 and FIPS 197.

3.1.1.5.13 Digital dual band mobile radio key storage shall be in tamper protected volatile or non-volatile memory.

3.1.1.5.14 Digital dual band mobile radio shall be capable of the following transmit power levels:

3.1.1.5.14.1 10-30 watts: 700 MHz
3.1.1.5.14.2 10-35 watts: 800 MHz
3.1.1.5.14.3 10-50 watts: VHF dash mount low-medium power
3.1.1.5.14.4 25-110 watts: VHF remote mount high power
3.1.1.5.14.5 10-40 watts: UHF 380-470 MHz dash mount low-medium power
3.1.1.5.14.6 25-110 watts: UHF 380-470 MHz remote mount high power

3.1.1.5.15 Digital dual band mobile radio transmitter channel spacing shall be:

3.1.1.4.15.1 25/12.5 kHz in the 700 MHz range
3.1.1.4.15.2 25/12.5 kHz in the 800 MHz range
3.1.1.4.15.3 30/25/12.5 kHz in the VHF range
3.1.1.4.15.4 25/12.5 kHz in the UHF range

3.1.1.5.16 Digital dual band mobile radio frequency separation shall be the full bandwidth.

3.1.1.5.17 Digital dual band mobile radio frequency stability shall be a minimum of 0.8 parts per million in all frequency ranges.

3.1.1.5.18 Digital dual band mobile radio audio distortion shall not exceed 0.50% in all frequency ranges.

3.1.1.5.19 Digital dual band mobile radio frequency modulation (FM) hum and noise at 25 kHz separation shall be:
3.1.1.5.19.1 50 dB in 700 MHz and 800 MHz
3.1.1.5.19.2 53 dB in VHF and UHF range

3.1.1.5.20 Digital dual band mobile radio frequency modulation (FM) hum and noise at 12.5 kHz separation shall be:
3.1.1.5.20.1 48 dB in 700 MHz and 800 MHz

3.1.1.5.20.2 52 dB in VHF
3.1.1.5.20.3 50 dB in UHF range

3.1.1.5.21 Digital dual band mobile radio continuous 4 level modulation fidelity (C4FM) for 12.5 kHz digital channel shall be a minimum of 1.10% in all frequency ranges.

3.1.1.5.22 Digital dual band mobile radio shall be P-25 compliant.

3.1.1.5.23 Digital dual band mobile radio shall include multi-band performance capable of operating in:
3.1.1.5.23.1 VHF: 136-174 MHz and UHF: 380-470 MHz
3.1.1.5.23.2 VHF: 136-174 MHz and 700/800: 764-870 MHz
3.1.1.5.23.3 UHF: 380-470 MHz and 700/800: 764-870 MHz

3.1.1.5.24 Digital dual band mobile radio shall be capable of scanning both bands of the radio at the same time.

3.1.1.5.25 Digital dual band mobile radio shall be capable of using and storing user defined scan lists that can be established, edited, and disabled on the radio keypad.

3.1.1.5.26 Digital dual band mobile radio shall be capable of simultaneously scanning both bands in the radio as well as "Trunking" and "Conventional" channels.

3.1.1.5.27 Digital dual band mobile radio shall be capable of Over-The-Air Programming (OTAP).

3.1.1.5.28 Digital dual band mobile radio shall be capable of Over-The-Air Rekeying (OTAR).

3.1.1.5.29 Digital dual band mobile radio receiver channel spacing shall be:
3.1.1.5.29.1 25/12.5 kHz in 700 MHz, 800 MHz, and UHF range
3.1.1.5.29.2 30/25/12.5 kHz in VHF range

3.1.1.5.30 Digital dual Band mobile radio receiver maximum frequency separation shall be full bandsplit for each frequency range.

3.1.1.5.31 Digital dual Band mobile radio receiver audio output power at 3% distortion shall be a minimum of 7.5 W frequency ranges.

3.1.1.5.32 Digital dual Band mobile radio receiver frequency stability shall be a minimum of 0.8 parts per million for all frequency ranges.

3.1.1.5.33 Digital dual Band mobile radio receiver analog sensitivity at 12dB SINAD shall be:

3.1.1.5.33.1 -121 dBm for 700 MHz and 800 MHz
3.1.1.5.33.2 -123 dBm using a pre-amp
3.1.1.5.33.3 -119 dBm standard for VHF and UHF

3.1.1.5.34 Digital dual Band mobile radio receiver digital sensitivity at 5% BER shall be:

3.1.1.5.34.1 -121.5 dBm for 700 MHz and 800 MHz
3.1.1.5.34.2 -123 dBm using a pre-amp
3.1.1.5.34.3 -119 dBm standard for VHF and UHF

3.1.1.5.35 Digital dual Band mobile radio receiver intermodulation at 25 kHz shall be:

3.1.1.5.35.1 82 dB for 700 MHz and 800 MHz
3.1.1.5.35.2 84 dB using a pre-amp in VHF
3.1.1.5.35.3 82 dB using a pre-amp in UHF
3.1.1.5.35.4 86 dB without using a pre-amp in VHF and UHF

3.1.1.5.36 Digital dual Band mobile radio receiver intermodulation at 12.5 kHz shall be:

3.1.1.5.36.1 82 dB for 700 MHz and 800 MHz
3.1.1.5.36.2 85 dB using a pre-amp in VHF
3.1.1.5.36.3 83 dB using a pre-amp in UHF
3.1.1.5.36.4 86 dB without using a pre-amp in VHF
3.1.1.5.36.5 85dB without using a pre-amp in UHF range 1 and UHF

3.1.1.5.37 Digital dual Band mobile radio receiver spurious rejection shall be:

3.1.1.5.37.1 91 dB for 700 MHz and 800 MHz

3.1.1.5.37.2 95 dB for VHF
3.1.1.5.37.3 93 dB for UHF

3.1.1.5.38 Digital dual Band mobile radio receiver audio distortion as rated shall be 1.2%.

3.1.1.5.39 Digital dual Band mobile radio receiver frequency modulation (FM) hum and noise at 25 kHz separation shall be:
3.1.1.5.39.1 59 dB for 700 MHz, 800 MHz, and VHF
3.1.1.5.39.2 55 dB for UHF

3.1.1.5.40 Digital dual Band mobile radio receiver frequency modulation (FM) hum and noise at 12.5 kHz separation shall be 50 dB for all frequency ranges.

3.1.1.5.41 Digital dual Band mobile radio receiver selectivity at 25 kHz shall be 85 dB for all frequency ranges.

3.1.1.5.42 Digital dual Band mobile radio receiver selectivity at 12.5 kHz shall be 75 dB for all frequency ranges.

3.1.1.5.43 Digital dual Band mobile radio receiver selectivity at 30 kHz shall be 90 dB for VHF.

3.1.1.5.44 Digital dual band mobile radio shall operate at 13.8 volt DC +/- 20% negative ground.

3.1.1.5.45 Digital dual band mobile radio standby current at 13.8 V shall be no greater than 0.85 amps for all frequency ranges.

3.1.1.5.46 Digital dual band mobile radio receive current at rated audio at 13.8 volt shall be no greater than 3.2 amps for all frequency ranges.

3.1.1.5.47 Digital dual band mobile radio transmit current at rated power shall be a maximum of:
3.1.1.5.47.1 136-174 MHz: 10-50 watts
3.1.1.5.47.2 13 amps: 50 watts
3.1.1.5.47.3 8 amps: 15 watts
3.1.1.5.47.4 380-470 MHz: 10-40 watts
3.1.1.5.47.5 11 amps: 40 watts
3.1.1.5.47.6 8 amps: 15 watts

3.1.1.5.47.7 764-870 MHz: 10-35 watts
3.1.1.5.47.8 12 amps: 50 watts
3.1.1.5.47.9 8 amps: 15 watts
3.1.1.5.47.10 136-174 MHz: 25-110 watts
3.1.1.5.47.11 20 amps: 110 watts
3.1.1.5.47.12 380-470 MHz: 25-110 watts
3.1.1.5.47.13 24 amps: 110 watts

3.1.1.5.48 Digital dual band mobile radio shall be integrated global positioning system (GPS) capable.

3.1.1.5.49 Digital dual band mobile radio GPS shall be capable of receiving a minimum of 12 channels.

3.1.1.5.50 Digital dual band mobile radio GPS tracking sensitivity shall be a minimum of -153 dBm.

3.1.1.5.51 Digital dual band mobile radio GPS accuracy shall be less than 10 meters.

3.1.1.5.52 Digital dual band mobile radio GPS cold start shall be less than 60 seconds.

3.1.1.5.53 Digital dual band mobile radio GPS hot start shall be less than 10 seconds.

3.1.1.5.54 Digital dual band mobile radio GPS mode of operation shall be autonomous.

3.1.1.5.54 Digital dual band mobile radio shall support clear or digital operation trunking.

3.1.1.5. **55** Digital dual band mobile radio shall have analog and digital systems configurations enabled.

3.1.1.5.56 Digital dual band mobile radio shall be capable of embedded digital signaling.

3.1.1.5.57 Digital dual band mobile radio signaling rate shall be 9.6 kbps.

3.1.1.5.58 Digital dual band mobile radio digital identification capacity shall be 10,000,000 conventional and 48,000 trunking.

3.1.1.5.59 Digital dual band mobile radio digital access codes shall be a minimum of 4,096 network site addresses.

3.1.1.5.60 Digital dual band mobile radio digital user group network site addresses shall be a minimum of 4,096 network addresses.

3.1.1.5.61 Digital dual band mobile radio digital user group addresses shall be a minimum of 65,000 conventional and 4,096 trunking.

3.1.1.5.62 Digital dual band mobile radio must use software for all programming functions which is compatible with Microsoft Windows XP or newer (to preserve compatibility with existing Agency software operating system).

3.1.1.5.63 Digital dual band mobile radio must include a USB cable to connect to a computer.

3.1.1.5.64 Digital dual band mobile radio channels shall be individually programmable for wideband and narrowband; digital and analog; trunking and non-trunking; encode and decode; encrypted and non-encrypted; transmit frequency; receive frequency; digital and analog continuous tone-coded squelch system tones and no tones operations.

3.1.1.5.65 Digital dual band mobile radio shall have an alphanumeric display with a minimum of two (2) rows with a minimum of fourteen (14) characters each.

3.1.1.5.66 Digital dual band mobile radio shall have a minimum of five (5) programmable function buttons.

3.1.1.5.67 Digital dual band mobile radio shall have an emergency button with a clearly distinctive color.

3.1.1.5.68 Digital dual band mobile radio shall have a menu navigation button capability.

3.1.1.6 Digital Dual Band Mobile Radio Base System will include a minimum of one (1) year parts and labor warranty that covers all of the components of the base system.

4. CONTRACT AWARD:

4.1 Contract Award: The Contract is intended to provide Agency with a purchase price for the Contract Item. The Contract shall be awarded to the Vendor that provides the Contract Item meeting the required specifications for the lowest overall Grant Total cost as shown on the Pricing Page.

4.2 Pricing Page: Vendor should complete the Pricing Page by completing the Unit Price, Extended Price, and Grand Total fields. The Extended Price should be calculated by multiplying the Quantity by the Unit Price. The Grand Total should be calculated by adding the Extended Price Column. Vendor should complete the Pricing Page in full as failure to complete the Pricing Page in its entirety may result in Vendor's bid being disqualified.

Vendor should type or electronically enter the information into the Pricing Page to prevent errors in the evaluation.

5. PAYMENT:

5.1 Payment: Vendor shall accept payment in accordance with the payment procedures of the State of West Virginia.

6. DELIVERY AND RETURN:

6.1 Shipment and Delivery: Upon receipt of a purchase order, Vendor must be able to deliver all Contract Items so that Agency receives by June 30, 2014. Contract Items shall be delivered to Agency at Bureau for Public Health, Center for Threat Preparedness, 505 Capitol Street, Suite 200, Charleston, West Virginia 25301.

6.2 Late Delivery: The Agency placing the order under this Contract must be notified in writing if the shipment of the Contract Item will be delayed for any reason. Any delay in delivery that could cause harm to an Agency will be grounds for cancellation of the Contract, and/or obtaining the Contract Item from a third party.

Any Agency seeking to obtain the Contract Item from a third party under this provision must first obtain approval of the Purchasing Division.

6.3 Delivery Payment/Risk of Loss: Vendor shall deliver the Contract Items F.O.B. destination to the Agency's location.

6.4 Return of Unacceptable Items: If the Agency deems the Contract Items to be unacceptable, the Contract Item shall be returned to Vendor at Vendor's expense and with no restocking charge. Vendor shall either make arrangements for the return within five (5) days of being notified that items are unacceptable, or permit the Agency to arrange for the return and reimburse Agency for delivery expenses. If the original packaging cannot be utilized for the return, Vendor will supply the Agency with appropriate return packaging upon request. All returns of unacceptable items shall be F.O.B. the Agency's location. The returned product shall either be replaced, or the Agency shall receive a full credit or refund for the purchase price, at the Agency's discretion.

6.5 Return Due to Agency Error: Items ordered in error by the Agency will be returned for credit within 30 days of receipt, F.O.B. Vendor's location. Vendor shall not charge a restocking fee if returned products are in a resalable condition. Items shall be deemed to be in a resalable condition if they are unused and in the original packaging. Any restocking fee for items not in a resalable condition shall be the lower of the Vendor's customary restocking fee or 5% of the total invoiced value of the returned items.

FIPS PUB 140-2

CHANGE NOTICES (12-03-2002)

FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION
(Supercedes FIPS PUB 140-1, 1994 January 11)

SECURITY REQUIREMENTS FOR CRYPTOGRAPHIC MODULES

CATEGORY: COMPUTER SECURITY **SUBCATEGORY: CRYPTOGRAPHY**

Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899-8900

Issued May 25, 2001

U.S. Department of Commerce
Donald L. Evans, Secretary

Technology Administration
Phillip J. Bond, Under Secretary for Technology

National Institute of Standards and Technology
Arden L. Bement, Jr., Director

Foreword

The Federal Information Processing Standards Publication Series of the National Institute of Standards and Technology (NIST) is the official series of publications relating to standards and guidelines adopted and promulgated under the provisions of Section 5131 of the Information Technology Management Reform Act of 1996 (Public Law 104-106) and the Computer Security Act of 1987 (Public Law 100-235). These mandates have given the Secretary of Commerce and NIST important responsibilities for improving the utilization and management of computer and related telecommunications systems in the Federal government. The NIST, through its Information Technology Laboratory, provides leadership, technical guidance, and coordination of government efforts in the development of standards and guidelines in these areas.

Comments concerning Federal Information Processing Standards Publications are welcomed and should be addressed to the Director, Information Technology Laboratory, National Institute of Standards and Technology, 100 Bureau Drive, Stop 8900, Gaithersburg, MD 20899-8900.

William Mehuron, Director
Information Technology Laboratory

Abstract

The selective application of technological and related procedural safeguards is an important responsibility of every Federal organization in providing adequate security in its computer and telecommunication systems. This publication provides a standard that will be used by Federal organizations when these organizations specify that cryptographic-based security systems are to be used to provide protection for sensitive or valuable data. Protection of a cryptographic module within a security system is necessary to maintain the confidentiality and integrity of the information protected by the module. This standard specifies the security requirements that will be satisfied by a cryptographic module. The standard provides four increasing, qualitative levels of security intended to cover a wide range of potential applications and environments. The security requirements cover areas related to the secure design and implementation of a cryptographic module. These areas include cryptographic module specification; cryptographic module ports and interfaces; roles, services, and authentication; finite state model; physical security; operational environment; cryptographic key management; electromagnetic interference/electromagnetic compatibility (EMI/EMC); self-tests; design assurance; and mitigation of other attacks.

Key words: computer security, telecommunication security, cryptography, cryptographic modules, Federal Information Processing Standard (FIPS).

National Institute of Standards
and Technology
FIPS PUB 140-2
64 pages (May 25, 2001)

U.S. Government Printing Office
Washington: 2001

For Sale by the National
Technical Information
Service
U.S. Department of Commerce

Federal Information
Processing Standards Publication 140-2

May 25, 2001

Announcing the Standard for

SECURITY REQUIREMENTS FOR CRYPTOGRAPHIC MODULES

Federal Information Processing Standards Publications (FIPS PUBS) are issued by the National Institute of Standards and Technology (NIST) after approval by the Secretary of Commerce pursuant to Section 5131 of the Information Technology Management Reform Act of 1996 (Public Law 104-106) and the Computer Security Act of 1987 (Public Law 100-235).

1. **Name of Standard.** Security Requirements for Cryptographic Modules (FIPS PUB 140-2).

2. **Category of Standard.** Computer Security Standard, Cryptography.

3. **Explanation.** This standard specifies the security requirements that will be satisfied by a cryptographic module utilized within a security system protecting sensitive but unclassified information (hereafter referred to as sensitive information). The standard provides four increasing, qualitative levels of security: Level 1, Level 2, Level 3, and Level 4. These levels are intended to cover the wide range of potential applications and environments in which cryptographic modules may be employed. The security requirements cover areas related to the secure design and implementation of a cryptographic module. These areas include cryptographic module specification, cryptographic module ports and interfaces; roles, services, and authentication; finite state model; physical security; operational environment; cryptographic key management; electromagnetic interference/electromagnetic compatibility (EMI/EMC); self-tests; design assurance; and mitigation of other attacks. This standard supersedes FIPS 140-1, *Security Requirements for Cryptographic Modules*, in its entirety.

The Cryptographic Module Validation Program (CMVP) validates cryptographic modules to Federal Information Processing Standard (FIPS) 140-2 and other cryptography based standards. The CMVP is a joint effort between NIST and the Communications Security Establishment (CSE) of the Government of Canada. Products validated as conforming to FIPS 140-2 are accepted by the Federal agencies of both countries for the protection of sensitive information (United States) or Designated Information (Canada). The goal of the CMVP is to promote the use of validated cryptographic modules and provide Federal agencies with a security metric to use in procuring equipment containing validated cryptographic modules.

In the CMVP, vendors of cryptographic modules use independent, accredited testing laboratories to have their modules tested. National Voluntary Laboratory Accreditation Program (NVLAP) accredited laboratories perform cryptographic module compliance/conformance testing.

4. **Approving Authority.** Secretary of Commerce.

5. **Maintenance Agency.** Department of Commerce, National Institute of Standards and Technology, Information Technology Laboratory (ITL).

6. **Cross Index.**

   a. FIPS PUB 46-3, Data Encryption Standard.
   b. FIPS PUB 74, Guidelines for Implementing and Using the NBS Data Encryption Standard.
   c. FIPS PUB 81, DES Modes of Operation.
   d. FIPS PUB 113, Computer Data Authentication.

e. FIPS PUB 171, Key Management Using ANSI X9.17.
f. FIPS PUB 180-1, Secure Hash Standard.
g. FIPS PUB 186-2, Digital Signature Standard.
h. Special Publication 800-2, Public Key Cryptography.
i. Special Publication 800-20, Modes of Operation Validation System for the Triple Data Encryption Algorithm (TMOVS): Requirements and Procedures

These documents may be found at the CMVP URL http://www.nist.gov/cmvp. Other NIST publications may be applicable to the implementation and use of this standard. A list (NIST Publications List 91) of currently available computer security publications, including ordering information, can be obtained from NIST.

7. Applicability. This standard is applicable to all Federal agencies that use cryptographic-based security systems to protect sensitive information in computer and telecommunication systems (including voice systems) as defined in Section 5131 of the Information Technology Management Reform Act of 1996, Public Law 104-106. This standard shall be used in designing and implementing cryptographic modules that Federal departments and agencies operate or are operated for them under contract. Cryptographic modules that have been approved for classified use may be used in lieu of modules that have been validated against this standard. The adoption and use of this standard is available to private and commercial organizations.

8. Applications. Cryptographic-based security systems may be utilized in various computer and telecommunication applications (e.g., data storage, access control and personal identification, network communications, radio, facsimile, and video) and in various environments (e.g., centralized computer facilities, office environments, and hostile environments). The cryptographic services (e.g., encryption, authentication, digital signature, and key management) provided by a cryptographic module are based on many factors that are specific to the application and environment. The security level to which a cryptographic module is validated must be chosen to provide a level of security appropriate for the security requirements of the application and environment in which the module will be utilized and the security services that the module will provide. The security requirements for a particular security level include both the security requirements specific to that level and the security requirements that apply to all modules regardless of the level.

9. Specifications. Federal Information Processing Standard (FIPS) 140-2, Security Requirements for Cryptographic Modules (affixed).

10. Implementations. This standard covers implementations of cryptographic modules including, but not limited to, hardware components or modules, software/firmware programs or modules or any combination thereof. Cryptographic modules that are validated under the CMVP will be considered as conforming to this standard. Information about the CMVP can be obtained from the

a. National Institute of Standards and Technology, Information Technology Laboratory, 100 Bureau Drive, Stop 8900, Gaithersburg, MD 20899-8900.
b. Communications Security Establishment, ITS Client Services, 1500 Bronson Ave., Ottawa, ON K1G 3Z4.
c. CMVP URL http://www.nist.gov/cmvp.

11. Approved Security Functions. Cryptographic modules that conform to this standard shall employ Approved security functions such as cryptographic algorithms, cryptographic key management techniques, and authentication techniques that have been approved for protecting Federal government sensitive information. Approved security functions include those that are either:

a. specified in a Federal Information Processing Standard (FIPS),
b. adopted in a FIPS and specified either in an appendix to the FIPS or in a document referenced by the FIPS, or
c. specified in the list of Approved security functions.

12. Interpretation. Questions concerning the content and specifications of this standard should be addressed to: Director, Information Technology Laboratory, ATTN: FIPS 140-2 Interpretation, National Institute of Standards and Technology, 100 Bureau Drive, Stop 8900, Gaithersburg, MD 20899-8900. Resolution of questions regarding this standard will be provided by the validation authorities at NIST and CSE.

13. Export Control. Certain cryptographic devices and technical data regarding them are subject to Federal export controls and exports of cryptographic modules implementing this standard and technical data regarding them must comply with these Federal regulations and be licensed by the Bureau of Export Administration of the U.S. Department of Commerce. Applicable Federal government export controls are specified in Title 15, Code of Federal Regulations (CFR) Part 740.17; Title 15, CFR Part 742; and Title 15, CFR Part 774, Category 5, Part 2.

14. Implementation Schedule. This standard becomes effective six months after approval by the Secretary of Commerce. A transition period from November 25, 2001 until six months after the effective date is provided to enable all agencies to develop plans for the acquisition of products that are compliant with FIPS 140-2. Agencies may retain and use FIPS 140-1 validated products that have been purchased before the end of the transition period. After the transition period, modules will no longer be tested against the FIPS 140-1 requirements. After the transition period, all previous validations against FIPS 140-1 will still be recognized. Figure 1 summarizes the FIPS 140-2 implementation schedule.

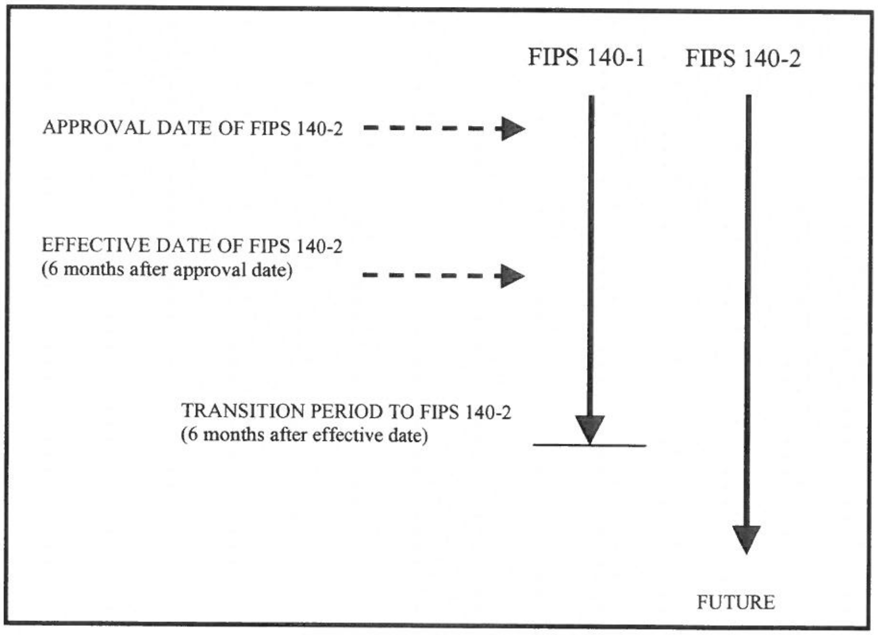

FIPS 140-1
FIPS 140-2
APPROVAL DATE OF FIPS 140-2
EFFECTIVE DATE OF FIPS 140-2
(6 months after approval date)
TRANSITION PERIOD TO FIPS 140-2
(6 months after effective date)
FUTURE

Figure 1. *FIPS 140-2 Implementation Schedule*

15. Qualifications. The security requirements specified in this standard are based upon information provided by many sources within the Federal government and private industry. The requirements are designed to protect against adversaries mounting cost-effective attacks on sensitive government or commercial data (e.g., hackers, organized crime, and economic competitors). The primary goal in designing an effective security system is to make the cost of any attack greater than the possible payoff.

While the security requirements specified in this standard are intended to maintain the security provided by a cryptographic module, conformance to this standard is not sufficient to ensure that a particular module is secure. The operator of a cryptographic module is responsible for ensuring that the security provided by a module is sufficient and acceptable to the owner of the information that is being protected and that any residual risk is acknowledged and accepted.

Similarly, the use of a validated cryptographic module in a computer or telecommunications system does not guarantee the security of the overall system. The responsible authority in each agency shall ensure that the security of the system is sufficient and acceptable.

Since a standard of this nature must be flexible enough to adapt to advancements and innovations in science and technology, this standard will be reviewed every five years in order to consider new or revised requirements that may be needed to meet technological and economic changes.

16. Waiver Procedure. Under certain exceptional circumstances, the heads of Federal agencies, or their delegates, may approve waivers to Federal Information Processing Standards (FIPS), for their agency. The heads of such agencies may redelegate such authority only to a senior official designated pursuant to Section 3506(b) of Title 44, U.S. Code. Waivers shall be granted only when compliance with a standard would

a. adversely affect the accomplishment of the mission of an operator of Federal computer system or

b. cause a major adverse financial impact on the operator that is not offset by government-wide savings.

Agency heads may act upon a written waiver request containing the information detailed above. Agency heads may also act without a written waiver request when they determine which conditions for meeting the standard cannot be met. Agency heads may approve waivers only by a written decision that explains the basis on which the agency head made the required finding(s). A copy of each such decision, with procurement sensitive or classified portions clearly identified, shall be sent to: National Institute of Standards and Technology; ATTN: FIPS Waiver Decision, Information Technology Laboratory, 100 Bureau Drive, Stop 8900, Gaithersburg, MD 20899-8900.

In addition, notice of each waiver granted and each delegation of authority to approve waivers shall be sent promptly to the Committee on Government Operations of the House of Representatives and the Committee on Government Affairs of the Senate and shall be published promptly in the Federal Register.

When the determination on a waiver applies to the procurement of equipment and/or services, a notice of the waiver determination must be published in the Commerce Business Daily as a part of the notice of solicitation for offers of an acquisition or, if the waiver determination is made after that notice is published, by amendment to such notice.

A copy of the waiver, any supporting documents, the document approving the waiver and any supporting and accompanying documents, with such deletions as the agency is authorized and decides to make under Section 552(b) of Title 5, U.S. Code, shall be part of the procurement documentation and retained by the agency.

17. Where to obtain copies. Copies of this publication are available from the URL: http://csrc.nist.gov/publications. Copies are available for sale by the National Technical Information Service, U.S. Department of Commerce, Springfield, VA 22161. When ordering, refer to Federal Information Processing Standards Publication 140-2 (FIPSPUB1402) and identify the title. When microfiche is desired, this should be specified. Prices are published by NTIS in current catalogs and other issuances. Payment may be made by check, money order, deposit account, or charged to a credit card accepted by NTIS.

18. CHANGE NOTICE. See important change notice at the end of this document.

TABLE OF CONTENTS

1. OVERVIEW

This standard specifies the security requirements for a cryptographic module utilized within a security system protecting sensitive information in computer and telecommunication systems (including voice systems) as defined in Section 5131 of the Information Technology Management Reform Act of 1996, Public Law 104-106.

FIPS 140-1 was developed by a government and industry working group composed of both operators and vendors. The working group identified requirements for four security levels for cryptographic modules to provide for a wide spectrum of data sensitivity (e.g., low value administrative data, million dollar funds transfers, and life protecting data) and a diversity of application environments (e.g., a guarded facility, an office, and a completely unprotected location). Four security levels are specified for each of 11 requirement areas. Each security level offers an increase in security over the preceding level. These four increasing levels of security allow cost-effective solutions that are appropriate for different degrees of data sensitivity and different application environments. FIPS 140-2 incorporates changes in applicable standards and technology since the development of FIPS 140-1 as well as changes that are based on comments received from the vendor, laboratory, and user communities.

While the security requirements specified in this standard are intended to maintain the security provided by a cryptographic module, conformance to this standard is not sufficient to ensure that a particular module is secure. The operator of a cryptographic module is responsible for ensuring that the security provided by the module is sufficient and acceptable to the owner of the information that is being protected, and that any residual risk is acknowledged and accepted.

Similarly, the use of a validated cryptographic module in a computer or telecommunications system is not sufficient to ensure the security of the overall system. The overall security level of a cryptographic module must be chosen to provide a level of security appropriate for the security requirements of the application and environment in which the module is to be utilized and for the security services that the module is to provide. The responsible authority in each organization should ensure that their computer and telecommunication systems that utilize cryptographic modules provide an acceptable level of security for the given application and environment.

The importance of security awareness and of making information security a management priority should be communicated to all users. Since information security requirements vary for different applications, organizations should identify their information resources and determine the sensitivity to and the potential impact of losses. Controls should be based on the potential risks and should be selected from available controls, including administrative policies and procedures, physical and environmental controls, information and data controls, software development and acquisition controls, and backup and contingency planning.

The following sections provide an overview of the four security levels. Common examples, given to illustrate how the requirements might be met, are not intended to be restrictive or exhaustive.

The location of Annexes A, B, C, and D can be found in APPENDIX D: SELECTED BIBLIOGRAPHY.

1.1 Security Level 1

Security Level 1 provides the lowest level of security. Basic security requirements are specified for a cryptographic module (e.g., at least one Approved algorithm or Approved security function shall be used). No specific physical security mechanisms are required in a Security Level 1 cryptographic module beyond the basic requirement for production-grade components. An example of a Security Level 1 cryptographic module is a personal computer (PC) encryption board.

Security Level 1 allows the software and firmware components of a cryptographic module to be executed on a general purpose computing system using an unevaluated operating system. Such implementations may be appropriate for some low-level security applications when other controls, such as physical security, network security, and administrative procedures are limited or nonexistent. The implementation of cryptographic software may be more cost-effective than corresponding hardware-based mechanisms, enabling organizations to select from alternative cryptographic solutions to meet lower-level security requirements.

1.2 Security Level 2

Security Level 2 enhances the physical security mechanisms of a Security Level 1 cryptographic module by adding the requirement for tamper-evidence, which includes the use of tamper-evident coatings or seals or for pick-resistant locks on removable covers or doors of the module. Tamper-evident coatings or seals are placed on a cryptographic module so that the coating or seal must be broken to attain physical access to the plaintext cryptographic keys and critical security parameters (CSPs) within the module. Tamper-evident seals or pick-resistant locks are placed on covers or doors to protect against unauthorized physical access.

Security Level 2 requires, at a minimum, role-based authentication in which a cryptographic module authenticates the authorization of an operator to assume a specific role and perform a corresponding set of services.

Security Level 2 allows the software and firmware components of a cryptographic module to be executed on a general purpose computing system using an operating system that

- meets the functional requirements specified in the Common Criteria (CC) Protection Profiles (PPs) listed in Annex B and

- is evaluated at the CC evaluation assurance level EAL2 (or higher).

An equivalent evaluated trusted operating system may be used. A trusted operating system provides a level of trust so that cryptographic modules executing on general purpose computing platforms are comparable to cryptographic modules implemented using dedicated hardware systems.

1.3 Security Level 3

In addition to the tamper-evident physical security mechanisms required at Security Level 2, Security Level 3 attempts to prevent the intruder from gaining access to CSPs held within the cryptographic module. Physical security mechanisms required at Security Level 3 are intended to have a high probability of detecting and responding to attempts at physical access, use or modification of the cryptographic module. The physical security mechanisms may include the use of strong enclosures and tamper detection/response circuitry that zeroizes all plaintext CSPs when the removable covers/doors of the cryptographic module are opened.

Security Level 3 requires identity-based authentication mechanisms, enhancing the security provided by the role-based authentication mechanisms specified for Security Level 2. A cryptographic module authenticates the identity of an operator and verifies that the identified operator is authorized to assume a specific role and perform a corresponding set of services.

Security Level 3 requires the entry or output of plaintext CSPs (including the entry or output of plaintext CSPs using split knowledge procedures) be performed using ports that are physically separated from other ports, or interfaces that are logically separated using a trusted path from other interfaces. Plaintext CSPs may be entered into or output from the cryptographic module in encrypted form (in which case they may travel through enclosing or intervening systems).

Security Level 3 allows the software and firmware components of a cryptographic module to be executed on a general purpose computing system using an operating system that

- meets the functional requirements specified in the PPs listed in Annex B with the additional functional requirement of a Trusted Path (FTP_TRP.1) and

- is evaluated at the CC evaluation assurance level EAL3 (or higher) with the additional assurance requirement of an Informal Target of Evaluation (TOE) Security Policy Model (ADV_SPM.1).

An equivalent evaluated trusted operating system may be used. The implementation of a trusted path protects plaintext CSPs and the software and firmware components of the cryptographic module from other untrusted software or firmware that may be executing on the system.

1.4 Security Level 4

Security Level 4 provides the highest level of security defined in this standard. At this security level, the physical security mechanisms provide a complete envelope of protection around the cryptographic module with the intent of detecting and responding to all unauthorized attempts at physical access. Penetration of the cryptographic module enclosure from any direction has a very high probability of being detected, resulting in the immediate zeroization of all plaintext CSPs. Security Level 4 cryptographic modules are useful for operation in physically unprotected environments.

Security Level 4 also protects a cryptographic module against a security compromise due to environmental conditions or fluctuations outside of the module's normal operating ranges for voltage and temperature. Intentional excursions beyond the normal operating ranges may be used by an attacker to thwart a cryptographic module's defenses. A cryptographic module is required to either include special environmental protection features designed to detect fluctuations and zeroize CSPs, or to undergo rigorous environmental failure testing to provide a reasonable assurance that the module will not be affected by fluctuations outside of the normal operating range in a manner that can compromise the security of the module.

Security Level 4 allows the software and firmware components of a cryptographic module to be executed on a general purpose computing system using an operating system that

- meets the functional requirements specified for Security Level 3 and

- is evaluated at the CC evaluation assurance level EAL4 (or higher).

An equivalent evaluated trusted operating system may be used.

2. GLOSSARY OF TERMS AND ACRONYMS

2.1 Glossary of Terms

The following definitions are tailored for use in this standard:

Approved: FIPS-Approved and/or NIST-recommended.

Approved mode of operation: a mode of the cryptographic module that employs only Approved security functions (not to be confused with a specific mode of an Approved security function, e.g., DES CBC mode).

Approved security function: for this standard, a security function (e.g., cryptographic algorithm, cryptographic key management technique, or authentication technique) that is either

a) specified in an Approved standard,
b) adopted in an Approved standard and specified either in an appendix of the Approved standard or in a document referenced by the Approved standard, or
c) specified in the list of Approved security functions.

Authentication code: a cryptographic checksum based on an Approved security function (also known as a Message Authentication Code).

Automated key transport: the transport of cryptographic keys, usually in encrypted form, using electronic means such as a computer network (e.g., key transport/agreement protocols).

Compromise: the unauthorized disclosure, modification, substitution, or use of sensitive data (including plaintext cryptographic keys and other CSPs).

Confidentiality: the property that sensitive information is not disclosed to unauthorized individuals, entities, or processes.

Control information: information that is entered into a cryptographic module for the purposes of directing the operation of the module.

Critical security parameter (CSP): security-related information (e.g., secret and private cryptographic keys, and authentication data such as passwords and PINs) whose disclosure or modification can compromise the security of a cryptographic module.

Cryptographic boundary: an explicitly defined continuous perimeter that establishes the physical bounds of a cryptographic module and contains all the hardware, software, and/or firmware components of a cryptographic module.

Cryptographic key (key): a parameter used in conjunction with a cryptographic algorithm that determines

- the transformation of plaintext data into ciphertext data,
- the transformation of ciphertext data into plaintext data,
- a digital signature computed from data,
- the verification of a digital signature computed from data,
- an authentication code computed from data, or
- an exchange agreement of a shared secret.

Cryptographic key component (key component): a parameter used in conjunction with other key components in an Approved security function to form a plaintext cryptographic key or perform a cryptographic function.

Cryptographic module: the set of hardware, software, and/or firmware that implements Approved security functions (including cryptographic algorithms and key generation) and is contained within the cryptographic boundary.

Cryptographic module security policy: a precise specification of the security rules under which a cryptographic module will operate, including the rules derived from the requirements of this standard and additional rules imposed by the vendor. (See Appendix C.)

Crypto officer: an operator or process (subject), acting on behalf of the operator, performing cryptographic initialization or management functions.

Data path: the physical or logical route over which data passes; a physical data path may be shared by multiple logical data paths.

Differential power analysis (DPA): an analysis of the variations of the electrical power consumption of a cryptographic module, using advanced statistical methods and/or other techniques, for the purpose of extracting information correlated to cryptographic keys used in a cryptographic algorithm.

Digital signature: the result of a cryptographic transformation of data which, when properly implemented, provides the services of:

1. origin authentication
2. data integrity, and
3. signer non-repudiation.

Electromagnetic compatibility (EMC): the ability of electronic devices to function satisfactorily in an electromagnetic environment without introducing intolerable electromagnetic disturbances to other devices in that environment.

Electromagnetic interference (EMI): electromagnetic emissions from a device, equipment, or system that interfere with the normal operation of another device, equipment, or system.

Electronic key entry: the entry of cryptographic keys into a cryptographic module using electronic methods such as a smart card or a key-loading device. (The operator of the key may have no knowledge of the value of the key being entered.)

Encrypted key: a cryptographic key that has been encrypted using an Approved security function with a key encrypting key, a PIN, or a password in order to disguise the value of the underlying plaintext key.

Environmental failure protection (EFP): the use of features to protect against a compromise of the security of a cryptographic module due to environmental conditions or fluctuations outside of the module's normal operating range.

Environmental failure testing (EFT): the use of testing to provide a reasonable assurance that the security of a cryptographic module will not be compromised by environmental conditions or fluctuations outside of the module's normal operating range.

Error detection code (EDC): a code computed from data and comprised of redundant bits of information designed to detect, but not correct, unintentional changes in the data.

Finite state model: a mathematical model of a sequential machine that is comprised of a finite set of input events, a finite set of output events, a finite set of states, a function that maps states and input to output, a function that maps states and inputs to states (a state transition function), and a specification that describes the initial state.

Firmware: the programs and data components of a cryptographic module that are stored in hardware (e.g., ROM, PROM, EPROM, EEPROM or FLASH) within the cryptographic boundary and cannot be dynamically written or modified during execution.

Hardware: the physical equipment within the cryptographic boundary used to process programs and data.

Hash-based message authentication code (HMAC): a message authentication code that utilizes a keyed hash.

Initialization vector (IV): a vector used in defining the starting point of an encryption process within a cryptographic algorithm.

Input data: information that is entered into a cryptographic module for the purposes of transformation or computation using an Approved security function.

Integrity: the property that sensitive data has not been modified or deleted in an unauthorized and undetected manner.

Interface: a logical entry or exit point of a cryptographic module that provides access to the module for logical information flows representing physical signals.

Key encrypting key: a cryptographic key that is used for the encryption or decryption of other keys.

Key establishment: the process by which cryptographic keys are securely distributed among cryptographic modules using manual transport methods (e.g., key loaders), automated methods (e.g., key transport and/or key agreement protocols), or a combination of automated and manual methods (consists of key transport plus key agreement).

Key loader: a self-contained unit that is capable of storing at least one plaintext or encrypted cryptographic key or key component that can be transferred, upon request, into a cryptographic module.

Key management: the activities involving the handling of cryptographic keys and other related security parameters (e.g., IVs and passwords) during the entire life cycle of the keys, including their generation, storage, establishment, entry and output, and zeroization.

Key transport: secure transport of cryptographic keys from one cryptographic module to another module.

Manual key transport: a non-electronic means of transporting cryptographic keys.

Manual key entry: the entry of cryptographic keys into a cryptographic module, using devices such as a keyboard.

Microcode: the elementary processor instructions that correspond to an executable program instruction.

Operator: an individual accessing a cryptographic module or a process (subject) operating on behalf of the individual, regardless of the assumed role.

Output data: information that is produced from a cryptographic module.

Password: a string of characters (letters, numbers, and other symbols) used to authenticate an identity or to verify access authorization.

Personal identification number (PIN): an alphanumeric code or password used to authenticate an identity.

Physical protection: the safeguarding of a cryptographic module, cryptographic keys, or CSPs using physical means.

Plaintext key: an unencrypted cryptographic key.

Port: a physical entry or exit point of a cryptographic module that provides access to the module for physical signals, represented by logical information flows (physically separated ports do not share the same physical pin or wire).

Private key: a cryptographic key, used with a public key cryptographic algorithm, that is uniquely associated with an entity and is not made public.

Protection Profile: an implementation-independent set of security requirements for a category of Targets of Evaluation (TOEs) that meet specific consumer needs.

Public key: a cryptographic key used with a public key cryptographic algorithm that is uniquely associated with an entity and that may be made public. (Public keys are not considered CSPs.)

Public key certificate: a set of data that uniquely identifies an entity, contains the entity's public key, and is digitally signed by a trusted party, thereby binding the public key to the entity.

Public key (asymmetric) cryptographic algorithm: a cryptographic algorithm that uses two related keys, a public key and a private key. The two keys have the property that deriving the private key from the public key is computationally infeasible.

Random Number Generator: Random Number Generators (RNGs) used for cryptographic applications typically produce a sequence of zero and one bits that may be combined into sub-sequences or blocks of random numbers. There are two basic classes: deterministic and nondeterministic. A deterministic RNG consists of an algorithm that produces a sequence of bits from an initial value called a seed. A nondeterministic RNG produces output that is dependent on some unpredictable physical source that is outside human control.

Removable cover: a cover designed to permit physical access to the contents of a cryptographic module.

Secret key: a cryptographic key, used with a secret key cryptographic algorithm, that is uniquely associated with one or more entities and should not be made public.

Secret key (symmetric) cryptographic algorithm: a cryptographic algorithm that uses a single secret key for both encryption and decryption.

Security policy: see Cryptographic module security policy.

Seed key: a secret value used to initialize a cryptographic function or operation.

Simple power analysis (SPA): a direct (primarily visual) analysis of patterns of instruction execution (or execution of individual instructions), obtained through monitoring the variations in electrical power consumption of a cryptographic module, for the purpose of revealing the features and implementations of cryptographic algorithms and subsequently the values of cryptographic keys.

Software: the programs and data components within the cryptographic boundary, usually stored on erasable media (e.g., disk), that can be dynamically written and modified during execution.

Split knowledge: a process by which a cryptographic key is split into multiple key components, individually sharing no knowledge of the original key, that can be subsequently input into, or output from, a cryptographic module by separate entities and combined to recreate the original cryptographic key.

Status information: information that is output from a cryptographic module for the purposes of indicating certain operational characteristics or states of the module.

System software: the special software within the cryptographic boundary (e.g., operating system, compilers or utility programs) designed for a specific computer system or family of computer systems to facilitate the operation and maintenance of the computer system, and associated programs, and data.

Tamper detection: the automatic determination by a cryptographic module that an attempt has been made to compromise the physical security of the module.

Tamper evidence: the external indication that an attempt has been made to compromise the physical security of a cryptographic module. (The evidence of the tamper attempt should be observable by an operator subsequent to the attempt.)

Tamper response: the automatic action taken by a cryptographic module when a tamper detection has occurred (the minimum response action is the zeroization of plaintext keys and CSPs).

Target of Evaluation (TOE): an information technology product or system and associated administrator and user guidance documentation that is the subject of an evaluation.

TEMPEST: a name referring to the investigation, study, and control of unintentional compromising emanations from telecommunications and automated information systems equipment.

TOE Security Functions (TSF): used in the Common Criteria, a set of the TOE consisting of all hardware, software, and firmware that must be relied upon for the correct enforcement of the TOE Security Policy.

TOE Security Policy (TSP): used in the Common Criteria, a set of rules that regulate how assets are managed, protected, and distributed within a Target of Evaluation.

Trusted path: a means by which an operator and a TOE Security Function can communicate with the necessary confidence to support the TOE Security Policy.

User: an individual or a process (subject) acting on behalf of the individual that accesses a cryptographic module in order to obtain cryptographic services.

Validation authorities: NIST and CSE.

Zeroization: a method of erasing electronically stored data, cryptographic keys, and CSPs by altering or deleting the contents of the data storage to prevent recovery of the data.

2.2 Acronyms

The following acronyms and abbreviations are used throughout this standard:

ANSI	American National Standards Institute
API	Application Program Interface

CAPP	Controlled Access Protection Profile
CBC	Cipher Block Chaining
CC	Common Criteria
CMVP	Cryptographic Module Validation Program
CSE	Communications Security Establishment of the Government of Canada
CSP	Critical Security Parameter
DES	Data Encryption Standard
DOD	Department of Defense
DPA	Differential Power Analysis
DTR	Derived Test Requirements
EAL	Common Criteria Evaluation Assurance Level
EDC	Error Detection Code
EEPROM	Electronically-Erasable Programmable Read-Only Memory
EFP	Environmental Failure Protection
EFT	Environmental Failure Testing
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
EPROM	Erasable Programmable Read-Only Memory
FCC	Federal Communications Commission
FIPS	Federal Information Processing Standard
FIPS PUB	FIPS Publication
HDL	Hardware Description Language
HMAC	Hash-Based Message Authentication Code
IC	Integrated Circuit
IG	Implementation Guidance
ISO	International Organization for Standardization
ITSEC	Information Technology Security Evaluation Criteria
IV	Initialization Vector

NIST	National Institute of Standards and Technology
NTIS	National Technical Information Service
PIN	Personal Identification Number
PROM	Programmable Read-Only Memory
RAM	Random Access Memory
RNG	Random Number Generator
ROM	Read-Only Memory
SPA	Simple Power Analysis
TOE	Target of Evaluation
TSF	Target of Evaluation Security Functions
TSP	Target of Evaluation Security Policy
URL	Uniform Resource Locator

3. FUNCTIONAL SECURITY OBJECTIVES

The security requirements specified in this standard relate to the secure design and implementation of a cryptographic module. The requirements are derived from the following high-level functional security objectives for a cryptographic module:

- To employ and correctly implement the Approved security functions for the protection of sensitive information.
- To protect a cryptographic module from unauthorized operation or use.
- To prevent the unauthorized disclosure of the contents of the cryptographic module, including plaintext cryptographic keys and CSPs.
- To prevent the unauthorized and undetected modification of the cryptographic module and cryptographic algorithms, including the unauthorized modification, substitution, insertion, and deletion of cryptographic keys and CSPs.
- To provide indications of the operational state of the cryptographic module.
- To ensure that the cryptographic module performs properly when operating in an Approved mode of operation.
- To detect errors in the operation of the cryptographic module and to prevent the compromise of sensitive data and CSPs resulting from these errors.

4. SECURITY REQUIREMENTS

This section specifies the security requirements that shall be satisfied by cryptographic modules conforming to this standard. The security requirements cover areas related to the design and implementation of a cryptographic module. These areas include cryptographic module specification; module ports and interfaces; roles, services, and authentication; finite state model; physical security; operational environment; cryptographic key management; electromagnetic interference/electromagnetic compatibility (EMI/EMC); self-tests; and design assurance. An additional area concerned with the mitigation of other attacks is currently not tested but the vendor is required to document implemented controls (e.g., differential power analysis, and TEMPEST). Table 1 summarizes the security requirements in each of these areas.

	Security Level 1	*Security Level 2*	*Security Level 3*	*Security Level 4*
Cryptographic Module Specification	Specification of cryptographic module, cryptographic boundary, Approved algorithms, and Approved modes of operation. Description of cryptographic module, including all hardware, software, and firmware components. Statement of module security policy.			
Cryptographic Module Ports and Interfaces	Required and optional interfaces. Specification of all interfaces and of all input and output data paths.		Data ports for unprotected critical security parameters logically or physically separated from other data ports.	
Roles, Services, and Authentication	Logical separation of required and optional roles and services.	Role-based or identity-based operator authentication.	Identity-based operator authentication.	
Finite State Model	Specification of finite state model. Required states and optional states. State transition diagram and specification of state transitions.			
Physical Security	Production grade equipment.	Locks or tamper evidence.	Tamper detection and response for covers and doors.	Tamper detection and response envelope. EFP or EFT.
Operational Environment	Single operator. Executable code. Approved integrity technique.	Referenced PPs evaluated at EAL2 with specified discretionary access control mechanisms and auditing.	Referenced PPs plus trusted path evaluated at EAL3 plus security policy modeling.	Referenced PPs plus trusted path evaluated at EAL4.
Cryptographic Key Management	Key management mechanisms: random number and key generation, key establishment, key distribution, key entry/output, key storage, and key zeroization.			
	Secret and private keys established using manual methods may be entered or output in plaintext form.		Secret and private keys established using manual methods shall be entered or output encrypted or with split knowledge procedures.	
EMI/EMC	47 CFR FCC Part 15. Subpart B, Class A (Business use). Applicable FCC requirements (for radio).		47 CFR FCC Part 15. Subpart B, Class B (Home use).	
Self-Tests	Power-up tests: cryptographic algorithm tests, software/firmware integrity tests, critical functions tests. Conditional tests.			
Design Assurance	Configuration management (CM). Secure installation and generation. Design and policy correspondence. Guidance documents.	CM system. Secure distribution. Functional specification.	High-level language implementation.	Formal model. Detailed explanations (informal proofs). Preconditions and postconditions.
Mitigation of Other Attacks	Specification of mitigation of attacks for which no testable requirements are currently available.			

Table 1: *Summary of security requirements*

A cryptographic module shall be tested against the requirements of each area addressed in this section. The cryptographic module shall be independently rated in each area. Several areas provide for increasing levels of security with cumulative security requirements for each security level. In these areas, the cryptographic module will receive a rating that reflects the maximum security level for which the module fulfills all of the requirements of that area. In areas that do not provide for different levels of security (i.e., standard set of requirements), the cryptographic module will receive a rating commensurate with the overall level of security.

In addition to receiving independent ratings for each of the security areas, a cryptographic module will also receive an overall rating. The overall rating will indicate the minimum of the independent ratings received in the areas.

Many of the security requirements of this standard include specific documentation requirements that are summarized in Appendices A and C. All documentation, including copies of the user and installation manuals, shall be provided to the testing laboratory by the vendor.

4.1 Cryptographic Module Specification

A cryptographic module shall be a set of hardware, software, firmware, or some combination thereof that implements cryptographic functions or processes, including cryptographic algorithms and, optionally, key generation, and is contained within a defined cryptographic boundary. A cryptographic module shall implement at least one Approved security function used in an Approved mode of operation. Non-Approved security functions may also be included for use in non-Approved modes of operation. The operator shall be able to determine when an Approved mode of operation is selected. For Security Levels 1 and 2, the cryptographic module security policy may specify when a cryptographic module is performing in an Approved mode of operation. For Security Levels 3 and 4, a cryptographic module shall indicate when an Approved mode of operation is selected. (Approved security functions are listed in Annex A to this standard.)

A cryptographic boundary shall consist of an explicitly defined perimeter that establishes the physical bounds of a cryptographic module. If a cryptographic module consists of software or firmware components, the cryptographic boundary shall contain the processor(s) and other hardware components that store and protect the software and firmware components. Hardware, software, and firmware components of a cryptographic module can be excluded from the requirements of this standard if shown that these components do not affect the security of the module.

The following documentation requirements shall apply to all security-specific hardware, software, and firmware contained within a cryptographic module. These requirements do not apply to microcode or system software whose source code is not available to the vendor or to any hardware, software, or firmware components that can be shown not to affect the security of the cryptographic module.

- Documentation shall specify the hardware, software, and firmware components of a cryptographic module, specify the cryptographic boundary surrounding these components, and describe the physical configuration of the module (see Section 4.5).

- Documentation shall specify any hardware, software, or firmware components of a cryptographic module that are excluded from the security requirements of this standard and explain the rationale for the exclusion.

- Documentation shall specify the physical ports and logical interfaces and all defined input and output data paths of a cryptographic module.

- Documentation shall specify the manual or logical controls of a cryptographic module, physical or logical status indicators, and applicable physical, logical, and electrical characteristics.

- Documentation shall list all security functions, both Approved and non-Approved, that are employed by a cryptographic module and shall specify all modes of operation, both Approved and non-Approved.

- Documentation shall specify:

  - □ a block diagram depicting all of the major hardware components of a cryptographic module and component interconnections, including any microprocessors, input/output buffers, plaintext/ciphertext buffers, control buffers, key storage, working memory, and program memory, and

  - □ the design of the hardware, software, and firmware components of a cryptographic module. High-level specification languages for software/firmware or schematics for hardware shall be used to document the design.

- Documentation shall specify all security-related information, including secret and private cryptographic keys (both plaintext and encrypted), authentication data (e.g., passwords, PINs), CSPs, and other protected information (e.g., audited events, audit data) whose disclosure or modification can compromise the security of the cryptographic module.

- Documentation shall specify a cryptographic module security policy. The security policy shall include the rules derived from the requirements of this standard and the rules derived from any additional requirements imposed by the vendor (see Appendix C).

4.2 Cryptographic Module Ports and Interfaces

A cryptographic module shall restrict all information flow and physical access points to physical ports and logical interfaces that define all entry and exit points to and from the module. The cryptographic module interfaces shall be logically distinct from each other although they may share one physical port (e.g., input data may enter and output data may exit via the same port) or may be distributed over one or more physical ports (e.g., input data may enter via both a serial and a parallel port). An Application Program Interface (API) of a software component of a cryptographic module may be defined as one or more logical interfaces(s).

A cryptographic module shall have the following four logical interfaces ("input" and "output" are indicated from the perspective of the module):

> *Data input interface.* All data (except control data entered via the control input interface) that is input to and processed by a cryptographic module (including plaintext data, ciphertext data, cryptographic keys and CSPs, authentication data, and status information from another module) shall enter via the "data input" interface.
>
> *Data output interface.* All data (except status data output via the status output interface) that is output from a cryptographic module (including plaintext data, ciphertext data, cryptographic keys and CSPs, authentication data, and control information for another module) shall exit via the "data output" interface. All data output via the data output interface shall be inhibited when an error state exists and during self-tests (see Section 4.9).
>
> *Control input interface.* All input commands, signals, and control data (including function calls and manual controls such as switches, buttons, and keyboards) used to control the operation of a cryptographic module shall enter via the "control input" interface.
>
> *Status output interface.* All output signals, indicators, and status data (including return codes and physical indicators such as Light Emitting Diodes and displays) used to indicate the status of a cryptographic module shall exit via the "status output" interface.

All external electrical power that is input to a cryptographic module (including power from an external power source or batteries) shall enter via a power port. A power port is not required when all power is provided or maintained internally to the cryptographic boundary of the cryptographic module (e.g., an internal battery).

The cryptographic module shall distinguish between data and control for input and data and status for output. All input data entering the cryptographic module via the "data input" interface shall only pass through the input data path. All output data exiting the cryptographic module via the "data output" interface shall only pass through the output data path. The output data path shall be logically disconnected from the circuitry and processes while performing key generation, manual key entry, or key zeroization. To prevent the inadvertent output of sensitive information, two independent internal actions shall be required to output data via any output interface through which plaintext cryptographic keys or CSPs or sensitive data are output (e.g., two different software flags are set, one of which may be user initiated; or two hardware gates are set serially from two separate actions).

SECURITY LEVELS 1 AND 2

For Security Levels 1 and 2, the physical port(s) and logical interface(s) used for the input and output of plaintext cryptographic keys, cryptographic key components, authentication data, and CSPs may be shared physically and logically with other ports and interfaces of the cryptographic module.

SECURITY LEVELS 3 AND 4

For Security Levels 3 and 4,

- the physical port(s) used for the input and output of plaintext cryptographic key components, authentication data, and CSPs shall be physically separated from all other ports of the cryptographic module

or

- the logical interfaces used for the input and output of plaintext cryptographic key components, authentication data, and CSPs shall be logically separated from all other interfaces using a trusted path,

and

- plaintext cryptographic key components, authentication data, and other CSPs shall be directly entered into the cryptographic module (e.g., via a trusted path or directly attached cable). (See Section 4.7.4.)

4.3 Roles, Services, and Authentication

A cryptographic module shall support authorized roles for operators and corresponding services within each role. Multiple roles may be assumed by a single operator. If a cryptographic module supports concurrent operators, then the module shall internally maintain the separation of the roles assumed by each operator and the corresponding services. An operator is not required to assume an authorized role to perform services where cryptographic keys and CSPs are not modified, disclosed, or substituted (e.g., *show status, self-tests,* or other services that do not affect the security of the module).

Authentication mechanisms may be required within a cryptographic module to authenticate an operator accessing the module, and to verify that the operator is authorized to assume the requested role and perform the services within the role.

4.3.1 Roles

A cryptographic module shall support the following authorized roles for operators:

> *User Role.* The role assumed to perform general security services, including cryptographic operations and other Approved security functions.
>
> *Crypto Officer Role*: The role assumed to perform cryptographic initialization or management functions (e.g., module initialization, input/output of cryptographic keys and CSPs, and audit functions).

If the cryptographic module allows operators to perform maintenance services, then the module shall support the following authorized role:

> *Maintenance Role:* The role assumed to perform physical maintenance and/or logical maintenance services (e.g., hardware/software diagnostics). All plaintext secret and private keys and unprotected CSPs shall be zeroized when entering or exiting the maintenance role.

A cryptographic module may support other roles or sub-roles in addition to the roles specified above.

Documentation shall specify all authorized roles supported by the cryptographic module.

4.3.2 Services

Services shall refer to all of the services, operations, or functions that can be performed by a cryptographic module. *Service inputs* shall consist of all data or control inputs to the cryptographic module that initiate or obtain specific services, operations, or functions. *Service outputs* shall consist of all data and status outputs that result from services, operations, or functions initiated or obtained by service inputs. Each service input shall result in a service output.

A cryptographic module shall provide the following services to operators:

> *Show Status*. Output the current status of the cryptographic module.
>
> *Perform Self-Tests*. Initiate and run the self-tests as specified in Section 4.9.
>
> *Perform Approved Security Function*. Perform at least one Approved security function used in an Approved mode of operation, as specified in Section 4.1.

A cryptographic module may provide other services, operations, or functions, both Approved and non-Approved, in addition to the services specified above. Specific services may be provided in more than one role (e.g., key entry services may be provided in the user role and the crypto officer role).

If a cryptographic module implements a *bypass* capability, where services are provided without cryptographic processing (e.g., transferring plaintext through the module without encryption), then

- two independent internal actions shall be required to activate the capability to prevent the inadvertent bypass of plaintext data due to a single error (e.g., two different software or hardware flags are set, one of which may be user-initiated), and

- the module shall show status to indicate whether

1) the bypass capability *is not* activated, and the module is exclusively providing services *with* cryptographic processing (e.g., plaintext data *is* encrypted),

2) the bypass capability *is* activated and the module is exclusively providing services *without* cryptographic processing (e.g., plaintext data *is not* encrypted), or

3) the bypass capability *is alternately* activated and deactivated and the module is providing some services *with* cryptographic processing and some services *without* cryptographic processing (e.g., for modules with multiple communication channels, plaintext data *is* or *is not* encrypted depending on each channel configuration).

Documentation shall specify:

- the services, operations, or functions provided by the cryptographic module, both Approved and non-Approved,

- for each service provided by the module, the service inputs, corresponding service outputs, and the authorized role(s) in which the service can be performed, and

- any services provided by the cryptographic module for which the operator is not required to assume an authorized role, and how these services do not modify, disclose, or substitute cryptographic keys and CSPs, or otherwise affect the security of the module.

4.3.3 Operator Authentication

Authentication mechanisms may be required within a cryptographic module to authenticate an operator accessing the module and to verify that the operator is authorized to assume the requested role and perform services within that role. Depending on the security level, a cryptographic module shall support at least one of the following mechanisms to control access to the module:

Role-Based Authentication: If role-based authentication mechanisms are supported by a cryptographic module, the module shall require that one or more roles either be implicitly or explicitly selected by the operator and shall authenticate the assumption of the selected role (or set of roles). The cryptographic module is not required to authenticate the individual identity of the operator. The selection of roles and the authentication of the assumption of selected roles may be combined. If a cryptographic module permits an operator to change roles, then the module shall authenticate the assumption of any role that was not previously authenticated.

Identity-Based Authentication: If identity-based authentication mechanisms are supported by a cryptographic module, the module shall require that the operator be individually identified, shall require that one or more roles either be implicitly or explicitly selected by the operator, and shall authenticate the identity of the operator and the authorization of the operator to assume the selected role (or set of roles). The authentication of the identity of the operator, selection of roles, and the authorization of the assumption of the selected roles may be combined. If a cryptographic module permits an operator to change roles, then the module shall verify the authorization of the identified operator to assume any role that was not previously authorized.

A cryptographic module may permit an authenticated operator to perform all of the services allowed within an authorized role, or may require separate authentication for each service or for different sets of services. When a cryptographic module is powered off and subsequently powered on, the results of previous authentications shall not be retained and the module shall require the operator to be re-authenticated.

Various types of authentication data may be required by a cryptographic module to implement the supported authentication mechanisms, including (but not limited to) the knowledge or possession of a password, PIN, cryptographic key, or equivalent; possession of a physical key, token, or equivalent; or verification of personal characteristics (e.g., biometrics). Authentication data within a cryptographic module shall be protected against unauthorized disclosure, modification, and substitution.

The initialization of authentication mechanisms may warrant special treatment. If a cryptographic module does not contain the authentication data required to authenticate the operator for the first time the module is accessed, then other authorized methods (e.g., procedural controls or use of factory-set or default authentication data) shall be used to control access to the module and initialize the authentication mechanisms.

The strength of the authentication mechanism shall conform to the following specifications:

- For each attempt to use the authentication mechanism, the probability shall be less than one in 1,000,000 that a random attempt will succeed or a false acceptance will occur (e.g., guessing a password or PIN, false acceptance error rate of a biometric device, or some combination of authentication methods).

- For multiple attempts to use the authentication mechanism during a one-minute period, the probability shall be less than one in 100,000 that a random attempt will succeed or a false acceptance will occur.

- Feedback of authentication data to an operator shall be obscured during authentication (e.g., no visible display of characters when entering a password).

- Feedback provided to an operator during an attempted authentication shall not weaken the strength of the authentication mechanism.

Documentation shall specify:

- the authentication mechanisms supported by a cryptographic module,

- the types of authentication data required by the module to implement the supported authentication mechanisms,

- the authorized methods used to control access to the module for the first time and initialize the authentication mechanisms, and

- the strength of the authentication mechanisms supported by the module.

SECURITY LEVEL 1

For Security Level 1, a cryptographic module is not required to employ authentication mechanisms to control access to the module. If authentication mechanisms are not supported by a cryptographic module, the module shall require that one or more roles either be implicitly or explicitly selected by the operator.

SECURITY LEVEL 2

For Security Level 2, a cryptographic module shall employ *role-based* authentication to control access to the module.

SECURITY LEVELS 3 AND 4

For Security Levels 3 and 4, a cryptographic module shall employ *identity-based* authentication mechanisms to control access to the module.

4.4 Finite State Model

The operation of a cryptographic module shall be specified using a finite state model (or equivalent) represented by a state transition diagram and/or a state transition table.

The state transition diagram and/or state transition table includes:

- all operational and error states of a cryptographic module,
- the corresponding transitions from one state to another,
- the input events that cause transitions from one state to another, and
- the output events resulting from transitions from one state to another.

A cryptographic module shall include the following operational and error states:

Power on/off states. States for primary, secondary, or backup power. These states may distinguish between power sources being applied to a cryptographic module.

Crypto officer states. States in which the crypto officer services are performed (e.g., cryptographic initialization and key management).

Key/CSP entry states. States for entering cryptographic keys and CSPs into the cryptographic module.

User states. States in which authorized users obtain security services, perform cryptographic operations, or perform other Approved or non-Approved functions.

Self-test states. States in which the cryptographic module is performing self-tests.

Error states. States when the cryptographic module has encountered an error (e.g., failed a self-test or attempted to encrypt when missing operational keys or CSPs). Error states may include "hard" errors that indicate an equipment malfunction and that may require maintenance, service or repair of the cryptographic module, or recoverable "soft" errors that may require initialization or resetting of the module. Recovery from error states shall be possible except for those caused by hard errors that require maintenance, service, or repair of the cryptographic module.

A cryptographic module may contain other states including, but not limited to, the following:

Bypass states. States in which a bypass capability is activated and services are provided without cryptographic processing (e.g., transferring plaintext through the cryptographic module).

Maintenance states. States for maintaining and servicing a cryptographic module, including physical and logical maintenance testing. If a cryptographic module contains a maintenance role, then a maintenance state shall be included.

Documentation shall include a representation of the finite state model (or equivalent) using a state transition diagram and/or state transition table that shall specify:

- all operational and error states of a cryptographic module,
- the corresponding transitions from one state to another,

- the input events, including data inputs and control inputs, that cause transitions from one state to another, and

- the output events, including internal module conditions, data outputs, and status outputs resulting from transitions from one state to another.

4.5 Physical Security

A cryptographic module shall employ physical security mechanisms in order to restrict unauthorized physical access to the contents of the module and to deter unauthorized use or modification of the module (including substitution of the entire module) when installed. All hardware, software, firmware, and data components within the cryptographic boundary shall be protected.

A cryptographic module that is implemented completely in software such that the physical security is provided solely by the host platform is not subject to the physical security requirements of this standard.

Physical security requirements are specified for three defined physical embodiments of a cryptographic module:

- ***Single-chip cryptographic modules*** are physical embodiments in which a single integrated circuit (IC) chip may be used as a standalone device or may be embedded within an enclosure or a product that may not be physically protected. Examples of single-chip cryptographic modules include single IC chips or smart cards with a single IC chip.

- ***Multiple-chip embedded cryptographic modules*** are physical embodiments in which two or more IC chips are interconnected and are embedded within an enclosure or a product that may not be physically protected. Examples of multiple-chip embedded cryptographic modules include adapters and expansion boards.

- ***Multiple-chip standalone cryptographic modules*** are physical embodiments in which two or more IC chips are interconnected and the entire enclosure is physically protected. Examples of multiple-chip, standalone cryptographic modules include encrypting routers or secure radios.

Depending on the physical security mechanisms of a cryptographic module, unauthorized attempts at physical access, use, or modification will have a high probability of being detected

- subsequent to an attempt by leaving visible signs (i.e., tamper evidence)

and/or

- during an attempt so that appropriate actions can be taken by the cryptographic module to protect plaintext secret and private keys and CSPs (i.e., tamper response).

Table 2 summarizes the physical security requirements, both general and embodiment-specific, for each of the four security levels. The general physical security requirements at each security level are all three distinct physical embodiments of a cryptographic module. The embodiment-specific physical security requirements at each security level enhance the general requirements at the same level, and the embodiment-specific requirements of the previous level.

	General Requirements for all Embodiments	Single-Chip Cryptographic Modules	Multiple-Chip Embedded Cryptographic Modules	Multiple-Chip Standalone Cryptographic Modules
Security Level 1	Production-grade components (with standard passivation).	No additional requirements.	If applicable, production-grade enclosure or removable cover.	Production-grade enclosure.
Security Level 2	Evidence of tampering (e.g., cover, enclosure, or seal).	Opaque tamper-evident coating on chip or enclosure.	Opaque tamper-evident encapsulating material or enclosure with tamper-evident seals or pick-resistant locks for doors or removable covers.	Opaque enclosure with tamper-evident seals or pick-resistant locks for doors or removable covers.
Security Level 3	Automatic zeroization when accessing the maintenance access interface. Tamper response and zeroization circuitry. Protected vents.	Hard opaque tamper-evident coating on chip or strong removal-resistant and penetration resistant enclosure.	Hard opaque potting material encapsulation of multiple chip circuitry embodiment or applicable Multiple-Chip Standalone Security Level 3 requirements.	Hard opaque potting material encapsulation of multiple chip circuitry embodiment or strong enclosure with removal/penetration attempts causing serious damage.
Security Level 4	EFP or EFT for temperature and voltage.	Hard opaque removal-resistant coating on chip.	Tamper detection envelope with tamper response and zeroization circuitry.	Tamper detection/ response envelope with tamper response and zeroization circuitry.

Table 2: *Summary of physical security requirements*

In general, Security Level 1 requires minimal physical protection. Security Level 2 requires the addition of tamper-evident mechanisms. Security Level 3 adds requirements for the use of strong enclosures with tamper detection and response mechanisms for removable covers and doors. Security Level 4 adds requirements for the use of strong enclosures with tamper detection and response mechanisms for the entire enclosure. Environmental failure protection (EFP) or environmental failure testing (EFT) is required at Security Level 4. Tamper detection and tamper response are not substitutes for tamper evidence.

Security requirements are specified for a maintenance access interface when a cryptographic module is designed to permit physical access (e.g., by the module vendor or other authorized individuals).

4.5.1 General Physical Security Requirements

The following requirements shall apply to all physical embodiments.

- Documentation shall specify the physical embodiment and the security level for which the physical security mechanisms of a cryptographic module are implemented.

- Documentation shall specify the physical security mechanisms of a cryptographic module.

- If a cryptographic module includes a maintenance role that requires physical access to the contents of the module or if the module is designed to permit physical access (e.g., by the module vendor or other authorized individual), then:

  □ a maintenance access interface shall be defined,

- ❑ the maintenance access interface shall include all physical access paths to the contents of the cryptographic module, including any removable covers or doors,
- ❑ any removable covers or doors included within the maintenance access interface shall be safeguarded using the appropriate physical security mechanisms,
- ❑ all plaintext secret and private keys and CSPs shall be zeroized when the maintenance access interface is accessed, and
- ❑ documentation shall specify the maintenance access interface and how plaintext secret and private keys and CSPs are zeroized when the maintenance access interface is accessed.

SECURITY LEVEL 1

The following requirements shall apply to all cryptographic modules for Security Level 1.

- The cryptographic module shall consist of production-grade components that shall include standard passivation techniques (e.g., a conformal coating or a sealing coat applied over the module's circuitry to protect against environmental or other physical damage).
- When performing physical maintenance, all plaintext secret and private keys and other unprotected CSPs contained in the cryptographic module shall be zeroized. Zeroization shall either be performed procedurally by the operator or automatically by the cryptographic module.

SECURITY LEVEL 2

In addition to the general requirements for Security Level 1, the following requirement shall apply to all cryptographic modules for Security Level 2.

- The cryptographic module shall provide evidence of tampering (e.g., on the cover, enclosure, and seal) when physical access to the module is attempted.

SECURITY LEVEL 3

In addition to the general requirements for Security Levels 1 and 2, the following requirements shall apply to all cryptographic modules for Security Level 3.

- If the cryptographic module contains any doors or removable covers or if a maintenance access interface is defined, then the module shall contain tamper response and zeroization circuitry. The tamper response and zeroization circuitry shall immediately zeroize all plaintext secret and private keys and CSPs when a door is opened, a cover is removed, or when the maintenance access interface is accessed. The tamper response and zeroization circuitry shall remain operational when plaintext secret and private cryptographic keys or CSPs are contained within the cryptographic module.
- If the cryptographic module contains ventilation holes or slits, then the holes or slits shall be constructed in a manner that prevents undetected physical probing inside the enclosure (e.g., require at least one 90 degree bend or obstruction with a substantial blocking material).

SECURITY LEVEL 4

In addition to the general requirements for Security Levels 1, 2, and 3, the following requirement shall apply to all cryptographic modules for Security Level 4.

- The cryptographic module shall either include environmental failure protection (EFP) features or undergo environmental failure testing (EFT) as specified in Section 4.5.5.

4.5.2 Single-Chip Cryptographic Modules

In addition to the general security requirements specified in Section 4.5.1, the following requirements are specific to single-chip cryptographic modules.

SECURITY LEVEL 1

There are no additional Security Level 1 requirements for single-chip cryptographic modules.

SECURITY LEVEL 2

In addition to the requirements for Security Level 1, the following requirements shall apply to single-chip cryptographic modules for Security Level 2.

- The cryptographic module shall be covered with a tamper-evident coating (e.g., a tamper-evident passivation material or a tamper-evident material covering the passivation) or contained in a tamper-evident enclosure to deter direct observation, probing, or manipulation of the module and to provide evidence of attempts to tamper with or remove the module.
- The tamper-evident coating or tamper-evident enclosure shall be opaque within the visible spectrum.

SECURITY LEVEL 3

In addition to the requirements for Security Levels 1 and 2, the following requirements shall apply to single-chip cryptographic modules for Security Level 3.

Either

- the cryptographic module shall be covered with a hard opaque tamper-evident coating (e.g., a hard opaque epoxy covering the passivation)

or

- the enclosure shall be implemented so that attempts at removal or penetration of the enclosure shall have a high probability of causing serious damage to the cryptographic module (i.e., the module will not function).

SECURITY LEVEL 4

In addition to the requirements for Security Levels 1, 2, and 3, the following requirements shall apply to single-chip cryptographic modules for Security Level 4.

- The cryptographic module shall be covered with a hard, opaque removal-resistant coating with hardness and adhesion characteristics such that attempting to peel or pry the coating from the module will have a high probability of resulting in serious damage to the module (i.e., the module will not function).

- The removal-resistant coating shall have solvency characteristics such that dissolving the coating will have a high probability of dissolving or seriously damaging the module (i.e., the module will not function).

4.5.3 Multiple-Chip Embedded Cryptographic Modules

In addition to the general security requirements specified in Section 4.5.1, the following requirements are specific to multiple-chip embedded cryptographic modules.

SECURITY LEVEL 1

The following requirement shall apply to multiple-chip embedded cryptographic modules for Security Level 1.

- If the cryptographic module is contained within an enclosure or removable cover, a production-grade enclosure or removable cover shall be used.

SECURITY LEVEL 2

In addition to the requirement for Security Level 1, the following requirements shall apply to multiple-chip embedded cryptographic modules for Security Level 2.

Either

- the cryptographic module components shall be covered with a tamper-evident coating or potting material (e.g., etch-resistant coating or bleeding paint) or contained in a tamper-evident enclosure to deter direct observation, probing, or manipulation of module components and to provide evidence of attempts to tamper with or remove module components, and

- the tamper-evident coating or tamper-evident enclosure shall be opaque within the visible spectrum,

or

- the cryptographic module shall be entirely contained within a metal or hard plastic production-grade enclosure that may include doors or removable covers,

- the enclosure shall be opaque within the visible spectrum, and

- if the enclosure includes any doors or removable covers, then the doors or covers shall be locked with pick-resistant mechanical locks employing physical or logical keys or shall be protected with tamper-evident seals (e.g., evidence tape or holographic seals).

SECURITY LEVEL 3

In addition to the requirements for Security Levels 1 and 2, the following requirements shall apply to multiple-chip embedded cryptographic modules for Security Level 3.

Either

- the multiple-chip embodiment of the circuitry within the cryptographic module shall be covered with a hard coating or potting material (e.g., a hard epoxy material) that is opaque within the visible spectrum

or

- the applicable Security Level 3 requirements for multiple-chip standalone cryptographic modules shall apply. (Section 4.5.4)

SECURITY LEVEL 4

In addition to the requirements for Security Levels 1, 2, and 3, the following requirements shall apply to multiple-chip embedded cryptographic modules for Security Level 4.

- The cryptographic module components shall be covered by potting material or contained within an enclosure encapsulated by a tamper detection envelope (e.g., a flexible mylar printed circuit with a serpentine geometric pattern of conductors or a wire-wound package or a non-flexible, brittle circuit or a strong enclosure) that shall detect tampering by means such as cutting, drilling, milling, grinding, or dissolving of the potting material or enclosure to an extent sufficient for accessing plaintext secret and private keys cryptographic keys or CSPs.

- The cryptographic module shall contain tamper response and zeroization circuitry that shall continuously monitor the tamper detection envelope and, upon the detection of tampering, shall immediately zeroize all plaintext secret and private cryptographic keys and CSPs. The tamper response and zeroization circuitry shall remain operational when plaintext secret and private cryptographic keys or CSPs are contained within the cryptographic module.

4.5.4 Multiple-Chip Standalone Cryptographic Modules

In addition to the general security requirements specified in Section 4.5.1, the following requirements are specific to multiple-chip standalone cryptographic modules.

SECURITY LEVEL 1

The following requirement shall apply to multiple-chip standalone cryptographic modules for Security Level 1.

- The cryptographic module shall be entirely contained within a metal or hard plastic production-grade enclosure that may include doors or removable covers.

SECURITY LEVEL 2

In addition to the requirements for Security Level 1, the following requirements shall apply to multiple-chip standalone cryptographic modules for Security Level 2.

- The enclosure of the cryptographic module shall be opaque within the visible spectrum.

- If the enclosure of the cryptographic module includes any doors or removable covers, then the doors or covers shall be locked with pick-resistant mechanical locks employing physical or logical keys or shall be protected with tamper-evident seals (e.g., evidence tape or holographic seals).

SECURITY LEVEL 3

In addition to the requirements for Security Levels 1 and 2, the following requirements shall apply to multiple-chip standalone cryptographic modules for Security Level 3.

Either

- the multiple-chip embodiment of the circuitry within the cryptographic module shall be covered with a hard potting material (e.g., a hard epoxy material) that is opaque within the visible spectrum

or

- the cryptographic module shall be contained within a strong enclosure such that attempts at removal or penetration of the enclosure will have a high probability of causing serious damage to the module (i.e., the module will not function).

SECURITY LEVEL 4

In addition to the requirements for Security Levels 1, 2, and 3, the following requirements shall apply to multiple-chip standalone cryptographic modules for Security Level 4.

- The potting material or enclosure of the cryptographic module shall be encapsulated by a tamper detection envelope, by the use of tamper detection mechanisms such as cover switches (e.g., microswitches, magnetic Hall effect switches, permanent magnetic actuators, etc.), motion detectors (e.g., ultrasonic, infrared, or microwave), or other tamper detection mechanisms as described above for multiple-chip embedded cryptographic modules. The tamper detection mechanisms shall detect tampering by means such as cutting, drilling, milling, grinding, or dissolving of the potting material or enclosure, to an extent sufficient for accessing plaintext secret and private cryptographic keys and CSPs.
- The cryptographic module shall contain tamper response and zeroization circuitry that shall continuously monitor the tamper detection envelope and, upon the detection of tampering, shall immediately zeroize all plaintext secret and private cryptographic keys and CSPs. The tamper response and zeroization circuitry shall remain operational when plaintext cryptographic keys and CSPs are contained within the cryptographic module.

4.5.5 Environmental Failure Protection/Testing

The electronic devices and circuitry are designed to operate within a particular range of environmental conditions. Deliberate or accidental excursions outside the specified normal operating ranges of voltage and temperature can cause erratic operation or failure of the electronic devices or circuitry that can compromise the security of the cryptographic module. Reasonable assurance that the security of a cryptographic module cannot be compromised by extreme environmental conditions can be provided by having the module employ environmental failure protection (EFP) features or undergo environmental failure testing (EFT).

For Security Levels 1, 2, and 3, a cryptographic module is not required to employ environmental failure protection (EFP) features or undergo environmental failure testing (EFT). At Security Level 4, a cryptographic module shall either employ environmental failure protection (EFP) features or undergo environmental failure testing (EFT).

4.5.5.1 Environmental Failure Protection Features (Alternative 1)

Environmental failure protection (EFP) features shall protect a cryptographic module against unusual environmental conditions or fluctuations (accidental or induced) outside of the module's normal operating range that can compromise the security of the module. In particular, the cryptographic module shall monitor and correctly respond to fluctuations in the operating *temperature* and *voltage* outside of the specified normal operating ranges.

The EFP features shall involve electronic circuitry or devices that continuously measure the operating temperature and voltage of a cryptographic module. If the temperature or voltage fall outside of the cryptographic module's normal operating range, the protection circuitry shall either (1) shutdown the module to prevent further operation or (2) immediately zeroize all plaintext secret and private cryptographic keys and CSPs.

Documentation shall specify the normal operating ranges of a cryptographic module and the environmental failure protection features employed by the module.

4.5.5.2 Environmental Failure Testing Procedures (Alternative 2)

Environmental failure testing (EFT) shall involve a combination of analysis, simulation, and testing of a cryptographic module to provide reasonable assurance that environmental conditions or fluctuations (accidental or induced) outside the module's normal operating ranges for temperature and voltage will not compromise the security of the module.

EFT shall demonstrate that, if the operating temperature or voltage falls outside the normal operating range of the cryptographic module resulting in a failure of the electronic devices or circuitry within the module, at no time shall the security of the cryptographic module be compromised.

The temperature range to be tested shall be from -100° to +200° Celsius (-150° to +400° Fahrenheit). The voltage range to be tested shall be from the smallest negative voltage (with respect to ground) that causes the zeroization of the electronic devices or circuitry to the smallest positive voltage (with respect to ground) that causes the zeroization of the electronic devices or circuitry, including reversing the polarity of the voltages.

Documentation shall specify the normal operating ranges of the cryptographic module and the environmental failure tests performed.

4.6 Operational Environment

The *operational environment* of a cryptographic module refers to the management of the software, firmware, and/or hardware components required for the module to operate. The operational environment can be non-modifiable (e.g., firmware contained in ROM, or software contained in a computer with I/O devices disabled), or modifiable (e.g., firmware contained in RAM or software executed by a general purpose computer). An operating system is an important component of the operating environment of a cryptographic module.

A *general purpose operational environment* refers to the use of a commercially-available general purpose operating system (i.e., resource manager) that manages the software and firmware components within the cryptographic boundary, and also manages system and operator(s) processes/thread(s), including general-purpose application software such as word processors.

A *limited operational environment* refers to a static non-modifiable virtual operational environment (e.g., JAVA virtual machine on a non-programmable PC card) with no underlying general purpose operating system upon which the operational environment uniquely resides.

A *modifiable operational environment* refers to an operating environment that *may* be reconfigured to add/delete/modify functionality, and/or *may* include general purpose operating system capabilities (e.g., use of a computer O/S, configurable smart card O/S, or programmable firmware). Operating systems are considered to be modifiable operational environments if software/firmware components can be modified by the operator and/or the operator can load and execute software or firmware (e.g., a word processor) that was not included as part of the validation of the module.

If the operational environment is a modifiable operational environment, the operating system requirements in Section 4.6.1 shall apply. If the operational environment is a limited operational environment, the operating system requirements in Section 4.6.1 do not apply.

Documentation shall specify the operational environment for a cryptographic module, including, if applicable, the operating system employed by the module, and for Security Levels 2, 3, and 4, the Protection Profile and the CC assurance level.

4.6.1 Operating System Requirements

SECURITY LEVEL 1

The following requirements shall apply to operating systems for Security Level 1.

- For Security Level 1 only, the operating system shall be restricted to a single operator mode of operation (i.e., concurrent operators are explicitly excluded).

- For Security Level 1 only, the cryptographic module shall prevent access by other processes to plaintext private and secret keys, CSPs, and intermediate key generation values during the time the cryptographic module is executing/operational. Processes that are spawned by the cryptographic module are owned by the module and are not owned by external processes/operators. Non-cryptographic processes shall not interrupt a cryptographic module during execution.

- All cryptographic software and firmware shall be installed in a form that protects the software and firmware source and executable code from unauthorized disclosure and modification.

- A cryptographic mechanism using an Approved integrity technique (e.g., an Approved message authentication code or digital signature algorithm) shall be applied to all cryptographic software and firmware components within the cryptographic module. This cryptographic mechanism requirement may be incorporated as part of the Software/Firmware Integrity Test (Section 4.9.1) if an Approved authentication technique is employed for that test.

SECURITY LEVEL 2

In addition to the applicable requirements for Security Level 1, the following requirements shall also apply for Security Level 2.

- All cryptographic software and firmware, cryptographic keys and CSPs, and control and status information shall be under the control of

  - ❑ an operating system that meets the functional requirements specified in the Protection Profiles listed in Annex B and is evaluated at the CC evaluation assurance level EAL2, or

  - ❑ an equivalent evaluated trusted operating system.

- To protect plaintext data, cryptographic software and firmware, cryptographic keys and CSPs, and authentication data, the discretionary access control mechanisms of the operating system shall be configured to:

  - ❑ Specify the set of roles that can *execute* stored cryptographic software and firmware.

  - ❑ Specify the set of roles that can *modify* (i.e., write, replace, and delete) the following cryptographic module software or firmware components stored within the cryptographic boundary: cryptographic programs, cryptographic data (e.g., cryptographic keys and audit data), CSPs, and plaintext data.

  - ❑ Specify the set of roles that can *read* the following cryptographic software components stored within the cryptographic boundary: cryptographic data (e.g., cryptographic keys and audit data), CSPs, and plaintext data.

  - ❑ Specify the set of roles that can *enter* cryptographic keys and CSPs.

- The operating system shall prevent all operators and executing processes from modifying executing cryptographic processes (i.e., loaded and executing cryptographic program images). In this case, executing processes refer to all non-operating system processes (i.e., operator-initiated), cryptographic or not.

- The operating system shall prevent operators and executing processes from reading cryptographic software stored within the cryptographic boundary.

- The operating system shall provide an audit mechanism to record modifications, accesses, deletions, and additions of cryptographic data and CSPs.

  - ❑ The following events shall be recorded by the audit mechanism:

    - attempts to provide invalid input for crypto officer functions, and
    - the addition or deletion of an operator to/from a crypto officer role.

  - ❑ The audit mechanism shall be capable of auditing the following events:

    - operations to process audit data stored in the audit trail,
    - requests to use authentication data management mechanisms,
    - use of a security-relevant crypto officer function,
    - requests to access user authentication data associated with the cryptographic module,
    - use of an authentication mechanism (e.g., login) associated with the cryptographic module,
    - explicit requests to assume a crypto officer role, and
    - the allocation of a function to a crypto officer role.

SECURITY LEVEL 3

In addition to the applicable requirements for Security Levels 1 and 2, the following requirements shall apply for Security Level 3.

- All cryptographic software and firmware, cryptographic keys and CSPs, and control and status information shall be under the control of

  - ❑ an operating system that meets the functional requirements specified in the Protection Profiles listed in Annex B. The operating system shall be evaluated at the CC evaluation assurance level EAL3 and include the following additional requirements: Trusted Path (FTP_TRP.1) and Informal TOE Security Policy Model (ADV_SPM.1), or

  - ❑ an equivalent evaluated trusted operating system.

- All cryptographic keys and CSPs, authentication data, control inputs, and status outputs shall be communicated via a trusted mechanism (e.g., a dedicated I/O physical port or a trusted path). If a trusted path is used, the Target of Evaluation Security Functions (TSF) shall support the trusted path between the TSF and the operator when a positive TSF-to-operator connection is required. Communications via this trusted path shall be activated exclusively by an operator or the TSF and shall be logically isolated from other paths.

- In addition to the audit requirements of Security Level 2, the following events shall be recorded by the audit mechanism:

  - ❑ attempts to use the trusted path function, and

  - ❑ identification of the initiator and target of a trusted path.

SECURITY LEVEL 4

In addition to the applicable requirements for Security Levels 1, 2, and 3, the following requirements shall also apply to operating systems for Security Level 4.

- All cryptographic software, cryptographic keys and CSPs, and control and status information shall be under the control of

  □ an operating system that meets the functional requirements specified in the Protection Profiles listed in Annex B. The operating system shall be evaluated at the CC evaluation assurance level EAL4, or

  □ an equivalent evaluated trusted operating system.

4.7 Cryptographic Key Management

The security requirements for cryptographic key management encompass the entire lifecycle of cryptographic keys, cryptographic key components, and CSPs employed by the cryptographic module. Key management includes random number and key generation, key establishment, key distribution, key entry/output, key storage, and key zeroization. A cryptographic module may also employ the key management mechanisms of another cryptographic module. Encrypted cryptographic keys and CSPs refer to keys and CSPs that are encrypted using an Approved algorithm or Approved security function. Cryptographic keys and CSPs encrypted using a non-Approved algorithm or proprietary algorithm or method are considered in plaintext form, within the scope of this standard

Secret keys, private keys, and CSPs shall be protected within the cryptographic module from unauthorized disclosure, modification, and substitution. Public keys shall be protected within the cryptographic module against unauthorized modification and substitution.

Documentation shall specify all cryptographic keys, cryptographic key components, and CSPs employed by a cryptographic module.

4.7.1 Random Number Generators (RNGs)

A cryptographic module may employ random number generators (RNGs). If a cryptographic module employs Approved or non-Approved RNGs in an Approved mode of operation, the data output from the RNG shall pass the continuous random number generator test as specified in Section 4.9.2. Approved RNGs shall be subject to the cryptographic algorithm test in Section 4.9.1. Approved RNGs are listed in Annex C to this standard.

Until such time as an Approved nondeterministic RNG standard exists, nondeterministic RNGs approved for use in classified applications may be used for key generation or to seed Approved deterministic RNGs used in key generation. Commercially available nondeterministic RNGs may be used for the purpose of generating seeds for Approved deterministic RNGs. Nondeterministic RNGs shall comply with all applicable RNG requirements of this standard.

An Approved RNG shall be used for the generation of cryptographic keys used by an Approved security function. The output from a non-Approved RNG may be used 1) as input (e.g., seed, and seed key) to an Approved deterministic RNG or 2) to generate initialization vectors (IVs) for Approved security function(s). The seed and seed key shall not have the same value.

Documentation shall specify each RNG (Approved and non-Approved) employed by a cryptographic module.

4.7.2 Key Generation

A cryptographic module may generate cryptographic keys internally. Cryptographic keys generated by the cryptographic module for use by an Approved algorithm or security function shall be generated using an Approved key generation method. Approved key generation methods are listed in Annex C to this standard. If an Approved key generation method requires input from a RNG, then an Approved RNG that meets the requirements specified in Section 4.7.1 shall be used.

Compromising the security of the key generation method (e.g., guessing the seed value to initialize the deterministic RNG) shall require as least as many operations as determining the value of the generated key.

If a seed key is entered during the key generation process, entry of the key shall meet the key entry requirements specified in Section 4.7.4. If intermediate key generation values are output from the cryptographic module, the values shall be output either 1) in encrypted form or 2) under split knowledge procedures.

Documentation shall specify each of the key generation methods (Approved and non-Approved) employed by a cryptographic module.

4.7.3 Key Establishment

Key establishment may be performed by automated methods (e.g., use of a public key algorithm), manual methods (use of a manually-transported key loading device), or a combination of automated and manual methods. If key establishment methods are employed by a cryptographic module, only Approved key establishment methods shall be used. Approved key establishment methods are listed in Annex D to this standard.

If, in lieu of an Approved key establishment method, a radio communications cryptographic module implements Over-The-Air-Rekeying (OTAR), it shall be implemented as specified in the TIA/EIA Telecommunications Systems Bulletin, APCO Project 25, *Over-The-Air-Rekeying (OTAR) Protocol*, New Technology Standards Project, Digital Radio Technical Standards, TSB102.AACA, January, 1996, Telecommunications Industry Association.

Compromising the security of the key establishment method (e.g., compromising the security of the algorithm used for key establishment) shall require at least as many operations as determining the value of the cryptographic key being transported or agreed upon.

If a *key transport* method is used, the cryptographic key being transported shall meet the key entry/output requirements of Section 4.7.4. If a *key agreement* method is used (e.g., a cryptographic key is derived from shared intermediate values), the shared values are not required to meet the key entry/output requirements of Section 4.7.4.

Documentation shall specify the key establishment methods employed by a cryptographic module.

4.7.4 Key Entry and Output

Cryptographic keys may be entered into or output from a cryptographic module. If cryptographic keys are entered into or output from a cryptographic module, the entry or output of keys shall be performed using either manual (e.g., via a keyboard) or electronic methods (e.g., smart cards/tokens, PC cards, or other electronic key loading devices).

A seed key, if entered during key generation, shall be entered in the same manner as cryptographic keys.

All encrypted secret and private keys, entered into or output from a cryptographic module and used in an Approved mode of operation, shall be encrypted using an Approved algorithm. Public keys may be entered into or output from a cryptographic module in plaintext form. A cryptographic module shall associate a key (secret, private, or public) entered into or output from the module with the correct entity (i.e., person, group, or process) to which the key is assigned.

Manually-entered cryptographic keys (keys entered using manual methods) shall be verified during entry into a cryptographic module for accuracy using the manual key entry test specified in Section 4.9.2. During key entry, the manually entered values may be temporarily displayed to allow visual verification and to improve accuracy. If encrypted cryptographic keys or key components are manually entered into the cryptographic module, then the plaintext values of the cryptographic keys or key components shall not be displayed.

Documentation shall specify the key entry and output methods employed by a cryptographic module.

SECURITY LEVELS 1 AND 2

For Security Levels 1 and 2, secret and private keys established using *automated methods* shall be entered into and output from a cryptographic module in encrypted form. Secret and private keys established using *manual methods* may be entered into or output from a cryptographic module in plaintext form.

SECURITY LEVELS 3 AND 4

For Security Levels 3 and 4:

- Secret and private keys established using *automated methods* shall be entered into and output from a cryptographic module in encrypted form.

- Secret and private keys established using *manual methods* shall be entered into or output from a cryptographic module either (1) in encrypted form or (2) using split knowledge procedures (i.e., as two or more plaintext cryptographic key components).

  If split knowledge procedures are used:

  □ the cryptographic module shall separately authenticate the operator entering or outputting each key component,

  □ plaintext cryptographic key components shall be directly entered into or output from the cryptographic module (e.g., via a trusted path or directly attached cable) without traveling through any enclosing or intervening systems where the key components may inadvertently be stored, combined, or otherwise processed (see Section 4.2),

  □ at least two key components shall be required to reconstruct the original cryptographic key,

  □ documentation shall prove that if knowledge of *n* key components is required to reconstruct the original key, then knowledge of any *n*-1 key components provides no information about the original key other than the length, and

  □ documentation shall specify the procedures employed by a cryptographic module.

4.7.5 Key Storage

Cryptographic keys stored within a cryptographic module shall be stored either in plaintext form or encrypted form. Plaintext secret and private keys shall not be accessible from outside the cryptographic module to unauthorized operators.

A cryptographic module shall associate a cryptographic key (secret, private, or public) stored within the module with the correct entity (e.g., person, group, or process) to which the key is assigned.

Documentation shall specify the key storage methods employed by a cryptographic module.

4.7.6 Key Zeroization

A cryptographic module shall provide methods to zeroize all plaintext secret and private cryptographic keys and CSPs within the module. Zeroization of encrypted cryptographic keys and CSPs or keys otherwise physically or logically protected within an additional embedded validated module (meeting the requirements of this standard) is not required.

Documentation shall specify the key zeroization methods employed by a cryptographic module.

4.8 Electromagnetic Interference/Electromagnetic Compatibility (EMI/EMC)

Cryptographic modules shall meet the following requirements for EMI/EMC. Radios are explicitly excluded from these requirements but shall meet all applicable FCC requirements.

Documentation shall include proof of conformance to EMI/EMC requirements.

SECURITY LEVELS 1 AND 2

For Security Levels 1 and 2, a cryptographic module shall (at a minimum) conform to the EMI/EMC requirements specified by 47 Code of Federal Regulations, Part 15, Subpart B, Unintentional Radiators, Digital Devices, Class A (i.e., for business use).

SECURITY LEVELS 3 AND 4

For Security Levels 3 and 4, a cryptographic module shall (at a minimum) conform to the EMI/EMC requirements specified by 47 Code of Federal Regulations, Part 15, Subpart B, Unintentional Radiators, Digital Devices, Class B (i.e., for home use).

4.9 Self-Tests

A cryptographic module shall perform power-up self-tests and conditional self-tests to ensure that the module is functioning properly. *Power-up self-tests* shall be performed when the cryptographic module is powered up. *Conditional self-tests* shall be performed when an applicable security function or operation is invoked (i.e., security functions for which self-tests are required). A cryptographic module may perform other power-up or conditional self-tests in addition to the tests specified in this standard.

If a cryptographic module fails a self-test, the module shall enter an error state and output an error indicator via the status output interface. The cryptographic module shall not perform any cryptographic operations while in an error state. All data output via the data output interface shall be inhibited when an error state exists.

Documentation shall specify:

- the self-tests performed by a cryptographic module, including power-up and conditional tests,
- the error states that a cryptographic module can enter when a self-test fails, and
- the conditions and actions necessary to exit the error states and resume normal operation of a cryptographic module (i.e., this may include maintenance of the module, or returning the module to the vendor for servicing.)

4.9.1 Power-Up Tests

Power-up tests shall be performed by a cryptographic module when the module is powered up (after being powered off, reset, rebooted, etc.). The power-up tests shall be initiated automatically and shall not require operator intervention. When the power-up tests are completed, the results (i.e., indications of success or failure) shall be output via the "status output" interface. All data output via the data output interface shall be inhibited when the power-up tests are performed.

In addition to performing the power-up tests when powered up, a cryptographic module shall permit operators to initiate the tests on demand for periodic testing of the module. Resetting, rebooting, and power cycling are acceptable means for the on-demand initiation of power-up tests.

A cryptographic module shall perform the following power-up tests: cryptographic algorithm test, software/firmware integrity test, and critical functions test.

Cryptographic algorithm test. A cryptographic algorithm test using a known answer shall be conducted for all cryptographic functions (e.g., encryption, decryption, authentication, and random number generation) of each Approved cryptographic algorithm implemented by a cryptographic module. A known-answer test involves operating the cryptographic algorithm on data for which the correct output is already known and comparing the calculated output with the previously generated output (the known answer). If the calculated output does not equal the known answer, the known-answer test shall fail.

Cryptographic algorithms whose outputs vary for a given set of inputs (e.g., the Digital Signature Algorithm) shall be tested using a known-answer test or shall be tested using a pair-wise consistency test (specified below). Message digest algorithms shall have an independent known-answer test or the known-answer test shall be included with the associated cryptographic algorithm test (e.g., the Digital Signature Standard).

If a cryptographic module includes two independent implementations of the same cryptographic algorithm, then:

- the known-answer test may be omitted,
- the outputs of two implementations shall be continuously compared, and
- if the outputs of two implementations are not equal, the cryptographic algorithm test shall fail.

Software/firmware integrity test. A software/firmware integrity test using an error detection code (EDC) or Approved authentication technique (e.g., an Approved message authentication code or digital signature algorithm) shall be applied to all validated software and firmware components within a cryptographic module when the module is powered up. The software/firmware integrity test is not required for any software and firmware components excluded from the security requirements of this standard (refer to

Section 4.1). If the calculated result does not equal the previously generated result, the software/firmware test shall fail.

If an EDC is used, the EDC shall be at least 16 bits in length.

Critical functions test. Other security functions critical to the secure operation of a cryptographic module shall be tested when the module is powered up as part of the power-up tests. Other critical security functions performed under specific conditions shall be tested as conditional tests.

Documentation shall specify all security functions critical to the secure operation of a cryptographic module and shall identify the applicable power-up tests and conditional tests performed by the module.

4.9.2 Conditional Tests

Conditional tests shall be performed by a cryptographic module when the conditions specified for the following tests occur: pair-wise consistency test, software/firmware load test, manual key entry test, continuous random number generator test, and bypass test.

Pair-wise consistency test (for public and private keys). If a cryptographic module generates public or private keys, then the following pair-wise consistency tests for public and private keys shall be performed:

1. If the keys are used to perform an approved key transport method, then the public key shall encrypt a plaintext value. The resulting ciphertext value shall be compared to the original plaintext value. If the two values are equal, then the test shall fail. If the two values differ, then the private key shall be used to decrypt the ciphertext and the resulting value shall be compared to the original plaintext value. If the two values are not equal, the test shall fail.

2. If the keys are used to perform the calculation and verification of digital signatures, then the consistency of the keys shall be tested by the calculation and verification of a digital signature. If the digital signature cannot be verified, the test shall fail.

Software/firmware load test. If software or firmware components can be externally loaded into a cryptographic module, then the following software/firmware load tests shall be performed:

1. An Approved authentication technique (e.g., an Approved message authentication code, digital signature algorithm, or HMAC) shall be applied to all validated software and firmware components when the components are externally loaded into a cryptographic module. The software/firmware load test is not required for any software and firmware components excluded from the security requirements of this standard (refer to Section 4.1).

2. The calculated result shall be compared with a previously generated result. If the calculated result does not equal the previously generated result, the software/firmware load test shall fail.

Manual key entry test. If cryptographic keys or key components are manually entered into a cryptographic module, then the following manual key entry tests shall be performed:

1. The cryptographic key or key components shall have an EDC applied, or shall be entered using duplicate entries.

2. If an EDC is used, the EDC shall be at least 16 bits in length.

3. If the EDC cannot be verified, or the duplicate entries do not match, the test shall fail.

Continuous random number generator test. If a cryptographic module employs Approved or non-Approved RNGs in an Approved mode of operation, the module shall perform the following continuous random number generator test on each RNG that tests for failure to a constant value.

1. If each call to a RNG produces blocks of n bits (where $n > 15$), the first n-bit block generated after power-up, initialization, or reset shall not be used, but shall be saved for comparison with the next n-bit block to be generated. Each subsequent generation of an n-bit block shall be compared with the previously generated block. The test shall fail if any two compared n-bit blocks are equal.

2. If each call to a RNG produces fewer than 16 bits, the first n bits generated after power-up, initialization, or reset (for some $n > 15$) shall not be used, but shall be saved for comparison with the next n generated bits. Each subsequent generation of n bits shall be compared with the previously generated n bits. The test fails if any two compared n-bit sequences are equal.

Bypass test. If a cryptographic module implements a *bypass* capability where the services may be provided without cryptographic processing (e.g., transferring plaintext through the module), then the following bypass tests shall be performed to ensure that a single point of failure of module components will not result in the unintentional output of plaintext:

1. A cryptographic module shall test for the correct operation of the services providing cryptographic processing when a switch takes place between an exclusive bypass service and an exclusive cryptographic service.

2. If a cryptographic module can automatically alternate between a bypass service and a cryptographic service, providing some services *with* cryptographic processing and some services *without* cryptographic processing, then the module shall test for the correct operation of the services providing cryptographic processing when the mechanism governing the switching procedure is modified (e.g., an IP address source/destination table).

Documentation shall specify the mechanism or logic governing the switching procedure.

4.10 Design Assurance

Design assurance refers to the use of best practices by the vendor of a cryptographic module during the design, deployment, and operation of a cryptographic module, providing assurance that the module is properly tested, configured, delivered, installed, and developed, and that the proper operator guidance documentation is provided. Security requirements are specified for configuration management, delivery and operation, development, and guidance documents.

4.10.1 Configuration Management

Configuration management specifies the security requirements for a configuration management system implemented by a cryptographic module vendor, providing assurance that the functional requirements and specifications are realized in the implementation.

A configuration management system shall be implemented for a cryptographic module and module components within the cryptographic boundary, and for associated module documentation. Each version of each configuration item (e.g., cryptographic module, module components, user guidance, security policy, and operating system) that comprises the module and associated documentation shall be assigned and labeled with a unique identification number.

4.10.2 Delivery and Operation

Delivery and operation specifies the security requirements for the secure delivery, installation, and startup of a cryptographic module, providing assurance that the module is securely delivered to authorized operators, and is installed and initialized in a correct and secure manner.

SECURITY LEVEL 1

For Security Level 1, documentation shall specify the procedures for secure installation, initialization, and startup of a cryptographic module.

SECURITY LEVELS 2, 3, AND 4

For Security Levels 2, 3, and 4, in addition to the requirements of Security Level 1, documentation shall specify the procedures required for maintaining security while distributing and delivering versions of a cryptographic module to authorized operators.

4.10.3 Development

Development specifies the security requirements for the representation of a cryptographic module security functionality at various levels of abstraction from the functional interface to the implementation representation. Development provides assurance that the implementation of a cryptographic module corresponds to the module security policy and functional specification.

Functional specification refers to a high-level description of the ports and interfaces visible to the operator and a high-level description of the behavior of the cryptographic module.

SECURITY LEVEL 1

The following requirements shall apply to cryptographic modules for Security Level 1.

- Documentation shall specify the correspondence between the design of the hardware, software, and firmware components of a cryptographic module and the cryptographic module security policy (see Section 4.1).

- If a cryptographic module contains software or firmware components, documentation shall specify the source code for the software and firmware components, annotated with comments that clearly depict the correspondence of the components to the design of the module.

- If a cryptographic module contains hardware components, documentation shall specify the schematics and/or Hardware Description Language (HDL) listings for the hardware components.

SECURITY LEVEL 2

In addition to the requirements for Security Level 1, the following requirement shall apply to cryptographic modules for Security Level 2.

- Documentation shall specify a functional specification that informally describes a cryptographic module, the external ports and interfaces of the module, and the purpose of the interfaces.

SECURITY LEVEL 3

In addition to the requirements for Security Levels 1 and 2, the following requirements shall apply to cryptographic modules for Security Level 3.

- All software and firmware components within a cryptographic module shall be implemented using a high-level language, except that the limited use of a low-level language (e.g., assembly language or microcode) is allowed if essential to the performance of the module or when a high-level language is not available.

- If HDL is used, all hardware components within a cryptographic module shall be implemented using a high-level specification language.

SECURITY LEVEL 4

In addition to the requirements for Security Levels 1, 2, and 3, the following requirements shall apply to cryptographic modules for Security Level 4.

- Documentation shall specify a formal model that describes the rules and characteristics of the cryptographic module security policy. The formal model shall be specified using a formal specification language that is a rigorous notation based on established mathematics, such as first order logic or set theory.

- Documentation shall specify a rationale that demonstrates the consistency and completeness of the formal model with respect to the cryptographic module security policy.

- Documentation shall specify an informal proof of the correspondence between the formal model and the functional specification.

- For each cryptographic module hardware, software, and firmware component, the source code shall be annotated with comments that specify (1) the preconditions required upon entry into the module component, function, or procedure in order to execute correctly and (2) the postconditions expected to be true when execution of the module component, function, or procedure is complete. The preconditions and postconditions may be specified using any notation that is sufficiently detailed to completely and unambiguously explain the behavior of the cryptographic module component, function, or procedure.

- Documentation shall specify an informal proof of the correspondence between the design of the cryptographic module (as reflected by the precondition and postcondition annotations) and the functional specification.

RECOMMENDED SOFTWARE DEVELOPMENT PRACTICES FOR ALL LEVELS

Implementation of software and firmware components within a cryptographic module using recommended development practices listed in Appendix B will facilitate the analysis of the components for conformance to the requirements in this standard and will reduce the chance of design errors.

4.10.4 Guidance Documents

Crypto officer guidance is concerned with the correct configuration, maintenance, and administration of the cryptographic module. *User guidance* describes the security functions of the cryptographic module along with instructions, guidelines, and warnings for the secure use of the module. If a cryptographic module supports a maintenance role, user/crypto officer guidance describes the physical and/or logical maintenance services for operators assuming the maintenance role.

Crypto officer guidance shall specify:

- the administrative functions, security events, security parameters (and parameter values, as appropriate), physical ports, and logical interfaces of the cryptographic module available to the crypto officer,

- procedures on how to administer the cryptographic module in a secure manner, and

- assumptions regarding user behavior that are relevant to the secure operation of the cryptographic module.

User guidance shall specify:

- the Approved security functions, physical ports, and logical interfaces available to the users of a cryptographic module, and

- all user responsibilities necessary for the secure operation of a cryptographic module.

4. 11 Mitigation of Other Attacks

Cryptographic modules may be susceptible to other attacks for which testable security requirements were not available at the time this version of the standard was issued (e.g., power analysis, timing analysis, and/or fault induction) or the attacks were outside of the scope of the standard (e.g., TEMPEST). Susceptibility of a cryptographic module to such attacks depends on module type, implementation, and implementation environment. Such attacks may be of particular concern for cryptographic modules implemented in hostile environments (e.g., where the attackers may be the authorized operators of the module). Such types of attacks generally rely on the analysis of information obtained from sources physically external to the module. In all cases, the attacks attempt to determine some knowledge about the cryptographic keys and CSPs within the cryptographic module. Brief summaries of currently known attacks are provided below.

Power Analysis: Attacks based on the analysis of power consumption can be divided into two general categories, Simple Power Analysis (SPA) and Differential Power Analysis (DPA). SPA involves a direct (primarily visual) analysis of electrical power consumption patterns and timings derived from the execution of individual instructions carried out by a cryptographic module during a cryptographic process. The patterns are obtained through monitoring the variations in electrical power consumption of a cryptographic module for the purpose of revealing the features and implementations of cryptographic algorithms and subsequently values of cryptographic keys. DPA has the same goals but utilizes advanced statistical methods and/or other techniques to analyze the variations of the electrical power consumption of a cryptographic module. Cryptographic modules that utilize external power (direct current) sources appear to be at greatest risk. Methods that may reduce the overall risk of Power Analysis attacks include the use of capacitors to level the power consumption, the use of internal power sources, and the manipulation of the individual operations of the algorithms or processes to level the rate of power consumption during cryptographic processing.

Timing Analysis: Timing Analysis attacks rely on precisely measuring the time required by a cryptographic module to perform specific mathematical operations associated with a cryptographic algorithm or process. The timing information collected is analyzed to determine the relationship between the inputs to the module and the cryptographic keys used by the underlying algorithms or processes. The analysis of the relationship may be used to exploit the timing measurements to reveal the cryptographic key or CSPs. Timing Analysis attacks assume that the attacker has knowledge of the design of the cryptographic module. Manipulation of the individual operations of the algorithms or processes to reduce timing fluctuations during processing is one method to reduce the risk of this attack.

Fault Induction: Fault Induction attacks utilize external forces such as microwaves, temperature extremes, and voltage manipulation to cause processing errors within the cryptographic module. An analysis of these errors and their patterns can be used in an attempt to reverse engineer the cryptographic module, revealing certain features and implementations of cryptographic algorithms and subsequently revealing the values of cryptographic keys. Cryptographic modules with limited physical security appear to be at greatest risk. Proper selection of physical security features may be used to reduce the risk of this attack.

TEMPEST: TEMPEST attacks involve the remote or external detection and collection of the electromagnetic signals emitted from a cryptographic module and associated equipment during processing. Such an attack can be used to obtain keystroke information, messages displayed on a video screen, and other forms of critical security information (e.g., cryptographic keys). Special shielding of all components, including network cabling, is the mechanism used to reduce the risk of such an attack. Shielding reduces and, in some cases, prevents the emission of electromagnetic signals.

If a cryptographic module is designed to mitigate one or more specific attacks, then the module's security policy shall specify the security mechanisms employed by the module to mitigate the attack(s). The existence and proper functioning of the security mechanisms will be validated when requirements and associated tests are developed.

APPENDIX A: SUMMARY OF DOCUMENTATION REQUIREMENTS

The following check list summarizes the documentation requirements of this standard. All documentation shall be provided to the validation facility by the vendor of a cryptographic module.

CRYPTOGRAPHIC MODULE SPECIFICATION

- Specification of the hardware, software, and firmware components of a cryptographic module, specification of the cryptographic boundary surrounding these components, and description of the physical configuration of the module. *(Security Levels 1, 2, 3, and 4)*

- Specification of any hardware, software, or firmware components of a cryptographic module that are excluded from the security requirements of this standard and an explanation of the rationale for the exclusion. *(Security Levels 1, 2, 3, and 4)*

- Specification of the physical ports and logical interfaces of a cryptographic module. *(Security Levels 1, 2, 3, and 4)*

- Specification of the manual or logical controls of a cryptographic module, physical or logical status indicators, and applicable physical, logical, and electrical characteristics. *(Security Levels 1, 2, 3, and 4)*

- List of all security functions, both Approved and non-Approved, that are employed by a cryptographic module and specification of all modes of operation, both Approved and non-Approved. *(Security Levels 1, 2, 3, and 4)*

- Block diagram depicting all of the major hardware components of a cryptographic module and component interconnections, including any microprocessors, input/output buffers, plaintext/ciphertext buffers, control buffers, key storage, working memory, and program memory. *(Security Levels 1, 2, 3, and 4)*

- Specification of the design of the hardware, software, and firmware components of a cryptographic module. *(Security Levels 1, 2, 3, and 4)*

- Specification of all security-related information, including secret and private cryptographic keys (both plaintext and encrypted), authentication data (e.g., passwords, PINs), CSPs, and other protected information (e.g., audited events, audit data) whose disclosure or modification can compromise the security of the cryptographic module.

- Specification of a cryptographic module security policy including the rules derived from the requirements of this standard and the rules derived from any additional requirements imposed by the vendor). *(Security Levels 1, 2, 3, and 4)*

CRYPTOGRAPHIC MODULE PORTS AND INTERFACES

- Specification of the physical ports and logical interfaces of a cryptographic module and all defined input and output data paths. *(Security Levels 1, 2, 3, and 4)*

ROLES, SERVICES, AND AUTHENTICATION

- Specification of all authorized roles supported by a cryptographic module. *(Security Levels 1, 2, 3, and 4)*

- Specification of the services, operations, or functions provided by a cryptographic module, both Approved and non-Approved. For each service, specification of the service inputs, corresponding service outputs, and the authorized role(s) in which the service can be performed. *(Security Levels 1, 2, 3, and 4)*

- Specification of any services provided by a cryptographic module for which the operator is not required to assume an authorized role, and how these services do not modify, disclose, or substitute cryptographic keys and CSPs, or otherwise affect the security of the module.

- Specification of the authentication mechanisms supported by a cryptographic module, the types of authentication data required to implement supported authentication mechanisms, the authorized methods used to control access to the module for the first time and initialize the authentication mechanism, and the corresponding strength of the mechanisms supported by the module. *(Security Levels 2, 3, and 4)*

FINITE STATE MODEL

- Representation of a finite state model (or equivalent) using the state transition diagram and/or state transition table that specifies all operational and error states, corresponding transitions from one state to another, input events (including data inputs and control outputs) that cause transitions from one state to another, and output events (including internal module conditions, data outputs, and status outputs) resulting from transitions from one state to another. *(Security Levels 1, 2, 3, and 4)*

PHYSICAL SECURITY

- Specification of the physical embodiment and security level for which the physical security mechanisms of a cryptographic module are implemented. Specification of the physical security mechanisms that are employed by a module. *(Security Levels 1, 2, 3, and 4)*

- If a cryptographic module includes a maintenance role that requires physical access to the contents of the module or if the module is designed to permit physical access, specification of the maintenance access interface and how plaintext secret and private keys and CSPs are to be zeroized when the maintenance access interface is accessed. *(Security Levels 1, 2, 3, and 4)*

- Specification of the normal operating ranges of a cryptographic module. Specification of the environmental failure protection features employed by a cryptographic module or specification of the environmental failure tests performed. *(Security Level 4)*

OPERATIONAL ENVIRONMENT

- Specification of the operational environment for the cryptographic module. *(Security Levels 1, 2, 3, and 4)*

- Identification of the operating system employed by a cryptographic module, the applicable Protection Profile, and the CC assurance level. *(Security Levels 2, 3, and 4)*

CRYPTOGRAPHIC KEY MANAGEMENT

- Specification of all cryptographic keys, cryptographic key components, and CSPs employed by a cryptographic module.

- Specification of each RNG (Approved and non-Approved) employed by a cryptographic module. *(Security Levels 1, 2, 3, and 4)*

- Specification of each of the key generation methods (Approved and non-Approved) employed by a cryptographic module. *(Security Levels 1, 2, 3, and 4)*

- Specification of the key establishment methods employed by a cryptographic module. *(Security Levels 1, 2, 3, and 4)*

- Specification of the key entry and output methods employed by a cryptographic module. *(Security Levels 1, 2, 3, and 4)*

- If split knowledge procedures are used, proof that if knowledge of *n* key components is required to reconstruct the original key, then knowledge that any *n*-1 key components provides no information about the original key other than length, and specification of the split-knowledge procedures employed by a cryptographic module. *(Security Levels 3 and 4)*

- Specification of the key storage methods employed by a cryptographic module. *(Security Levels 1, 2, 3, and 4)*

- Specification of the key zeroization methods employed by a cryptographic module. *(Security Levels 1, 2, 3, and 4)*

ELECTROMAGNETIC INTERFERENCE/ELECTROMAGNETIC COMPATIBILITY

- Proof of conformance to EMI/EMC requirements. *(Security Levels 1, 2, 3, and 4)*

SELF-TESTS

- Specification of the self-tests performed by a cryptographic module including power-up and conditional tests. *(Security Levels 1, 2, 3, and 4)*

- Specification of the error states that a cryptographic module can enter when a self-test fails, and the conditions and actions necessary to exit the error states and resume normal operation of a module. *(Security Levels 1, 2, 3, and 4)*

- Specification of all security functions critical to the secure operation of a cryptographic module and identification of the applicable power-up tests and conditional tests performed by the module. *(Security Levels 1, 2, 3, and 4)*

- If a cryptographic module implements a bypass capability, specification of the mechanism or logic governing the switching procedure. *(Security Levels 1, 2, 3, and 4)*

DESIGN ASSURANCE

- Specification of procedures for secure installation, generation, and start-up of a cryptographic module. *(Security Levels 1, 2, 3, and 4)*

- Specification of the procedures for maintaining security while distributing and delivering versions of a cryptographic module to authorized operators. *(Security Level 2, 3, and 4)*

- Specification of the correspondence between the design of the hardware, software, and firmware components of a cryptographic module and the cryptographic module security policy (i.e., the rules of operation). *(Security Levels 1, 2, 3, and 4)*

- If a cryptographic module contains software or firmware components, specification of the source code for the software and firmware components, annotated with comments that clearly depict the correspondence of the components to the design of the module. *(Security Levels 1, 2, 3, and 4)*

- If a cryptographic module contains hardware components, specification of the schematics and/or Hardware Description Language (HDL) listings for the hardware components. *(Security Levels 1, 2, 3, and 4)*

- Functional specification that informally describes a cryptographic module, the external ports and interfaces of the module, and the purpose of the interfaces. *(Security Levels 2, 3, and 4)*

- Specification of a formal model that describes the rules and characteristics of the cryptographic module security policy, using a formal specification language that is a rigorous notation based on established mathematics, such as first order logic or set theory. *(Security Level 4)*

- Specification of a rationale that demonstrates the consistency and completeness of the formal model with respect to the cryptographic module security policy. *(Security Level 4)*

- Specification of an informal proof of the correspondence between the formal model and the functional specification. *(Security Level 4)*

- For each hardware, software, and firmware component, source code annotation with comments that specify (1) the preconditions required upon entry into the module component, function or procedure in order to execute correctly and (2) the postconditions expected to be true when execution of the module component, function, or procedure is complete. *(Security Level 4)*

- Specification of an informal proof of the correspondence between the design of the cryptographic module (as reflected by the precondition and postcondition annotations) and the functional specification. *(Security Level 4)*

- For crypto officer guidance, specification of:

  - ❑ the administrative functions, security events, security parameters (and parameter values, as appropriate), physical ports, and logical interfaces of the cryptographic module available to the crypto officer *(Security Levels 1, 2, 3, and 4)*,

  - ❑ procedures on how to administer the cryptographic module in a secure manner *(Security Levels 1, 2, 3, and 4)*, and

  - ❑ assumptions regarding user behavior that is relevant to the secure operation of the cryptographic module. *(Security Levels 1, 2, 3, and 4)*

- For user guidance, specification of

  - ❑ the Approved security functions, physical ports, and logical interfaces available to the users of the cryptographic module *(Security Levels 1, 2, 3, and 4)*, and

  - ❑ all user responsibilities necessary for the secure operation of the module. *(Security Levels 1, 2, 3, and 4)*

MITIGATION OF OTHER ATTACKS

- If a cryptographic module is designed to mitigate one or more specific attacks, specification in the module's security policy of the security mechanisms employed by the cryptographic module to mitigate the attack(s). *(Security Levels 1, 2, 3, and 4)*

SECURITY POLICY

- See Appendix C. *(Security Levels 1, 2, 3, and 4)*

APPENDIX B: RECOMMENDED SOFTWARE DEVELOPMENT PRACTICES

This Appendix is provided for informational purposes only and does not contain security requirements applicable to cryptographic modules within the scope of the standard.

Life-cycle software engineering recommendations (dealing with the specification, construction, verification, testing, maintenance, and documentation of software) should be followed. Software engineering practices may include documented unit testing, code reviews, explicit high-level and low-level design documents, explicit requirements and functional specifications, structure charts and data flow diagrams, function-point analysis, defect and resolution tracking, configuration management, and a documented software development process.

For all software development, both large and small, the following programming techniques are consistent with current practices and should be used to facilitate analysis of software components of a cryptographic module and to reduce chances of programming errors.

MODULAR DESIGN

- A modular design is recommended, especially for moderate to large-scale software development efforts. Each software module should have well-defined and readily understood logical interfaces.

- Software components should be constructed using the principles of data abstraction. If available, an object-oriented, high-level language that supports the construction of abstract data types should be used.

- The software should be hierarchically structured as a series of layers.

SOFTWARE MODULE/PROCEDURE INTERFACES

- Entries to a software module or procedure should be through external calls on explicitly defined interfaces.

- Each procedure should have only one entry point and at most two exit points, one for normal exits and one for error exits.

- Data should be communicated between software modules and between procedures through the use of argument lists and/or explicit return values. Global variables should not be used among procedures except where necessary for the implementation of abstract data types. Input values should be checked for range errors using assertion statements (if provided by the programming language in use).

INTERNAL CONSTRUCTION

- Each procedure should perform only a single, well-defined function.

- Control flow within a single thread of execution should be defined using only sequencing, structured programming constructs for conditionals (e.g., if-then-else or case), and structured constructs for loops (e.g., while-do or repeat-until).

- If concurrent execution is employed (e.g., via multiple threads, tasks, or processes), the software components should enforce limits on the maximum allowable degree of concurrency and should use structured synchronization constructs to control access to shared data.

- Equivalence of variables should not be used to permit multiple memory usage for conflicting purposes.

- Robust command parsing and range checking mechanisms should be implemented to guard against malformed requests, out-of-range parameters, and I/O buffer overflows.

IN-LINE DOCUMENTATION

- Each software module, procedure, and major programming construct should be documented specifying the functions performed along with a (formal or informal) specification of preconditions and postconditions.

- Each loop should be preceded by a convincing argument (as a comment) that termination is guaranteed.

- Variable names should be used in only one context within the same procedure.

- Each variable should have an associated comment identifying the purpose of the variable and noting the range of allowable values, including if the range is unrestricted.

- If concurrency is employed, the documentation should specify how limits are enforced on the maximum allowable degree of concurrency and how accesses to shared data are synchronized in order to avoid (possibly undetected) run-time errors.

ASSEMBLY LANGUAGE

The following additional programming practices should be used when the implementation is in assembly language.

- All code should be position independent except where appropriate security concerns, efficiency, or hardware constraints require position dependency.

- All register references should use symbolic register names.

- Self-modifying code should not be used.

- All procedures should be responsible for saving and restoring the contents of any register that is used within the procedure.

- Control transfer instructions should not use numeric literals.

- Each unit should contain comments describing register use in the unit.

APPENDIX C: CRYPTOGRAPHIC MODULE SECURITY POLICY

A cryptographic module security policy shall be included in the documentation provided by the vendor. The following paragraphs outline the required contents of the security policy.

C.1 Definition of Cryptographic Module Security Policy

A cryptographic module security policy shall consist of:

- a specification of the security rules, under which a cryptographic module shall operate, including the security rules derived from the requirements of the standard and the additional security rules imposed by the vendor.

The specification shall be sufficiently detailed to answer the following questions:

- What access does operator *X*, performing service *Y* while in role *Z*, have to security-relevant data item *W* for every role, service, and security-relevant data item contained in the cryptographic module?

- What physical security mechanisms are implemented to protect a cryptographic module and what actions are required to ensure that the physical security of a module is maintained?

- What security mechanisms are implemented in a cryptographic module to mitigate against attacks for which testable requirements are not defined in the standard?

C.2 Purpose of Cryptographic Module Security Policy

There are two major reasons for developing and following a precise cryptographic module security policy:

- To provide a specification of the cryptographic security that will allow individuals and organizations to determine whether a cryptographic module, as implemented, satisfies a stated security policy.

- To describe to individuals and organizations the capabilities, protection, and access rights provided by the cryptographic module, thereby allowing an assessment of whether the module will adequately serve the individual or organizational security requirements.

C.3 Specification of a Cryptographic Module Security Policy

A cryptographic module security policy shall be expressed in terms of roles, services, and cryptographic keys and CSPs. At a minimum, the following shall be specified:

- an identification and authentication (I&A) policy,

- an access control policy,

- a physical security policy, and

- a security policy for mitigation of other attacks.

C.3.1 Identification and Authentication Policy

The cryptographic module security policy shall specify an identification and authentication policy, including

- all roles (e.g., user, crypto officer, and maintenance) and associated type of authentication (e.g., identity-based, role-based, or none) and

- the authentication data required of each role or operator (e.g., password or biometric data) and the corresponding strength of the authentication mechanism.

C.3.2 Access Control Policy

The cryptographic module security policy shall specify an access control policy. The specification shall be of sufficient detail to identify the cryptographic keys and CSPs that the operator has access to while performing a service, and the type(s) of access the operator has to the parameters.

The security policy shall specify:

- all roles supported by a cryptographic module,

- all services provided by a cryptographic module,

- all cryptographic keys and CSPs employed by the cryptographic module, including

  - ❑ secret, private, and public cryptographic keys (both plaintext and encrypted),

  - ❑ authentication data such as passwords or PINs, and

  - ❑ other security-relevant information (e.g., audited events and audit data),

- for each role, the services an operator is authorized to perform within that role, and

- for each service within each role, the type(s) of access to the cryptographic keys and CSPs.

C.3.3 Physical Security Policy

The cryptographic module security policy shall specify a physical security policy, including:

- the physical security mechanisms that are implemented in a cryptographic module (e.g., tamper-evident seals, locks, tamper response and zeroization switches, and alarms) and

- the actions required by the operator(s) to ensure that physical security is maintained (e.g., periodic inspection of tamper-evident seals or testing of tamper response and zeroization switches).

C.3.4 Mitigation of Other Attacks Policy

The cryptographic module security policy shall specify a security policy for mitigation of other attacks, including the security mechanisms implemented to mitigate the attacks.

C.4 Security Policy Check List Tables

The following check list tables may be used as guides to ensure the security policy is complete and contains the appropriate details:

Role	Type of Authentication	Authentication Data
...	...	... ...
...	...	... ...

Table C1. *Roles and Required Identification and Authentication*

Authentication Mechanism	Strength of Mechanism
...	...
...	...

Table C2. *Strengths of Authentication Mechanisms*

Role	Authorized Services
...	... ...
...	... ...

Table C3. *Services Authorized for Roles*

Service	Cryptographic Keys and CSPs	Type(s) of Access (e.g., RWE)
...	... ...	... ...
...	... ...	... ...

Table C4. *Access Rights within Services*

Physical Security Mechanisms	Recommended Frequency of Inspection/Test	Inspection/Test Guidance Details
...	... ...	... ...
...	... ...	... ...

Table C5. *Inspection/Testing of Physical Security Mechanisms*

Other Attacks	Mitigation Mechanism	Specific Limitations
...	... ...	... ...
...	... ...	... ...

Table C6. *Mitigation of Other Attacks*

APPENDIX D: SELECTED BIBLIOGRAPHY

American Bankers Association, *Digital Signatures using Reversible Public Key Cryptography for the Financial Services Industry (rDSA),* ANSI X9.31-1998, Washington, D.C., 1998.

American Bankers Association, *Triple Data Encryption Algorithm Modes of Operation*, ANSI X9.52-1998, Washington, D.C., 1998.

American Bankers Association, *Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm*, American National Standard X9.62-1998, Washington, D.C., 1998.

Common Criteria Implementation Board (CCIB), *International Standard (IS) 15408*, *Common Criteria for Information Technology Security Evaluation*, Version 2, May 1998, ISO/IEC JTC 1 and Common Criteria Implementation Board.

Computer Security Act of 1987, 40 U.S. Code 759, (Public Law 100-235), January 8, 1988.

Department of Defense, *Department of Defense Trusted Computer System Evaluation Criteria*, DOD 5200.28-STD, December 1985.

Information Technology Management Reform Act of 1996, U.S. Code, (Public Law 104-106), 10 February *1996.*

Information Technology Security Evaluation Criteria (ITSEC), Harmonized Criteria of France – Germany - the Netherlands - the United Kingdom, Version 1.1, January 1991.

Keller, Sharon and Smid, Miles, *Modes of Operation Validation System (MOVS): Requirements and Procedures*, Special Publication 800-17, Gaithersburg, MD, National Institute of Standards and Technology, February 1998.

Keller, Sharon, *Modes of Operation Validation System for the Triple Data Encryption Algorithm (TMOVS): Requirements and Procedures,* Special Publication 800-20, Gaithersburg, MD, National Institute of Standards and Technology, October 1999.

Lee, Annabelle, *Guideline for Implementing Cryptography in the Federal Government*, Special Publication 800-21, Gaithersburg, MD, National Institute of Standards and Technology, November, 1999.

National Institute of Standards and Technology, *FIPS 140-2 Annex A: Approved Security Functions*, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *FIPS 140-2 Annex B: Approved Protection Profiles*, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *FIPS 140-2 Annex C: Approved Random Number Generators*, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *FIPS 140-2 Annex D: Approved Key Establishment Techniques*, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *Computer Data Authentication*, Federal Information Processing Standards Publication 113, 30 May 1985.

National Institute of Standards and Technology, *Data Encryption Standard*, Federal Information Processing Standards Publication 46-3, October 25, 1999.

National Institute of Standards and Technology and Communications Security Establishment, *Derived Test Requirements(DTR) for FIPS PUB 140-2, Security Requirements for Cryptographic Modules*, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *DES Modes of Operation*, Federal Information Processing Standards Publication 81, December 2, 1980.

National Institute of Standards and Technology, *Digital Signature Standard (DSS)*, Federal Information Processing Standards Publication 186-2, January 27, 2000.

National Institute of Standards and Technology, *Digital Signature Standard Validation System (DSSVS) User's Guide*, June 20, 1997.

National Institute of Standards and Technology, *Entity Authentication Using Public Key Cryptography*, Federal Information Processing Standards Publication 196, February 18, 1997.

National Institute of Standards and Technology, *Guideline for the Use of Advanced Authentication Technology Alternatives*, Federal Information Processing Standards Publication 190, September 28, 1994.

National Institute of Standards and Technology and Communications Security Establishment, *Implementation Guidance (IG) for FIPS 140-2*, available at URL: http://www.nist.gov/cmvp.

National Institute of Standards and Technology, *Key Management using ANSI X9.17*, Federal Information Processing Standards Publication 171, April 27, 1992.

National Institute of Standards and Technology, *Secure Hash Standard*, Federal Information Processing Standards Publication 180-1, April 17, 1995.

National Institute of Standards and Technology, *Security Requirements for Cryptographic Modules*, Federal Information Processing Standards Publication 140-1, January 11, 1994.

Office of Management and Budget, *Security of Federal Automated Information Resources*, Appendix III to OMB Circular No. A-130, February 8, 1996.

Telecommunications Industry Association, *Over-The-Air-Rekeying (OTAR) Protocol*, New Technology Standards Project, Digital Radio Technical Standards, TIA/EIA Telecommunications Systems Bulletin, APCO Project 25, TSB102.AACA, January 1996.

APPENDIX E: APPLICABLE INTERNET UNIFORM RESOURCE LOCATORS (URL)

Communications Security Establishment (CSE): http://www.cse-cst.gc.ca

Cryptographic Module Validation Program (CMVP): http://www.nist.gov/cmvp

NIST Information Technology Laboratory (NIST ITL): http://www.nist.gov/itl

NIST Security Publications including FIPS and Special Publications: http://csrc.nist.gov/publications

National Technical Information Service (NTIS): http://www.ntis.gov

National Voluntary Laboratory Accreditation Program (NVLAP): http://ts.nist.gov/nvlap

National Information Assurance Partnership® (NIAP): http://niap.nist.gov/

Validated Protection Profiles: http://niap.nist.gov/cc-scheme/PPRegistry.html

CHANGE NOTICES

Change Notice 1 (Superseded by Change Notice 2)

FIPS PUB 140-2, SECURITY REQUIREMENTS FOR CRYPTOGRAPHIC MODULES

U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
Gaithersburg, MD 20899

DATE OF CHANGE: 2001 October 10

Federal Information Processing Standard (FIPS) 140-2, *Security Requirements for Cryptographic Modules*, specifies the security requirements that will be satisfied by a cryptographic module utilized within a security system protecting sensitive but unclassified information (hereafter referred to as sensitive information).

This change notice provides a correction to the required intervals for the length of runs test indicated in Table 3 in Section 4.9.1 Power-Up Tests.

Table 3 as originally published, incorrectly specified the required intervals. The correct intervals are indicated.

Incorrect	
Length of Run	Required Interval
1	2,343 – 2,657
2	1,135 – 1,365
3	542 – 708
4	251 – 373
5	111 – 201
6+	111 - 201

Correct	
Length of Run	Required Interval
1	2,315 – 2,685
2	1,114 – 1,386
3	527 - 723
4	240 - 384
5	103 - 209
6+	103 - 209

Questions regarding this change notice may be directed to Annabelle Lee (annabelle.lee@nist.gov, 301-975-2941).

Change Notice 2

FIPS PUB 140-2, SECURITY REQUIREMENTS FOR CRYPTOGRAPHIC MODULES

U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
Gaithersburg, MD 20899

DATE OF CHANGE: 2002 December 03

TITLE: Random Number Generator Requirements

Federal Information Processing Standard (FIPS) 140-2, *Security Requirements for Cryptographic Modules*, specifies the security requirements that will be satisfied by a cryptographic module utilized within a security system protecting sensitive but unclassified information (hereafter referred to as sensitive information).

This change notice provides corrections to the requirements for random number generator used by cryptographic modules. These corrections involve paragraphs 4.7.1 and 4.9.1 of FIPS 140-2. Table 1 – *Summary of security requirements* has also been corrected and involves the random number generator requirements.

This change notice also provides a correction to the Table 1 – *Summary of security requirements*. The correction involves text found in the requirements of Physical Security at Security Level 4.

Finally, this change notice replaces the term "modes" used in paragraph 4.9.1 *Cryptographic algorithm test* with "cryptographic functions" which clarifies the standard.

In the corrected paragraphs and table below, the deleted text is struck out and the added text is underlined.

Change Notice 2 supersedes Change Notice 1.

The *Derived Test Requirements for FIPS 140-2* are also affected by these corrections.

Questions regarding this change notice may be directed to Annabelle Lee (annabelle.lee@nist.gov, 301-975-2941).

4.7.1 Random Number Generators (RNGs)

A cryptographic module may employ random number generators (RNGs). If a cryptographic module employs Approved or non-Approved RNGs in an Approved mode of operation, the data output from the RNG shall pass the continuous random number generator test as specified in Section 4.9.2. ~~Depending on the security level, the data output from an Approved RNG shall pass all statistical tests for randomness as specified in Section 4.9.1.~~ Approved ~~deterministic~~ RNGs shall be subject to the cryptographic algorithm test in Section 4.9.1. Approved RNGs are listed in Annex C to this standard.

Until such time as an Approved nondeterministic RNG standard exists, nondeterministic RNGs approved for use in classified applications may be used for key generation or to seed Approved deterministic RNGs used in key generation. Commercially available nondeterministic RNGs may be used for the purpose of generating seeds for Approved deterministic RNGs. Nondeterministic RNGs shall comply with all applicable RNG requirements of this standard.

An Approved RNG shall be used for the generation of cryptographic keys used by an Approved security function. The output from a non-Approved RNG may be used 1) as input (e.g., seed, and seed key) to an

Approved deterministic RNG or 2) to generate initialization vectors (IVs) for Approved security function(s). The seed and seed key shall not have the same value.

Documentation shall specify each RNG (Approved and non-Approved) employed by a cryptographic module.

4.9.1 Power-Up Tests

Power-up tests shall be performed by a cryptographic module when the module is powered up (after being powered off, reset, rebooted, etc.). The power-up tests shall be initiated automatically and shall not require operator intervention. When the power-up tests are completed, the results (i.e., indications of success or failure) shall be output via the "status output" interface. All data output via the data output interface shall be inhibited when the power-up tests are performed.

In addition to performing the power-up tests when powered up, a cryptographic module shall permit operators to initiate the tests on demand for periodic testing of the module. Resetting, rebooting, and power cycling are acceptable means for the on-demand initiation of power-up tests.

A cryptographic module shall perform the following power-up tests: cryptographic algorithm test, software/firmware integrity test, and critical functions test.

~~SECURITY LEVELS 1 AND 2~~

~~For Security Levels 1 and 2, a cryptographic module shall perform the following power-up tests: cryptographic algorithm test, software/firmware integrity test, and critical functions test. Statistical random number generator tests may be performed by the cryptographic module but are not required at Security Levels 1 and 2.~~

~~SECURITY LEVEL 3~~

~~For Security Level 3, in addition to the tests specified for Security Levels 1 and 2, a cryptographic module shall perform all of the statistical random number tests on demand by the operator and *may* perform the tests when the module is powered up.~~

~~SECURITY LEVEL 4~~

~~For Security Level 4, in addition to the tests specified for Security Levels 1,2 and 3, a cryptographic module shall also perform all of the statistical random number generator tests when the module is powered up.~~

Cryptographic algorithm test. A cryptographic algorithm test using a known answer shall be conducted for all ~~modes~~ cryptographic functions (e.g., encryption, decryption, authentication, and ~~deterministic~~ random number generation) of each Approved cryptographic algorithm implemented by a cryptographic module. A known-answer test involves operating the cryptographic algorithm on data for which the correct output is already known and comparing the calculated output with the previously generated output (the known answer). If the calculated output does not equal the known answer, the known-answer test shall fail.

Cryptographic algorithms whose outputs vary for a given set of inputs (e.g., the Digital Signature Algorithm) shall be tested using a known-answer test or shall be tested using a pair-wise consistency test (specified below). Message digest algorithms shall have an independent known-answer test or the known-answer test shall be included with the associated cryptographic algorithm test (e.g., the Digital Signature Standard).

If a cryptographic module includes two independent implementations of the same cryptographic algorithm, then:

- the known-answer test may be omitted,
- the outputs of two implementations shall be continuously compared, and
- if the outputs of two implementations are not equal, the cryptographic algorithm test shall fail.

Software/firmware integrity test. A software/firmware integrity test using an error detection code (EDC) or Approved authentication technique (e.g., an Approved message authentication code or digital signature algorithm) shall be applied to all validated software and firmware components within a cryptographic module when the module is powered up. The software/firmware integrity test is not required for any software and firmware components excluded from the security requirements of this standard (refer to Section 4.1). If the calculated result does not equal the previously generated result, the software/firmware test shall fail.

If an EDC is used, the EDC shall be at least 16 bits in length.

Critical functions test. Other security functions critical to the secure operation of a cryptographic module shall be tested when the module is powered up as part of the power-up tests. Other critical security functions performed under specific conditions shall be tested as conditional tests.

Documentation shall specify all security functions critical to the secure operation of a cryptographic module and shall identify the applicable power-up tests and conditional tests performed by the module.

~~*Statistical random number generator tests.* If statistical random number generator tests are required (i.e., depending on the security level), a cryptographic module employing RNGs shall perform the following statistical tests for randomness. A single bit stream of 20,000 consecutive bits of output from each RNG shall be subjected to the following four tests: monobit test, poker test, runs test, and long runs test.~~

~~*The monobit test*~~

~~Count the number of ones in the 20,000 bit stream. Denote this quantity by X.~~

~~The test is passed if 9,725 < X < 10,275.~~

~~*The poker test*~~

~~Divide the 20,000 bit stream into 5,000 consecutive 4 bit segments. Count and store the number of occurrences of the 16 possible 4 bit values. Denote *f(i)* as the number of each 4 bit value *i*, where $0 \leq i \leq 15$.~~

~~Evaluate the following:~~

$$\cancel{X = (16/5000) * \left(\sum_{i=0}^{15} [f(i)]^2\right) - 5000}$$

~~The test is passed if 2.16 < X < 46.17.~~

~~*The runs test*~~

~~A run is defined as a maximal sequence of consecutive bits of either all ones or all zeros that is part of the 20,000 bit sample stream. The incidences of runs (for both consecutive zeros and consecutive ones) of all lengths (≥ 1) in the sample stream should be counted and stored.~~

~~The test is passed if the runs that occur (of lengths 1 through 6) are each within the corresponding interval specified in the table below. This must hold for both the zeros and ones (i.e., all 12 counts must lie in the specified interval). For the purposes of this test, runs of greater than 6 are considered to be of length 6.~~

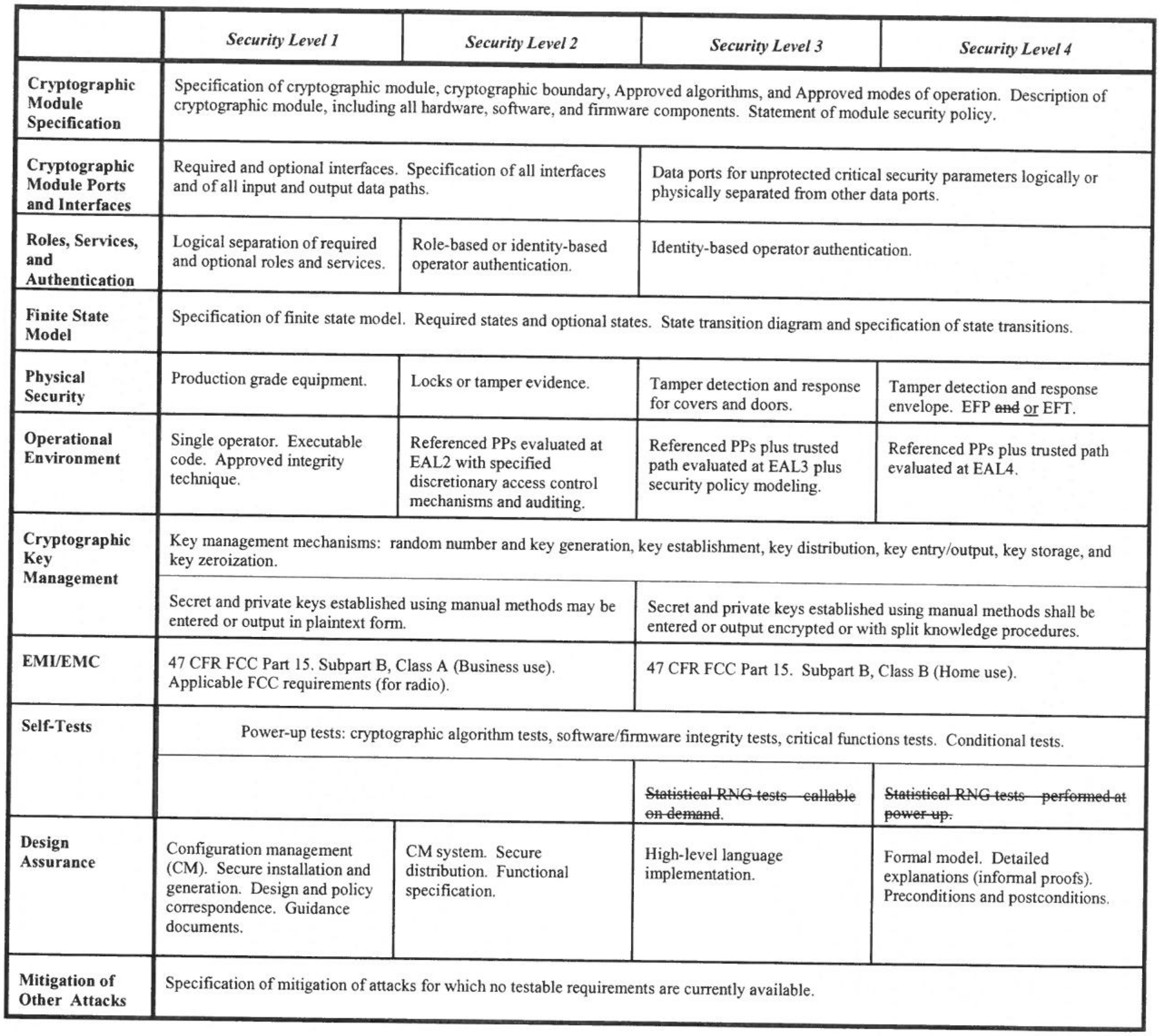
~~Length of Run~~	~~Required Interval~~
~~1~~	~~2,343 – 2,657~~
~~2~~	~~1,135 – 1,365~~
~~3~~	~~542 – 708~~
4	~~251 – 373~~
~~5~~	~~111 – 201~~
~~6+~~	~~111 – 201~~

~~Table 3. *Required intervals for length of runs test*~~

~~*The long runs test*~~

~~A long run is defined to be a run of length 26 or more (of either zeros or ones).~~

~~On the sample of 20,000 bits, the test is passed if there are no long runs.~~



	Security Level 1	*Security Level 2*	*Security Level 3*	*Security Level 4*
Cryptographic Module Specification	Specification of cryptographic module, cryptographic boundary, Approved algorithms, and Approved modes of operation. Description of cryptographic module, including all hardware, software, and firmware components. Statement of module security policy.			
Cryptographic Module Ports and Interfaces	Required and optional interfaces. Specification of all interfaces and of all input and output data paths.		Data ports for unprotected critical security parameters logically or physically separated from other data ports.	
Roles, Services, and Authentication	Logical separation of required and optional roles and services.	Role-based or identity-based operator authentication.	Identity-based operator authentication.	
Finite State Model	Specification of finite state model. Required states and optional states. State transition diagram and specification of state transitions.			
Physical Security	Production grade equipment.	Locks or tamper evidence.	Tamper detection and response for covers and doors.	Tamper detection and response envelope. EFP ~~and~~ or EFT.
Operational Environment	Single operator. Executable code. Approved integrity technique.	Referenced PPs evaluated at EAL2 with specified discretionary access control mechanisms and auditing.	Referenced PPs plus trusted path evaluated at EAL3 plus security policy modeling.	Referenced PPs plus trusted path evaluated at EAL4.
Cryptographic Key Management	Key management mechanisms: random number and key generation, key establishment, key distribution, key entry/output, key storage, and key zeroization.			
	Secret and private keys established using manual methods may be entered or output in plaintext form.		Secret and private keys established using manual methods shall be entered or output encrypted or with split knowledge procedures.	
EMI/EMC	47 CFR FCC Part 15. Subpart B, Class A (Business use). Applicable FCC requirements (for radio).		47 CFR FCC Part 15. Subpart B, Class B (Home use).	
Self-Tests	Power-up tests: cryptographic algorithm tests, software/firmware integrity tests, critical functions tests. Conditional tests.			
			~~Statistical RNG tests – callable on demand~~.	~~Statistical RNG tests – performed at power-up.~~
Design Assurance	Configuration management (CM). Secure installation and generation. Design and policy correspondence. Guidance documents.	CM system. Secure distribution. Functional specification.	High-level language implementation.	Formal model. Detailed explanations (informal proofs). Preconditions and postconditions.
Mitigation of Other Attacks	Specification of mitigation of attacks for which no testable requirements are currently available.			

Table 1: *Summary of security requirements*

Change Notice 3

FIPS PUB 140-2, SECURITY REQUIREMENTS FOR CRYPTOGRAPHIC MODULES

U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
Gaithersburg, MD 20899

DATE OF CHANGE: 2002 December 03

TITLE: Pair-Wise Consistency Test

Federal Information Processing Standard (FIPS) 140-2, *Security Requirements for Cryptographic Modules*, specifies the security requirements that will be satisfied by a cryptographic module utilized within a security system protecting sensitive but unclassified information (hereafter referred to as sensitive information).

This change notice provides corrections to the requirements for pair-wise consistency test for public/ private keys used for key agreement. These corrections involve paragraphs 4.9.2 of FIPS 140-2.

In the corrected paragraphs below, the deleted text is struck out and the added text is underlined.

The *Derived Test Requirements for FIPS 140-2* is also affected by these corrections.

Questions regarding this change notice may be directed to Annabelle Lee (annabelle.lee@nist.gov, 301-975-2941).

4.9.2 Conditional Tests

Conditional tests shall be performed by a cryptographic module when the conditions specified for the following tests occur: pair-wise consistency test, software/firmware load test, manual key entry test, continuous random number generator test, and bypass test.

Pair-wise consistency test (for public and private keys). If a cryptographic module generates public or private keys, then the following pair-wise consistency tests for public and private keys shall be performed:

1. If the keys are used to perform an approved key transport method ~~or encryption~~, then the public key shall encrypt a plaintext value. The resulting ciphertext value shall be compared to the original plaintext value. If the two values are equal, then the test shall fail. If the two values differ, then the private key shall be used to decrypt the ciphertext and the resulting value shall be compared to the original plaintext value. If the two values are not equal, the test shall fail.

~~2. If the keys are used to perform key agreement, then the cryptographic module shall create a second, compatible key pair. The cryptographic module shall perform both sides of the key agreement algorithm and shall compare the resulting shared values. If the shared values are not equal, the test shall fail.~~

2. If the keys are used to perform the calculation and verification of digital signatures, then the consistency of the keys shall be tested by the calculation and verification of a digital signature. If the digital signature cannot be verified, the test shall fail.

Change Notice 4

FIPS PUB 140-2, SECURITY REQUIREMENTS FOR CRYPTOGRAPHIC MODULES

U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
Gaithersburg, MD 20899

DATE OF CHANGE: 2002 December 03

TITLE: Limited Operational Environment

Federal Information Processing Standard (FIPS) 140-2, *Security Requirements for Cryptographic Modules*, specifies the security requirements that will be satisfied by a cryptographic module utilized within a security system protecting sensitive but unclassified information (hereafter referred to as sensitive information).

This change notice provides a correction to the definition of a *Limited Operational Environment*. This correction involves paragraph 4.6 of FIPS 140-2.

In the corrected paragraph below, the deleted text is struck out and the added text is underlined.

Questions regarding this change notice may be directed to Annabelle Lee (annabelle.lee@nist.gov, 301-975-2941).

4.6 Operational Environment

A *limited operational environment* refers to a static non-modifiable virtual operational environment (e.g., JAVA virtual machine ~~or~~ on a non-programmable PC card) with no underlying general purpose operating system upon which the operational environment uniquely resides.

Federal Information

Processing Standards Publication 197

November 26, 2001

Announcing the

ADVANCED ENCRYPTION STANDARD (AES)

Federal Information Processing Standards Publications (FIPS PUBS) are issued by the National Institute of Standards and Technology (NIST) after approval by the Secretary of Commerce pursuant to Section 5131 of the Information Technology Management Reform Act of 1996 (Public Law 104-106) and the Computer Security Act of 1987 (Public Law 100-235).

1. **Name of Standard.** Advanced Encryption Standard (AES) (FIPS PUB 197).

2. **Category of Standard.** Computer Security Standard, Cryptography.

3. **Explanation.** The Advanced Encryption Standard (AES) specifies a FIPS-approved cryptographic algorithm that can be used to protect electronic data. The AES algorithm is a symmetric block cipher that can encrypt (encipher) and decrypt (decipher) information. Encryption converts data to an unintelligible form called ciphertext; decrypting the ciphertext converts the data back into its original form, called plaintext.

The AES algorithm is capable of using cryptographic keys of 128, 192, and 256 bits to encrypt and decrypt data in blocks of 128 bits.

4. **Approving Authority.** Secretary of Commerce.

5. **Maintenance Agency.** Department of Commerce, National Institute of Standards and Technology, Information Technology Laboratory (ITL).

6. **Applicability.** This standard may be used by Federal departments and agencies when an agency determines that sensitive (unclassified) information (as defined in P. L. 100-235) requires cryptographic protection.

Other FIPS-approved cryptographic algorithms may be used in addition to, or in lieu of, this standard. Federal agencies or departments that use cryptographic devices for protecting classified information can use those devices for protecting sensitive (unclassified) information in lieu of this standard.

In addition, this standard may be adopted and used by non-Federal Government organizations. Such use is encouraged when it provides the desired security for commercial and private organizations.

7. Specifications. Federal Information Processing Standard (FIPS) 197, Advanced Encryption Standard (AES) (affixed).

8. Implementations. The algorithm specified in this standard may be implemented in software, firmware, hardware, or any combination thereof. The specific implementation may depend on several factors such as the application, the environment, the technology used, etc. The algorithm shall be used in conjunction with a FIPS approved or NIST recommended mode of operation. Object Identifiers (OIDs) and any associated parameters for AES used in these modes are available at the Computer Security Objects Register (CSOR), located at http://csrc.nist.gov/csor/ [2].

Implementations of the algorithm that are tested by an accredited laboratory and validated will be considered as complying with this standard. Since cryptographic security depends on many factors besides the correct implementation of an encryption algorithm, Federal Government employees, and others, should also refer to NIST Special Publication 800-21, *Guideline for Implementing Cryptography in the Federal Government*, for additional information and guidance (NIST SP 800-21 is available at http://csrc.nist.gov/publications/).

9. Implementation Schedule. This standard becomes effective on May 26, 2002.

10. Patents. Implementations of the algorithm specified in this standard may be covered by U.S. and foreign patents.

11. Export Control. Certain cryptographic devices and technical data regarding them are subject to Federal export controls. Exports of cryptographic modules implementing this standard and technical data regarding them must comply with these Federal regulations and be licensed by the Bureau of Export Administration of the U.S. Department of Commerce. Applicable Federal government export controls are specified in Title 15, Code of Federal Regulations (CFR) Part 740.17; Title 15, CFR Part 742; and Title 15, CFR Part 774, Category 5, Part 2.

12. Qualifications. NIST will continue to follow developments in the analysis of the AES algorithm. As with its other cryptographic algorithm standards, NIST will formally reevaluate this standard every five years.

Both this standard and possible threats reducing the security provided through the use of this standard will undergo review by NIST as appropriate, taking into account newly available analysis and technology. In addition, the awareness of any breakthrough in technology or any mathematical weakness of the algorithm will cause NIST to reevaluate this standard and provide necessary revisions.

13. Waiver Procedure. Under certain exceptional circumstances, the heads of Federal agencies, or their delegates, may approve waivers to Federal Information Processing Standards (FIPS). The heads of such agencies may redelegate such authority only to a senior official designated pursuant to Section 3506(b) of Title 44, U.S. Code. Waivers shall be granted only when compliance with this standard would

a. adversely affect the accomplishment of the mission of an operator of Federal computer system or

b. cause a major adverse financial impact on the operator that is not offset by government-wide savings.

Agency heads may act upon a written waiver request containing the information detailed above. Agency heads may also act without a written waiver request when they determine that conditions for meeting the standard cannot be met. Agency heads may approve waivers only by a written decision that explains the basis on which the agency head made the required finding(s). A copy of each such decision, with procurement sensitive or classified portions clearly identified, shall be sent to: National Institute of Standards and Technology; ATTN: FIPS Waiver Decision, Information Technology Laboratory, 100 Bureau Drive, Stop 8900, Gaithersburg, MD 20899-8900.

In addition, notice of each waiver granted and each delegation of authority to approve waivers shall be sent promptly to the Committee on Government Operations of the House of Representatives and the Committee on Government Affairs of the Senate and shall be published promptly in the Federal Register.

When the determination on a waiver applies to the procurement of equipment and/or services, a notice of the waiver determination must be published in the Commerce Business Daily as a part of the notice of solicitation for offers of an acquisition or, if the waiver determination is made after that notice is published, by amendment to such notice.

A copy of the waiver, any supporting documents, the document approving the waiver and any supporting and accompanying documents, with such deletions as the agency is authorized and decides to make under Section 552(b) of Title 5, U.S. Code, shall be part of the procurement documentation and retained by the agency.

14. Where to obtain copies. This publication is available electronically by accessing http://csrc.nist.gov/publications/. A list of other available computer security publications, including ordering information, can be obtained from NIST Publications List 91, which is available at the same web site. Alternatively, copies of NIST computer security publications are available from: National Technical Information Service (NTIS), 5285 Port Royal Road, Springfield, VA 22161.

Federal Information
Processing Standards Publication 197

November 26, 2001

Specification for the

ADVANCED ENCRYPTION STANDARD (AES)

Table of Contents

Table of Figures

1. Introduction

This standard specifies the **Rijndael** algorithm ([3] and [4]), a symmetric block cipher that can process **data blocks** of **128 bits**, using cipher **keys** with lengths of **128**, **192**, and **256 bits**. Rijndael was designed to handle additional block sizes and key lengths, however they are not adopted in this standard.

Throughout the remainder of this standard, the algorithm specified herein will be referred to as "the AES algorithm." The algorithm may be used with the three different key lengths indicated above, and therefore these different "flavors" may be referred to as "AES-128", "AES-192", and "AES-256".

This specification includes the following sections:

2. Definitions of terms, acronyms, and algorithm parameters, symbols, and functions;
3. Notation and conventions used in the algorithm specification, including the ordering and numbering of bits, bytes, and words;
4. Mathematical properties that are useful in understanding the algorithm;
5. Algorithm specification, covering the key expansion, encryption, and decryption routines;
6. Implementation issues, such as key length support, keying restrictions, and additional block/key/round sizes.

The standard concludes with several appendices that include step-by-step examples for Key Expansion and the Cipher, example vectors for the Cipher and Inverse Cipher, and a list of references.

2. Definitions

2.1 Glossary of Terms and Acronyms

The following definitions are used throughout this standard:

AES	Advanced Encryption Standard
Affine Transformation	A transformation consisting of multiplication by a matrix followed by the addition of a vector.
Array	An enumerated collection of identical entities (e.g., an array of bytes).
Bit	A binary digit having a value of 0 or 1.
Block	Sequence of binary bits that comprise the input, output, State, and Round Key. The length of a sequence is the number of bits it contains. Blocks are also interpreted as arrays of bytes.
Byte	A group of eight bits that is treated either as a single entity or as an array of 8 individual bits.

Cipher
Series of transformations that converts plaintext to ciphertext using the Cipher Key.

Cipher Key
Secret, cryptographic key that is used by the Key Expansion routine to generate a set of Round Keys; can be pictured as a rectangular array of bytes, having four rows and ***Nk*** columns.

Ciphertext
Data output from the Cipher or input to the Inverse Cipher.

Inverse Cipher
Series of transformations that converts ciphertext to plaintext using the Cipher Key.

Key Expansion
Routine used to generate a series of Round Keys from the Cipher Key.

Plaintext
Data input to the Cipher or output from the Inverse Cipher.

Rijndael
Cryptographic algorithm specified in this Advanced Encryption Standard (AES).

Round Key
Round keys are values derived from the Cipher Key using the Key Expansion routine; they are applied to the State in the Cipher and Inverse Cipher.

State
Intermediate Cipher result that can be pictured as a rectangular array of bytes, having four rows and ***Nb*** columns.

S-box
Non-linear substitution table used in several byte substitution transformations and in the Key Expansion routine to perform a one-for-one substitution of a byte value.

Word
A group of 32 bits that is treated either as a single entity or as an array of 4 bytes.

2.2 Algorithm Parameters, Symbols, and Functions

The following algorithm parameters, symbols, and functions are used throughout this standard:

`AddRoundKey()`
Transformation in the Cipher and Inverse Cipher in which a Round Key is added to the State using an XOR operation. The length of a Round Key equals the size of the State (i.e., for *Nb* = 4, the Round Key length equals 128 bits/16 bytes).

`InvMixColumns()`
Transformation in the Inverse Cipher that is the inverse of **`MixColumns()`**.

`InvShiftRows()`
Transformation in the Inverse Cipher that is the inverse of **`ShiftRows()`**.

`InvSubBytes()`
Transformation in the Inverse Cipher that is the inverse of **`SubBytes()`**.

K
Cipher Key.

`MixColumns()` Transformation in the Cipher that takes all of the columns of the State and mixes their data (independently of one another) to produce new columns.

Nb Number of columns (32-bit words) comprising the State. For this standard, ***Nb*** = 4. (Also see Sec. 6.3.)

Nk Number of 32-bit words comprising the Cipher Key. For this standard, ***Nk*** = 4, 6, or 8. (Also see Sec. 6.3.)

Nr Number of rounds, which is a function of ***Nk*** and ***Nb*** (which is fixed). For this standard, ***Nr*** = 10, 12, or 14. (Also see Sec. 6.3.)

Rcon[] The round constant word array.

`RotWord()` Function used in the Key Expansion routine that takes a four-byte word and performs a cyclic permutation.

`ShiftRows()` Transformation in the Cipher that processes the State by cyclically shifting the last three rows of the State by different offsets.

`SubBytes()` Transformation in the Cipher that processes the State using a non-linear byte substitution table (S-box) that operates on each of the State bytes independently.

`SubWord()` Function used in the Key Expansion routine that takes a four-byte input word and applies an S-box to each of the four bytes to produce an output word.

XOR Exclusive-OR operation.

$\oplus$ Exclusive-OR operation.

$\otimes$ Multiplication of two polynomials (each with degree < 4) modulo $x^4 + 1$.

$\bullet$ Finite field multiplication.

3. Notation and Conventions

3.1 Inputs and Outputs

The **input** and **output** for the AES algorithm each consist of **sequences of 128 bits** (digits with values of 0 or 1). These sequences will sometimes be referred to as **blocks** and the number of bits they contain will be referred to as their length. The **Cipher Key** for the AES algorithm is a **sequence of 128, 192 or 256 bits**. Other input, output and Cipher Key lengths are not permitted by this standard.

The bits within such sequences will be numbered starting at zero and ending at one less than the sequence length (block length or key length). The number i attached to a bit is known as its index and will be in one of the ranges $0 \le i < 128$, $0 \le i < 192$ or $0 \le i < 256$ depending on the block length and key length (specified above).

3.2 Bytes

The basic unit for processing in the AES algorithm is a **byte,** a sequence of eight bits treated as a single entity. The input, output and Cipher Key bit sequences described in Sec. 3.1 are processed as arrays of bytes that are formed by dividing these sequences into groups of eight contiguous bits to form arrays of bytes (see Sec. 3.3). For an input, output or Cipher Key denoted by a, the bytes in the resulting array will be referenced using one of the two forms, a_n or $a[n]$, where n will be in one of the following ranges:

Key length = 128 bits, $0 \leq n < 16$; Block length = 128 bits, $0 \leq n < 16$;

Key length = 192 bits, $0 \leq n < 24$;

Key length = 256 bits, $0 \leq n < 32$.

All byte values in the AES algorithm will be presented as the concatenation of its individual bit values (0 or 1) between braces in the order $\{b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0\}$. These bytes are interpreted as finite field elements using a polynomial representation:

$$b_7x^7 + b_6x^6 + b_5x^5 + b_4x^4 + b_3x^3 + b_2x^2 + b_1x + b_0 = \sum_{i=0}^{7} b_i x^i . \qquad (3.1)$$

For example, {01100011} identifies the specific finite field element $x^6 + x^5 + x + 1$.

It is also convenient to denote byte values using hexadecimal notation with each of two groups of four bits being denoted by a single character as in Fig. 1.

Bit Pattern	Character	Bit Pattern	Character	Bit Pattern	Character	Bit Pattern	Character
0000	0	0100	4	1000	8	1100	c
0001	1	0101	5	1001	9	1101	d
0010	2	0110	6	1010	a	1110	e
0011	3	0111	7	1011	b	1111	f

Figure 1. Hexadecimal representation of bit patterns.

Hence the element {01100011} can be represented as {63}, where the character denoting the four-bit group containing the higher numbered bits is again to the left.

Some finite field operations involve one additional bit (b_8) to the left of an 8-bit byte. Where this extra bit is present, it will appear as '{01}' immediately preceding the 8-bit byte; for example, a 9-bit sequence will be presented as {01}{1b}.

3.3 Arrays of Bytes

Arrays of bytes will be represented in the following form:

$$a_0 a_1 a_2 \ldots a_{15}$$

The bytes and the bit ordering within bytes are derived from the 128-bit input sequence

$$input_0\ input_1\ input_2\ \ldots\ input_{126}\ input_{127}$$

as follows:

$$a_0 = \{input_0, input_1, \ldots, input_7\};$$

$$a_1 = \{input_8, input_9, \ldots, input_{15}\};$$

$$\vdots$$

$$a_{15} = \{input_{120}, input_{121}, \ldots, input_{127}\}.$$

The pattern can be extended to longer sequences (i.e., for 192- and 256-bit keys), so that, in general,

$$a_n = \{input_{8n}, input_{8n+1}, \ldots, input_{8n+7}\}. \qquad (3.2)$$

Taking Sections 3.2 and 3.3 together, Fig. 2 shows how bits within each byte are numbered.

Input bit sequence	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	...
Byte number	0								1								2								...
Bit numbers in byte	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	...

Figure 2. Indices for Bytes and Bits.

3.4 The State

Internally, the AES algorithm's operations are performed on a two-dimensional array of bytes called the **State**. The State consists of four rows of bytes, each containing ***Nb*** bytes, where ***Nb*** is the block length divided by 32. In the State array denoted by the symbol *s*, each individual byte has two indices, with its row number *r* in the range $0 \le r < 4$ and its column number *c* in the range $0 \le c < \boldsymbol{Nb}$. This allows an individual byte of the State to be referred to as either $s_{r,c}$ or $s[r,c]$. For this standard, ***Nb***=4, i.e., $0 \le c < 4$ (also see Sec. 6.3).

At the start of the Cipher and Inverse Cipher described in Sec. 5, the input – the array of bytes in_0, in_1, ... in_{15} – is copied into the State array as illustrated in Fig. 3. The Cipher or Inverse Cipher operations are then conducted on this State array, after which its final value is copied to the output – the array of bytes out_0, out_1, ... out_{15}.

input bytes

in_0	in_4	in_8	in_{12}
in_1	in_5	in_9	in_{13}
in_2	in_6	in_{10}	in_{14}
in_3	in_7	in_{11}	in_{15}

→

State array

$s_{0,0}$	$s_{0,1}$	$s_{0,2}$	$s_{0,3}$
$s_{1,0}$	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$
$s_{2,0}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$
$s_{3,0}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$

→

output bytes

out_0	out_4	out_8	out_{12}
out_1	out_5	out_9	out_{13}
out_2	out_6	out_{10}	out_{14}
out_3	out_7	out_{11}	out_{15}

Figure 3. State array input and output.

Hence, at the beginning of the Cipher or Inverse Cipher, the input array, *in*, is copied to the State array according to the scheme:

$$s[r, c] = in[r + 4c] \qquad \text{for } 0 \le r < 4 \text{ and } 0 \le c < Nb, \qquad (3.3)$$

and at the end of the Cipher and Inverse Cipher, the State is copied to the output array *out* as follows:

$$out[r + 4c] = s[r, c] \qquad \text{for } 0 \le r < 4 \text{ and } 0 \le c < Nb. \qquad (3.4)$$

3.5 The State as an Array of Columns

The four bytes in each column of the State array form 32-bit **words**, where the row number r provides an index for the four bytes within each word. The state can hence be interpreted as a one-dimensional array of 32 bit words (columns), $w_0...w_3$, where the column number c provides an index into this array. Hence, for the example in Fig. 3, the State can be considered as an array of four words, as follows:

$$\begin{array}{ll} w_0 = s_{0,0}\, s_{1,0}\, s_{2,0}\, s_{3,0} & w_2 = s_{0,2}\, s_{1,2}\, s_{2,2}\, s_{3,2} \\ w_1 = s_{0,1}\, s_{1,1}\, s_{2,1}\, s_{3,1} & w_3 = s_{0,3}\, s_{1,3}\, s_{2,3}\, s_{3,3}\,. \end{array} \qquad (3.5)$$

4. Mathematical Preliminaries

All bytes in the AES algorithm are interpreted as finite field elements using the notation introduced in Sec. 3.2. Finite field elements can be added and multiplied, but these operations are different from those used for numbers. The following subsections introduce the basic mathematical concepts needed for Sec. 5.

4.1 Addition

The addition of two elements in a finite field is achieved by "adding" the coefficients for the corresponding powers in the polynomials for the two elements. The addition is performed with the XOR operation (denoted by $\oplus$) - i.e., modulo 2 - so that $1 \oplus 1 = 0$, $1 \oplus 0 = 1$, and $0 \oplus 0 = 0$. Consequently, subtraction of polynomials is identical to addition of polynomials.

Alternatively, addition of finite field elements can be described as the modulo 2 addition of corresponding bits in the byte. For two bytes $\{a_7a_6a_5a_4a_3a_2a_1a_0\}$ and $\{b_7b_6b_5b_4b_3b_2b_1b_0\}$, the sum is $\{c_7c_6c_5c_4c_3c_2c_1c_0\}$, where each $c_i = a_i \oplus b_i$ (i.e., $c_7 = a_7 \oplus b_7$, $c_6 = a_6 \oplus b_6$, $...c_0 = a_0 \oplus b_0$).

For example, the following expressions are equivalent to one another:

$$(x^6 + x^4 + x^2 + x + 1) + (x^7 + x + 1) = x^7 + x^6 + x^4 + x^2 \qquad \text{(polynomial notation);}$$

$$\{\texttt{01010111}\} \oplus \{\texttt{10000011}\} = \{\texttt{11010100}\} \qquad \text{(binary notation);}$$

$$\{\texttt{57}\} \oplus \{\texttt{83}\} = \{\texttt{d4}\} \qquad \text{(hexadecimal notation).}$$

4.2 Multiplication

In the polynomial representation, multiplication in GF(2^8) (denoted by $\bullet$) corresponds with the multiplication of polynomials modulo an **irreducible polynomial** of degree 8. A polynomial is irreducible if its only divisors are one and itself. **For the AES algorithm, this irreducible polynomial is**

$$m(x) = x^8 + x^4 + x^3 + x + 1, \qquad (4.1)$$

or {01}{1b} in hexadecimal notation.

For example, {57} • {83} = {c1}, because

$$\begin{aligned}(x^6+x^4+x^2+x+1)\,(x^7+x+1) &= x^{13}+x^{11}+x^9+x^8+x^7+\\ &\quad x^7+x^5+x^3+x^2+x+\\ &\quad x^6+x^4+x^2+x+1\\ &= x^{13}+x^{11}+x^9+x^8+x^6+x^5+x^4+x^3+1\end{aligned}$$

and

$$\begin{aligned}x^{13}+x^{11}+x^9+x^8+x^6+x^5+x^4+x^3+1 \texttt{ modulo } (x^8+x^4+x^3+x+1)\\ = x^7+x^6+1.\end{aligned}$$

The modular reduction by $m(x)$ ensures that the result will be a binary polynomial of degree less than 8, and thus can be represented by a byte. Unlike addition, there is no simple operation at the byte level that corresponds to this multiplication.

The multiplication defined above is associative, and the element {01} is the multiplicative identity. For any non-zero binary polynomial $b(x)$ of degree less than 8, the multiplicative inverse of $b(x)$, denoted $b^{-1}(x)$, can be found as follows: the extended Euclidean algorithm [7] is used to compute polynomials $a(x)$ and $c(x)$ such that

$$b(x)a(x)+m(x)c(x)=1. \qquad (4.2)$$

Hence, $a(x)\bullet b(x) \bmod m(x)=1$, which means

$$b^{-1}(x)=a(x) \bmod m(x). \qquad (4.3)$$

Moreover, for any $a(x)$, $b(x)$ and $c(x)$ in the field, it holds that

$$a(x)\bullet(b(x)+c(x))=a(x)\bullet b(x)+a(x)\bullet c(x).$$

It follows that the set of 256 possible byte values, with XOR used as addition and the multiplication defined as above, has the structure of the finite field GF(2^8).

4.2.1 Multiplication by x

Multiplying the binary polynomial defined in equation (3.1) with the polynomial x results in

$$b_7x^8+b_6x^7+b_5x^6+b_4x^5+b_3x^4+b_2x^3+b_1x^2+b_0x. \qquad (4.4)$$

The result $x\bullet b(x)$ is obtained by reducing the above result modulo $m(x)$, as defined in equation (4.1). If $b_7 = 0$, the result is already in reduced form. If $b_7 = 1$, the reduction is accomplished by subtracting (i.e., XORing) the polynomial $m(x)$. It follows that multiplication by x (i.e., {00000010} or {02}) can be implemented at the byte level as a left shift and a subsequent conditional bitwise XOR with {1b}. This operation on bytes is denoted by xtime(). Multiplication by higher powers of x can be implemented by repeated application of xtime(). By adding intermediate results, multiplication by any constant can be implemented.

For example, {57} • {13} = {fe} because

$$\{57\} \bullet \{02\} = \mathtt{xtime}(\{57\}) = \{ae\}$$
$$\{57\} \bullet \{04\} = \mathtt{xtime}(\{ae\}) = \{47\}$$
$$\{57\} \bullet \{08\} = \mathtt{xtime}(\{47\}) = \{8e\}$$
$$\{57\} \bullet \{10\} = \mathtt{xtime}(\{8e\}) = \{07\},$$

thus,

$$\begin{aligned}\{57\} \bullet \{13\} &= \{57\} \bullet (\{01\} \oplus \{02\} \oplus \{10\}) \\ &= \{57\} \oplus \{ae\} \oplus \{07\} \\ &= \{fe\}.\end{aligned}$$

4.3 Polynomials with Coefficients in GF(2^8)

Four-term polynomials can be defined - with coefficients that are finite field elements - as:

$$a(x) = a_3x^3 + a_2x^2 + a_1x + a_0 \tag{4.5}$$

which will be denoted as a word in the form $[a_0 , a_1 , a_2 , a_3]$. Note that the polynomials in this section behave somewhat differently than the polynomials used in the definition of finite field elements, even though both types of polynomials use the same indeterminate, x. The coefficients in this section are themselves finite field elements, i.e., bytes, instead of bits; also, the multiplication of four-term polynomials uses a different reduction polynomial, defined below. The distinction should always be clear from the context.

To illustrate the addition and multiplication operations, let

$$b(x) = b_3x^3 + b_2x^2 + b_1x + b_0 \tag{4.6}$$

define a second four-term polynomial. Addition is performed by adding the finite field coefficients of like powers of x. This addition corresponds to an XOR operation between the corresponding bytes in each of the words – in other words, the XOR of the complete word values.

Thus, using the equations of (4.5) and (4.6),

$$a(x) + b(x) = (a_3 \oplus b_3)x^3 + (a_2 \oplus b_2)x^2 + (a_1 \oplus b_1)x + (a_0 \oplus b_0) \tag{4.7}$$

Multiplication is achieved in two steps. In the first step, the polynomial product $c(x) = a(x) \bullet b(x)$ is algebraically expanded, and like powers are collected to give

$$c(x) = c_6x^6 + c_5x^5 + c_4x^4 + c_3x^3 + c_2x^2 + c_1x + c_0 \tag{4.8}$$

where

$$\begin{aligned} c_0 &= a_0 \bullet b_0 & c_4 &= a_3 \bullet b_1 \oplus a_2 \bullet b_2 \oplus a_1 \bullet b_3 \\ c_1 &= a_1 \bullet b_0 \oplus a_0 \bullet b_1 & c_5 &= a_3 \bullet b_2 \oplus a_2 \bullet b_3 \\ c_2 &= a_2 \bullet b_0 \oplus a_1 \bullet b_1 \oplus a_0 \bullet b_2 & c_6 &= a_3 \bullet b_3 \end{aligned} \tag{4.9}$$

$$c_3 = a_3 \bullet b_0 \oplus a_2 \bullet b_1 \oplus a_1 \bullet b_2 \oplus a_0 \bullet b_3 .$$

The result, $c(x)$, does not represent a four-byte word. Therefore, the second step of the multiplication is to reduce $c(x)$ modulo a polynomial of degree 4; the result can be reduced to a polynomial of degree less than 4. **For the AES algorithm, this is accomplished with the polynomial $x^4 + 1$**, so that

$$x^i \bmod(x^4 + 1) = x^{i \bmod 4} . \tag{4.10}$$

The modular product of $a(x)$ and $b(x)$, denoted by $a(x) \otimes b(x)$, is given by the four-term polynomial $d(x)$, defined as follows:

$$d(x) = d_3 x^3 + d_2 x^2 + d_1 x + d_0 \tag{4.11}$$

with

$$\begin{aligned} d_0 &= (a_0 \bullet b_0) \oplus (a_3 \bullet b_1) \oplus (a_2 \bullet b_2) \oplus (a_1 \bullet b_3) \\ d_1 &= (a_1 \bullet b_0) \oplus (a_0 \bullet b_1) \oplus (a_3 \bullet b_2) \oplus (a_2 \bullet b_3) \\ d_2 &= (a_2 \bullet b_0) \oplus (a_1 \bullet b_1) \oplus (a_0 \bullet b_2) \oplus (a_3 \bullet b_3) \\ d_3 &= (a_3 \bullet b_0) \oplus (a_2 \bullet b_1) \oplus (a_1 \bullet b_2) \oplus (a_0 \bullet b_3) \end{aligned} \tag{4.12}$$

When $a(x)$ is a fixed polynomial, the operation defined in equation (4.11) can be written in matrix form as:

$$\begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} a_0 & a_3 & a_2 & a_1 \\ a_1 & a_0 & a_3 & a_2 \\ a_2 & a_1 & a_0 & a_3 \\ a_3 & a_2 & a_1 & a_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} \tag{4.13}$$

Because $x^4 + 1$ is not an irreducible polynomial over GF(2^8), multiplication by a fixed four-term polynomial is not necessarily invertible. However, the AES algorithm specifies a fixed four-term polynomial that *does* have an inverse (see Sec. 5.1.3 and Sec. 5.3.3):

$$a(x) = \{\texttt{03}\}x^3 + \{\texttt{01}\}x^2 + \{\texttt{01}\}x + \{\texttt{02}\} \tag{4.14}$$

$$a^{-1}(x) = \{\texttt{0b}\}x^3 + \{\texttt{0d}\}x^2 + \{\texttt{09}\}x + \{\texttt{0e}\}. \tag{4.15}$$

Another polynomial used in the AES algorithm (see the **`RotWord()`** function in Sec. 5.2) has $a_0 = a_1 = a_2 = \{\texttt{00}\}$ and $a_3 = \{\texttt{01}\}$, which is the polynomial x^3. Inspection of equation (4.13) above will show that its effect is to form the output word by rotating bytes in the input word. This means that $[b_0, b_1, b_2, b_3]$ is transformed into $[b_1, b_2, b_3, b_0]$.

5. Algorithm Specification

For the AES algorithm, **the length of the input block, the output block and the State is 128 bits.** This is represented by ***Nb*** = 4, which reflects the number of 32-bit words (number of columns) in the State.

For the AES algorithm, **the length of the Cipher Key, *K*, is 128, 192, or 256 bits.** The key length is represented by ***Nk*** = 4, 6, or 8, which reflects the number of 32-bit words (number of columns) in the Cipher Key.

For the AES algorithm, the number of rounds to be performed during the execution of the algorithm is dependent on the key size. The number of rounds is represented by ***Nr***, where ***Nr*** = 10 when ***Nk*** = 4, ***Nr*** = 12 when ***Nk*** = 6, and ***Nr*** = 14 when ***Nk*** = 8.

The only Key-Block-Round combinations that conform to this standard are given in Fig. 4. For implementation issues relating to the key length, block size and number of rounds, see Sec. 6.3.

	Key Length ***(Nk words)***	**Block Size** ***(Nb words)***	**Number of Rounds** ***(Nr)***
AES-128	4	4	10
AES-192	6	4	12
AES-256	8	4	14

Figure 4. Key-Block-Round Combinations.

For both its Cipher and Inverse Cipher, the AES algorithm uses a round function that is composed of four different byte-oriented transformations: 1) byte substitution using a substitution table (S-box), 2) shifting rows of the State array by different offsets, 3) mixing the data within each column of the State array, and 4) adding a Round Key to the State. These transformations (and their inverses) are described in Sec. 5.1.1-5.1.4 and 5.3.1-5.3.4.

The Cipher and Inverse Cipher are described in Sec. 5.1 and Sec. 5.3, respectively, while the Key Schedule is described in Sec. 5.2.

5.1 Cipher

At the start of the Cipher, the input is copied to the State array using the conventions described in Sec. 3.4. After an initial Round Key addition, the State array is transformed by implementing a round function 10, 12, or 14 times (depending on the key length), with the final round differing slightly from the first ***Nr*** −1 rounds. The final State is then copied to the output as described in Sec. 3.4.

The round function is parameterized using a key schedule that consists of a one-dimensional array of four-byte words derived using the Key Expansion routine described in Sec. 5.2.

The Cipher is described in the pseudo code in Fig. 5. The individual transformations - **`SubBytes()`**, **`ShiftRows()`**, **`MixColumns()`**, and **`AddRoundKey()`** – process the State and are described in the following subsections. In Fig. 5, the array **`w[]`** contains the key schedule, which is described in Sec. 5.2.

As shown in Fig. 5, all ***Nr*** rounds are identical with the exception of the final round, which does not include the **`MixColumns()`** transformation.

Appendix B presents an example of the Cipher, showing values for the State array at the beginning of each round and after the application of each of the four transformations described in the following sections.

```
Cipher(byte in[4*Nb], byte out[4*Nb], word w[Nb*(Nr+1)])
begin
   byte  state[4,Nb]

   state = in

   AddRoundKey(state, w[0, Nb-1])            // See Sec. 5.1.4

   for round = 1 step 1 to Nr-1
      SubBytes(state)                        // See Sec. 5.1.1
      ShiftRows(state)                       // See Sec. 5.1.2
      MixColumns(state)                      // See Sec. 5.1.3
      AddRoundKey(state, w[round*Nb, (round+1)*Nb-1])
   end for

   SubBytes(state)
   ShiftRows(state)
   AddRoundKey(state, w[Nr*Nb, (Nr+1)*Nb-1])

   out = state
end
```

Figure 5. Pseudo Code for the Cipher.[1]

5.1.1 SubBytes() Transformation

The **SubBytes()** transformation is a non-linear byte substitution that operates independently on each byte of the State using a substitution table (S-box). This S-box (Fig. 7), which is invertible, is constructed by composing two transformations:

1. Take the multiplicative inverse in the finite field GF(2^8), described in Sec. 4.2; the element {00} is mapped to itself.
2. Apply the following affine transformation (over GF(2)):

$$b'_i = b_i \oplus b_{(i+4)\bmod 8} \oplus b_{(i+5)\bmod 8} \oplus b_{(i+6)\bmod 8} \oplus b_{(i+7)\bmod 8} \oplus c_i \qquad (5.1)$$

for $0 \le i < 8$, where b_i is the i^{th} bit of the byte, and c_i is the i^{th} bit of a byte c with the value {63} or {01100011}. Here and elsewhere, a prime on a variable (e.g., b') indicates that the variable is to be updated with the value on the right.

In matrix form, the affine transformation element of the S-box can be expressed as:

[1] The various transformations (e.g., **SubBytes()**, **ShiftRows()**, etc.) act upon the State array that is addressed by the 'state' pointer. **AddRoundKey()** uses an additional pointer to address the Round Key.

$$\begin{bmatrix} b_0' \\ b_1' \\ b_2' \\ b_3' \\ b_4' \\ b_5' \\ b_6' \\ b_7' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} . \qquad (5.2)$$

Figure 6 illustrates the effect of the **SubBytes()** transformation on the State.

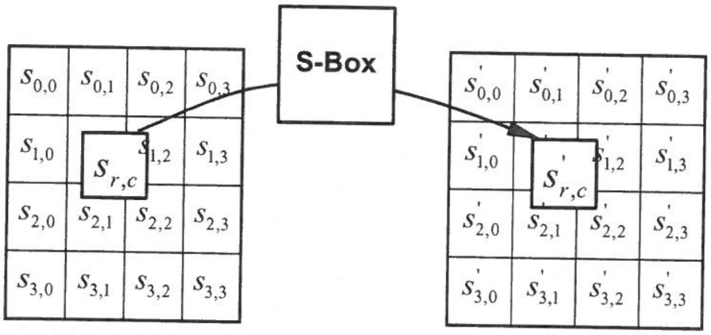


Figure 6. SubBytes() applies the S-box to each byte of the State.

The S-box used in the **SubBytes()** transformation is presented in hexadecimal form in Fig. 7. For example, if $s_{1,1} = \{53\}$, then the substitution value would be determined by the intersection of the row with index '5' and the column with index '3' in Fig. 7. This would result in $s'_{1,1}$ having a value of {ed}.

		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Figure 7. S-box: substitution values for the byte xy (in hexadecimal format).

5.1.2 `ShiftRows()` Transformation

In the **`ShiftRows()`** transformation, the bytes in the last three rows of the State are cyclically shifted over different numbers of bytes (offsets). The first row, $r = 0$, is not shifted.

Specifically, the **`ShiftRows()`** transformation proceeds as follows:

$$s'_{r,c} = s_{r,(c+shift(r,Nb)) \bmod Nb} \quad \text{for } 0 < r < 4 \quad \text{and} \quad 0 \le c < \boldsymbol{Nb}, \qquad (5.3)$$

where the shift value $shift(r,Nb)$ depends on the row number, r, as follows (recall that $\boldsymbol{Nb} = 4$):

$$shift(1,4) = 1; \quad shift(2,4) = 2; \quad shift(3,4) = 3. \qquad (5.4)$$

This has the effect of moving bytes to "lower" positions in the row (i.e., lower values of c in a given row), while the "lowest" bytes wrap around into the "top" of the row (i.e., higher values of c in a given row).

Figure 8 illustrates the **`ShiftRows()`** transformation.

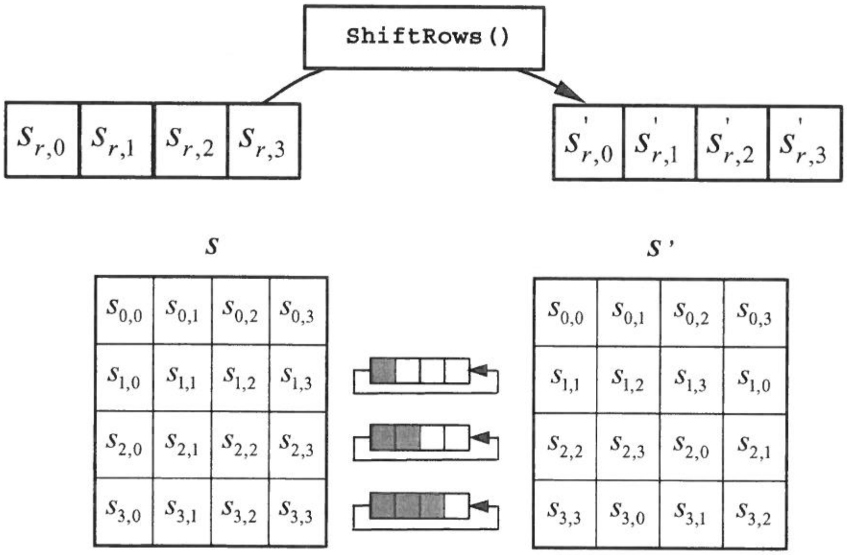

ShiftRows()
$s_{r,0}$ $s_{r,1}$ $s_{r,2}$ $s_{r,3}$
$s'_{r,0}$ $s'_{r,1}$ $s'_{r,2}$ $s'_{r,3}$
S
S'
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,1}$ $s_{1,2}$ $s_{1,3}$ $s_{1,0}$
$s_{2,2}$ $s_{2,3}$ $s_{2,0}$ $s_{2,1}$
$s_{3,3}$ $s_{3,0}$ $s_{3,1}$ $s_{3,2}$

Figure 8. `ShiftRows()` cyclically shifts the last three rows in the State.

5.1.3 `MixColumns()` Transformation

The **`MixColumns()`** transformation operates on the State column-by-column, treating each column as a four-term polynomial as described in Sec. 4.3. The columns are considered as polynomials over GF(2^8) and multiplied modulo $x^4 + 1$ with a fixed polynomial $a(x)$, given by

$$a(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}. \qquad (5.5)$$

As described in Sec. 4.3, this can be written as a matrix multiplication. Let

$s'(x) = a(x) \otimes s(x)$:

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix} \quad \text{for } 0 \le c < \boldsymbol{Nb}. \qquad (5.6)$$

As a result of this multiplication, the four bytes in a column are replaced by the following:

$$s'_{0,c} = (\{02\} \bullet s_{0,c}) \oplus (\{03\} \bullet s_{1,c}) \oplus s_{2,c} \oplus s_{3,c}$$

$$s'_{1,c} = s_{0,c} \oplus (\{02\} \bullet s_{1,c}) \oplus (\{03\} \bullet s_{2,c}) \oplus s_{3,c}$$

$$s'_{2,c} = s_{0,c} \oplus s_{1,c} \oplus (\{02\} \bullet s_{2,c}) \oplus (\{03\} \bullet s_{3,c})$$

$$s'_{3,c} = (\{03\} \bullet s_{0,c}) \oplus s_{1,c} \oplus s_{2,c} \oplus (\{02\} \bullet s_{3,c}).$$

Figure 9 illustrates the **`MixColumns()`** transformation.

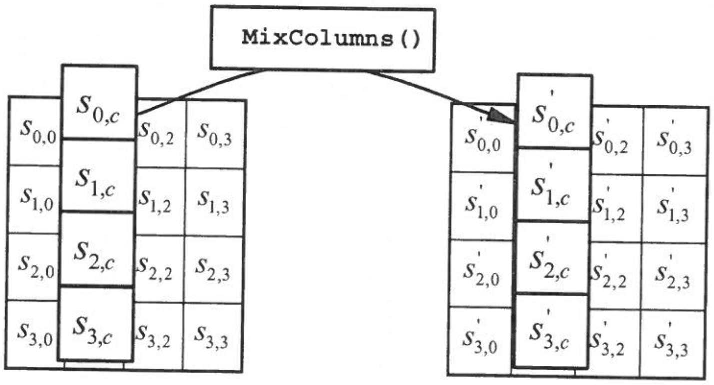


Figure 9. `MixColumns()` operates on the State column-by-column.

5.1.4 `AddRoundKey()` Transformation

In the **`AddRoundKey()`** transformation, a Round Key is added to the State by a simple bitwise XOR operation. Each Round Key consists of ***Nb*** words from the key schedule (described in Sec. 5.2). Those ***Nb*** words are each added into the columns of the State, such that

$$[s'_{0,c}, s'_{1,c}, s'_{2,c}, s'_{3,c}] = [s_{0,c}, s_{1,c}, s_{2,c}, s_{3,c}] \oplus [w_{round*Nb+c}] \quad \text{for } 0 \le c < \boldsymbol{Nb}, \qquad (5.7)$$

where $[w_i]$ are the key schedule words described in Sec. 5.2, and *round* is a value in the range $0 \le$ *round* $\le$***Nr***. In the Cipher, the initial Round Key addition occurs when *round* = 0, prior to the first application of the round function (see Fig. 5). The application of the **`AddRoundKey()`** transformation to the ***Nr*** rounds of the Cipher occurs when $1 \le$ *round* $\le$***Nr***.

The action of this transformation is illustrated in Fig. 10, where *l* = *round* * ***Nb***. The byte address within words of the key schedule was described in Sec. 3.1.

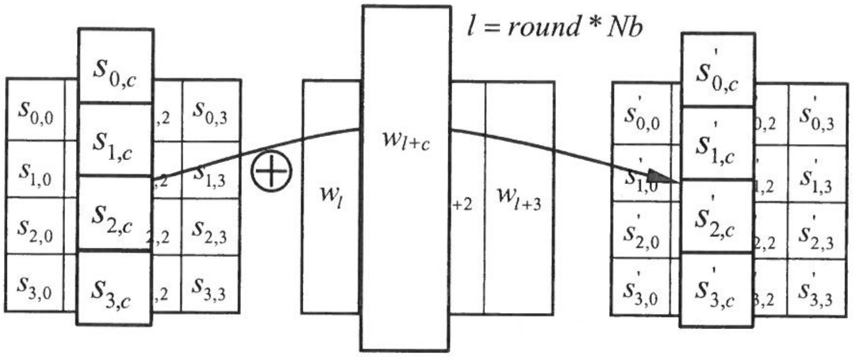


Figure 10. `AddRoundKey()` XORs each column of the State with a word from the key schedule.

5.2 Key Expansion

The AES algorithm takes the Cipher Key, ***K***, and performs a Key Expansion routine to generate a key schedule. The Key Expansion generates a total of ***Nb*** (***Nr*** + 1) words: the algorithm requires an initial set of ***Nb*** words, and each of the ***Nr*** rounds requires ***Nb*** words of key data. The resulting key schedule consists of a linear array of 4-byte words, denoted $[w_i]$, with i in the range $0 \leq i < Nb(Nr + 1)$.

The expansion of the input key into the key schedule proceeds according to the pseudo code in Fig. 11.

`SubWord()` is a function that takes a four-byte input word and applies the S-box (Sec. 5.1.1, Fig. 7) to each of the four bytes to produce an output word. The function **`RotWord()`** takes a word $[a_0,a_1,a_2,a_3]$ as input, performs a cyclic permutation, and returns the word $[a_1,a_2,a_3,a_0]$. The round constant word array, **`Rcon[i]`**, contains the values given by $[x^{i-1},\{00\},\{00\},\{00\}]$, with x^{i-1} being powers of x (x is denoted as {02}) in the field GF(2^8), as discussed in Sec. 4.2 (note that i starts at 1, not 0).

From Fig. 11, it can be seen that the first ***Nk*** words of the expanded key are filled with the Cipher Key. Every following word, **w[i]**, is equal to the XOR of the previous word, **w[i-1]**, and the word ***Nk*** positions earlier, **w[i-*Nk*]**. For words in positions that are a multiple of ***Nk***, a transformation is applied to **w[i-1]** prior to the XOR, followed by an XOR with a round constant, **`Rcon[i]`**. This transformation consists of a cyclic shift of the bytes in a word (**`RotWord()`**), followed by the application of a table lookup to all four bytes of the word (**`SubWord()`**).

It is important to note that the Key Expansion routine for 256-bit Cipher Keys (***Nk*** = 8) is slightly different than for 128- and 192-bit Cipher Keys. If ***Nk*** = 8 and `i-4` is a multiple of ***Nk***, then **`SubWord()`** is applied to **w[i-1]** prior to the XOR.

```
KeyExpansion(byte key[4*Nk], word w[Nb*(Nr+1)], Nk)
begin
   word  temp

   i = 0

   while (i < Nk)
      w[i] = word(key[4*i], key[4*i+1], key[4*i+2], key[4*i+3])
      i = i+1
   end while

   i = Nk

   while (i < Nb * (Nr+1)]
      temp = w[i-1]
      if (i mod Nk = 0)
         temp = SubWord(RotWord(temp)) xor Rcon[i/Nk]
      else if (Nk > 6 and i mod Nk = 4)
         temp = SubWord(temp)
      end if
      w[i] = w[i-Nk] xor temp
      i = i + 1
   end while
end
```

Note that *Nk*=4, 6, and 8 do not all have to be implemented; they are all included in the conditional statement above for conciseness. Specific implementation requirements for the Cipher Key are presented in Sec. 6.1.

Figure 11. Pseudo Code for Key Expansion.[2]

Appendix A presents examples of the Key Expansion.

5.3 Inverse Cipher

The Cipher transformations in Sec. 5.1 can be inverted and then implemented in reverse order to produce a straightforward Inverse Cipher for the AES algorithm. The individual transformations used in the Inverse Cipher - **`InvShiftRows()`**, **`InvSubBytes()`**,**`InvMixColumns()`**, and **`AddRoundKey()`** – process the State and are described in the following subsections.

The Inverse Cipher is described in the pseudo code in Fig. 12. In Fig. 12, the array **`w[]`** contains the key schedule, which was described previously in Sec. 5.2.

[2] The functions **`SubWord()`** and **`RotWord()`** return a result that is a transformation of the function input, whereas the transformations in the Cipher and Inverse Cipher (e.g., **`ShiftRows()`**, **`SubBytes()`**, etc.) transform the State array that is addressed by the '`state`' pointer.

```
InvCipher(byte in[4*Nb], byte out[4*Nb], word w[Nb*(Nr+1)])
begin
   byte  state[4,Nb]

   state = in

   AddRoundKey(state, w[Nr*Nb, (Nr+1)*Nb-1]) // See Sec. 5.1.4

   for round = Nr-1 step -1 downto 1
      InvShiftRows(state)                    // See Sec. 5.3.1
      InvSubBytes(state)                     // See Sec. 5.3.2
      AddRoundKey(state, w[round*Nb, (round+1)*Nb-1])
      InvMixColumns(state)                   // See Sec. 5.3.3
   end for

   InvShiftRows(state)
   InvSubBytes(state)
   AddRoundKey(state, w[0, Nb-1])

   out = state
end
```

Figure 12. Pseudo Code for the Inverse Cipher.[3]

5.3.1 `InvShiftRows()` Transformation

`InvShiftRows()` is the inverse of the **`ShiftRows()`** transformation. The bytes in the last three rows of the State are cyclically shifted over different numbers of bytes (offsets). The first row, $r = 0$, is not shifted. The bottom three rows are cyclically shifted by $\boldsymbol{Nb} - shift(r, Nb)$ bytes, where the shift value *shift(r,Nb)* depends on the row number, and is given in equation (5.4) (see Sec. 5.1.2).

Specifically, the **`InvShiftRows()`** transformation proceeds as follows:

$$s'_{r,(c+shift(r,Nb)) \bmod Nb} = s_{r,c} \quad \text{for } 0 < r < 4 \quad \text{and} \quad 0 \le c < \boldsymbol{Nb} \qquad (5.8)$$

Figure 13 illustrates the **`InvShiftRows()`** transformation.

[3] The various transformations (e.g., **`InvSubBytes()`**, **`InvShiftRows()`**, etc.) act upon the State array that is addressed by the '`state`' pointer. **`AddRoundKey()`** uses an additional pointer to address the Round Key.

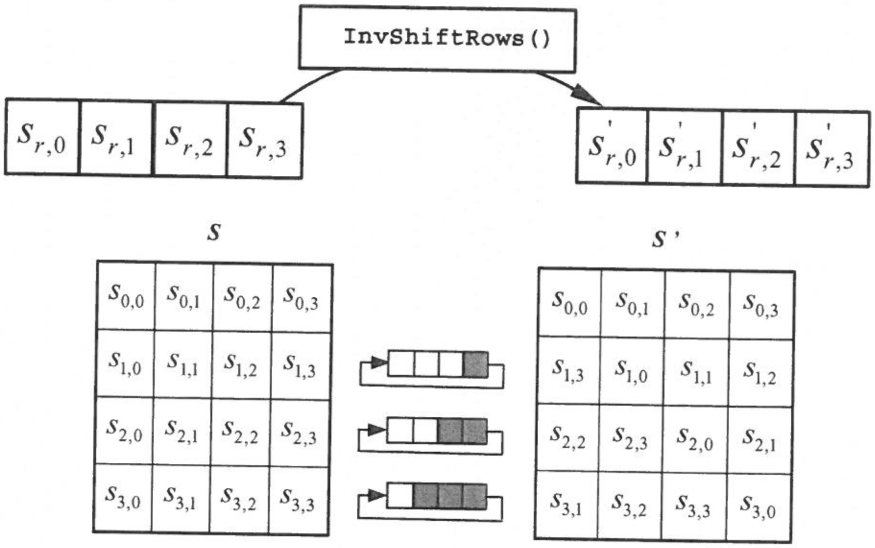

InvShiftRows()
$s_{r,0}$
$s_{r,1}$
$s_{r,2}$
$s_{r,3}$
$s'_{r,0}$
$s'_{r,1}$
$s'_{r,2}$
$s'_{r,3}$
S
S'
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,0}$ $s_{1,1}$ $s_{1,2}$ $s_{1,3}$
$s_{2,0}$ $s_{2,1}$ $s_{2,2}$ $s_{2,3}$
$s_{3,0}$ $s_{3,1}$ $s_{3,2}$ $s_{3,3}$
$s_{0,0}$ $s_{0,1}$ $s_{0,2}$ $s_{0,3}$
$s_{1,3}$ $s_{1,0}$ $s_{1,1}$ $s_{1,2}$
$s_{2,2}$ $s_{2,3}$ $s_{2,0}$ $s_{2,1}$
$s_{3,1}$ $s_{3,2}$ $s_{3,3}$ $s_{3,0}$

Figure 13. `InvShiftRows()` **cyclically shifts the last three rows in the State.**

5.3.2 `InvSubBytes()` Transformation

`InvSubBytes()` is the inverse of the byte substitution transformation, in which the inverse S-box is applied to each byte of the State. This is obtained by applying the inverse of the affine transformation (5.1) followed by taking the multiplicative inverse in GF(2^8).

The inverse S-box used in the `InvSubBytes()` transformation is presented in Fig. 14:

		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	52	09	6a	d5	30	36	a5	38	bf	40	a3	9e	81	f3	d7	fb
	1	7c	e3	39	82	9b	2f	ff	87	34	8e	43	44	c4	de	e9	cb
	2	54	7b	94	32	a6	c2	23	3d	ee	4c	95	0b	42	fa	c3	4e
	3	08	2e	a1	66	28	d9	24	b2	76	5b	a2	49	6d	8b	d1	25
	4	72	f8	f6	64	86	68	98	16	d4	a4	5c	cc	5d	65	b6	92
	5	6c	70	48	50	fd	ed	b9	da	5e	15	46	57	a7	8d	9d	84
	6	90	d8	ab	00	8c	bc	d3	0a	f7	e4	58	05	b8	b3	45	06
	7	d0	2c	1e	8f	ca	3f	0f	02	c1	af	bd	03	01	13	8a	6b
	8	3a	91	11	41	4f	67	dc	ea	97	f2	cf	ce	f0	b4	e6	73
	9	96	ac	74	22	e7	ad	35	85	e2	f9	37	e8	1c	75	df	6e
	a	47	f1	1a	71	1d	29	c5	89	6f	b7	62	0e	aa	18	be	1b
	b	fc	56	3e	4b	c6	d2	79	20	9a	db	c0	fe	78	cd	5a	f4
	c	1f	dd	a8	33	88	07	c7	31	b1	12	10	59	27	80	ec	5f
	d	60	51	7f	a9	19	b5	4a	0d	2d	e5	7a	9f	93	c9	9c	ef
	e	a0	e0	3b	4d	ae	2a	f5	b0	c8	eb	bb	3c	83	53	99	61
	f	17	2b	04	7e	ba	77	d6	26	e1	69	14	63	55	21	0c	7d

Figure 14. Inverse S-box: substitution values for the byte `xy` **(in hexadecimal format).**

5.3.3 `InvMixColumns()` Transformation

`InvMixColumns()` is the inverse of the **`MixColumns()`** transformation. **`InvMixColumns()`** operates on the State column-by-column, treating each column as a four-term polynomial as described in Sec. 4.3. The columns are considered as polynomials over GF(2^8) and multiplied modulo $x^4 + 1$ with a fixed polynomial $a^{-1}(x)$, given by

$$a^{-1}(x) = \{\texttt{0b}\}x^3 + \{\texttt{0d}\}x^2 + \{\texttt{09}\}x + \{\texttt{0e}\}. \qquad (5.9)$$

As described in Sec. 4.3, this can be written as a matrix multiplication. Let

$s'(x) = a^{-1}(x) \otimes s(x)$:

$$\begin{bmatrix} s'_{0,c} \\ s'_{1,c} \\ s'_{2,c} \\ s'_{3,c} \end{bmatrix} = \begin{bmatrix} 0e & 0b & 0d & 09 \\ 09 & 0e & 0b & 0d \\ 0d & 09 & 0e & 0b \\ 0b & 0d & 09 & 0e \end{bmatrix} \begin{bmatrix} s_{0,c} \\ s_{1,c} \\ s_{2,c} \\ s_{3,c} \end{bmatrix} \quad \text{for } 0 \le c < \boldsymbol{Nb}. \qquad (5.10)$$

As a result of this multiplication, the four bytes in a column are replaced by the following:

$$s'_{0,c} = (\{\texttt{0e}\} \bullet s_{0,c}) \oplus (\{\texttt{0b}\} \bullet s_{1,c}) \oplus (\{\texttt{0d}\} \bullet s_{2,c}) \oplus (\{\texttt{09}\} \bullet s_{3,c})$$

$$s'_{1,c} = (\{\texttt{09}\} \bullet s_{0,c}) \oplus (\{\texttt{0e}\} \bullet s_{1,c}) \oplus (\{\texttt{0b}\} \bullet s_{2,c}) \oplus (\{\texttt{0d}\} \bullet s_{3,c})$$

$$s'_{2,c} = (\{\texttt{0d}\} \bullet s_{0,c}) \oplus (\{\texttt{09}\} \bullet s_{1,c}) \oplus (\{\texttt{0e}\} \bullet s_{2,c}) \oplus (\{\texttt{0b}\} \bullet s_{3,c})$$

$$s'_{3,c} = (\{\texttt{0b}\} \bullet s_{0,c}) \oplus (\{\texttt{0d}\} \bullet s_{1,c}) \oplus (\{\texttt{09}\} \bullet s_{2,c}) \oplus (\{\texttt{0e}\} \bullet s_{3,c})$$

5.3.4 Inverse of the `AddRoundKey()` Transformation

`AddRoundKey()`, which was described in Sec. 5.1.4, is its own inverse, since it only involves an application of the XOR operation.

5.3.5 Equivalent Inverse Cipher

In the straightforward Inverse Cipher presented in Sec. 5.3 and Fig. 12, the sequence of the transformations differs from that of the Cipher, while the form of the key schedules for encryption and decryption remains the same. However, several properties of the AES algorithm allow for an Equivalent Inverse Cipher that has the same sequence of transformations as the Cipher (with the transformations replaced by their inverses). This is accomplished with a change in the key schedule.

The two properties that allow for this Equivalent Inverse Cipher are as follows:

1. The **`SubBytes()`** and **`ShiftRows()`** transformations commute; that is, a **`SubBytes()`** transformation immediately followed by a **`ShiftRows()`** transformation is equivalent to a **`ShiftRows()`** transformation immediately followed buy a **`SubBytes()`** transformation. The same is true for their inverses, **`InvSubBytes()`** and **`InvShiftRows.`**

2. The column mixing operations - **MixColumns()** and **InvMixColumns()** - are linear with respect to the column input, which means

```
InvMixColumns(state XOR Round Key) =
                InvMixColumns(state) XOR InvMixColumns(Round Key).
```

These properties allow the order of **InvSubBytes()** and **InvShiftRows()** transformations to be reversed. The order of the **AddRoundKey()** and **InvMixColumns()** transformations can also be reversed, provided that the columns (words) of the decryption key schedule are modified using the **InvMixColumns()** transformation.

The equivalent inverse cipher is defined by reversing the order of the **InvSubBytes()** and **InvShiftRows()** transformations shown in Fig. 12, and by reversing the order of the **AddRoundKey()** and **InvMixColumns()** transformations used in the "round loop" after first modifying the decryption key schedule for *round* = 1 to *Nr*-1 using the **InvMixColumns()** transformation. The first and last *Nb* words of the decryption key schedule shall *not* be modified in this manner.

Given these changes, the resulting Equivalent Inverse Cipher offers a more efficient structure than the Inverse Cipher described in Sec. 5.3 and Fig. 12. Pseudo code for the Equivalent Inverse Cipher appears in Fig. 15. (The word array **dw[]** contains the modified decryption key schedule. The modification to the Key Expansion routine is also provided in Fig. 15.)

```
EqInvCipher(byte in[4*Nb], byte out[4*Nb], word dw[Nb*(Nr+1)])
begin
   byte  state[4,Nb]

   state = in

   AddRoundKey(state, dw[Nr*Nb, (Nr+1)*Nb-1])

   for round = Nr-1 step -1 downto 1
      InvSubBytes(state)
      InvShiftRows(state)
      InvMixColumns(state)
      AddRoundKey(state, dw[round*Nb, (round+1)*Nb-1])
   end for

   InvSubBytes(state)
   InvShiftRows(state)
   AddRoundKey(state, dw[0, Nb-1])

   out = state
end
```

For the Equivalent Inverse Cipher, the following pseudo code is added at the end of the Key Expansion routine (Sec. 5.2):

```
   for i = 0 step 1 to (Nr+1)*Nb-1
      dw[i] = w[i]
   end for


   for round = 1 step 1 to Nr-1
      InvMixColumns(dw[round*Nb, (round+1)*Nb-1])     // note change of
type
   end for
```

Note that, since InvMixColumns operates on a two-dimensional array of bytes while the Round Keys are held in an array of words, the call to InvMixColumns in this code sequence involves a change of type (i.e. the input to InvMixColumns() is normally the State array, which is considered to be a two-dimensional array of bytes, whereas the input here is a Round Key computed as a one-dimensional array of words).

Figure 15. Pseudo Code for the Equivalent Inverse Cipher.

6. Implementation Issues

6.1 Key Length Requirements

An implementation of the AES algorithm shall support *at least one* of the three key lengths specified in Sec. 5: 128, 192, or 256 bits (i.e., Nk = 4, 6, or 8, respectively). Implementations may optionally support two or three key lengths, which may promote the interoperability of algorithm implementations.

6.2 Keying Restrictions

No weak or semi-weak keys have been identified for the AES algorithm, and there is no restriction on key selection.

6.3 Parameterization of Key Length, Block Size, and Round Number

This standard explicitly defines the allowed values for the key length (***Nk***), block size (***Nb***), and number of rounds (***Nr***) – see Fig. 4. However, future reaffirmations of this standard could include changes or additions to the allowed values for those parameters. Therefore, implementers may choose to design their AES implementations with future flexibility in mind.

6.4 Implementation Suggestions Regarding Various Platforms

Implementation variations are possible that may, in many cases, offer performance or other advantages. Given the same input key and data (plaintext or ciphertext), any implementation that produces the same output (ciphertext or plaintext) as the algorithm specified in this standard is an acceptable implementation of the AES.

Reference [3] and other papers located at Ref. [1] include suggestions on how to efficiently implement the AES algorithm on a variety of platforms.

Appendix A - Key Expansion Examples

This appendix shows the development of the key schedule for various key sizes. Note that multi-byte values are presented using the notation described in Sec. 3. The intermediate values produced during the development of the key schedule (see Sec. 5.2) are given in the following table (all values are in hexadecimal format, with the exception of the index column (i)).

A.1 Expansion of a 128-bit Cipher Key

This section contains the key expansion of the following cipher key:

```
Cipher Key = 2b 7e 15 16 28 ae d2 a6 ab f7 15 88 09 cf 4f 3c
```

for *Nk* = 4, which results in

w_0 = 2b7e1516 w_1 = 28aed2a6 w_2 = abf71588 w_3 = 09cf4f3c

i (dec)	temp	After RotWord()	After SubWord()	Rcon[i/Nk]	After XOR with Rcon	w[i-Nk]	w[i]= temp XOR w[i-Nk]
4	09cf4f3c	cf4f3c09	8a84eb01	01000000	8b84eb01	2b7e1516	a0fafe17
5	a0fafe17					28aed2a6	88542cb1
6	88542cb1					abf71588	23a33939
7	23a33939					09cf4f3c	2a6c7605
8	2a6c7605	6c76052a	50386be5	02000000	52386be5	a0fafe17	f2c295f2
9	f2c295f2					88542cb1	7a96b943
10	7a96b943					23a33939	5935807a
11	5935807a					2a6c7605	7359f67f
12	7359f67f	59f67f73	cb42d28f	04000000	cf42d28f	f2c295f2	3d80477d
13	3d80477d					7a96b943	4716fe3e
14	4716fe3e					5935807a	1e237e44
15	1e237e44					7359f67f	6d7a883b
16	6d7a883b	7a883b6d	dac4e23c	08000000	d2c4e23c	3d80477d	ef44a541
17	ef44a541					4716fe3e	a8525b7f
18	a8525b7f					1e237e44	b671253b
19	b671253b					6d7a883b	db0bad00
20	db0bad00	0bad00db	2b9563b9	10000000	3b9563b9	ef44a541	d4d1c6f8
21	d4d1c6f8					a8525b7f	7c839d87
22	7c839d87					b671253b	caf2b8bc
23	caf2b8bc					db0bad00	11f915bc

24	11f915bc	f915bc11	99596582	20000000	b9596582	d4d1c6f8	6d88a37a
25	6d88a37a					7c839d87	110b3efd
26	110b3efd					caf2b8bc	dbf98641
27	dbf98641					11f915bc	ca0093fd
28	ca0093fd	0093fdca	63dc5474	40000000	23dc5474	6d88a37a	4e54f70e
29	4e54f70e					110b3efd	5f5fc9f3
30	5f5fc9f3					dbf98641	84a64fb2
31	84a64fb2					ca0093fd	4ea6dc4f
32	4ea6dc4f	a6dc4f4e	2486842f	80000000	a486842f	4e54f70e	ead27321
33	ead27321					5f5fc9f3	b58dbad2
34	b58dbad2					84a64fb2	312bf560
35	312bf560					4ea6dc4f	7f8d292f
36	7f8d292f	8d292f7f	5da515d2	1b000000	46a515d2	ead27321	ac7766f3
37	ac7766f3					b58dbad2	19fadc21
38	19fadc21					312bf560	28d12941
39	28d12941					7f8d292f	575c006e
40	575c006e	5c006e57	4a639f5b	36000000	7c639f5b	ac7766f3	d014f9a8
41	d014f9a8					19fadc21	c9ee2589
42	c9ee2589					28d12941	e13f0cc8
43	e13f0cc8					575c006e	b6630ca6

A.2 Expansion of a 192-bit Cipher Key

This section contains the key expansion of the following cipher key:

```
Cipher Key =        8e 73 b0 f7 da 0e 64 52 c8 10 f3 2b
                    80 90 79 e5 62 f8 ea d2 52 2c 6b 7b
```

for $Nk = 6$, which results in

w_0 = 8e73b0f7 w_1 = da0e6452 w_2 = c810f32b w_3 = 809079e5

w_4 = 62f8ead2 w_5 = 522c6b7b

i (dec)	temp	After RotWord()	After SubWord()	Rcon[i/Nk]	After XOR with Rcon	w[i-Nk]	w[i]= temp XOR w[i-Nk]
6	522c6b7b	2c6b7b52	717f2100	01000000	707f2100	8e73b0f7	fe0c91f7
7	fe0c91f7					da0e6452	2402f5a5
8	2402f5a5					c810f32b	ec12068e

9	ec12068e					809079e5	6c827f6b
10	6c827f6b					62f8ead2	0e7a95b9
11	0e7a95b9					522c6b7b	5c56fec2
12	5c56fec2	56fec25c	b1bb254a	02000000	b3bb254a	fe0c91f7	4db7b4bd
13	4db7b4bd					2402f5a5	69b54118
14	69b54118					ec12068e	85a74796
15	85a74796					6c827f6b	e92538fd
16	e92538fd					0e7a95b9	e75fad44
17	e75fad44					5c56fec2	bb095386
18	bb095386	095386bb	01ed44ea	04000000	05ed44ea	4db7b4bd	485af057
19	485af057					69b54118	21efb14f
20	21efb14f					85a74796	a448f6d9
21	a448f6d9					e92538fd	4d6dce24
22	4d6dce24					e75fad44	aa326360
23	aa326360					bb095386	113b30e6
24	113b30e6	3b30e611	e2048e82	08000000	ea048e82	485af057	a25e7ed5
25	a25e7ed5					21efb14f	83b1cf9a
26	83b1cf9a					a448f6d9	27f93943
27	27f93943					4d6dce24	6a94f767
28	6a94f767					aa326360	c0a69407
29	c0a69407					113b30e6	d19da4e1
30	d19da4e1	9da4e1d1	5e49f83e	10000000	4e49f83e	a25e7ed5	ec1786eb
31	ec1786eb					83b1cf9a	6fa64971
32	6fa64971					27f93943	485f7032
33	485f7032					6a94f767	22cb8755
34	22cb8755					c0a69407	e26d1352
35	e26d1352					d19da4e1	33f0b7b3
36	33f0b7b3	f0b7b333	8ca96dc3	20000000	aca96dc3	ec1786eb	40beeb28
37	40beeb28					6fa64971	2f18a259
38	2f18a259					485f7032	6747d26b
39	6747d26b					22cb8755	458c553e
40	458c553e					e26d1352	a7e1466c
41	a7e1466c					33f0b7b3	9411f1df
42	9411f1df	11f1df94	82a19e22	40000000	c2a19e22	40beeb28	821f750a
43	821f750a					2f18a259	ad07d753

44	ad07d753					6747d26b	ca400538
45	ca400538					458c553e	8fcc5006
46	8fcc5006					a7e1466c	282d166a
47	282d166a					9411f1df	bc3ce7b5
48	bc3ce7b5	3ce7b5bc	eb94d565	80000000	6b94d565	821f750a	e98ba06f
49	e98ba06f					ad07d753	448c773c
50	448c773c					ca400538	8ecc7204
51	8ecc7204					8fcc5006	01002202

A.3 Expansion of a 256-bit Cipher Key

This section contains the key expansion of the following cipher key:

```
Cipher Key =    60 3d eb 10 15 ca 71 be 2b 73 ae f0 85 7d 77 81
                1f 35 2c 07 3b 61 08 d7 2d 98 10 a3 09 14 df f4
```

for *Nk* = 8, which results in

w_0 = 603deb10	w_1 = 15ca71be	w_2 = 2b73aef0	w_3 = 857d7781
w_4 = 1f352c07	w_5 = 3b6108d7	w_6 = 2d9810a3	w_7 = 0914dff4

i (dec)	temp	After RotWord()	After SubWord()	Rcon[i/Nk]	After XOR with Rcon	w[i-Nk]	w[i]= temp XOR w[i-Nk]
8	0914dff4	14dff409	fa9ebf01	01000000	fb9ebf01	603deb10	9ba35411
9	9ba35411					15ca71be	8e6925af
10	8e6925af					2b73aef0	a51a8b5f
11	a51a8b5f					857d7781	2067fcde
12	2067fcde		b785b01d			1f352c07	a8b09c1a
13	a8b09c1a					3b6108d7	93d194cd
14	93d194cd					2d9810a3	be49846e
15	be49846e					0914dff4	b75d5b9a
16	b75d5b9a	5d5b9ab7	4c39b8a9	02000000	4e39b8a9	9ba35411	d59aecb8
17	d59aecb8					8e6925af	5bf3c917
18	5bf3c917					a51a8b5f	fee94248
19	fee94248					2067fcde	de8ebe96
20	de8ebe96		1d19ae90			a8b09c1a	b5a9328a
21	b5a9328a					93d194cd	2678a647
22	2678a647					be49846e	98312229

23	98312229					b75d5b9a	2f6c79b3
24	2f6c79b3	6c79b32f	50b66d15	04000000	54b66d15	d59aecb8	812c81ad
25	812c81ad					5bf3c917	dadf48ba
26	dadf48ba					fee94248	24360af2
27	24360af2					de8ebe96	fab8b464
28	fab8b464		2d6c8d43			b5a9328a	98c5bfc9
29	98c5bfc9					2678a647	bebd198e
30	bebd198e					98312229	268c3ba7
31	268c3ba7					2f6c79b3	09e04214
32	09e04214	e0421409	e12cfa01	08000000	e92cfa01	812c81ad	68007bac
33	68007bac					dadf48ba	b2df3316
34	b2df3316					24360af2	96e939e4
35	96e939e4					fab8b464	6c518d80
36	6c518d80		50d15dcd			98c5bfc9	c814e204
37	c814e204					bebd198e	76a9fb8a
38	76a9fb8a					268c3ba7	5025c02d
39	5025c02d					09e04214	59c58239
40	59c58239	c5823959	a61312cb	10000000	b61312cb	68007bac	de136967
41	de136967					b2df3316	6ccc5a71
42	6ccc5a71					96e939e4	fa256395
43	fa256395					6c518d80	9674ee15
44	9674ee15		90922859			c814e204	5886ca5d
45	5886ca5d					76a9fb8a	2e2f31d7
46	2e2f31d7					5025c02d	7e0af1fa
47	7e0af1fa					59c58239	27cf73c3
48	27cf73c3	cf73c327	8a8f2ecc	20000000	aa8f2ecc	de136967	749c47ab
49	749c47ab					6ccc5a71	18501dda
50	18501dda					fa256395	e2757e4f
51	e2757e4f					9674ee15	7401905a
52	7401905a		927c60be			5886ca5d	cafaaae3
53	cafaaae3					2e2f31d7	e4d59b34
54	e4d59b34					7e0af1fa	9adf6ace
55	9adf6ace					27cf73c3	bd10190d
56	bd10190d	10190dbd	cad4d77a	40000000	8ad4d77a	749c47ab	fe4890d1
57	fe4890d1					18501dda	e6188d0b

58	e6188d0b					e2757e4f	046df344
59	046df344					7401905a	706c631e

Appendix B – Cipher Example

The following diagram shows the values in the State array as the Cipher progresses for a block length and a Cipher Key length of 16 bytes each (i.e., *Nb* = 4 and *Nk* = 4).

```
Input =        32 43 f6 a8 88 5a 30 8d 31 31 98 a2 e0 37 07 34

Cipher Key = 2b 7e 15 16 28 ae d2 a6 ab f7 15 88 09 cf 4f 3c
```

The Round Key values are taken from the Key Expansion example in Appendix A.

Round Number	Start of Round	After SubBytes	After ShiftRows	After MixColumns		Round Key Value	
input	32 88 31 e0				⊕	2b 28 ab 09	=
	43 5a 31 37					7e ae f7 cf	
	f6 30 98 07					15 d2 15 4f	
	a8 8d a2 34					16 a6 88 3c	
1	19 a0 9a e9	d4 e0 b8 1e	d4 e0 b8 1e	04 e0 48 28	⊕	a0 88 23 2a	=
	3d f4 c6 f8	27 bf b4 41	bf b4 41 27	66 cb f8 06		fa 54 a3 6c	
	e3 e2 8d 48	11 98 5d 52	5d 52 11 98	81 19 d3 26		fe 2c 39 76	
	be 2b 2a 08	ae f1 e5 30	30 ae f1 e5	e5 9a 7a 4c		17 b1 39 05	
2	a4 68 6b 02	49 45 7f 77	49 45 7f 77	58 1b db 1b	⊕	f2 7a 59 73	=
	9c 9f 5b 6a	de db 39 02	db 39 02 de	4d 4b e7 6b		c2 96 35 59	
	7f 35 ea 50	d2 96 87 53	87 53 d2 96	ca 5a ca b0		95 b9 80 f6	
	f2 2b 43 49	89 f1 1a 3b	3b 89 f1 1a	f1 ac a8 e5		f2 43 7a 7f	
3	aa 61 82 68	ac ef 13 45	ac ef 13 45	75 20 53 bb	⊕	3d 47 1e 6d	=
	8f dd d2 32	73 c1 b5 23	c1 b5 23 73	ec 0b c0 25		80 16 23 7a	
	5f e3 4a 46	cf 11 d6 5a	d6 5a cf 11	09 63 cf d0		47 fe 7e 88	
	03 ef d2 9a	7b df b5 b8	b8 7b df b5	93 33 7c dc		7d 3e 44 3b	
4	48 67 4d d6	52 85 e3 f6	52 85 e3 f6	0f 60 6f 5e	⊕	ef a8 b6 db	=
	6c 1d e3 5f	50 a4 11 cf	a4 11 cf 50	d6 31 c0 b3		44 52 71 0b	
	4e 9d b1 58	2f 5e c8 6a	c8 6a 2f 5e	da 38 10 13		a5 5b 25 ad	
	ee 0d 38 e7	28 d7 07 94	94 28 d7 07	a9 bf 6b 01		41 7f 3b 00	
5	e0 c8 d9 85	e1 e8 35 97	e1 e8 35 97	25 bd b6 4c	⊕	d4 7c ca 11	=
	92 63 b1 b8	4f fb c8 6c	fb c8 6c 4f	d1 11 3a 4c		d1 83 f2 f9	
	7f 63 35 be	d2 fb 96 ae	96 ae d2 fb	a9 d1 33 c0		c6 9d b8 15	
	e8 c0 50 01	9b ba 53 7c	7c 9b ba 53	ad 68 8e b0		f8 87 bc bc	

$$6\quad \begin{array}{|c|c|c|c|}\hline f1 & c1 & 7c & 5d \\ \hline 00 & 92 & c8 & b5 \\ \hline 6f & 4c & 8b & d5 \\ \hline 55 & ef & 32 & 0c \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline a1 & 78 & 10 & 4c \\ \hline 63 & 4f & e8 & d5 \\ \hline a8 & 29 & 3d & 03 \\ \hline fc & df & 23 & fe \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline a1 & 78 & 10 & 4c \\ \hline 4f & e8 & d5 & 63 \\ \hline 3d & 03 & a8 & 29 \\ \hline fe & fc & df & 23 \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline 4b & 2c & 33 & 37 \\ \hline 86 & 4a & 9d & d2 \\ \hline 8d & 89 & f4 & 18 \\ \hline 6d & 80 & e8 & d8 \\ \hline \end{array} \oplus \begin{array}{|c|c|c|c|}\hline 6d & 11 & db & ca \\ \hline 88 & 0b & f9 & 00 \\ \hline a3 & 3e & 86 & 93 \\ \hline 7a & fd & 41 & fd \\ \hline \end{array} =$$

$$7\quad \begin{array}{|c|c|c|c|}\hline 26 & 3d & e8 & fd \\ \hline 0e & 41 & 64 & d2 \\ \hline 2e & b7 & 72 & 8b \\ \hline 17 & 7d & a9 & 25 \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline f7 & 27 & 9b & 54 \\ \hline ab & 83 & 43 & b5 \\ \hline 31 & a9 & 40 & 3d \\ \hline f0 & ff & d3 & 3f \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline f7 & 27 & 9b & 54 \\ \hline 83 & 43 & b5 & ab \\ \hline 40 & 3d & 31 & a9 \\ \hline 3f & f0 & ff & d3 \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline 14 & 46 & 27 & 34 \\ \hline 15 & 16 & 46 & 2a \\ \hline b5 & 15 & 56 & d8 \\ \hline bf & ec & d7 & 43 \\ \hline \end{array} \oplus \begin{array}{|c|c|c|c|}\hline 4e & 5f & 84 & 4e \\ \hline 54 & 5f & a6 & a6 \\ \hline f7 & c9 & 4f & dc \\ \hline 0e & f3 & b2 & 4f \\ \hline \end{array} =$$

$$8\quad \begin{array}{|c|c|c|c|}\hline 5a & 19 & a3 & 7a \\ \hline 41 & 49 & e0 & 8c \\ \hline 42 & dc & 19 & 04 \\ \hline b1 & 1f & 65 & 0c \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline be & d4 & 0a & da \\ \hline 83 & 3b & e1 & 64 \\ \hline 2c & 86 & d4 & f2 \\ \hline c8 & c0 & 4d & fe \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline be & d4 & 0a & da \\ \hline 3b & e1 & 64 & 83 \\ \hline d4 & f2 & 2c & 86 \\ \hline fe & c8 & c0 & 4d \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline 00 & b1 & 54 & fa \\ \hline 51 & c8 & 76 & 1b \\ \hline 2f & 89 & 6d & 99 \\ \hline d1 & ff & cd & ea \\ \hline \end{array} \oplus \begin{array}{|c|c|c|c|}\hline ea & b5 & 31 & 7f \\ \hline d2 & 8d & 2b & 8d \\ \hline 73 & ba & f5 & 29 \\ \hline 21 & d2 & 60 & 2f \\ \hline \end{array} =$$

$$9\quad \begin{array}{|c|c|c|c|}\hline ea & 04 & 65 & 85 \\ \hline 83 & 45 & 5d & 96 \\ \hline 5c & 33 & 98 & b0 \\ \hline f0 & 2d & ad & c5 \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline 87 & f2 & 4d & 97 \\ \hline ec & 6e & 4c & 90 \\ \hline 4a & c3 & 46 & e7 \\ \hline 8c & d8 & 95 & a6 \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline 87 & f2 & 4d & 97 \\ \hline 6e & 4c & 90 & ec \\ \hline 46 & e7 & 4a & c3 \\ \hline a6 & 8c & d8 & 95 \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline 47 & 40 & a3 & 4c \\ \hline 37 & d4 & 70 & 9f \\ \hline 94 & e4 & 3a & 42 \\ \hline ed & a5 & a6 & bc \\ \hline \end{array} \oplus \begin{array}{|c|c|c|c|}\hline ac & 19 & 28 & 57 \\ \hline 77 & fa & d1 & 5c \\ \hline 66 & dc & 29 & 00 \\ \hline f3 & 21 & 41 & 6e \\ \hline \end{array} =$$

$$10\quad \begin{array}{|c|c|c|c|}\hline eb & 59 & 8b & 1b \\ \hline 40 & 2e & a1 & c3 \\ \hline f2 & 38 & 13 & 42 \\ \hline 1e & 84 & e7 & d2 \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline e9 & cb & 3d & af \\ \hline 09 & 31 & 32 & 2e \\ \hline 89 & 07 & 7d & 2c \\ \hline 72 & 5f & 94 & b5 \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline e9 & cb & 3d & af \\ \hline 31 & 32 & 2e & 09 \\ \hline 7d & 2c & 89 & 07 \\ \hline b5 & 72 & 5f & 94 \\ \hline \end{array} \quad \begin{array}{|c|c|c|c|}\hline & & & \\ \hline & & & \\ \hline & & & \\ \hline & & & \\ \hline \end{array} \oplus \begin{array}{|c|c|c|c|}\hline d0 & c9 & e1 & b6 \\ \hline 14 & ee & 3f & 63 \\ \hline f9 & 25 & 0c & 0c \\ \hline a8 & 89 & c8 & a6 \\ \hline \end{array} =$$

$$\text{output}\quad \begin{array}{|c|c|c|c|}\hline 39 & 02 & dc & 19 \\ \hline 25 & dc & 11 & 6a \\ \hline 84 & 09 & 85 & 0b \\ \hline 1d & fb & 97 & 32 \\ \hline \end{array}$$

Appendix C – Example Vectors

This appendix contains example vectors, including intermediate values – for all three AES key lengths (*Nk* = 4, 6, and 8), for the Cipher, Inverse Cipher, and Equivalent Inverse Cipher that are described in Sec. 5.1, 5.3, and 5.3.5, respectively. Additional examples may be found at [1] and [5].

All vectors are in hexadecimal notation, with each pair of characters giving a byte value in which the left character of each pair provides the bit pattern for the 4 bit group containing the higher numbered bits using the notation explained in Sec. 3.2, while the right character provides the bit pattern for the lower-numbered bits. The array index for all bytes (groups of two hexadecimal digits) within these test vectors starts at zero and increases from left to right.

```
Legend for CIPHER (ENCRYPT) (round number r = 0 to 10, 12 or 14):

   input:   cipher input
   start:   state at start of round[r]
   s_box:   state after SubBytes()
   s_row:   state after ShiftRows()
   m_col:   state after MixColumns()
   k_sch:   key schedule value for round[r]
   output:  cipher output


Legend for INVERSE CIPHER (DECRYPT) (round number r = 0 to 10, 12 or 14):
   iinput:  inverse cipher input
   istart:  state at start of round[r]
   is_box:  state after InvSubBytes()
   is_row:  state after InvShiftRows()
   ik_sch:  key schedule value for round[r]
   ik_add:  state after AddRoundKey()
   ioutput: inverse cipher output


Legend for EQUIVALENT INVERSE CIPHER (DECRYPT) (round number r = 0 to 10, 12
   or 14):

   iinput:  inverse cipher input
   istart:  state at start of round[r]
   is_box:  state after InvSubBytes()
   is_row:  state after InvShiftRows()
   im_col:  state after InvMixColumns()
   ik_sch:  key schedule value for round[r]
   ioutput: inverse cipher output
```

C.1 AES-128 (*Nk*=4, *Nr*=10)

```
PLAINTEXT:          00112233445566778899aabbccddeeff
KEY:                000102030405060708090a0b0c0d0e0f

CIPHER (ENCRYPT):
```

```
round[ 0].input     00112233445566778899aabbccddeeff
round[ 0].k_sch     000102030405060708090a0b0c0d0e0f
round[ 1].start     00102030405060708090a0b0c0d0e0f0
round[ 1].s_box     63cab7040953d051cd60e0e7ba70e18c
round[ 1].s_row     6353e08c0960e104cd70b751bacad0e7
round[ 1].m_col     5f72641557f5bc92f7be3b291db9f91a
round[ 1].k_sch     d6aa74fdd2af72fadaa678f1d6ab76fe
round[ 2].start     89d810e8855ace682d1843d8cb128fe4
round[ 2].s_box     a761ca9b97be8b45d8ad1a611fc97369
round[ 2].s_row     a7be1a6997ad739bd8c9ca451f618b61
round[ 2].m_col     ff87968431d86a51645151fa773ad009
round[ 2].k_sch     b692cf0b643dbdf1be9bc5006830b3fe
round[ 3].start     4915598f55e5d7a0daca94fa1f0a63f7
round[ 3].s_box     3b59cb73fcd90ee05774222dc067fb68
round[ 3].s_row     3bd92268fc74fb735767cbe0c0590e2d
round[ 3].m_col     4c9c1e66f771f0762c3f868e534df256
round[ 3].k_sch     b6ff744ed2c2c9bf6c590cbf0469bf41
round[ 4].start     fa636a2825b339c940668a3157244d17
round[ 4].s_box     2dfb02343f6d12dd09337ec75b36e3f0
round[ 4].s_row     2d6d7ef03f33e334093602dd5bfb12c7
round[ 4].m_col     6385b79ffc538df997be478e7547d691
round[ 4].k_sch     47f7f7bc95353e03f96c32bcfd058dfd
round[ 5].start     247240236966b3fa6ed2753288425b6c
round[ 5].s_box     36400926f9336d2d9fb59d23c42c3950
round[ 5].s_row     36339d50f9b539269f2c092dc4406d23
round[ 5].m_col     f4bcd45432e554d075f1d6c51dd03b3c
round[ 5].k_sch     3caaa3e8a99f9deb50f3af57adf622aa
round[ 6].start     c81677bc9b7ac93b25027992b0261996
round[ 6].s_box     e847f56514dadde23f77b64fe7f7d490
round[ 6].s_row     e8dab6901477d4653ff7f5e2e747dd4f
round[ 6].m_col     9816ee7400f87f556b2c049c8e5ad036
round[ 6].k_sch     5e390f7df7a69296a7553dc10aa31f6b
round[ 7].start     c62fe109f75eedc3cc79395d84f9cf5d
round[ 7].s_box     b415f8016858552e4bb6124c5f998a4c
round[ 7].s_row     b458124c68b68a014b99f82e5f15554c
round[ 7].m_col     c57e1c159a9bd286f05f4be098c63439
round[ 7].k_sch     14f9701ae35fe28c440adf4d4ea9c026
round[ 8].start     d1876c0f79c4300ab45594add66ff41f
round[ 8].s_box     3e175076b61c04678dfc2295f6a8bfc0
round[ 8].s_row     3e1c22c0b6fcbf768da85067f6170495
round[ 8].m_col     baa03de7a1f9b56ed5512cba5f414d23
round[ 8].k_sch     47438735a41c65b9e016baf4aebf7ad2
round[ 9].start     fde3bad205e5d0d73547964ef1fe37f1
round[ 9].s_box     5411f4b56bd9700e96a0902fa1bb9aa1
round[ 9].s_row     54d990a16ba09ab596bbf40ea111702f
round[ 9].m_col     e9f74eec023020f61bf2ccf2353c21c7
round[ 9].k_sch     549932d1f08557681093ed9cbe2c974e
round[10].start     bd6e7c3df2b5779e0b61216e8b10b689
round[10].s_box     7a9f102789d5f50b2beffd9f3dca4ea7
round[10].s_row     7ad5fda789ef4e272bca100b3d9ff59f
round[10].k_sch     13111d7fe3944a17f307a78b4d2b30c5
round[10].output    69c4e0d86a7b0430d8cdb78070b4c55a

INVERSE CIPHER (DECRYPT):
round[ 0].iinput    69c4e0d86a7b0430d8cdb78070b4c55a
round[ 0].ik_sch    13111d7fe3944a17f307a78b4d2b30c5
round[ 1].istart    7ad5fda789ef4e272bca100b3d9ff59f
```

```
round[ 1].is_row    7a9f102789d5f50b2beffd9f3dca4ea7
round[ 1].is_box    bd6e7c3df2b5779e0b61216e8b10b689
round[ 1].ik_sch    549932d1f08557681093ed9cbe2c974e
round[ 1].ik_add    e9f74eec023020f61bf2ccf2353c21c7
round[ 2].istart    54d990a16ba09ab596bbf40ea111702f
round[ 2].is_row    5411f4b56bd9700e96a0902fa1bb9aa1
round[ 2].is_box    fde3bad205e5d0d73547964ef1fe37f1
round[ 2].ik_sch    47438735a41c65b9e016baf4aebf7ad2
round[ 2].ik_add    baa03de7a1f9b56ed5512cba5f414d23
round[ 3].istart    3e1c22c0b6fcbf768da85067f6170495
round[ 3].is_row    3e175076b61c04678dfc2295f6a8bfc0
round[ 3].is_box    d1876c0f79c4300ab45594add66ff41f
round[ 3].ik_sch    14f9701ae35fe28c440adf4d4ea9c026
round[ 3].ik_add    c57e1c159a9bd286f05f4be098c63439
round[ 4].istart    b458124c68b68a014b99f82e5f15554c
round[ 4].is_row    b415f8016858552e4bb6124c5f998a4c
round[ 4].is_box    c62fe109f75eedc3cc79395d84f9cf5d
round[ 4].ik_sch    5e390f7df7a69296a7553dc10aa31f6b
round[ 4].ik_add    9816ee7400f87f556b2c049c8e5ad036
round[ 5].istart    e8dab6901477d4653ff7f5e2e747dd4f
round[ 5].is_row    e847f56514dadde23f77b64fe7f7d490
round[ 5].is_box    c81677bc9b7ac93b25027992b0261996
round[ 5].ik_sch    3caaa3e8a99f9deb50f3af57adf622aa
round[ 5].ik_add    f4bcd45432e554d075f1d6c51dd03b3c
round[ 6].istart    36339d50f9b539269f2c092dc4406d23
round[ 6].is_row    36400926f9336d2d9fb59d23c42c3950
round[ 6].is_box    247240236966b3fa6ed2753288425b6c
round[ 6].ik_sch    47f7f7bc95353e03f96c32bcfd058dfd
round[ 6].ik_add    6385b79ffc538df997be478e7547d691
round[ 7].istart    2d6d7ef03f33e334093602dd5bfb12c7
round[ 7].is_row    2dfb02343f6d12dd09337ec75b36e3f0
round[ 7].is_box    fa636a2825b339c940668a3157244d17
round[ 7].ik_sch    b6ff744ed2c2c9bf6c590cbf0469bf41
round[ 7].ik_add    4c9c1e66f771f0762c3f868e534df256
round[ 8].istart    3bd92268fc74fb735767cbe0c0590e2d
round[ 8].is_row    3b59cb73fcd90ee05774222dc067fb68
round[ 8].is_box    4915598f55e5d7a0daca94fa1f0a63f7
round[ 8].ik_sch    b692cf0b643dbdf1be9bc5006830b3fe
round[ 8].ik_add    ff87968431d86a51645151fa773ad009
round[ 9].istart    a7be1a6997ad739bd8c9ca451f618b61
round[ 9].is_row    a761ca9b97be8b45d8ad1a611fc97369
round[ 9].is_box    89d810e8855ace682d1843d8cb128fe4
round[ 9].ik_sch    d6aa74fdd2af72fadaa678f1d6ab76fe
round[ 9].ik_add    5f72641557f5bc92f7be3b291db9f91a
round[10].istart    6353e08c0960e104cd70b751bacad0e7
round[10].is_row    63cab7040953d051cd60e0e7ba70e18c
round[10].is_box    00102030405060708090a0b0c0d0e0f0
round[10].ik_sch    000102030405060708090a0b0c0d0e0f
round[10].ioutput   00112233445566778899aabbccddeeff

EQUIVALENT INVERSE CIPHER (DECRYPT):
round[ 0].iinput    69c4e0d86a7b0430d8cdb78070b4c55a
round[ 0].ik_sch    13111d7fe3944a17f307a78b4d2b30c5
round[ 1].istart    7ad5fda789ef4e272bca100b3d9ff59f
round[ 1].is_box    bdb52189f261b63d0b107c9e8b6e776e
round[ 1].is_row    bd6e7c3df2b5779e0b61216e8b10b689
round[ 1].im_col    4773b91ff72f354361cb018ea1e6cf2c
```

```
round[ 1].ik_sch   13aa29be9c8faff6f770f58000f7bf03
round[ 2].istart   54d990a16ba09ab596bbf40ea111702f
round[ 2].is_box   fde596f1054737d235febad7f1e3d04e
round[ 2].is_row   fde3bad205e5d0d73547964ef1fe37f1
round[ 2].im_col   2d7e86a339d9393ee6570a1101904e16
round[ 2].ik_sch   1362a4638f2586486bff5a76f7874a83
round[ 3].istart   3e1c22c0b6fcbf768da85067f6170495
round[ 3].is_box   d1c4941f7955f40fb46f6c0ad68730ad
round[ 3].is_row   d1876c0f79c4300ab45594add66ff41f
round[ 3].im_col   39daee38f4f1a82aaf432410c36d45b9
round[ 3].ik_sch   8d82fc749c47222be4dadc3e9c7810f5
round[ 4].istart   b458124c68b68a014b99f82e5f15554c
round[ 4].is_box   c65e395df779cf09ccf9e1c3842fed5d
round[ 4].is_row   c62fe109f75eedc3cc79395d84f9cf5d
round[ 4].im_col   9a39bf1d05b20a3a476a0bf79fe51184
round[ 4].ik_sch   72e3098d11c5de5f789dfe1578a2cccb
round[ 5].istart   e8dab6901477d4653ff7f5e2e747dd4f
round[ 5].is_box   c87a79969b0219bc2526773bb016c992
round[ 5].is_row   c81677bc9b7ac93b25027992b0261996
round[ 5].im_col   18f78d779a93eef4f6742967c47f5ffd
round[ 5].ik_sch   2ec410276326d7d26958204a003f32de
round[ 6].istart   36339d50f9b539269f2c092dc4406d23
round[ 6].is_box   2466756c69d25b236e4240fa8872b332
round[ 6].is_row   247240236966b3fa6ed2753288425b6c
round[ 6].im_col   85cf8bf472d124c10348f545329c0053
round[ 6].ik_sch   a8a2f5044de2c7f50a7ef79869671294
round[ 7].istart   2d6d7ef03f33e334093602dd5bfb12c7
round[ 7].is_box   fab38a1725664d2840246ac957633931
round[ 7].is_row   fa636a2825b339c940668a3157244d17
round[ 7].im_col   fc1fc1f91934c98210fbfb8da340eb21
round[ 7].ik_sch   c7c6e391e54032f1479c306d6319e50c
round[ 8].istart   3bd92268fc74fb735767cbe0c0590e2d
round[ 8].is_box   49e594f755ca638fda0a59a01f15d7fa
round[ 8].is_row   4915598f55e5d7a0daca94fa1f0a63f7
round[ 8].im_col   076518f0b52ba2fb7a15c8d93be45e00
round[ 8].ik_sch   a0db02992286d160a2dc029c2485d561
round[ 9].istart   a7be1a6997ad739bd8c9ca451f618b61
round[ 9].is_box   895a43e485188fe82d121068cbd8ced8
round[ 9].is_row   89d810e8855ace682d1843d8cb128fe4
round[ 9].im_col   ef053f7c8b3d32fd4d2a64ad3c93071a
round[ 9].ik_sch   8c56dff0825dd3f9805ad3fc8659d7fd
round[10].istart   6353e08c0960e104cd70b751bacad0e7
round[10].is_box   0050a0f04090e03080d02070c01060b0
round[10].is_row   00102030405060708090a0b0c0d0e0f0
round[10].ik_sch   000102030405060708090a0b0c0d0e0f
round[10].ioutput  00112233445566778899aabbccddeeff
```

C.2 AES-192 (*Nk*=6, *Nr*=12)

```
PLAINTEXT:    00112233445566778899aabbccddeeff
KEY:          000102030405060708090a0b0c0d0e0f1011121314151617

CIPHER (ENCRYPT):
round[ 0].input    00112233445566778899aabbccddeeff
round[ 0].k_sch    000102030405060708090a0b0c0d0e0f
round[ 1].start    00102030405060708090a0b0c0d0e0f0
```

```
round[ 1].s_box     63cab7040953d051cd60e0e7ba70e18c
round[ 1].s_row     6353e08c0960e104cd70b751bacad0e7
round[ 1].m_col     5f72641557f5bc92f7be3b291db9f91a
round[ 1].k_sch     10111213141516175846f2f95c43f4fe
round[ 2].start     4f63760643e0aa85aff8c9d041fa0de4
round[ 2].s_box     84fb386f1ae1ac977941dd70832dd769
round[ 2].s_row     84e1dd691a41d76f792d389783fbac70
round[ 2].m_col     9f487f794f955f662afc86abd7f1ab29
round[ 2].k_sch     544afef55847f0fa4856e2e95c43f4fe
round[ 3].start     cb02818c17d2af9c62aa64428bb25fd7
round[ 3].s_box     1f770c64f0b579deaaac432c3d37cf0e
round[ 3].s_row     1fb5430ef0accf64aa370cde3d77792c
round[ 3].m_col     b7a53ecbbf9d75a0c40efc79b674cc11
round[ 3].k_sch     40f949b31cbabd4d48f043b810b7b342
round[ 4].start     f75c7778a327c8ed8cfebfc1a6c37f53
round[ 4].s_box     684af5bc0acce85564bb0878242ed2ed
round[ 4].s_row     68cc08ed0abbd2bc642ef555244ae878
round[ 4].m_col     7a1e98bdacb6d1141a6944dd06eb2d3e
round[ 4].k_sch     58e151ab04a2a5557effb5416245080c
round[ 5].start     22ffc916a81474416496f19c64ae2532
round[ 5].s_box     9316dd47c2fa92834390a1de43e43f23
round[ 5].s_row     93faa123c2903f4743e4dd83431692de
round[ 5].m_col     aaa755b34cffe57cef6f98e1f01c13e6
round[ 5].k_sch     2ab54bb43a02f8f662e3a95d66410c08
round[ 6].start     80121e0776fd1d8a8d8c31bc965d1fee
round[ 6].s_box     cdc972c53854a47e5d64c765904cc028
round[ 6].s_row     cd54c7283864c0c55d4c727e90c9a465
round[ 6].m_col     921f748fd96e937d622d7725ba8ba50c
round[ 6].k_sch     f501857297448d7ebdf1c6ca87f33e3c
round[ 7].start     671ef1fd4e2a1e03dfdcb1ef3d789b30
round[ 7].s_box     8572a1542fe5727b9e86c8df27bc1404
round[ 7].s_row     85e5c8042f8614549ebca17b277272df
round[ 7].m_col     e913e7b18f507d4b227ef652758acbcc
round[ 7].k_sch     e510976183519b6934157c9ea351f1e0
round[ 8].start     0c0370d00c01e622166b8accd6db3a2c
round[ 8].s_box     fe7b5170fe7c8e93477f7e4bf6b98071
round[ 8].s_row     fe7c7e71fe7f807047b95193f67b8e4b
round[ 8].m_col     6cf5edf996eb0a069c4ef21cbfc25762
round[ 8].k_sch     1ea0372a995309167c439e77ff12051e
round[ 9].start     7255dad30fb80310e00d6c6b40d0527c
round[ 9].s_box     40fc5766766c7bcae1d7507f09700010
round[ 9].s_row     406c501076d70066e17057ca09fc7b7f
round[ 9].m_col     7478bcdce8a50b81d4327a9009188262
round[ 9].k_sch     dd7e0e887e2fff68608fc842f9dcc154
round[10].start     a906b254968af4e9b4bdb2d2f0c44336
round[10].s_box     d36f3720907ebf1e8d7a37b58c1c1a05
round[10].s_row     d37e3705907a1a208d1c371e8c6fbfb5
round[10].m_col     0d73cc2d8f6abe8b0cf2dd9bb83d422e
round[10].k_sch     859f5f237a8d5a3dc0c02952beefd63a
round[11].start     88ec930ef5e7e4b6cc32f4c906d29414
round[11].s_box     c4cedcabe694694e4b23bfdd6fb522fa
round[11].s_row     c494bffae62322ab4bb5dc4e6fce69dd
round[11].m_col     71d720933b6d677dc00b8f28238e0fb7
round[11].k_sch     de601e7827bcdf2ca223800fd8aeda32
round[12].start     afb73eeb1cd1b85162280f27fb20d585
round[12].s_box     79a9b2e99c3e6cd1aa3476cc0fb70397
round[12].s_row     793e76979c3403e9aab7b2d10fa96ccc
```

```
round[12].k_sch     a4970a331a78dc09c418c271e3a41d5d
round[12].output    dda97ca4864cdfe06eaf70a0ec0d7191

INVERSE CIPHER (DECRYPT):
round[ 0].iinput    dda97ca4864cdfe06eaf70a0ec0d7191
round[ 0].ik_sch    a4970a331a78dc09c418c271e3a41d5d
round[ 1].istart    793e76979c3403e9aab7b2d10fa96ccc
round[ 1].is_row    79a9b2e99c3e6cd1aa3476cc0fb70397
round[ 1].is_box    afb73eeb1cd1b85162280f27fb20d585
round[ 1].ik_sch    de601e7827bcdf2ca223800fd8aeda32
round[ 1].ik_add    71d720933b6d677dc00b8f28238e0fb7
round[ 2].istart    c494bffae62322ab4bb5dc4e6fce69dd
round[ 2].is_row    c4cedcabe694694e4b23bfdd6fb522fa
round[ 2].is_box    88ec930ef5e7e4b6cc32f4c906d29414
round[ 2].ik_sch    859f5f237a8d5a3dc0c02952beefd63a
round[ 2].ik_add    0d73cc2d8f6abe8b0cf2dd9bb83d422e
round[ 3].istart    d37e3705907a1a208d1c371e8c6fbfb5
round[ 3].is_row    d36f3720907ebf1e8d7a37b58c1c1a05
round[ 3].is_box    a906b254968af4e9b4bdb2d2f0c44336
round[ 3].ik_sch    dd7e0e887e2fff68608fc842f9dcc154
round[ 3].ik_add    7478bcdce8a50b81d4327a9009188262
round[ 4].istart    406c501076d70066e17057ca09fc7b7f
round[ 4].is_row    40fc5766766c7bcae1d7507f09700010
round[ 4].is_box    7255dad30fb80310e00d6c6b40d0527c
round[ 4].ik_sch    1ea0372a995309167c439e77ff12051e
round[ 4].ik_add    6cf5edf996eb0a069c4ef21cbfc25762
round[ 5].istart    fe7c7e71fe7f807047b95193f67b8e4b
round[ 5].is_row    fe7b5170fe7c8e93477f7e4bf6b98071
round[ 5].is_box    0c0370d00c01e622166b8accd6db3a2c
round[ 5].ik_sch    e510976183519b6934157c9ea351f1e0
round[ 5].ik_add    e913e7b18f507d4b227ef652758acbcc
round[ 6].istart    85e5c8042f8614549ebca17b277272df
round[ 6].is_row    8572a1542fe5727b9e86c8df27bc1404
round[ 6].is_box    671ef1fd4e2a1e03dfdcb1ef3d789b30
round[ 6].ik_sch    f501857297448d7ebdf1c6ca87f33e3c
round[ 6].ik_add    921f748fd96e937d622d7725ba8ba50c
round[ 7].istart    cd54c7283864c0c55d4c727e90c9a465
round[ 7].is_row    cdc972c53854a47e5d64c765904cc028
round[ 7].is_box    80121e0776fd1d8a8d8c31bc965d1fee
round[ 7].ik_sch    2ab54bb43a02f8f662e3a95d66410c08
round[ 7].ik_add    aaa755b34cffe57cef6f98e1f01c13e6
round[ 8].istart    93faa123c2903f4743e4dd83431692de
round[ 8].is_row    9316dd47c2fa92834390a1de43e43f23
round[ 8].is_box    22ffc916a81474416496f19c64ae2532
round[ 8].ik_sch    58e151ab04a2a5557effb5416245080c
round[ 8].ik_add    7a1e98bdacb6d1141a6944dd06eb2d3e
round[ 9].istart    68cc08ed0abbd2bc642ef555244ae878
round[ 9].is_row    684af5bc0acce85564bb0878242ed2ed
round[ 9].is_box    f75c7778a327c8ed8cfebfc1a6c37f53
round[ 9].ik_sch    40f949b31cbabd4d48f043b810b7b342
round[ 9].ik_add    b7a53ecbbf9d75a0c40efc79b674cc11
round[10].istart    1fb5430ef0accf64aa370cde3d77792c
round[10].is_row    1f770c64f0b579deaaac432c3d37cf0e
round[10].is_box    cb02818c17d2af9c62aa64428bb25fd7
round[10].ik_sch    544afef55847f0fa4856e2e95c43f4fe
round[10].ik_add    9f487f794f955f662afc86abd7f1ab29
round[11].istart    84e1dd691a41d76f792d389783fbac70
```

```
round[11].is_row    84fb386f1ae1ac977941dd70832dd769
round[11].is_box    4f63760643e0aa85aff8c9d041fa0de4
round[11].ik_sch    101112131415161758 46f2f95c43f4fe
round[11].ik_add    5f72641557f5bc92f7be3b291db9f91a
round[12].istart    6353e08c0960e104cd70b751bacad0e7
round[12].is_row    63cab7040953d051cd60e0e7ba70e18c
round[12].is_box    00102030405060708090a0b0c0d0e0f0
round[12].ik_sch    000102030405060708090a0b0c0d0e0f
round[12].ioutput   00112233445566778899aabbccddeeff

EQUIVALENT INVERSE CIPHER (DECRYPT):
round[ 0].iinput    dda97ca4864cdfe06eaf70a0ec0d7191
round[ 0].ik_sch    a4970a331a78dc09c418c271e3a41d5d
round[ 1].istart    793e76979c3403e9aab7b2d10fa96ccc
round[ 1].is_box    afd10f851c28d5eb62203e51fbb7b827
round[ 1].is_row    afb73eeb1cd1b85162280f27fb20d585
round[ 1].im_col    122a02f7242ac8e20605afce51cc7264
round[ 1].ik_sch    d6bebd0dc209ea494db073803e021bb9
round[ 2].istart    c494bffae62322ab4bb5dc4e6fce69dd
round[ 2].is_box    88e7f414f532940eccd293b606ece4c9
round[ 2].is_row    88ec930ef5e7e4b6cc32f4c906d29414
round[ 2].im_col    5cc7aecce3c872194ae5ef8309a933c7
round[ 2].ik_sch    8fb999c973b26839c7f9d89d85c68c72
round[ 3].istart    d37e3705907a1a208d1c371e8c6fbfb5
round[ 3].is_box    a98ab23696bd4354b4c4b2e9f006f4d2
round[ 3].is_row    a906b254968af4e9b4bdb2d2f0c44336
round[ 3].im_col    b7113ed134e85489b20866b51d4b2c3b
round[ 3].ik_sch    f77d6ec1423f54ef5378317f14b75744
round[ 4].istart    406c501076d70066e17057ca09fc7b7f
round[ 4].is_box    72b86c7c0f0d52d3e0d0da104055036b
round[ 4].is_row    7255dad30fb80310e00d6c6b40d0527c
round[ 4].im_col    ef3b1be1b9b0e64bdcb79f1e0a707fbb
round[ 4].ik_sch    1147659047cf663b9b0ece8dfc0bf1f0
round[ 5].istart    fe7c7e71fe7f807047b95193f67b8e4b
round[ 5].is_box    0c018a2c0c6b3ad016db7022d603e6cc
round[ 5].is_row    0c0370d00c01e622166b8accd6db3a2c
round[ 5].im_col    592460b248832b2952e0b831923048f1
round[ 5].ik_sch    dcc1a8b667053f7dcc5c194ab5423a2e
round[ 6].istart    85e5c8042f8614549ebca17b277272df
round[ 6].is_box    672ab1304edc9bfddf78f1033d1e1eef
round[ 6].is_row    671ef1fd4e2a1e03dfdcb1ef3d789b30
round[ 6].im_col    0b8a7783417ae3a1f9492dc0c641a7ce
round[ 6].ik_sch    c6deb0ab791e2364a4055fbe568803ab
round[ 7].istart    cd54c7283864c0c55d4c727e90c9a465
round[ 7].is_box    80fd31ee768c1f078d5d1e8a96121dbc
round[ 7].is_row    80121e0776fd1d8a8d8c31bc965d1fee
round[ 7].im_col    4ee1ddf9301d6352c9ad769ef8d20515
round[ 7].ik_sch    dd1b7cdaf28d5c158a49ab1dbbc497cb
round[ 8].istart    93faa123c2903f4743e4dd83431692de
round[ 8].is_box    2214f132a896251664aec94164ff749c
round[ 8].is_row    22ffc916a81474416496f19c64ae2532
round[ 8].im_col    1008ffe53b36ee6af27b42549b8a7bb7
round[ 8].ik_sch    78c4f708318d3cd69655b701bfc093cf
round[ 9].istart    68cc08ed0abbd2bc642ef555244ae878
round[ 9].is_box    f727bf53a3fe7f788cc377eda65cc8c1
round[ 9].is_row    f75c7778a327c8ed8cfebfc1a6c37f53
round[ 9].im_col    7f69ac1ed939ebaac8ece3cb12e159e3
```

```
round[ 9].ik_sch    60dcef10299524ce62dbef152f9620cf
round[10].istart    1fb5430ef0accf64aa370cde3d77792c
round[10].is_box    cbd264d717aa5f8c62b2819c8b02af42
round[10].is_row    cb02818c17d2af9c62aa64428bb25fd7
round[10].im_col    cfaf16b2570c18b52e7fef50cab267ae
round[10].ik_sch    4b4ecbdb4d4dcfda5752d7c74949cbde
round[11].istart    84e1dd691a41d76f792d389783fbac70
round[11].is_box    4fe0c9e443f80d06affa76854163aad0
round[11].is_row    4f63760643e0aa85aff8c9d041fa0de4
round[11].im_col    794cf891177bfd1d8a327086f3831b39
round[11].ik_sch    1a1f181d1e1b1c194742c7d74949cbde
round[12].istart    6353e08c0960e104cd70b751bacad0e7
round[12].is_box    0050a0f04090e03080d02070c01060b0
round[12].is_row    00102030405060708090a0b0c0d0e0f0
round[12].ik_sch    000102030405060708090a0b0c0d0e0f
round[12].ioutput   00112233445566778899aabbccddeeff
```

C.3 AES-256 (Nk=8, Nr=14)

```
PLAINTEXT:   00112233445566778899aabbccddeeff
KEY:         000102030405060708090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f

CIPHER (ENCRYPT):
round[ 0].input    00112233445566778899aabbccddeeff
round[ 0].k_sch    000102030405060708090a0b0c0d0e0f
round[ 1].start    00102030405060708090a0b0c0d0e0f0
round[ 1].s_box    63cab7040953d051cd60e0e7ba70e18c
round[ 1].s_row    6353e08c0960e104cd70b751bacad0e7
round[ 1].m_col    5f72641557f5bc92f7be3b291db9f91a
round[ 1].k_sch    101112131415161718191a1b1c1d1e1f
round[ 2].start    4f63760643e0aa85efa7213201a4e705
round[ 2].s_box    84fb386f1ae1ac97df5cfd237c49946b
round[ 2].s_row    84e1fd6b1a5c946fdf4938977cfbac23
round[ 2].m_col    bd2a395d2b6ac438d192443e615da195
round[ 2].k_sch    a573c29fa176c498a97fce93a572c09c
round[ 3].start    1859fbc28a1c00a078ed8aadc42f6109
round[ 3].s_box    adcb0f257e9c63e0bc557e951c15ef01
round[ 3].s_row    ad9c7e017e55ef25bc150fe01ccb6395
round[ 3].m_col    810dce0cc9db8172b3678c1e88a1b5bd
round[ 3].k_sch    1651a8cd0244beda1a5da4c10640bade
round[ 4].start    975c66c1cb9f3fa8a93a28df8ee10f63
round[ 4].s_box    884a33781fdb75c2d380349e19f876fb
round[ 4].s_row    88db34fb1f807678d3f833c2194a759e
round[ 4].m_col    b2822d81abe6fb275faf103a078c0033
round[ 4].k_sch    ae87dff00ff11b68a68ed5fb03fc1567
round[ 5].start    1c05f271a417e04ff921c5c104701554
round[ 5].s_box    9c6b89a349f0e18499fda678f2515920
round[ 5].s_row    9cf0a62049fd59a399518984f26be178
round[ 5].m_col    aeb65ba974e0f822d73f567bdb64c877
round[ 5].k_sch    6de1f1486fa54f9275f8eb5373b8518d
round[ 6].start    c357aae11b45b7b0a2c7bd28a8dc99fa
round[ 6].s_box    2e5bacf8af6ea9e73ac67a34c286ee2d
round[ 6].s_row    2e6e7a2dafc6eef83a86ace7c25ba934
round[ 6].m_col    b951c33c02e9bd29ae25cdb1efa08cc7
round[ 6].k_sch    c656827fc9a799176f294cec6cd5598b
round[ 7].start    7f074143cb4e243ec10c815d8375d54c
round[ 7].s_box    d2c5831a1f2f36b278fe0c4cec9d0329
```

```
round[ 7].s_row    d22f0c291ffe031a789d83b2ecc5364c
round[ 7].m_col    ebb19e1c3ee7c9e87d7535e9ed6b9144
round[ 7].k_sch    3de23a75524775e727bf9eb45407cf39
round[ 8].start    d653a4696ca0bc0f5acaab5db96c5e7d
round[ 8].s_box    f6ed49f950e06576be74624c565058ff
round[ 8].s_row    f6e062ff507458f9be50497656ed654c
round[ 8].m_col    5174c8669da98435a8b3e62ca974a5ea
round[ 8].k_sch    0bdc905fc27b0948ad5245a4c1871c2f
round[ 9].start    5aa858395fd28d7d05e1a38868f3b9c5
round[ 9].s_box    bec26a12cfb55dff6bf80ac4450d56a6
round[ 9].s_row    beb50aa6cff856126b0d6aff45c25dc4
round[ 9].m_col    0f77ee31d2ccadc05430a83f4ef96ac3
round[ 9].k_sch    45f5a66017b2d387300d4d33640a820a
round[10].start    4a824851c57e7e47643de50c2af3e8c9
round[10].s_box    d61352d1a6f3f3a04327d9fee50d9bdd
round[10].s_row    d6f3d9dda6279bd1430d52a0e513f3fe
round[10].m_col    bd86f0ea748fc4f4630f11c1e9331233
round[10].k_sch    7ccff71cbeb4fe5413e6bbf0d261a7df
round[11].start    c14907f6ca3b3aa070e9aa313b52b5ec
round[11].s_box    783bc54274e280e0511eacc7e200d5ce
round[11].s_row    78e2acce741ed5425100c5e0e23b80c7
round[11].m_col    af8690415d6e1dd387e5fbedd5c89013
round[11].k_sch    f01afafee7a82979d7a5644ab3afe640
round[12].start    5f9c6abfbac634aa50409fa766677653
round[12].s_box    cfde0208f4b418ac5309db5c338538ed
round[12].s_row    cfb4dbedf4093808538502ac33de185c
round[12].m_col    7427fae4d8a695269ce83d315be0392b
round[12].k_sch    2541fe719bf500258813bbd55a721c0a
round[13].start    516604954353950314fb86e401922521
round[13].s_box    d133f22a1aed2a7bfa0f44697c4f3ffd
round[13].s_row    d1ed44fd1a0f3f2afa4ff27b7c332a69
round[13].m_col    2c21a820306f154ab712c75eee0da04f
round[13].k_sch    4e5a6699a9f24fe07e572baacdf8cdea
round[14].start    627bceb9999d5aaac945ecf423f56da5
round[14].s_box    aa218b56ee5ebeacdd6ecebf26e63c06
round[14].s_row    aa5ece06ee6e3c56dde68bac2621bebf
round[14].k_sch    24fc79ccbf0979e9371ac23c6d68de36
round[14].output   8ea2b7ca516745bfeafc49904b496089

INVERSE CIPHER (DECRYPT):
round[ 0].iinput   8ea2b7ca516745bfeafc49904b496089
round[ 0].ik_sch   24fc79ccbf0979e9371ac23c6d68de36
round[ 1].istart   aa5ece06ee6e3c56dde68bac2621bebf
round[ 1].is_row   aa218b56ee5ebeacdd6ecebf26e63c06
round[ 1].is_box   627bceb9999d5aaac945ecf423f56da5
round[ 1].ik_sch   4e5a6699a9f24fe07e572baacdf8cdea
round[ 1].ik_add   2c21a820306f154ab712c75eee0da04f
round[ 2].istart   d1ed44fd1a0f3f2afa4ff27b7c332a69
round[ 2].is_row   d133f22a1aed2a7bfa0f44697c4f3ffd
round[ 2].is_box   516604954353950314fb86e401922521
round[ 2].ik_sch   2541fe719bf500258813bbd55a721c0a
round[ 2].ik_add   7427fae4d8a695269ce83d315be0392b
round[ 3].istart   cfb4dbedf4093808538502ac33de185c
round[ 3].is_row   cfde0208f4b418ac5309db5c338538ed
round[ 3].is_box   5f9c6abfbac634aa50409fa766677653
round[ 3].ik_sch   f01afafee7a82979d7a5644ab3afe640
round[ 3].ik_add   af8690415d6e1dd387e5fbedd5c89013
```

```
round[ 4].istart    78e2acce741ed5425100c5e0e23b80c7
round[ 4].is_row    783bc54274e280e0511eacc7e200d5ce
round[ 4].is_box    c14907f6ca3b3aa070e9aa313b52b5ec
round[ 4].ik_sch    7ccff71cbeb4fe5413e6bbf0d261a7df
round[ 4].ik_add    bd86f0ea748fc4f4630f11c1e9331233
round[ 5].istart    d6f3d9dda6279bd1430d52a0e513f3fe
round[ 5].is_row    d61352d1a6f3f3a04327d9fee50d9bdd
round[ 5].is_box    4a824851c57e7e47643de50c2af3e8c9
round[ 5].ik_sch    45f5a66017b2d387300d4d33640a820a
round[ 5].ik_add    0f77ee31d2ccadc05430a83f4ef96ac3
round[ 6].istart    beb50aa6cff856126b0d6aff45c25dc4
round[ 6].is_row    bec26a12cfb55dff6bf80ac4450d56a6
round[ 6].is_box    5aa858395fd28d7d05e1a38868f3b9c5
round[ 6].ik_sch    0bdc905fc27b0948ad5245a4c1871c2f
round[ 6].ik_add    5174c8669da98435a8b3e62ca974a5ea
round[ 7].istart    f6e062ff507458f9be50497656ed654c
round[ 7].is_row    f6ed49f950e06576be74624c565058ff
round[ 7].is_box    d653a4696ca0bc0f5acaab5db96c5e7d
round[ 7].ik_sch    3de23a75524775e727bf9eb45407cf39
round[ 7].ik_add    ebb19e1c3ee7c9e87d7535e9ed6b9144
round[ 8].istart    d22f0c291ffe031a789d83b2ecc5364c
round[ 8].is_row    d2c5831a1f2f36b278fe0c4cec9d0329
round[ 8].is_box    7f074143cb4e243ec10c815d8375d54c
round[ 8].ik_sch    c656827fc9a799176f294cec6cd5598b
round[ 8].ik_add    b951c33c02e9bd29ae25cdb1efa08cc7
round[ 9].istart    2e6e7a2dafc6eef83a86ace7c25ba934
round[ 9].is_row    2e5bacf8af6ea9e73ac67a34c286ee2d
round[ 9].is_box    c357aae11b45b7b0a2c7bd28a8dc99fa
round[ 9].ik_sch    6de1f1486fa54f9275f8eb5373b8518d
round[ 9].ik_add    aeb65ba974e0f822d73f567bdb64c877
round[10].istart    9cf0a62049fd59a399518984f26be178
round[10].is_row    9c6b89a349f0e18499fda678f2515920
round[10].is_box    1c05f271a417e04ff921c5c104701554
round[10].ik_sch    ae87dff00ff11b68a68ed5fb03fc1567
round[10].ik_add    b2822d81abe6fb275faf103a078c0033
round[11].istart    88db34fb1f807678d3f833c2194a759e
round[11].is_row    884a33781fdb75c2d380349e19f876fb
round[11].is_box    975c66c1cb9f3fa8a93a28df8ee10f63
round[11].ik_sch    1651a8cd0244beda1a5da4c10640bade
round[11].ik_add    810dce0cc9db8172b3678c1e88a1b5bd
round[12].istart    ad9c7e017e55ef25bc150fe01ccb6395
round[12].is_row    adcb0f257e9c63e0bc557e951c15ef01
round[12].is_box    1859fbc28a1c00a078ed8aadc42f6109
round[12].ik_sch    a573c29fa176c498a97fce93a572c09c
round[12].ik_add    bd2a395d2b6ac438d192443e615da195
round[13].istart    84e1fd6b1a5c946fdf4938977cfbac23
round[13].is_row    84fb386f1ae1ac97df5cfd237c49946b
round[13].is_box    4f63760643e0aa85efa7213201a4e705
round[13].ik_sch    101112131415161718191a1b1c1d1e1f
round[13].ik_add    5f72641557f5bc92f7be3b291db9f91a
round[14].istart    6353e08c0960e104cd70b751bacad0e7
round[14].is_row    63cab7040953d051cd60e0e7ba70e18c
round[14].is_box    00102030405060708090a0b0c0d0e0f0
round[14].ik_sch    000102030405060708090a0b0c0d0e0f
round[14].ioutput   00112233445566778899aabbccddeeff

EQUIVALENT INVERSE CIPHER (DECRYPT):
```

```
round[ 0].iinput   8ea2b7ca516745bfeafc49904b496089
round[ 0].ik_sch   24fc79ccbf0979e9371ac23c6d68de36
round[ 1].istart   aa5ece06ee6e3c56dde68bac2621bebf
round[ 1].is_box   629deca599456db9c9f5ceaa237b5af4
round[ 1].is_row   627bceb9999d5aaac945ecf423f56da5
round[ 1].im_col   e51c9502a5c1950506a61024596b2b07
round[ 1].ik_sch   34f1d1ffbfceaa2ffce9e25f2558016e
round[ 2].istart   d1ed44fd1a0f3f2afa4ff27b7c332a69
round[ 2].is_box   5153862143fb259514920403016695e4
round[ 2].is_row   516604954353950314fb86e401922521
round[ 2].im_col   91a29306cc450d0226f4b5eaef5efed8
round[ 2].ik_sch   5e1648eb384c350a7571b746dc80e684
round[ 3].istart   cfb4dbedf4093808538502ac33de185c
round[ 3].is_box   5fc69f53ba4076bf50676aaa669c34a7
round[ 3].is_row   5f9c6abfbac634aa50409fa766677653
round[ 3].im_col   b041a94eff21ae9212278d903b8a63f6
round[ 3].ik_sch   c8a305808b3f7bd043274870d9b1e331
round[ 4].istart   78e2acce741ed5425100c5e0e23b80c7
round[ 4].is_box   c13baaeccae9b5f6705207a03b493a31
round[ 4].is_row   c14907f6ca3b3aa070e9aa313b52b5ec
round[ 4].im_col   638357cec07de6300e30d0ec4ce2a23c
round[ 4].ik_sch   b5708e13665a7de14d3d824ca9f151c2
round[ 5].istart   d6f3d9dda6279bd1430d52a0e513f3fe
round[ 5].is_box   4a7ee5c9c53de85164f348472a827e0c
round[ 5].is_row   4a824851c57e7e47643de50c2af3e8c9
round[ 5].im_col   ca6f71058c642842a315595fdf54f685
round[ 5].ik_sch   74da7ba3439c7e50c81833a09a96ab41
round[ 6].istart   beb50aa6cff856126b0d6aff45c25dc4
round[ 6].is_box   5ad2a3c55fe1b93905f3587d68a88d88
round[ 6].is_row   5aa858395fd28d7d05e1a38868f3b9c5
round[ 6].im_col   ca46f5ea835eab0b9537b6dbb221b6c2
round[ 6].ik_sch   3ca69715d32af3f22b67ffade4ccd38e
round[ 7].istart   f6e062ff507458f9be50497656ed654c
round[ 7].is_box   d6a0ab7d6cca5e695a6ca40fb953bc5d
round[ 7].is_row   d653a4696ca0bc0f5acaab5db96c5e7d
round[ 7].im_col   2a70c8da28b806e9f319ce42be4baead
round[ 7].ik_sch   f85fc4f3374605f38b844df0528e98e1
round[ 8].istart   d22f0c291ffe031a789d83b2ecc5364c
round[ 8].is_box   7f4e814ccb0cd543c175413e8307245d
round[ 8].is_row   7f074143cb4e243ec10c815d8375d54c
round[ 8].im_col   f0073ab7404a8a1fc2cba0b80df08517
round[ 8].ik_sch   de69409aef8c64e7f84d0c5fcfab2c23
round[ 9].istart   2e6e7a2dafc6eef83a86ace7c25ba934
round[ 9].is_box   c345bdfa1bc799e1a2dcaab0a857b728
round[ 9].is_row   c357aae11b45b7b0a2c7bd28a8dc99fa
round[ 9].im_col   3225fe3686e498a32593c1872b613469
round[ 9].ik_sch   aed55816cf19c100bcc24803d90ad511
round[10].istart   9cf0a62049fd59a399518984f26be178
round[10].is_box   1c17c554a4211571f970f24f0405e0c1
round[10].is_row   1c05f271a417e04ff921c5c104701554
round[10].im_col   9d1d5c462e655205c4395b7a2eac55e2
round[10].ik_sch   15c668bd31e5247d17c168b837e6207c
round[11].istart   88db34fb1f807678d3f833c2194a759e
round[11].is_box   979f2863cb3a0fc1a9e166a88e5c3fdf
round[11].is_row   975c66c1cb9f3fa8a93a28df8ee10f63
round[11].im_col   d24bfb0e1f997633cfce86e37903fe87
round[11].ik_sch   7fd7850f61cc991673db890365c89d12
```

```
round[12].istart   ad9c7e017e55ef25bc150fe01ccb6395
round[12].is_box   181c8a098aed61c2782ffba0c45900ad
round[12].is_row   1859fbc28a1c00a078ed8aadc42f6109
round[12].im_col   aec9bda23e7fd8aff96d74525cdce4e7
round[12].ik_sch   2a2840c924234cc026244cc5202748c4
round[13].istart   84e1fd6b1a5c946fdf4938977cfbac23
round[13].is_box   4fe0210543a7e706efa476850163aa32
round[13].is_row   4f63760643e0aa85efa7213201a4e705
round[13].im_col   794cf891177bfd1ddf67a744acd9c4f6
round[13].ik_sch   1a1f181d1e1b1c191217101516131411
round[14].istart   6353e08c0960e104cd70b751bacad0e7
round[14].is_box   0050a0f04090e03080d02070c01060b0
round[14].is_row   00102030405060708090a0b0c0d0e0f0
round[14].ik_sch   000102030405060708090a0b0c0d0e0f
round[14].ioutput  00112233445566778899aabbccddeeff
```

Appendix D - References

[1] AES page available via http://www.nist.gov/CryptoToolkit.[4]

[2] Computer Security Objects Register (CSOR): http://csrc.nist.gov/csor/.

[3] J. Daemen and V. Rijmen, *AES Proposal: Rijndael*, AES Algorithm Submission, September 3, 1999, available at [1].

[4] J. Daemen and V. Rijmen, *The block cipher Rijndael*, Smart Card research and Applications, LNCS 1820, Springer-Verlag, pp. 288-296.

[5] B. Gladman's AES related home page http://fp.gladman.plus.com/cryptography_technology/.

[6] A. Lee, NIST Special Publication 800-21, *Guideline for Implementing Cryptography in the Federal Government*, National Institute of Standards and Technology, November 1999.

[7] A. Menezes, P. van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, New York, 1997, p. 81-83.

[8] J. Nechvatal, et. al., *Report on the Development of the Advanced Encryption Standard (AES)*, National Institute of Standards and Technology, October 2, 2000, available at [1].

[4] A complete set of documentation from the AES development effort – including announcements, public comments, analysis papers, conference proceedings, etc. – is available from this site.

EXHIBIT A
BPH14178 - PRICING PAGE

CONTRACT ITEM #	DESCRIPTION	QUANTITY	UNIT PRICE	EXTENDED PRICE
3.1.1	Digital Dual Band Mobile Radio Base System, Motorola APX 7500 Multi-Band Mobile Radio Base System, or Equal.	12 Each		
GRAND TOTAL:				

Contract shall be awarded to the Vendor that provides the Contract Items meeting the required specifications for the lowest overall total cost.

Upon receipt of a purchase order, Vendor must be able to deliver all Contract Items so that Agency receives by June 30, 2014.
Contract Items shall be delivered to Agency at Bureau for Public Health, Center for Threat Preparedness, 505 Capitol Street, Suite 200, Charleston, West Virginia 25301.

VENDOR SECTION:

Vendor Name:
Physical Address:
Remit to Address:
Telephone:
Fax:
Email:
Vendor Representative (print name):

Signature:	**Date:**

CERTIFICATION AND SIGNATURE PAGE

By signing below, I certify that I have reviewed this Solicitation in its entirety; understand the requirements, terms and conditions, and other information contained herein; that I am submitting this bid or proposal for review and consideration; that I am authorized by the bidder to execute this bid or any documents related thereto on bidder's behalf; that I am authorized to bind the bidder in a contractual relationship; and that to the best of my knowledge, the bidder has properly registered with any State agency that may require registration.

____

(Company)

____

(Authorized Signature)

____

(Representative Name, Title)

____

(Phone Number) (Fax Number)

____

(Date)

RFQ No. BPH14178

STATE OF WEST VIRGINIA
Purchasing Division

PURCHASING AFFIDAVIT

MANDATE: Under W. Va. Code §5A-3-10a, no contract or renewal of any contract may be awarded by the state or any of its political subdivisions to any vendor or prospective vendor when the vendor or prospective vendor or a related party to the vendor or prospective vendor is a debtor and: (1) the debt owed is an amount greater than one thousand dollars in the aggregate; or (2) the debtor is in employer default.

EXCEPTION: The prohibition listed above does not apply where a vendor has contested any tax administered pursuant to chapter eleven of the W. Va. Code, workers' compensation premium, permit fee or environmental fee or assessment and the matter has not become final or where the vendor has entered into a payment plan or agreement and the vendor is not in default of any of the provisions of such plan or agreement.

DEFINITIONS:

"Debt" means any assessment, premium, penalty, fine, tax or other amount of money owed to the state or any of its political subdivisions because of a judgment, fine, permit violation, license assessment, defaulted workers' compensation premium, penalty or other assessment presently delinquent or due and required to be paid to the state or any of its political subdivisions, including any interest or additional penalties accrued thereon.

"Employer default" means having an outstanding balance or liability to the old fund or to the uninsured employers' fund or being in policy default, as defined in W. Va. Code § 23-2c-2, failure to maintain mandatory workers' compensation coverage, or failure to fully meet its obligations as a workers' compensation self-insured employer. An employer is not in employer default if it has entered into a repayment agreement with the Insurance Commissioner and remains in compliance with the obligations under the repayment agreement.

"Related party" means a party, whether an individual, corporation, partnership, association, limited liability company or any other form or business association or other entity whatsoever, related to any vendor by blood, marriage, ownership or contract through which the party has a relationship of ownership or other interest with the vendor so that the party will actually or by effect receive or control a portion of the benefit, profit or other consideration from performance of a vendor contract with the party receiving an amount that meets or exceed five percent of the total contract amount.

AFFIRMATION: By signing this form, the vendor's authorized signer affirms and acknowledges under penalty of law for false swearing (*W. Va. Code* §61-5-3) that neither vendor nor any related party owe a debt as defined above and that neither vendor nor any related party are in employer default as defined above, unless the debt or employer default is permitted under the exception above.

WITNESS THE FOLLOWING SIGNATURE:

Vendor's Name: ______________________________

Authorized Signature: ______________________________ Date: ______________

State of ______________

County of ______________, to-wit:

Taken, subscribed, and sworn to before me this ___ day of ______________, 20___.

My Commission expires ______________, 20___.

AFFIX SEAL HERE **NOTARY PUBLIC** ______________

Rev. 04/14

State of West Virginia

VENDOR PREFERENCE CERTIFICATE

Certification and application* is hereby made for Preference in accordance with ***West Virginia Code***, §5A-3-37. (Does not apply to construction contracts). ***West Virginia Code***, §5A-3-37, provides an opportunity for qualifying vendors to request (at the time of bid) preference for their residency status. Such preference is an evaluation method only and will be applied only to the cost bid in accordance with the ***West Virginia Code***. This certificate for application is to be used to request such preference. The Purchasing Division will make the determination of the Vendor Preference, if applicable.

1. **Application is made for 2.5% vendor preference for the reason checked:**

____ Bidder is an individual resident vendor and has resided continuously in West Virginia for four (4) years immediately preceding the date of this certification; **or,**

____ Bidder is a partnership, association or corporation resident vendor and has maintained its headquarters or principal place of business continuously in West Virginia for four (4) years immediately preceding the date of this certification; or 80% of the ownership interest of Bidder is held by another individual, partnership, association or corporation resident vendor who has maintained its headquarters or principal place of business continuously in West Virginia for four (4) years immediately preceding the date of this certification; **or,**

____ Bidder is a nonresident vendor which has an affiliate or subsidiary which employs a minimum of one hundred state residents and which has maintained its headquarters or principal place of business within West Virginia continuously for the four (4) years immediately preceding the date of this certification; **or,**

2. **Application is made for 2.5% vendor preference for the reason checked:**

____ Bidder is a resident vendor who certifies that, during the life of the contract, on average at least 75% of the employees working on the project being bid are residents of West Virginia who have resided in the state continuously for the two years immediately preceding submission of this bid; **or,**

3. **Application is made for 2.5% vendor preference for the reason checked:**

____ Bidder is a nonresident vendor employing a minimum of one hundred state residents or is a nonresident vendor with an affiliate or subsidiary which maintains its headquarters or principal place of business within West Virginia employing a minimum of one hundred state residents who certifies that, during the life of the contract, on average at least 75% of the employees or Bidder's affiliate's or subsidiary's employees are residents of West Virginia who have resided in the state continuously for the two years immediately preceding submission of this bid; **or,**

4. **Application is made for 5% vendor preference for the reason checked:**

____ Bidder meets either the requirement of both subdivisions (1) and (2) or subdivision (1) and (3) as stated above; **or,**

5. **Application is made for 3.5% vendor preference who is a veteran for the reason checked:**

____ Bidder is an individual resident vendor who is a veteran of the United States armed forces, the reserves or the National Guard and has resided in West Virginia continuously for the four years immediately preceding the date on which the bid is submitted; **or,**

6. **Application is made for 3.5% vendor preference who is a veteran for the reason checked:**

____ Bidder is a resident vendor who is a veteran of the United States armed forces, the reserves or the National Guard, if, for purposes of producing or distributing the commodities or completing the project which is the subject of the vendor's bid and continuously over the entire term of the project, on average at least seventy-five percent of the vendor's employees are residents of West Virginia who have resided in the state continuously for the two immediately preceding years.

7. **Application is made for preference as a non-resident small, women- and minority-owned business, in accordance with *West Virginia Code* §5A-3-59 and *West Virginia Code of State Rules.***

____ Bidder has been or expects to be approved prior to contract award by the Purchasing Division as a certified small, women- and minority-owned business.

Bidder understands if the Secretary of Revenue determines that a Bidder receiving preference has failed to continue to meet the requirements for such preference, the Secretary may order the Director of Purchasing to: (a) reject the bid; or (b) assess a penalty against such Bidder in an amount not to exceed 5% of the bid amount and that such penalty will be paid to the contracting agency or deducted from any unpaid balance on the contract or purchase order.

By submission of this certificate, Bidder agrees to disclose any reasonably requested information to the Purchasing Division and authorizes the Department of Revenue to disclose to the Director of Purchasing appropriate information verifying that Bidder has paid the required business taxes, provided that such information does not contain the amounts of taxes paid nor any other information deemed by the Tax Commissioner to be confidential.

Under penalty of law for false swearing (West Virginia Code, §61-5-3), Bidder hereby certifies that this certificate is true and accurate in all respects; and that if a contract is issued to Bidder and if anything contained within this certificate changes during the term of the contract, Bidder will notify the Purchasing Division in writing immediately.

Bidder: ______________________ **Signed:** ______________________

Date: ______________________ **Title:** ______________________