Network Working Group
Request for Comments: 2420
Category: Standards Track

H. Kummert
Nentec GmbH

September 1998

# The PPP Triple-DES Encryption Protocol (3DESE)

## Status of this Memo



## Copyright Notice



## Abstract

The Point-to-Point Protocol (PPP) [1] provides a standard method for transporting multi-protocol datagrams over point-to-point links.

The PPP Encryption Control Protocol (ECP) [2] provides a method to negotiate and utilize encryption protocols over PPP encapsulated links.

This document provides specific details for the use of the Triple-DES standard (3DES) [6] for encrypting PPP encapsulated packets.

## Table of Contents

## 1. Introduction

The purpose of encrypting packets exchanged between two PPP implementations is to attempt to insure the privacy of communication conducted via the two implementations. There exists a large variety of encryption algorithms, where one is the DES algorithm. The DES encryption algorithm is a well studied, understood and widely implemented encryption algorithm. Triple-DES means that this algorithm is applied three times on the data to be encrypted before it is sent over the line. The variant used is the DES-EDE3-CBC, which is described in more detail in the text. It was also chosen to be applied in the security section of IP [5].

This document shows how to send via the Triple-DES algorithm encrypted packets over a point-to-point-link. It lies in the context of the generic PPP Encryption Control Protocol [2].

Because of the use of the CBC-mode a sequence number is provided to ensure the right order of transmitted packets. So lost packets can be detected.

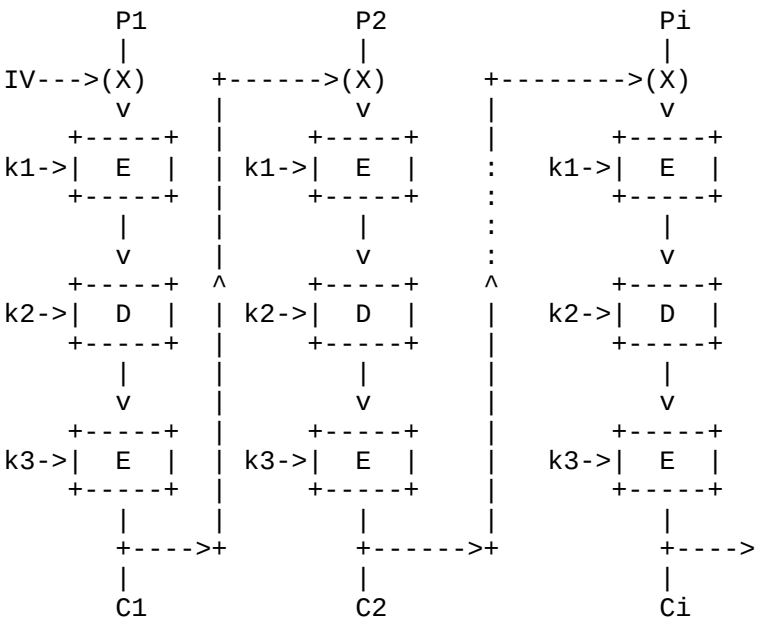
The padding section reflects the result of the discussion on this topic on the ppp mailing list.

In this document, the key words "MUST", "SHOULD", and "recommended" are to be interpreted as described in [3].

### 1.1 Algorithm

The DES-EDE3-CBC algorithm is a simple variant of the DES-CBC algorithm. In DES-EDE3-CBC, an Initialization Vector (IV) is XOR'd with the first 64-bit (8 octet) plaintext block (P1). The keyed DES function is iterated three times, an encryption (E) followed by a decryption (D) followed by an encryption (E), and generates the ciphertext (C1) for the block. Each iteration uses an independent key: k1, k2 and k3.

For successive blocks, the previous ciphertext block is XOR'd with the current 8-octet plaintext block (Pi). The keyed DES-EDE3 encryption function generates the ciphertext (Ci) for that block.




To decrypt, the order of the functions is reversed: decrypt with k3, encrypt with k2, decrypt with k1, and XOR with the previous cipher-text block.

When all three keys (k1, k2 and k3) are the same, DES-EDE3-CBC is equivalent to DES-CBC.

## 1.2 Keys

The secret DES-EDE3 key shared between the communicating parties is effectively 168-bits long. This key consists of three independent 56-bit quantities used by the DES algorithm. Each of the three 56-bit subkeys is stored as a 64-bit (8 octet) quantity, with the least significant bit of each octet used as a parity bit.

When configuring keys for 3DESE those with incorrect parity or so-called weak keys [6] SHOULD be rejected.

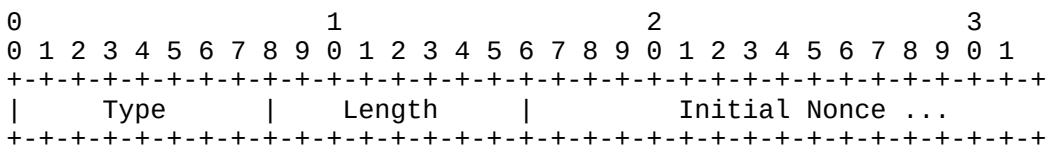
# 2. 3DESE Configuration Option for ECP

Description

The ECP 3DESE Configuration Option indicates that the issuing implementation is offering to employ this specification for decrypting communications on the link, and may be thought of as a request for its peer to encrypt packets in this manner. The

ECP 3DESE Configuration Option has the following fields, which are transmitted from left to right:

Figure 1: ECP 3DESE Configuration Option




Type

2, to indicate the 3DESE protocol.

Length

10

Initial Nonce

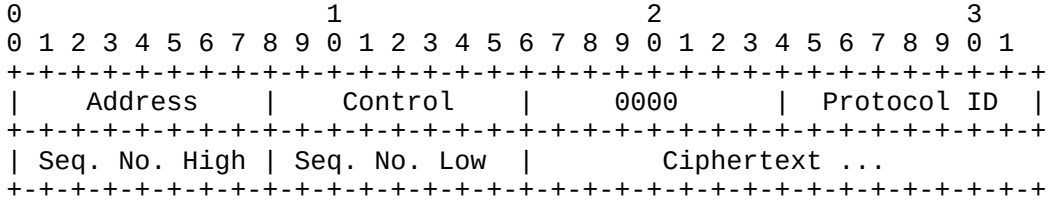
This field is an 8 byte quantity which is used by the peer implementation to encrypt the first packet transmitted after the sender reaches the opened state. To guard against replay attacks, the implementation SHOULD offer a different value during each ECP negotiation.

## 3. Packet format for 3DESE

Description

The 3DESE packets that contain the encrypted payload have the following fields:

Figure 2: 3DESE Encryption Protocol Packet Format




Address and Control

These fields MUST be present unless the PPP Address and Control Field Compression option (ACFC) has been

negotiated.

Protocol ID

The value of this field is 0x53 or 0x55; the latter indicates the use of the Individual Link Encryption Control Protocol and that the ciphertext contains a Multilink fragment. Protocol Field Compression MAY be applied to the leading zero if negotiated.

Sequence Number

These 16-bit numbers are assigned by the encryptor sequentially starting with 0 (for the first packet transmitted once ECP has reached the opened state).

Ciphertext

The generation of this data is described in the next section.

## 4. Encryption

Once the ECP has reached the Opened state, the sender MUST NOT apply the encryption procedure to LCP packets nor ECP packets.

If the async control character map option has been negotiated on the link, the sender applies mapping after the encryption algorithm has been run.

The encryption algorithm is generally to pad the Protocol and Information fields of a PPP packet to some multiple of 8 bytes, and apply 3DES as described in section 1.1 with the three 56-bit keys k1, k2 and k3.

The encryption procedure is only applied to that portion of the packet excluding the address and control field.

When encrypting the first packet after ECP stepped into opened state the Initial Nonce is encrypted via 3DES-algorithm before its use.

### 4.1 Padding

Since the 3DES algorithm operates on blocks of 8 octets, plain text packets which are of length not a multiple of 8 octets must be padded prior to encrypting. If this padding is not removed after decryption this can be injurious to the interpretation of some protocols which do not contain an explicit length field in their protocol headers.

DOCKET ALARM

# Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

## Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

## Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

## Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

## API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

**LAW FIRMS**
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

**FINANCIAL INSTITUTIONS**
Litigation and bankruptcy checks for companies and debtors.

**E-DISCOVERY AND LEGAL VENDORS**
Sync your system to PACER to automate legal marketing.