岐阜大学工学部
電気電子・情報工学科
令和 6 年度卒業論文

# セキュアな V2V アドホックネットワーク ルーティングプロトコルのための EdDSA 署名方式の評価

三嶋研究室

学籍番号：1213033107

永野 正剛

指導教員：三嶋 美和子 教授

# 目次

# はじめに

# 第 1 章　準備

## 1.1　VANET

vanet 書くよ

## 1.2　GPSR

GPSR 書くよ

## 1.3　デジタル署名

### 1.3.1　EdDSA

実験に導入した Ed25519 のプロトコル内で使用されるリトルエンディアン、エンコーディング、プルーニングについて説明する.

鍵生成

1. 法とする素数 $p$、楕円曲線 $E$、基準点 $G$、鍵のサイズ $b$、ハッシュ関数 $H$、コファクター $c$、位数 $L$ を定める.
2. $b$ バイトのランダムな値 $sk$ を生成し、秘密鍵とする.
3. $h = H(sk)$ を計算し、$h$（オクテット文字列）を前半部分 h[0] から h[31] と後半部分 h[32] から h[63] に分ける.
4. 前半部分 $s$[0] から $s$[31] を使ってプルーニングしたものをリトルエンディアンの整数として解釈し、スカラー $s \pmod{L}$ を生成する.
5. 基準点 $G$ を使って $A = sG$ を計算し、A のエンコードを公開鍵とする.

署名生成フェーズ

1. 秘密鍵 $sk$ を使って、ハッシュ値 $h = H(sk)$ を計算する.
2. $h$ の後半部分 $h[32]$ から $h[63]$ を使って、$r = DEC(H())$.
3.
4.

署名検証フェーズ

1.
2.

# 第 2 章　EdDSA

**ここに EdDSA の説明書くよ**

この章では, 本研究で用いる Ed25519 について概説する.

## 2.1　Ed25519

EdDSA には IETF の RFC8032 で推奨される二つのパラメーターが存在する. そのうちのひとつが本研究で使用する Ed25519 である. 現在、Ed25519 は EdDSA の最も一般的なインスタンスであり、約 128 ビットのセキュリティを提供する Edwards Curve25519 に基づいている.

## 2.2　データの変換

EdDSA のアルゴリズム内では、整数や点をオクテット列に変換するエンコードとその逆変換であるデコードが行われる.
以下で使用されるデータの変換について説明する.

**オクテット**

オクテットは $b_0b_1b_2b_3b_4b_5b_6b_7$ のような 8 ビットのビット列であり、$b_0$ を最下位ビット、$b_7$ を最上位ビットと呼ぶ.

例.　数値 $0d128$ のオクテットに対応するビット列は 00000001 である.

**リトルエンディアン**

リトルエンディアン形式では、データを格納する際に数値の下位バイト（最下位ビットに近い方）から順に配置する.

例.　数値 $0x12345678$ をリトルエンディアン形式で格納すると、$0x78, 0x56, 0x34, 0x12$ となる.

**エンコードとデコード**

1. $ENC(s)$
   整数 $s(0 < s < L-1)$ は、8 ビットずつをオクテットとみなすことに基づき、リトルエンディアン形式で $\frac{b}{8}$ オクテットに格納される.
2. $DEC(t)$
   $t$ はオクテット列であり、$ENC(s)$ の逆変換によって整数 $s$ に変換される.
3. $ENCE(A)$
   $E$ の点 $A$ は、元 $(x, y)$ の $y$ を $ENC(y)$ によりオクテット列に変換し、その最終オクテットの最上位ビットに $x$ 座標の符号（$x \geqq 0$ ならば 0、$x < 0$ ならば 1）が格納される.
4. $DECE(A)$
   $t$ は変換元の $\frac{b}{8}$ オクテットのオクテット列である.
   (a) $t$ の最終オクテットの最上位ビットを $x$ 座標の符号として取り出し $x_0$ に格納する.
   （$x_0 = 0$ または、$x_0 = 1$ とする.）
   (b) $t$ の最終オクテットの最上位ビットを 0 に設定する.
   (c) $y = DEC(t)$ を計算し、$0 \leq y < p$ でないならばデコード失敗.
   (d) 以下の処理を行う.
      i. $u = y^2 - 1, v = d * y^2 + 1$ として $x = uv^3(uv^7)^{\frac{p-5}{8}} \bmod p$ を計算する.
      ii. $vx^2 \neq \pm u \bmod p$ ならばデコード失敗.
      iii. $vx^2 = u \bmod p$ ならば、$x = 2^{\frac{p-1}{4}} x$

## 2.3 楕円曲線

## 2.4 パラメータ

EdDSA のパラメータは以下のようである.

- $p$ : 法となる素数. EdDSA は $\mathbb{F}_p$ 上の楕円曲線を使用する.
- $b$ : $p < 2^{b-1}$ となる正整数. 公開鍵の長さを表す.
- $E'$ : エンコーディング関数.
- $H$ : ハッシュ関数. 2b ビット長のハッシュ値を出力する.
- $(a, d, c, l)$ : 楕円曲線 $E$ を決定するパラメータ.

$$E := \{(x, y) \in \mathbb{F}_p \times \mathbb{F}_p \mid ax^2 + y^2 = 1 + dx^2y^2\}$$

– $a$ は $\mathbb{F}_p$ 上平方剰余、$d$ は非ゼロの非剰余.

– $c = 2$ または 3. $l$ は奇素数で $E$ の位数 $\#E = 2^c l$ となるような数.

- $n$ : $c \leq n < b$ となる整数.
- $B$ : $E$ 上のベースポイント. $B \neq (0, 1)$
- PH : プレハッシュ関数. （HashEdDSA の場合に用いる）

## 2.5 デジタル署名アルゴリズム

Ed25519 における 3 つのアルゴリズムの手順を以下に述べる.

**鍵生成**

1. 法とする素数 $p$、楕円曲線 $E$、基準点 $B$、鍵のサイズ $b$、ハッシュ関数 $H$、エンコーディング関数 $E'$、コファクター $c$、位数 $L$ を定める.
2. $b$ バイトのランダムな値 $sk$ を生成し、**秘密鍵**とする.
3. $h = H(sk)$ を計算し、$h$（オクテット文字列）を前半部分 $h[0]$ から $h[31]$ と後半部分 $h[32]$ から $h[63]$ に分ける.
4. 前半部分の最初のバイト $(h[0])$ の下位 3 ビットを 0 にクリアする. 最後のバイト $(h[31])$ の最上位ビットを 0 に、最上位 2 ビット目を 1 に設定したものをリトルエンディアンの整数として解釈し、スカラー $s \pmod L$ を生成する.
5. 基準点 $B$ を使って $A = sB$ を計算し、$ENCE(A)$ を**公開鍵**とする.

スカラー値 $s$ は 8 の倍数で，正確に 255 ビットとなる

**署名生成**

1. 秘密鍵 $sk$ を使って、ハッシュ値 $h = H(sk)$ を計算する.
2. $h$ の後半部分 $h[32]$ から $h[63]$ を使って、

$$r = DEC(H(h[32] \mid\mid ... \mid\mid h[63] \mid\mid M)) \bmod L.$$

を計算する.

3. $R = ENCE([r]B)$ を計算する.

4. $k = DEC(H(R \,||\, A \,||\, M))\ \mathrm{mod}L$ を計算する.

5. $S = ENC((r + k * s)\ \mathrm{mod}L)$ を計算する.

6. $(R, S)$ を署名とする.

秘密のスカラー値

**署名検証**

1. 署名 $(R, S)$ を受け取り、$R$ をデコードする.

2.

## 2.6 ECDSA と EdDSA の比較

# 第 3 章　提案手法

# 第 4 章　シミュレーション環境

# 第 5 章　シミュレーション実験

# 第 6 章　EdDSA に関する実装評価まとめ

# おわりに

# 謝辞

# 参考文献