Guidelines for Writing Cryptography Specifications

draft-xxx-cfrg-guidelines-00

IETF 116, Yokohama

Goals and Requirements for Specifications

Minimize ambiguity and misinterpretations

Leading to clearer specifications and more accurate implementation

Ensure consistent and correct implementation

Providing precise instructions and explanations

Facilitate review and analysis

Allowing for the verification of security properties and the identification of potential vulnerabilities

Enable standardization and interoperability

Promoting collaboration and compatibility between various systems and protocol

By following these guidelines, specification authors can create documents that facilitate the development, analysis, and implementation of cryptographic solutions.

Simplicity, Precision, and Consistency in Cryptography Specifications

Simplicity

Strive for simplicity in problem statements, technical content, and presentation to make documents more accessible and reduce the risk of misinterpretation.

Precision

Use clear and concise language, provide explicit instructions, address edge cases and potential pitfalls, employ formal notation or pseudocode, specify data formats and encodings, document assumptions and dependencies.

Consistency

Establish a consistent terminology, maintain a uniform style and tone, use a logical structure, provide consistent formatting, be consistent with conventions and notations, reference external documents consistently, keep the broader context in mind.

By focusing on simplicity, precision, and consistency in cryptography specifications, authors can create documents that are more accessible to a wider audience and reduce the risk of misinterpretation or implementation errors. This will ultimately lead to more secure, reliable, and interoperable cryptographic systems.

Representing Mathematical Operations in Cryptography Specification

- Notation Consistency

  Establish a clear notation system and use it consistently throughout the document

- Use of Standard Mathematical Symbols

  Use standard mathematical symbols to represent mathematical operations whenever possible

- Explicitly Defining Custom Operations

  Explicitly define custom operations and provide clear explanations and examples

- Pseudocode and Algorithmic Descriptions

  Provide pseudocode alongside mathematical expressions to improve clarity

- Visual Representations

  Include visual representations such as diagrams, tables, or visualizations to convey complex concepts

Catering to Different Audiences When Writing a Specification

Cater to Implementers

Provide step-by-step instructions, test vectors, and best practices for representing components of the specification in code

Cater to Researchers

Clearly define mathematical concepts and notations, provide security definitions, goals, and threat models, and present security proofs

Cater to Protocol Designers

Define interfaces, APIs, or functions exposed by the protocol or primitive, describe corner cases, and provide guidance on configuration and parameter selection

By considering the needs of implementers, researchers, and protocol designers when writing a specification, it is possible to create a document that is clear, concise, and unambiguous, leading to more secure and interoperable systems.

Defining Security Definitions and Threat Model

- Defining Security Goals

  Explicitly state security goals and clarify any trade-offs or limitations

- Formalizing Security Definitions

  Express security definitions in a formal language with clear explanations and examples

- Describing the Threat Model

  Detail the capabilities, resources, and motivations of adversaries

- Addressing Known Vulnerabilities and Attacks

  Discuss known vulnerabilities and explain how they are addressed or mitigated

- Providing Guidance on Secure Implementation and Deployment

  Provide guidance on secure implementation and deployment of the proposed algorithms and protocols

Promoting Reusability and Collaboration & Compromise

Promoting Reusability

Reusability is essential for creating efficient, interoperable, and widely adopted cryptographic systems

Collaboration and Compromise

Developing effective cryptography specifications requires collaboration between multiple stakeholders.

Reusability in cryptography specifications is essential for creating efficient, interoperable, and widely adopted cryptographic systems. Collaboration and compromise are key to developing effective cryptography specifications, and guidance should be provided to incorporate reusability principles into the specification development process.

A Foundation for Authors

Writing cryptography specifications is a complex process that requires both technical knowledge and creative problem-solving. This document intends to give authors a comprehensive set of guidelines for creating a useful cryptography specification.

Guidelines for Writing Cryptography Specifications

draft-xxx-cfrg-guidelines-00

IETF 116, Yokohama