情報通信実験（課題④　Wireshark を用いた TCP 通信のパケット内容の観測と解析）

実験レポート

出席番号：　16　名前：　　小形孔明　　実験日：　2025 年　10 月　16 日

１．課題目的

本課題の目的は、Wireshark を用いて実通信における TCP のコネクション管理（三者ハンドシェイク／四者揺手・RST 等）とデータ転送（シーケンス番号・ACK・ウィンドウ制御・再送）の挙動を可視化し、サーバ／クライアント間のイベント因果（入力操作→パケット→アプリ表示）を時系列で対応づけて理解することである。自作プログラムや telnet を対象に、SYN／ACK／FIN の役割、PSH/ACK の発生、MSS・遅延 ACK・再送の発生条件などを具体例で確認し、文字列送受（CRLF・エンコーディング）の実体もパケットレベルで把握することを狙いとする。

２．課題内容

サーバ（SimpleServer）とクライアント（telnet もしくは SimpleClient 系）を接続し、Wireshark で通信をキャプチャする。

コネクション管理の観測（④-1）：コネクションを確立させる際に「(サーバからの)確立要求に対する確認応答」を行うパケット受信を観測する。

データ転送の観測（④-2）：送受信メッセージの区切り（CRLF）と PSH/ACK の対応、セグメント分割など主要フィールドを読み解く。

３．Wireshark での観測時の重要箇所の解説

本実験で観測した 2 つのパケットは、TCP 通信の制御およびデータ転送を示すものである。

最初のパケットは ACK フラグのみを持つ制御パケットであり、送信元（クライアント）が受信確認を行っている。IP ヘッダの全長は 40 バイトで、TCP データ部を持たない。すなわち、通信の信頼性を保証するための確認応答パケットである。このパケットは、TCP 通信におけるコネクション確立処理の最終段階で送信される ACK パケットである。

Wireshark の観測結果では、フラグが ACK のみ（0x10）となっており、データ長（Len）は 0 であった。これは、アプリケーションデータを含まない純粋な制御用パケットであることを示している。IP ヘッダの全長は 40 バイト（0x0028）であり、内訳は IP ヘッダ 20 バイト、TCP ヘッダ 20 バイト、データ部 0 バイトである。

TCP ヘッダのシーケンス番号および確認応答番号（Seq=1, Ack=1）は、クライアントとサーバ間の同期が完了したことを意味し、これにより三者間ハンドシェイク（Three-way handshake）が成立していることが確認できる。このパケットには PSH や SYN といったデータ転送や接続要求のフラグが含まれていないため、通信の確立確認のみに用いられていることが分かる。

2 つ目のパケットは PSH および ACK フラグを持ち、送信元（サーバ）から宛先（クライアント）への実データ送信を示している。Wireshark の観測では、フラグが PSH および ACK（0x18）となっており、IP ヘッダの全長は 60 バイト（0x003C)、そのうち TCP データ長は 20 バイトであった。データ部にはアプリケーションから送信された文字列「ECHO: ogata(制御文字)¥r¥n」が格納されており、これはサーバからクライアントへの ECHO 応答メッセージに対応している。

また、TCP ヘッダのシーケンス番号および確認応答番号は、前回の ACK パケットの値から進んでおり、この変化によりデータ転送が実際に行われたことが確認できる。

このように、2 つ目のパケットは、通信が確立した後のデータ送信を担うパケットであり、IP ヘッダの全長や TCP デー

タ長の増加、PSH フラグの有効化、ASCII データ部の存在などから、アプリケーション層の通信が正常に行われていることが確認できる。

2 つのパケットを比較すると、前者は TCP における制御（確認応答）を担い、後者はアプリケーションデータの転送を担っている。これにより、TCP 通信が信頼性を確保するために制御パケットとデータパケットを明確に区別していることが確認できた。

４．結果

課題④－1 の観測結果は資料 1 に，課題④－2 の観測結果は資料 2 として添付。

５．感想・考察

転記シートを通じて、教科書的に理解していた「確立→データ転送→切断」の流れが、実パケットの順序・番号・長さとして対応づけられた点が有益であった。特に、確立時は SYN／SYN,ACK／ACK の三段階が連番のイベントとして整理され、切断時は FIN と ACK が往復で 4 パケット前後になることが、時刻と方向付きで明瞭になったことが印象的である。また、データ転送ではテキスト送受の 1 行＝CRLF 終端がペイロード長と一致して現れる様子が読み取れ、相対シーケンス番号（Seq）と ACK 番号の増分が文字数＋改行と整合している点が納得感を生んだ。操作側の体感（Enter 押下）と PSH,ACK を含む転送の発生が近接して並ぶ場面もあり、アプリ操作と TCP 挙動が時間軸で自然につながっていることを確認できた。

まず、1 本目の SYN,ACK を見て、ACK 番号が「クライアント SYN の Seq+1」になっていることを確認した。SYN はシーケンス番号を 1 消費するので、この+1 が成り立っていれば、確立の折り返しが正しく来ていると判断できる。同じ SYN,ACK の Seq はサーバ側の初期番号（ISN）を示す。ここまでで、三者ハンドシェイクの 2 段目まで届いていることを、パケットの数値だけで言える。

2 本目はクライアントからの PSH,ACK 付きデータで、Len>0 である。ACK が「サーバ ISN+1」を示していれば、最終 ACK はすでに終わっており、接続は ESTABLISHED だと分かる。つまり、確立が終わってすぐにアプリのデータ送信が始まっている。この 2 本目の Len は、そのまま「送ったバイト数」を表す。ペイロードの末尾に CRLF（0x0D,0x0A）があれば、本文バイト数+2 が Len になる。戻り ACK はここには無いが、TCP の基本どおりなら、次に返る ACK は Len ぶん前進するはずである。ここまで、行終端（CRLF）を含めて、送信量と番号の関係を 1 方向だけでも説明できる。

ヘッダを見ると、SYN,ACK はふつう Len=0 で小さく、IP+TCP の最小長で並ぶ。DF=1 でフラグメントオフセット=0 なら、断片化は起きていない。2 本目のデータも小さければ、この時点では分割や再構成の問題は無いと判断できる。受信ウィンドウ（Window 広告値）も見て、極端に小さい値やゼロではないことを確認する。少なくともこの瞬間は、受信側のバッファが詰まっておらず、フロー制御で止まっていないと言える。

この 2 本だけでは、再送があるか、遅延 ACK があるか、といった動きは判断できない。また、全角文字混在によるバイト長の増加と ACK 前進量の実測一致も、戻り ACK がないのでここでは確認できない。よって、結論は次の 3 点に限定する。

確立の妥当性：SYN,ACK の+1 関係から、折り返しが正しい。

データ送信の開始：PSH,ACK・Len>0 の出現で、確立直後に行単位送信が始まっている。CRLF があれば、次の ACK 前進は Len＝本文+2 になる見込み。

輸送条件の健全性（この瞬間）：フラグの並びも特に異常はない。

以上のように、2 パケットだけでも「確立が正しい」「行単位の送信に入った」「当面の妨げは見えない」という基本は、Seq/ACK・Len・フラグ・Δt の読み取りで十分示せる。

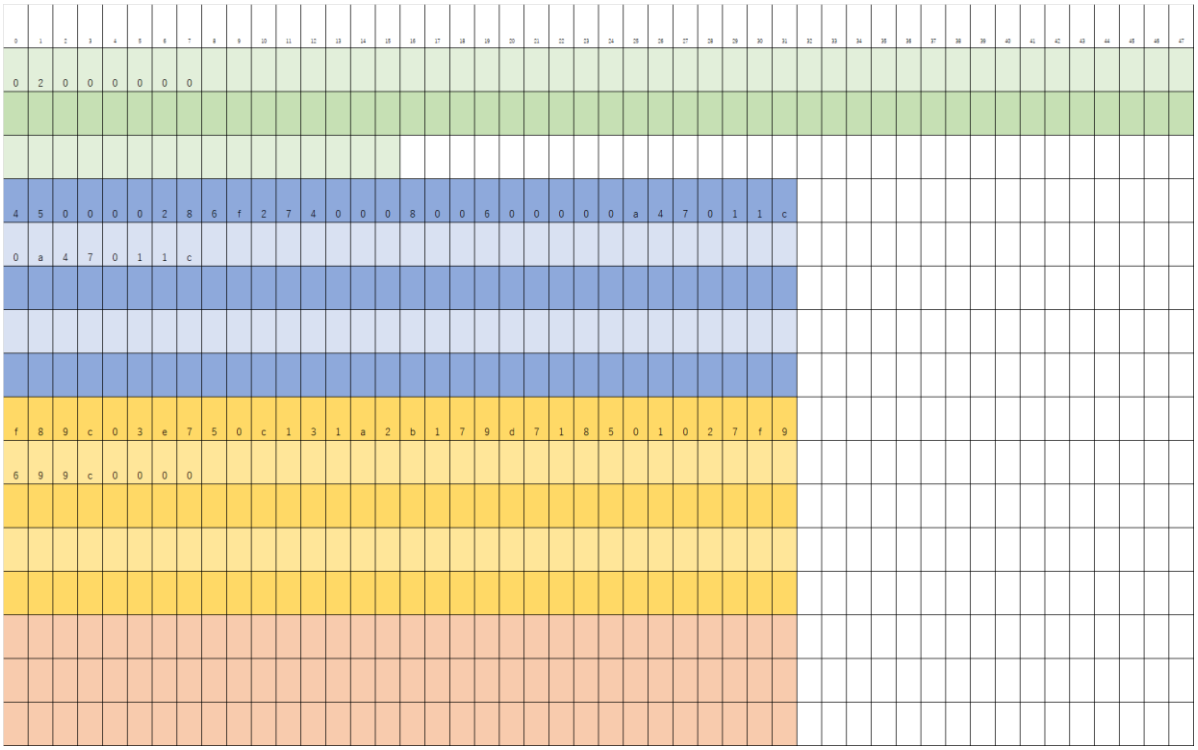
資料１

課題④－１：　コネクションの管理に伴うパケットの観測と解析

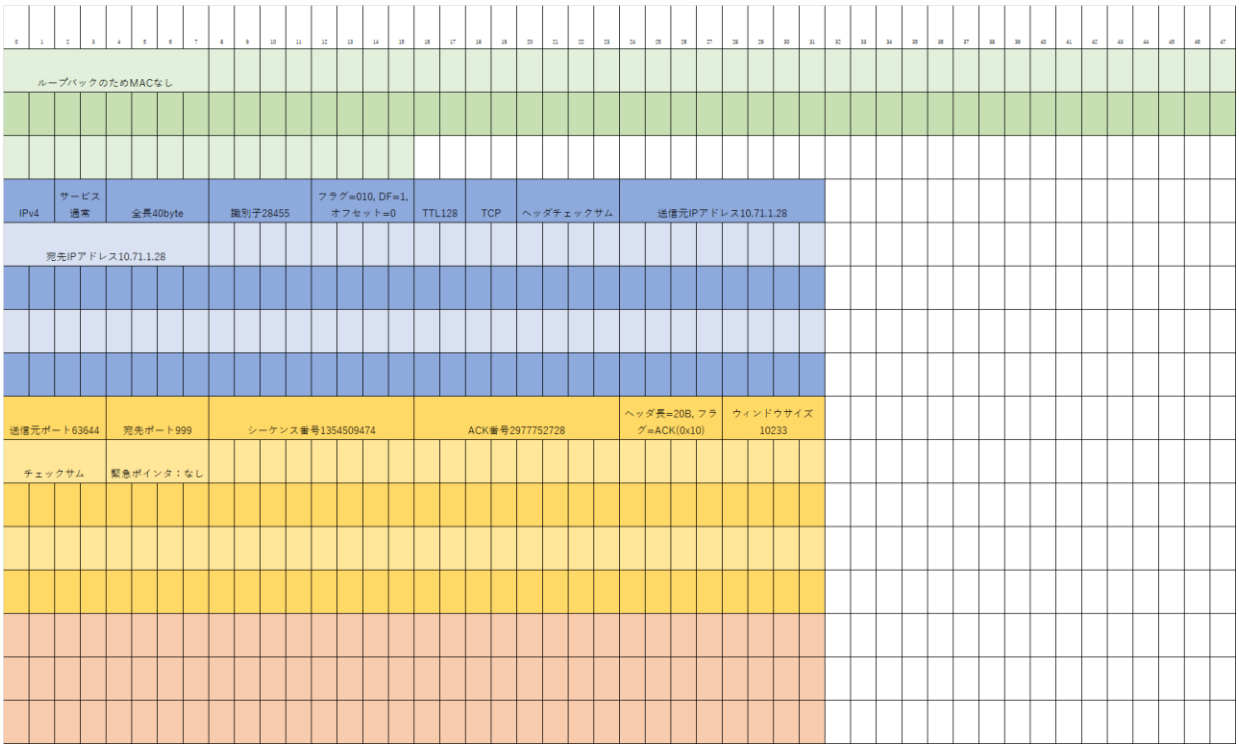
調査対象のパケット：コネクションを確立させる際に「(サーバからの)確立要求に対する確認応答」を行うパケット受信

※調査対象のパケットは「出席番号を７で割った余りの数字＋１」のパケットです。

【観測】全て 16 進数による表記



【解析】2 進数、10 進数、16 進数、アルファベットを適宜使用による表記

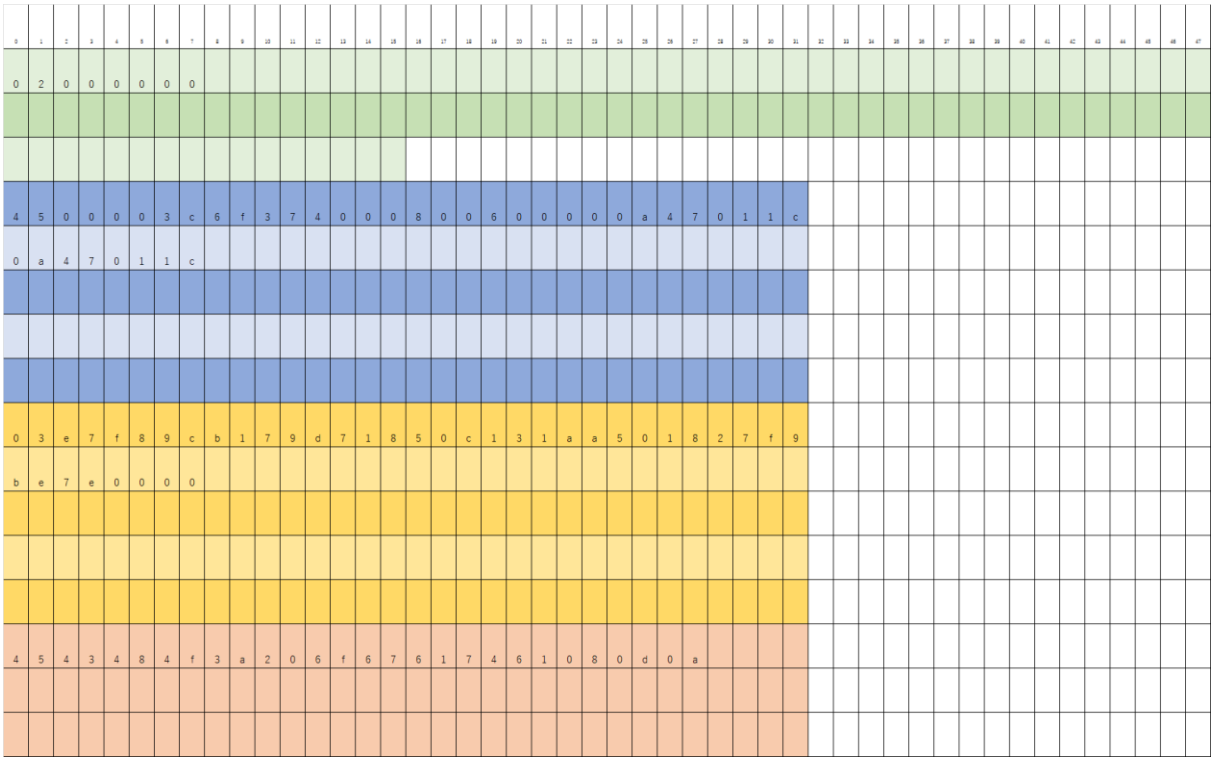


資料２

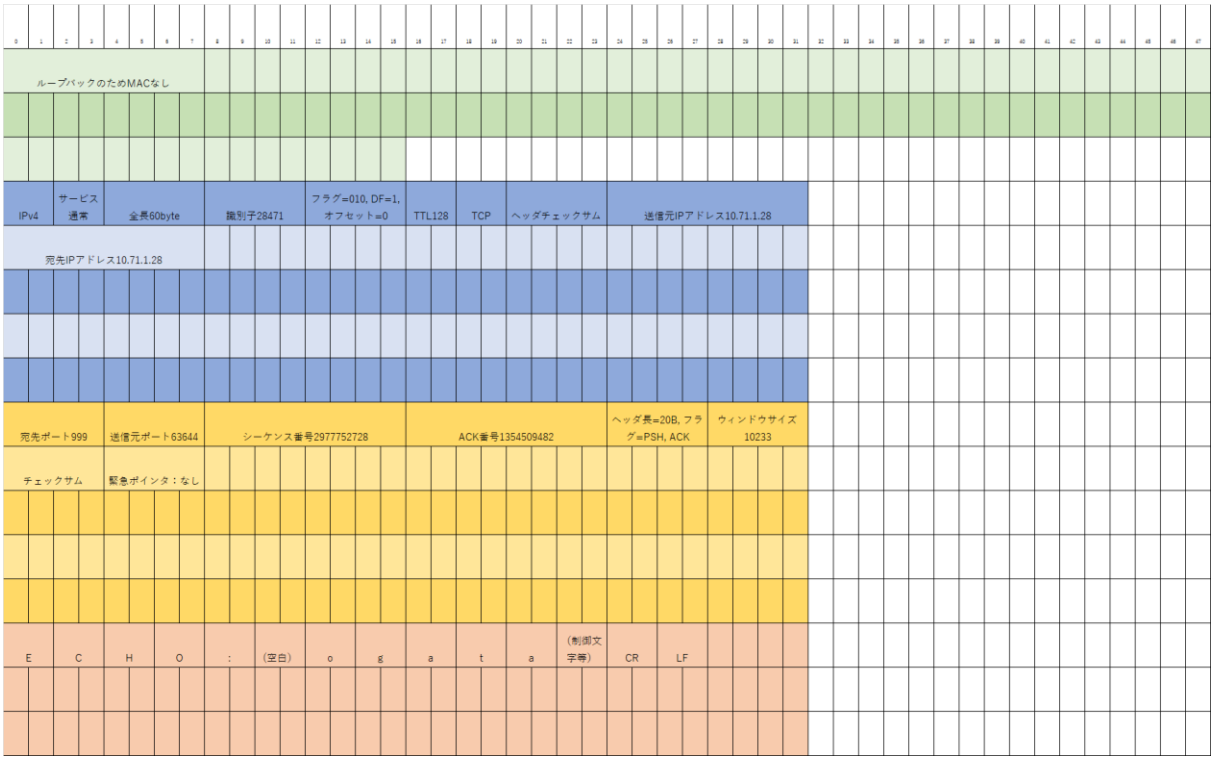
課題④－２：　クライアントから送信された文字が格納されたパケットの観測と解析

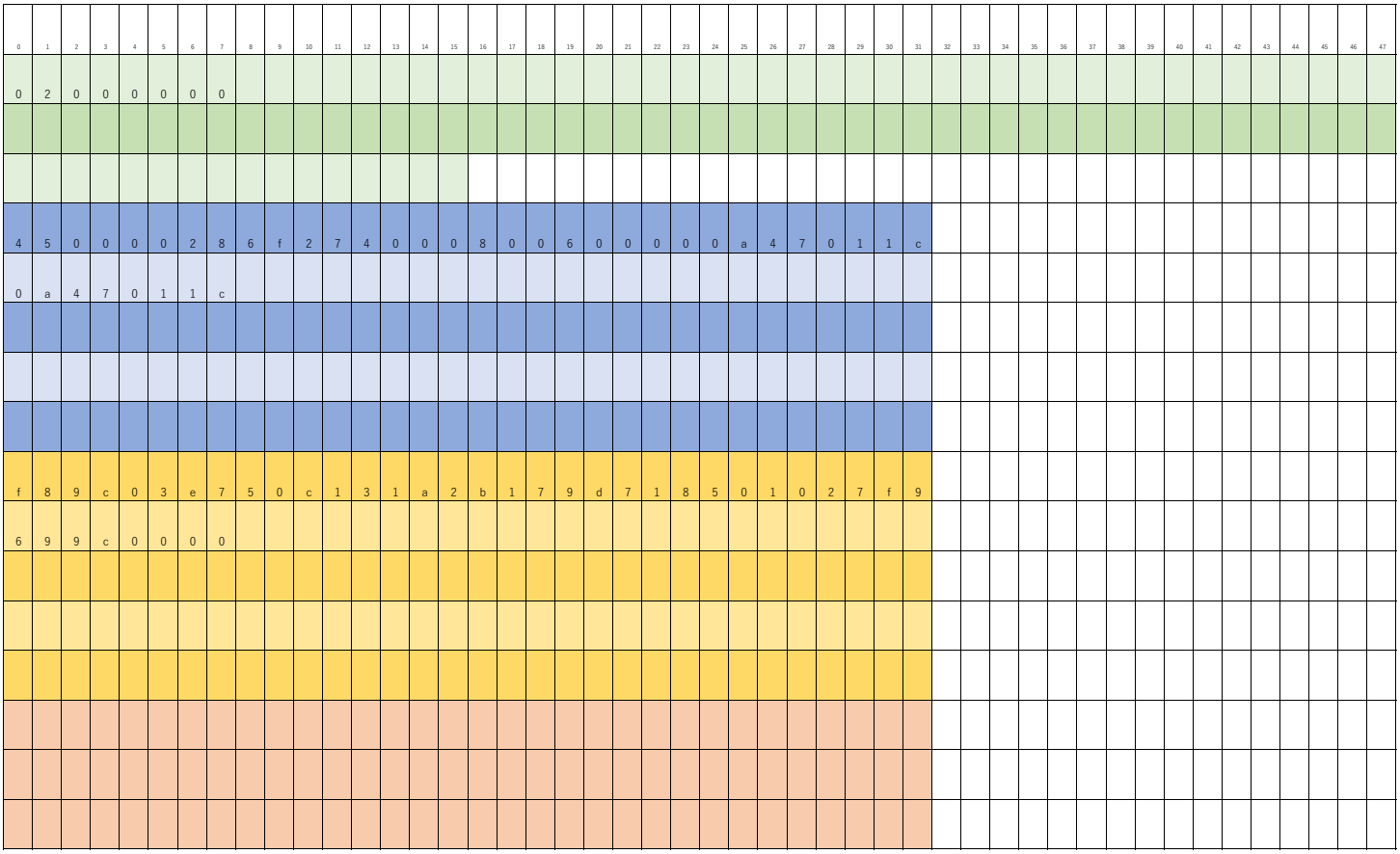
送信文字列：ogata

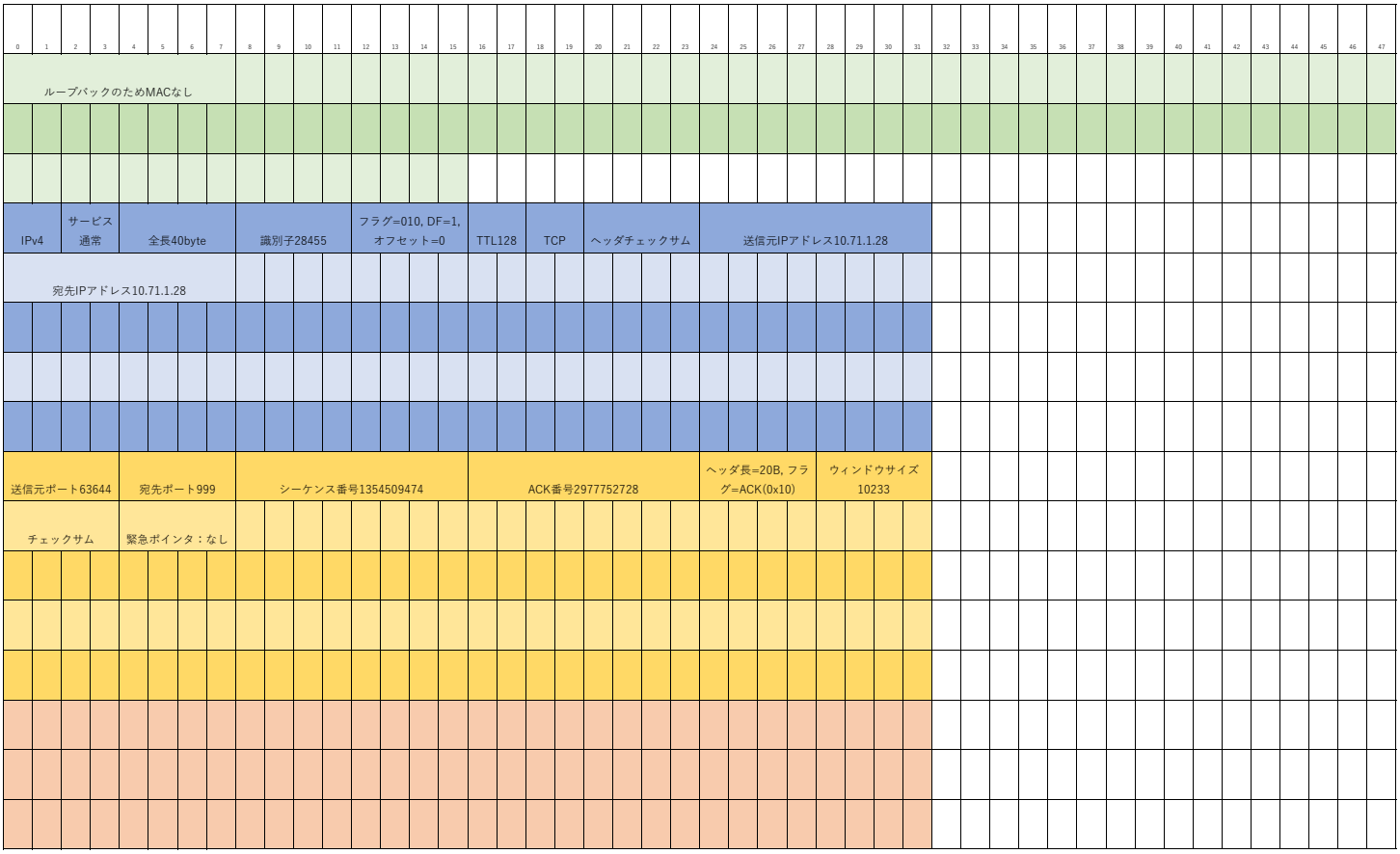
【観測】全て 16 進数による表記



【解析】2 進数、10 進数、16 進数、文字を適宜使用による表記




0 2 0 0 0 0 0 0
4 5 0 0 0 0 2 8 6 f 2 7 4 0 0 0 8 0 0 6 0 0 0 0 0 a 4 7 0 1 1 c
0 a 4 7 0 1 1 c
f 8 9 c 0 3 e 7 5 0 c 1 3 1 a 2 b 1 7 9 d 7 1 8 5 0 1 0 2 7 f 9
6 9 9 c 0 0 0 0


ループバックのためMACなし
IPv4
サービス 通常
全長40byte
識別子28455
フラグ=010, DF=1, オフセット=0
TTL128
TCP
ヘッダチェックサム
送信元IPアドレス10.71.1.28
宛先IPアドレス10.71.1.28
送信元ポート63644
宛先ポート999
シーケンス番号1354509474
ACK番号2977752728
ヘッダ長=20B, フラグ=ACK(0x10)
ウィンドウサイズ 10233
チェックサム
緊急ポインタ：なし

0	2	0	0	0	0	0	0																																								
4	5	0	0	0	0	3	c	6	f	3	7	4	0	0	0	8	0	0	6	0	0	0	0	0	a	4	7	0	1	1	c																
0	a	4	7	0	1	1	c																																								
0	3	e	7	f	8	9	c	b	1	7	9	d	7	1	8	5	0	c	1	3	1	a	a	5	0	1	8	2	7	f	9																
b	e	7	e	0	0	0	0																																								
4	5	4	3	4	8	4	f	3	a	2	0	6	f	6	7	6	1	7	4	6	1	0	8	0	d	0	a																				

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
ループバックのためMACなし																																															
IPv4		サービス 通常		全長60byte				識別子28471				フラグ=010, DF=1, オフセット=0				TTL128		TCP		ヘッダチェックサム				送信元IPアドレス10.71.1.28																							
宛先IPアドレス10.71.1.28																																															
宛先ポート999				送信元ポート63644				シーケンス番号2977752728								ACK番号1354509482								ヘッダ長=20B, フラグ=PSH, ACK				ウィンドウサイズ 10233																	
チェックサム				緊急ポインタ：なし																																											
E		C		H		O		:		(空白)		o		g		a		t		a		(制御文字等)		CR		LF																					