

# 郵便を用いた超低速IP通信システムの検討

上羽 未栄<sup>†a)</sup> 信濃 真伊<sup>†b)</sup> 佐伯 真紘<sup>†c)</sup> 一式 すみれ<sup>†d)</sup>

2025-11-07

<sup>†</sup> 東京広域電話網, <https://tkytel.github.io/>

a) a.k.a. KusaReMKN, mkn@kusaremkn.com

b) me@shinanomai.xyz

c) a.k.a. Nejikugi, me@scrwnl.eu.org

d) a.k.a. yude, i@yude.jp

2025-11-17

## 郵便を用いた超低速IP通信システムの検討

郵便を用いた超低速IP通信システムの検討

上羽 未栄<sup>†a)</sup> 信濃 真伊<sup>†b)</sup> 佐伯 真紘<sup>†c)</sup> 一式 すみれ<sup>†d)</sup>

2025-11-07

<sup>†</sup> 東京広域電話網, <https://tkytel.github.io/>

a) a.k.a. KusaReMKN, mkn@kusaremkn.com

b) me@shinanomai.xyz

c) a.k.a. Nejikugi, me@scrwnl.eu.org

d) a.k.a. yude, i@yude.jp

はじまるよ～。

郵便を用いた超低速IP通信システムの検討と題して、東京広域電話網のLoL-LIPoPチームを代表して上羽未栄が発表するよ。

# 今回のおはなし

インターネットを支える技術

鳥類キャリアによる IP 通信

郵便を用いた IP 通信

システムの実装

通信の実験

まとめ・検討事項

2025-11-17

2

郵便を用いた超低速 IP 通信システムの検討

└ 今回のおはなし

今回の発表の流れはこんな感じだよ。おおよそ 25 分くらいで進められたらいいな。

今回のおはなし

インターネットを支える技術  
鳥類キャリアによる IP 通信  
郵便を用いた IP 通信  
システムの実装  
通信の実験  
まとめ・検討事項

# 今回の発表に関するひとびと



実装



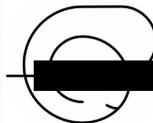
発案



命名



監修



被害者ら

3

2025-11-17 郵便を用いた超低速IP通信システムの検討

## └ 今回の発表に関するひとびと

今回の発表に関する人はスライドに示される通りだよ。誰かが発案したものを誰かが実装しようと試みて、その途中で誰かが命名したよ。実装にあたっては誰かによる監修があったよ。また、実験にあたっては二名の被害者による協力があったよ。この場を借りて御礼申し上げるよ。



# みかんちゃんについて

---

2025-11-17

郵便を用いた超低速 IP 通信システムの検討

└ みかんちゃんについて

みかんちゃんについて

まずは自己紹介するよ～。

# 自称・大天才美少女プログラミング初心者

うわばみかん  
「上羽未栞」あるいは「KusaReMKN」  
みかんちゃんって呼んでね！

17<sub>(18)</sub>歳のJK（超重要）

実はプログラマでもエンジニアでもない  
普段はホラを吹いて生活している  
古い計算機っぽいものが大好き

Twitterで思想を垂れ流すことが得意

<https://kusaremkn.com/>も見てね



郵便を用いた超低速IP通信システムの検討

└みかんちゃんについて

2025-11-17

└自称・大天才美少女プログラミング初心者

自称大天才美少女プログラミング初心者の上羽未栞だよ。みかんちゃんって呼ばれると大変喜ぶよ。

18歳のJKだよ。重要だよ。大天才とか偉ぶっているけれど、実はプログラマでもエンジニアでもないよ。普段はホラ吹きを生業としているよ。古い計算機っぽいものが大好きで、いろいろなものに手を出してはそれに掛けられた呪いを解くことを趣味にしているよ。

Twitterや自分のウェブサイトで思想を垂れ流すのが得意だよ。暇な人は眺めてみてね。

自称・大天才美少女プログラミング初心者

うわばみかん  
「上羽未栞」あるいは「KusaReMKN」  
みかんちゃんって呼んでね！

17<sub>(18)</sub>歳のJK（超重要）  
実はプログラマでもエンジニアでもない  
普段はホラを吹いて生活している  
古い計算機っぽいものが大好き

Twitterで思想を垂れ流すことが得意  
<https://kusaremkn.com/>も見てね

# インターネットを支える技術

2025-11-17

郵便を用いた超低速IP通信システムの検討  
└ インターネットを支える技術

インターネットを支える技術

内容に入る前に、まずはインターネットを支える技術について思いを馳せてみるよ。

# インターネットなしでは生きられない！

## 身の周りにある便利なもの

- ・Twitter や YouTube を支えている Web
- ・しゃべる洗濯機や冷蔵庫を支えている IoT
- ・睡眠時間を奪い心身の健康を蝕む VRChat

全てネットワークを使った通信のおかげ

それなのに……

ネットワークがなぜ繋がるのかあまり考えていない

ネットワークの通信にタダ乗りしているだけ

2025-11-17

郵便を用いた超低速 IP 通信システムの検討

└ インターネットを支える技術

└ インターネットなしでは生きられない！

現代社会においては、あらゆるもののがネットワークに接続されていて、何かしら通信していることが常になってきているよ。よく例として挙げられるものは Web があるし、最近では IoT なんかも生活の一部に感じられるようになってきたよ。今お前らがこうやって VRChat をしていられるのもネットワークのおかげだよ。いわんや、ネットワークを用いた通信は当然のものになっているよ。しかし、これを支える技術、特により物理側に近い層については関心を寄せる人はあまり多くないような印象があるよ。ネットワークがなぜ繋がるのかをあまり考えずに、その通信にタダ乗りしている風潮があるよ。これは由々しき事態であるから、積極的にネットワークに思いを馳せていいきたいね。

インターネットなしでは生きられない！

身の周りにある便利なもの

- ・Twitter や YouTube を支えている Web
- ・しゃべる洗濯機や冷蔵庫を支えている IoT
- ・睡眠時間を奪い心身の健康を蝕む VRChat

全てネットワークを使った通信のおかげ  
それなのに……  
ネットワークがなぜ繋がるのかあまり考えていない  
ネットワークの通信にタダ乗りしているだけ

# インターネットのしくみ<sup>[1]</sup>



サービスや機能の提供  
HTTP, OSC, DNS, DHCP, ...

エンドツーエンドの接続  
TCP, UDP, QUIC, ...

ネットワークを超えた接続  
IPv4, IPv6, ARP, NDP, ...

物理的で直接的な接続  
Ethernet, Wi-Fi, ...

## 2025-11-17 郵便を用いた超低速IP通信システムの検討 └ インターネットを支える技術

### └ インターネットのしくみ<sup>[1]</sup>

ネットワークの代表格であるとろのインターネットの基本原理について立ち返るよ。TCP/IPでは、より上位から順に、アプリケーション層、トランSPORT層、インターネット層、リンク層の4つの階層から成っているよ。

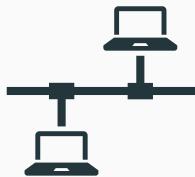
アプリケーション層は、例えばWebやVRChatといったサービス、名前解決やアドレス自動割り当てなどの機能を実現するよ。トランSPORT層は、Webのデータをブラウザへ、VRChatのデータをVRChatへといった具合に適切な通信を適切なアプリケーションに届けるよ。インターネット層では、世界中を張り巡らされた数多のネットワークを通り抜け、目的のホスト（コンピュータ）までデータを届けるよ。リンク層では、そのホストから物理的に直接通信できる範囲でデータを届けるよ。

これらの層が連携して仕事をすることで様々な通信を実現できるよ。



サービスや機能の提供  
HTTP, OSC, DNS, DHCP, ...  
エンドツーエンドの接続  
TCP, UDP, QUIC, ...  
ネットワークを超えた接続  
IPv4, IPv6, ARP, NDP, ...  
物理的で直接的な接続  
Ethernet, Wi-Fi, ...

# 情報を伝える根幹のしくみ



電気や光



電波



音



ハト



物理的で直接的な接続  
Ethernet, Wi-Fi, ...

2025-11-17 郵便を用いた超低速IP通信システムの検討  
└インターネットを支える技術

└情報を伝える根幹のしくみ

今回のお話では、これらの層の一番下位の部分、リンク層に着目するよ。リンク層で用いられるプロトコルとして、LANケーブルに電気を通したり、光ファイバに光を通したりして通信するEthernetや電波を遣って通信をするWi-Fiなどがあるよ。また、少し古めかしいところではダイヤルアップ通信（電話線とモデムとを使ったPPP）もリンク層の通信だよ。この場合、コンピュータは電話を使って「直接」通信していることになるよ。

TCP/IPでは下位の層は上位の層にデータを受け渡せば良いよ。つまり、リンク層では、同一リンク内で通信でき、適切にデータを伝送できれば良いので、ここで用いる伝送方式はかなり自由に選択できるよ。「味のあるリンク」で当たり前の通信を実現できると面白そうだね。



# 鳥類キャリアによるIP通信

---

2025-11-17

郵便を用いた超低速IP通信システムの検討  
└ 鳥類キャリアによるIP通信

鳥類キャリアによるIP通信

ここで、「味のあるリンク」であるところの鳥類キャリアについて思いを馳せていくよ。

# RFC 1149: 1990年4月1日発のジョーク RFC

鳥類キャリアを用いたIP通信の手法が検討されている<sup>[2]</sup>  
QoSの提供<sup>[3]</sup>やIPv6への対応<sup>[4]</sup>など改良・拡張されている  
2001年ノルウェーで実装実験が行われた

- 6羽のハトを使ってpingパケットを伝送
- 道中、別のハトの群れと一緒に寄り道するなど
- 4羽がパケットを持って戻ってきた（33%の損失）
- 往復通信時間は3211秒から6389秒までさまざま

2025-11-17

郵便を用いた超低速IP通信システムの検討  
└ 鳥類キャリアによるIP通信

RFC 1149: 1990年4月1日発のジョーク RFC

鳥類キャリアを用いたIP通信の手法が検討されている<sup>[2]</sup>  
QoSの提供<sup>[3]</sup>やIPv6への対応<sup>[4]</sup>など改良・拡張されている  
2001年ノルウェーで実装実験が行われた

- 6羽のハトを使ってpingパケットを伝送
- 道中、別のハトの群れと一緒に寄り道するなど
- 4羽がパケットを持って戻ってきた（33%の損失）
- 往復通信時間は3211秒から6389秒までさまざま

└ RFC 1149: 1990年4月1日発のジョーク RFC

興味深いリンクの手法として、「鳥類キャリア」があるよ。これは、情報を伝える手段として、電気や電波の代わりに鳥類（つまり伝書鳩など）を用いるものだよ。実験用のプロトコルとしてRFCに規格があるよ。当然、ジョークのRFCであるが、QoSの検討やIPv6への対応も図られているよ。

実装も試みられていて、2001年にはノルウェーで実験が行われているよ。この実験では6羽のハトを使ってpingパケットを伝送したよ。この日は現地では珍しい晴れの日であったようで、道中別のハトの群れと一緒に一時間ほど寄り道するなどしていたようだよ。パケットは無事に相手の端へ辿り着き、4羽が応答パケットを持って戻ってきたよ。パケロス率は33%、RTTは3211秒から6389秒まで（大体1-2時間程度）であったようだよ。

# 鳥類キャリアを用いたIP通信の手順

0. IPデータグラムが生成される
1. IPデータグラムを小さな細長い紙に16進数で印刷する
2. 鳥の脚に紙を巻き付け、テープで止める
3. 鳥を目的の端に飛ばす
  
4. 飛んできた鳥の脚に巻き付けられた紙切れを剥がす
5. 紙に印刷されているデータグラムを読み取る
6. コンピュータはデータグラムを処理する

2025-11-17

郵便を用いた超低速IP通信システムの検討

└ 鳥類キャリアによるIP通信

└ 鳥類キャリアを用いたIP通信の手順

## 鳥類キャリアを用いたIP通信の手順

0. IPデータグラムが生成される
1. IPデータグラムを小さな細長い紙に16進数で印刷する
2. 鳥の脚に紙を巻き付け、テープで止める
3. 鳥を目的の端に飛ばす
  
4. 飛んできた鳥の脚に巻き付けられた紙切れを剥がす
5. 紙に印刷されているデータグラムを読み取る
6. コンピュータはデータグラムを処理する

鳥類キャリアによるIP通信はスライドに示される手段で行われるよ。

IPデータグラムはそのまま（ヘッダなどを付けることなく）16進数で印刷されるよ。ヘッダを付けてしまうと、頭を切り落とされたときに伝送パフォーマンスが下がることが懸念されているよ。

データグラムの読み取りにはOCRを使っても良いし、人間を使っても良いよ。

# 鳥類キャリアを用いたIP通信の特徴

- ・通信の帯域幅は鳥の脚の長さによる（時間経過で変化）
- ・キャリアは喪失し得る（がIPとしては問題ない）
- ・MTUは鳥によって可変（典型的には256 mg）
- ・用いる鳥の種類によってQoSを設定できる
- ・ブロードキャスト・マルチキャストはむづかしい
- ・IPv4とIPv6とを区別するにはIPヘッダを見るしかない
- ・位置（地域）によってはキャリアのホップが短い
- ・盗聴（覗き見）によるセキュリティ上の懸念がある

2025-11-17

## 郵便を用いた超低速IP通信システムの検討

### └ 鳥類キャリアによるIP通信

#### └ 鳥類キャリアを用いたIP通信の特徴

RFCの本文で述べられている実用的な特徴はスライドに示す通りだよ。重要なところを紹介するよ。IPはベストエフォートの通信を提供するので、キャリアは喪失しても問題ないよ。MTUやQoSについては鳥の運用期間や種類によって設定されるよ。これらについては、人間の関与できるところではないので、キャリア特有の問題だと考えられるよ。また、鳥のヘッダを見てもパケットの種別を判断できないので、IPv4とIPv6との区別はこのレイヤで判別できないよ。これは、将来的にIPよりも素晴らしいプロトコルが登場したときに不便だよ。

#### 鳥類キャリアを用いたIP通信の特徴

- ・通信の帯域幅は鳥の脚の長さによる（時間経過で変化）
- ・キャリアは喪失し得る（がIPとしては問題ない）
- ・MTUは鳥によって可変（典型的には256 mg）
- ・用いる鳥の種類によってQoSを設定できる
- ・ブロードキャスト・マルチキャストはむづかしい
- ・IPv4とIPv6とを区別するにはIPヘッダを見るしかない
- ・位置（地域）によってはキャリアのホップが短い
- ・盗聴（覗き見）によるセキュリティ上の懸念がある

# 郵便を用いたIP通信

2025-11-17 郵便を用いた超低速IP通信システムの検討  
└ 郵便を用いたIP通信

郵便を用いたIP通信

ということで、鳥類キャリアのかわりに郵便を用いることについて考えてみるよ。

# 郵便を用いたIP通信

キャリアとして鳥類のかわりに郵便を用いる  
→ 鳥類キャリア特有の制約を取り扱える



2025-11-17 郵便を用いた超低速IP通信システムの検討  
└ 郵便を用いたIP通信

└ 郵便を用いたIP通信

この通信方式では鳥類キャリアによるIP通信をベースに、キャリアとして鳥類の代わりに郵便を用いることで鳥類キャリア特有の制約を取り扱うよ。  
つまり、鳥によって運ばれていたパケットを郵便によって運ぶほかは基本的に鳥類キャリアを用いたIP通信を基にしているよ。



# 郵便を用いたIP通信の手順

0. IPデータグラムが生成される
1. IPデータグラムを便箋などに印刷する
2. 便箋を封筒に入れ、必要な金額の切手を貼る
3. ポストに投函する
  
4. 家の郵便受けに届いた封筒から便箋を取り出す
5. 便箋に印刷されているデータグラムを読み取る
6. コンピュータはデータグラムを処理する

2025-11-17

郵便を用いた超低速IP通信システムの検討

└ 郵便を用いたIP通信

└ 郵便を用いたIP通信の手順

郵便によるIP通信はスライドに示される手段で行われるよ。

鳥類キャリアによる通信と比較してみると、鳥に関連している部分が郵便に置き換えられているだけだよ。

郵便を用いたIP通信の手順

0. IPデータグラムが生成される
1. IPデータグラムを便箋などに印刷する
2. 便箋を封筒に入れ、必要な金額の切手を貼る
3. ポストに投函する
  
4. 家の郵便受けに届いた封筒から便箋を取り出す
5. 便箋に印刷されているデータグラムを読み取る
6. コンピュータはデータグラムを処理する

# 鳥類キャリア vs. 郵便

## パケット喪失率を低減できる

郵便は鳥類よりもリンクの信頼性が高い

## QoS を確実にコントロールできる

速達やレターパックなどさまざまなサービスを選択できる  
特定記録や簡易書留などでパケットの到達を保証できる

## 通信の大容量化を実現できる

重さによる制限が緩和される ( $256\text{ mg} \rightarrow 50\text{ g}$ )

一度に送信可能なデータ量は封筒の大きさで選択できる

16進数だけでなく、Base64 や QR コードなどにも対応できる

2025-11-17

### 郵便を用いた超低速 IP 通信システムの検討

└ 郵便を用いた IP 通信

└ 鳥類キャリア vs. 郵便

#### 鳥類キャリア vs. 郵便

パケット喪失率を低減できる  
郵便は鳥類よりもリンクの信頼性が高い

QoS を確実にコントロールできる  
速達やレターパックなどさまざまなサービスを選択できる  
特定記録や簡易書留などでパケットの到達を保証できる

通信の大容量化を実現できる  
重さによる制限が緩和される ( $256\text{ mg} \rightarrow 50\text{ g}$ )  
一度に送信可能なデータ量は封筒の大きさで選択できる

16進数だけでなく、Base64 や QR コードなどにも対応できる

鳥類キャリアを用いた通信に比べた郵便を用いた通信の特徴を述べてみるよ。まず、パケット喪失率を低減できるようになるよ。郵便は鳥類キャリアよりもリンクの信頼性が高いからだよ。また、QoS を確実にコントロールできるようになるよ。速達などのオプションを利用すればパケットを早く届けられるし、特定記録などを利用すればパケットの到達を保証できるよ。さらに、通信の大容量化を実現できるよ。重さによる制限が約 200 倍改善されるので、より多くのデータを一度に送れるようになるよ。普通は紙でデータを送ることになると思うけれども、紙は鳥の脚に巻き付けられることかた解き放たれているので、この大きさは封筒の大きさに合わせて自由に選択できるよ。紙面が広がったことで、16進数だけでなく、Base64 や QR コードを印刷することもできるよ。

# システムの実装

2025-11-17

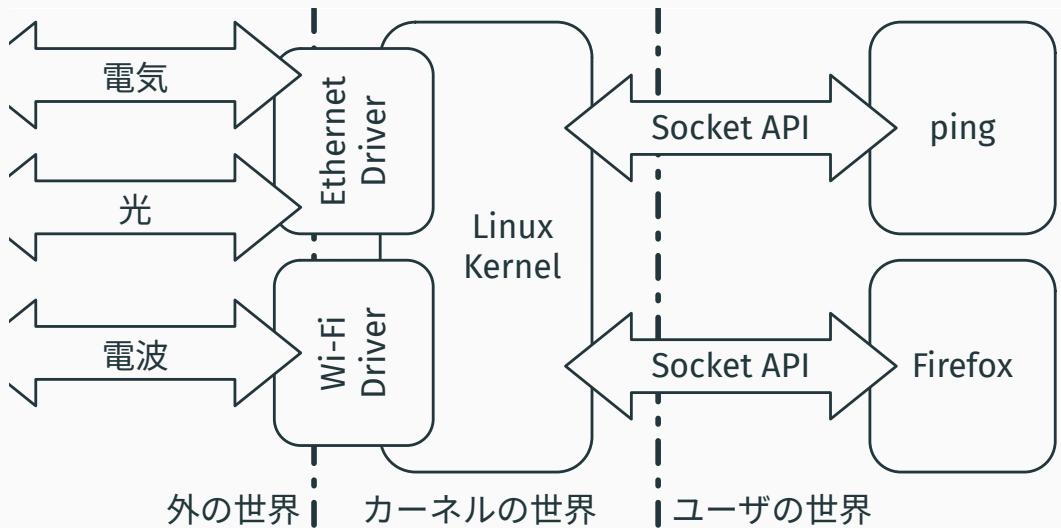
郵便を用いた超低速IP通信システムの検討

└ システムの実装

システムの実装

机上の空論ではつまらないので、これらについて実装を行ったよ。

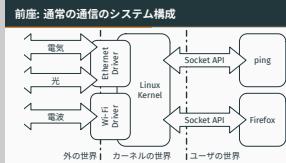
# 前座: 通常の通信のシステム構成



2025-11-17 郵便を用いた超低速IP通信システムの検討  
└ システムの実装

## └ 前座: 通常の通信のシステム構成

ここで、通常のデバイスによる通信に思いを馳せてみよう。通常、デバイスはカーネルが管理しているよ。当然、デバイスはカーネルと直接会話をするよ。そのため、より低レイヤ (OSI 参照モデルで L2 以下、リンク層) の通信は通常であればユーザの世界に出てこないよ。この方式によって愚直に実装しようと思えば、デバイスの開発やカーネルの修正など、膨大な作業が必要になってしまい、現実的ではないよ。



# ダイヤルアップ通信に思いを馳せる

ダイヤルアップ通信は本質的に PPP

PPP ではシリアルデバイスによってネットワーク接続を実現

シリアルデバイスはユーザの世界に露出している

Unix の世界ではシリアルデバイスをファイルと同様に扱える  
pppd のマネをすればユーザの世界から通信を吹き込める

pppd は TUN/TAP を用いることで仮想的な NIC を生やす

同様の手法で「普通のプログラム」でも通信を吹き込める

2025-11-17

郵便を用いた超低速 IP 通信システムの検討

└ システムの実装

└ ダイヤルアップ通信に思いを馳せる

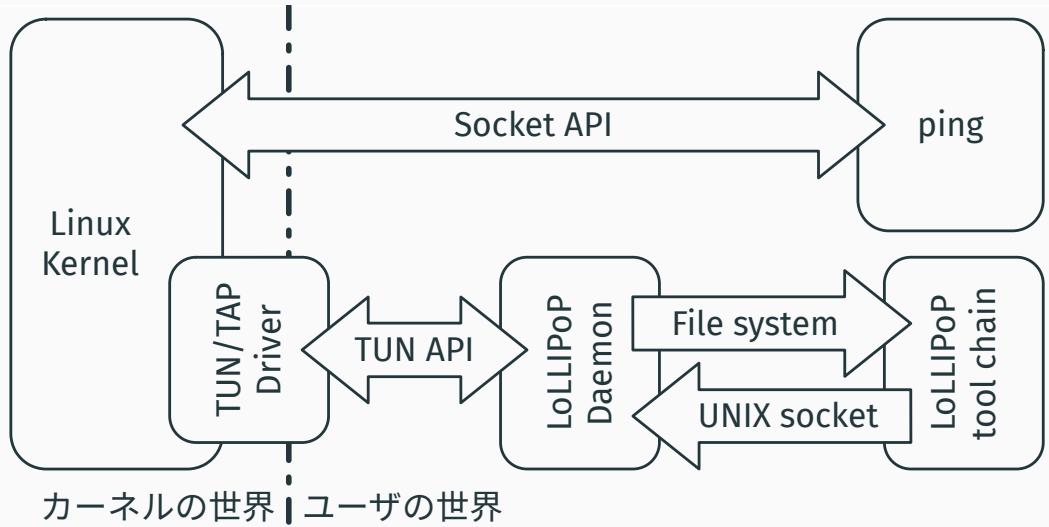
ここで、ダイヤルアップ通信に思いを馳せてみよう。ダイヤルアップ通信というのは本質的に PPP というプロトコルで通信しているよ。PPP ではシリアルデバイスによってネットワーク接続を実現しているよ。

ここで、よく考えてみると、シリアルデバイスというものはデバイスファイルとしてユーザの世界に露出していて、これはファイルのように読み書きできるよ。PPP を実現する pppd のマネをすればユーザの世界から通信を吹き込めるということなので、これについて調べてみると、TUN/TAP（トンネルデバイス）を利用して仮想的な NIC を生成しているようだよ。つまり、これを利用すると「普通のプログラム」からでも通信を実現できるようだよ。

ダイヤルアップ通信に思いを馳せる

ダイヤルアップ通信は本質的に PPP  
PPP ではシリアルデバイスによってネットワーク接続を実現  
シリアルデバイスはユーザの世界に露出している  
Unix の世界ではシリアルデバイスをファイルと同様に扱える  
pppd のマネをすればユーザの世界から通信を吹き込める  
pppd は TUN/TAP を用いることで仮想的な NIC を生やす  
同様の手法で「普通のプログラム」でも通信を吹き込める

# LoLLIPoP: Lots of Latency Letter IP over Post

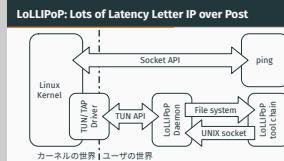


2025-11-17 郵便を用いた超低速IP通信システムの検討  
└ システムの実装

## └ LoLLIPoP: Lots of Latency Letter IP over Post

これを踏まえて郵便によるIP通信システムLoLLIPoPを開発したよ。LoLLIPoPでは、TUN/TAPを利用して仮想NICを用意し、さらにパケットの入出力のためのインターフェイスを用意するよ。システムはdaemonとCLI tool chainとに分かれている、daemonは入出力インターフェイスとNICとの間でデータを取り次ぐよ。CLI tool chainはユーザと入出力インターフェースとの間でデータを取り次ぐよ。ネットワークを利用して通信するプログラムからは通常のNICと同様に見えるから、pingやその他のプログラムをそのまま利用できるよ。

あと、パケットの先頭に疑似ヘッダを添えることで将来登場するより素晴らしいプロトコルにも対応できるようにしているよ。



# 通信の実験

---

2025-11-17

郵便を用いた超低速IP通信システムの検討

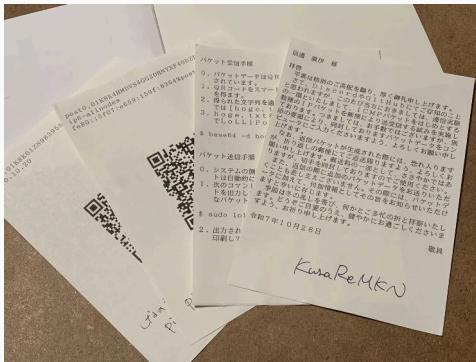
└ 通信の実験

通信の実験

開発したシステムをつかって、通信実験を行ってみたよ。

# 10月28日: パケットの生成、送信

各方面への ping パケットを QR コードの形式で印刷、送信



2025-11-17 郵便を用いた超低速 IP 通信システムの検討  
└ 通信の実験

└ 10月28日: パケットの生成、送信

10月28日、ping パケットが生成され、QR コードの形式で印刷、送信されたよ。このとき、パケットの形状としては郵便書簡が用いられたよ。郵便書簡ははがきと同額でありながら封筒のように使えるのでお得だよ。

10月28日: パケットの生成、送信  
各方面への ping パケットを QR コードの形式で印刷、送信



# 10月30日: パケットの着信報告

各方面からパケットの着信報告が寄せられる



yude ❤️ 346 2025/10/30, 20:15

受け取ったが、lollipopd から何も出力されなかった  
ログ  
tcpdump

```
$ sudo tcpdump -i post0 -vvv
tcpdump: listening on post0, link-type RAW (Raw IP), snapshot length
262144 bytes
20:12:20.142956 unknown_ip:2
```

18

2025-11-17 郵便を用いた超低速IP通信システムの検討

└ 通信の実験

└ 10月30日: パケットの着信報告

二日経って、パケットの着信報告が各方面から寄せられたよ。しかし、その報告は嬉しいものではなかったよ。

## 10月30日: パケットの着信報告

各方面からパケットの着信報告が寄せられる

yude ❤️ 346 2025/10/30, 20:15  
受け取ったが、lollipopd から何も出力されなかった  
ログ  
tcpdump

```
$ sudo tcpdump -i post0 -vvv
tcpdump: listening on post0, link-type RAW (Raw IP), snapshot length
262144 bytes
20:12:20.142956 unknown_ip:2
```

# 通信失敗の原因: パケット形式の不一致

送信側が古いバージョンのプログラムを利用していた  
パケットの先頭に疑似ヘッダのついていないものを送信した



19

2025-11-17 郵便を用いた超低速IP通信システムの検討

└ 通信の実験

└ 通信失敗の原因: パケット形式の不一致

アホだね～

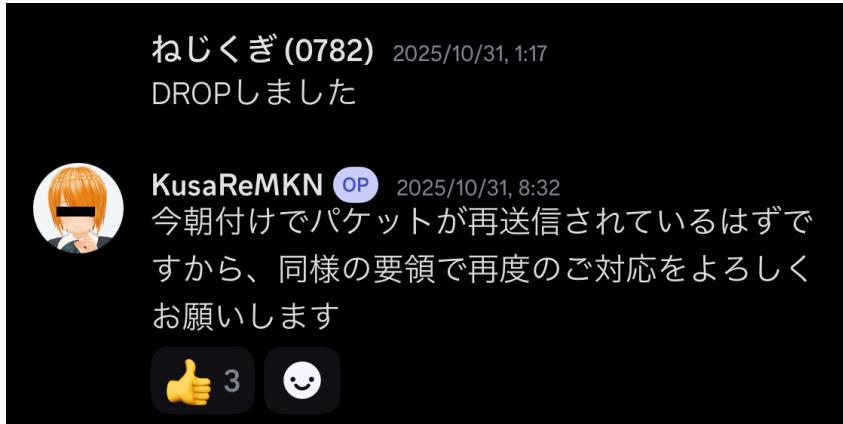
通信失敗の原因: パケット形式の不一致

送信側が古いバージョンのプログラムを利用していた  
パケットの先頭に疑似ヘッダのついてないものを送信した



# 10月31日: 速達でパケットを再送信

定型郵便（110円）に速達（300円）をつけると送料は410円  
→ 翌日以降、続々と着信報告があった



2025-11-17 郵便を用いた超低速IP通信システムの検討  
└ 通信の実験

└ 10月31日: 速達でパケットを再送信

原因に気付いてすぐパケットを再生成、再送信したよ。速達をつけたことで、  
翌日以降続々と着信報告があったよ。きちんと転送速度を制御できていることを確認できたよ。



# 11月4日: 初めての返信パケットを受信



21

2025-11-17 郵便を用いた超低速IP通信システムの検討  
└ 通信の実験

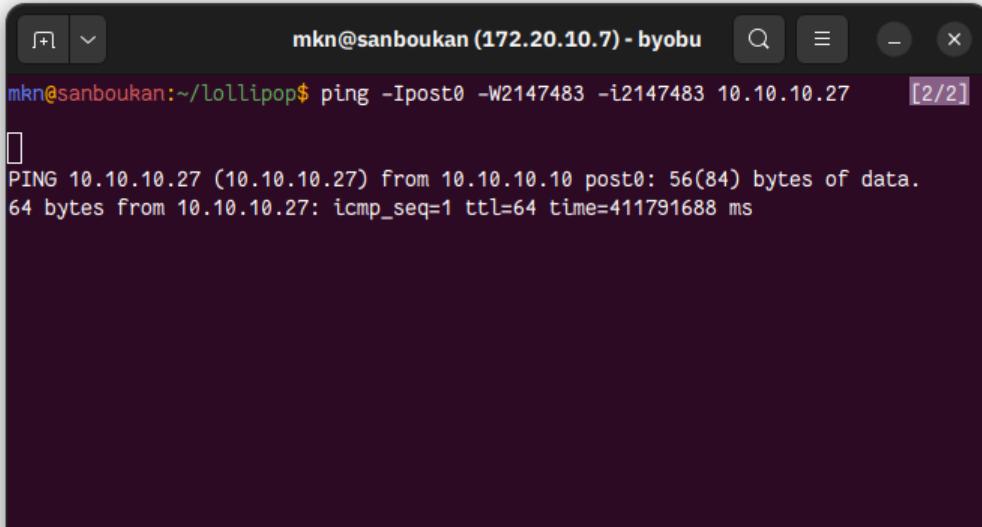
11月4日: 初めての返信パケットを受信



└ 11月4日: 初めての返信パケットを受信

しばらく待っていると、初めての返信パケットが到着したよ。QRコードに印刷されたパケットを読み込み、LoLLIPoPシステムに入力したところ、きちんと処理されたよ。そして……

# ping 実行結果



mkn@sanboukan (172.20.10.7) - byobu

```
mkn@sanboukan:~/lollipop$ ping -Ipost0 -W2147483 -i2147483 10.10.10.27 [2/2]
PING 10.10.10.27 (10.10.10.27) from 10.10.10.10 post0: 56(84) bytes of data.
64 bytes from 10.10.10.27: icmp_seq=1 ttl=64 time=411791688 ms
```

22

2025-11-17 郵便を用いた超低速IP通信システムの検討  
└ 通信の実験



ping 実行結果

```
mkn@sanboukan (172.20.10.7)-byobu
mkn@sanboukan:~/lollipop$ ping -Ipost0 -W2147483 -i2147483 10.10.10.27
PING 10.10.10.27 (10.10.10.27) from 10.10.10.10 post0: 56(84) bytes of data.
64 bytes from 10.10.10.27: icmp_seq=1 ttl=64 time=411791688 ms
```

└ ping 実行結果

ping コマンドの実行結果はこのようになったよ。

time = 411791688 ms

2025-11-17

郵便を用いた超低速IP通信システムの検討  
└通信の実験

time = 411791688 ms

4億1179万1688ミリ秒だよ

411 791 688 ms

||

**4.7661日**

2025-11-17

郵便を用いた超低速IP通信システムの検討  
└通信の実験

411 791 688 ms  
||  
**4.7661日**

これはなんと約4.7日に相当するよ。アホだね～。

# 往復通信の実験結果

全5件のうち3件が往復通信に成功

残りの2件は返信処理中

- 411 791 688 ms (郵便書簡)
- 581 116 715 ms (宅急便)
- 668 453 815 ms (はがき)

平均往復時間: 553 787 406 ms (**6.4095 日**)

2025-11-17

郵便を用いた超低速IP通信システムの検討

└ 通信の実験

└ 往復通信の実験結果

実験の結果をまとめるとこのようになるよ。全5件のうち3件が往復通信に成功したよ。残りの2件は現在返信処理中であるというウワサを聞いているよ。

平均の往復時間は5億5378万7406ミリ秒で、大体6日と半分だよ。

## 往復通信の実験結果

全5件のうち3件が往復通信に成功  
残りの2件は返信処理中

- 411 791 688 ms (郵便書簡)
- 581 116 715 ms (宅急便)
- 668 453 815 ms (はがき)

平均往復時間: 553 787 406 ms (**6.4095 日**)

# 実際に送信されてきた往復パケット



2025-11-17 郵便を用いた超低速IP通信システムの検討  
└ 通信の実験

└ 実際に送信されてきた往復パケット

実際に送信されてきた往復パケット



# まとめ・検討事項

2025-11-17 郵便を用いた超低速IP通信システムの検討  
└ まとめ・検討事項

まとめ・検討事項

まとめるよ～

# 郵便を用いた超低速IP通信システムの検討

鳥類キャリアを用いたIP通信に着想を得て  
郵便を用いたIP通信システムを検討した  
人間によってよりコントロールしやすい通信を実現  
Linux上で動作する実装を試作した  
通信実験ではパケットロス率40%（現時点；待てば届きそう）  
平均往復時間553 787 406 msを記録した

2025-11-17

郵便を用いた超低速IP通信システムの検討

└まとめ・検討事項

└郵便を用いた超低速IP通信システムの検討

郵便を用いた超低速IP通信システムの検討ということで、鳥類キャリアを用いたIP通信に着想を得て、郵便を用いたIP通信システムを検討したよ。キャリアとして郵便を用いることで、鳥類特有の制限を取り扱うことができ、人間によってよりコントロールしやすい通信を実現できるよ。

これについて、Linux上で動作するトンネルデバイスを用いた実装を試作したよ。通信実験ではパケットロス率（これは待てば届きそうだけど）40%を記録し、平均往復時間は5億5378万7406ミリ秒となったよ。

郵便を用いた超低速IP通信システムの検討

鳥類キャリアを用いたIP通信に着想を得て  
郵便を用いたIP通信システムを検討した  
人間によってよりコントロールしやすい通信を実現  
Linux上で動作する実装を試作した  
通信実験ではパケットロス率40%（現時点；待てば届きそう）  
平均往復時間553 787 406 msを記録した

# 今後の検討事項

PING以外の通信

TCPの3 way handshakeの実現

Wake on Letterなどのアプリケーション開発

ブロードキャスト・マルチキャストの問題

回覧板（トーカンリング）方式による実装

定期刊行物（第三種郵便）による実装

その他のリンクによる通信

電子メールによるe-LoLLIPoPの検討（MIME typeの策定）

2025-11-17

郵便を用いた超低速IP通信システムの検討

└まとめ・検討事項

└今後の検討事項

今後の検討事項

PING以外の通信  
TCPの3 way handshakeの実現  
Wake on Letterなどのアプリケーション開発  
ブロードキャスト・マルチキャストの問題  
回覧板（トーカンリング）方式による実装  
定期刊行物（第三種郵便）による実装  
その他のリンクによる通信  
電子メールによるe-LoLLIPoPの検討（MIME typeの策定）

あなたたちも郵便でIP通信をしなさい

# tkytel/lollipop

LoLLIPoP: Lots of Latency Letter IP over Post



3

Contributors

2

Issues

3

Stars

0

Forks



29

2025-11-17 郵便を用いた超低速IP通信システムの検討  
└まとめ・検討事項

└あなたたちも郵便でIP通信をしなさい

あなたたちも郵便でIP通信をしなさい

tkytel/lollipop

LoLLIPoP: Lots of Latency Letter IP over Post



3 Contributors

2 Issues

3 Stars

0 Forks

29

# おわりです

2025-11-17

郵便を用いた超低速IP通信システムの検討

おわりです

おわりだよ～。

# 参考資料

- [1] Braden, R.,  
**Requirements for Internet Hosts – Communication Layers,**  
RFC 1122, October 1989.
- [2] Waitzman, D.,  
**Standard for the transmission of IP datagrams on avian carriers,**  
RFC 1149, April 1990.
- [3] Waitzman, D.,  
**IP over Avian Carriers with Quality of Service,**  
RFC 2549, April 1999.
- [4] Carpenter B., Hinden R.,  
**Adaptation of RFC 1149 for IPv6,**  
RFC 6214, April 2011.

## 郵便を用いた超低速IP通信システムの検討

2025-11-17

参考資料

### 参考資料

- [1] Braden, R.,  
**Requirements for Internet Hosts – Communication Layers,**  
RFC 1122, October 1989.
- [2] Waitzman, D.,  
**Standard for the transmission of IP datagrams on avian carriers,**  
RFC 1149, April 1990.
- [3] Waitzman, D.,  
**IP over Avian Carriers with Quality of Service,**  
RFC 2549, April 1999.
- [4] Carpenter B., Hinden R.,  
**Adaptation of RFC 1149 for IPv6,**  
RFC 6214, April 2011.

# このスライドについて

Written in November 2025.

Permanent ID of this document: 2976cf5d5f923407.

Copyright © 2025 KusaReMKN.

特記無き場合、プログラムやソースコードは MIT License で、  
それ以外のコンテンツは CC-BY 4.0 で利用可能です。  
一部の画像には別のライセンスが適用されるかもしれません。

2025-11-17

郵便を用いた超低速 IP 通信システムの検討

← このスライドについて

このスライドについて

Written in November 2025.

Permanent ID of this document: 2976cf5d5f923407.

Copyright © 2025 KusaReMKN.

特記無き場合、プログラムやソースコードは MIT License で、

それ以外のコンテンツは CC-BY 4.0 で利用可能です。

一部の画像には別のライセンスが適用されるかもしれません。