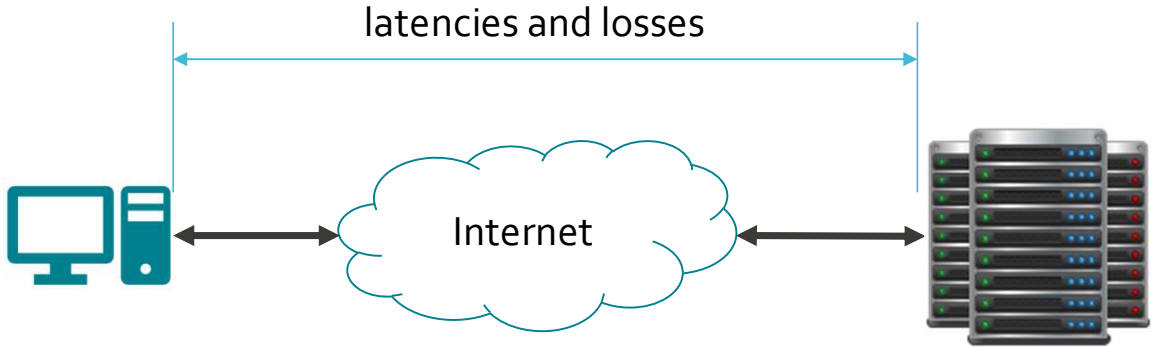
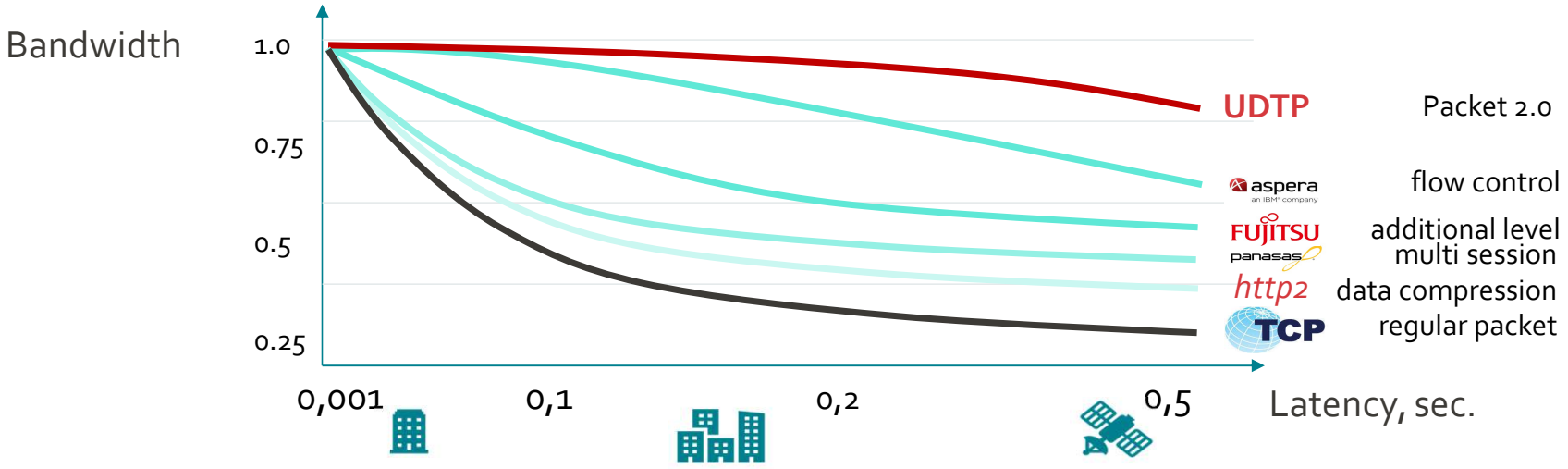
Transport protocol - UDTP

Sergey Valov
serg.valov@gmail.com



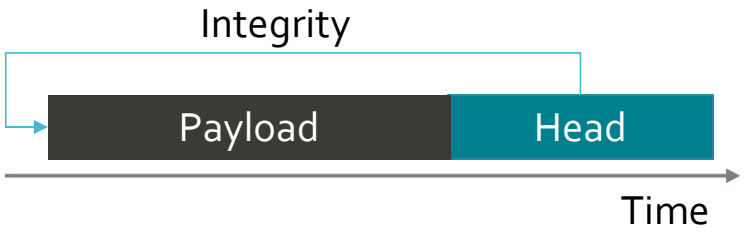
The greater the latency and the higher the losses, the lower the transmission speed!



Fundamental issue - Packet

The main advantage is asynchronous transmission.

The fundamental flaw is indivisibility.



Problems:

- ✓ Indefinite delay;
- ✓ Jitter delays;
- ✓ Traffic interference;
- ✓ Invalid traffic type information;
- ✓ Add-ons for traffic management;
- ✓ The need to harmonize management policies;

Up/Down Transport Protocol - UDTP aims to solve these problems

UDTP operating mode:

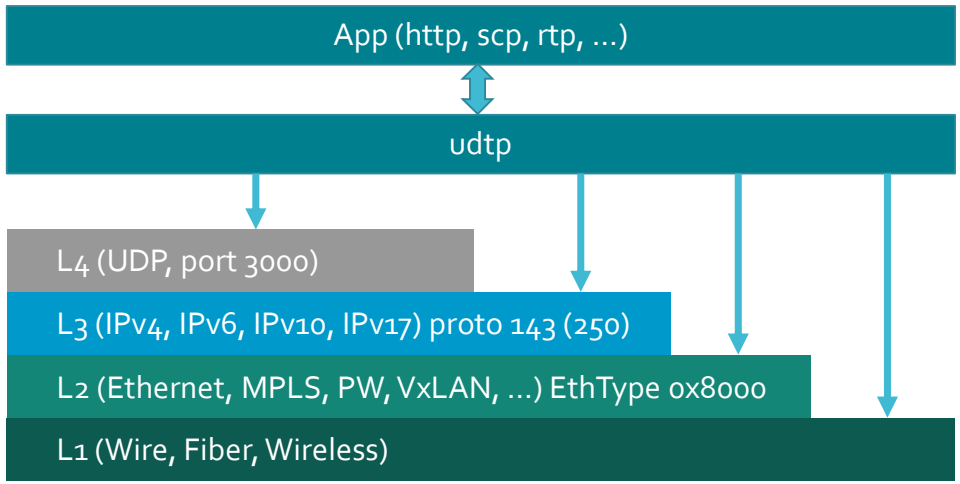
- ✓ udtp /udp/ip/eth, over NAT and Firewall;
- ✓ udtp /ip/eth, over Internet ;
- ✓ udtp /eth, over LAN, MPLS, VxLAN, PW;
- ✓ udtp /phy, over Eth, LTE, Sat.

UDTP in a new way:

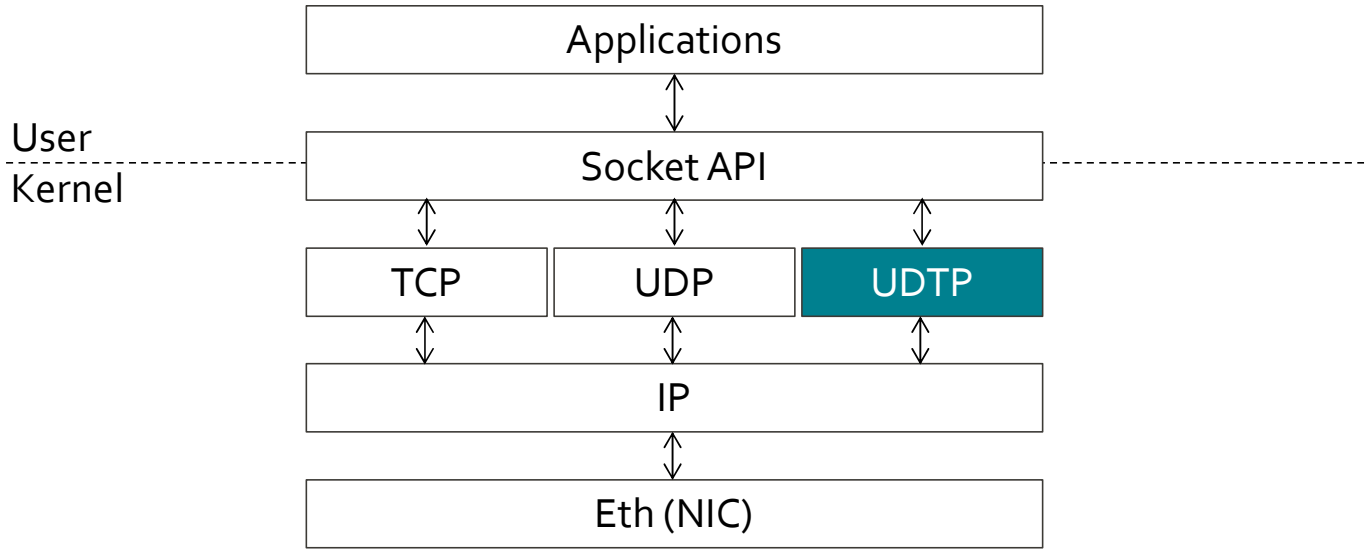
- ✓ serializing packages (Packet 2.0);
- ✓ maintains traffic integrity;
- ✓ manages transmission bitrate;
- ✓ establishes connection.

UDTP does not create:

- ✓ compression;
- ✓ duplication;
- ✓ break down into small parts;
- ✓ multiple patch.



Protocol stack



API (udtp 1.0)

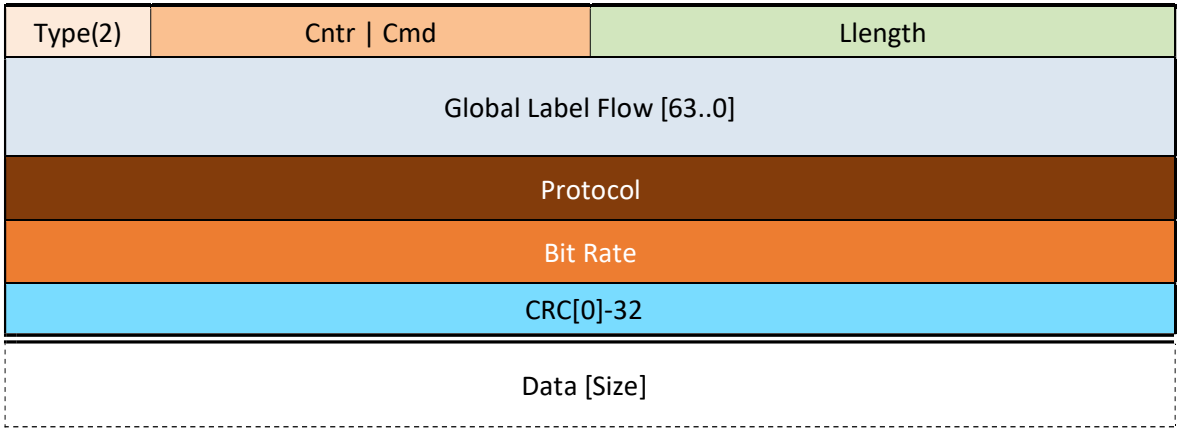
```
#define udtp_proto 250
…
if((sock=socket(AF_INET,SOCK_STREAM,udtp_proto))<0)
 perror(«udtp_err: socket");
```

Protocol number

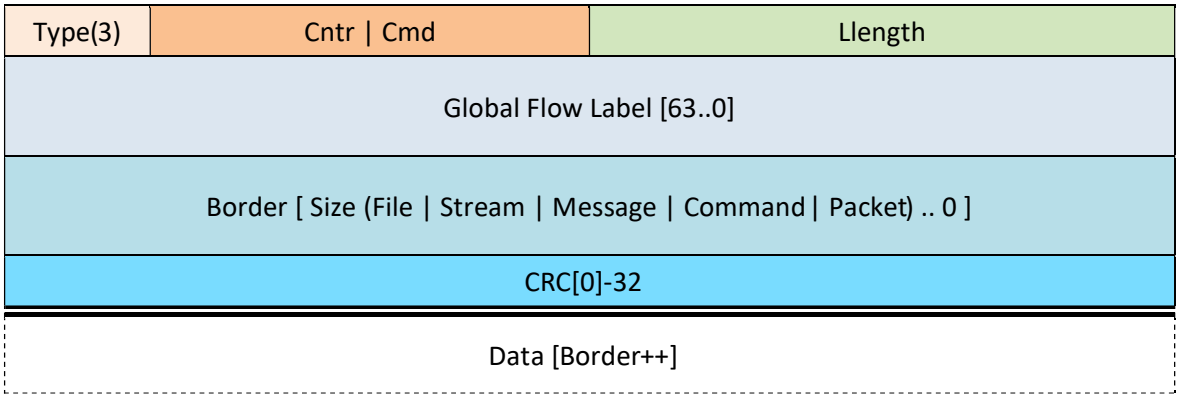
Bitrate

```
…
if(setsockopt(sock,udtp_proto,5,&bitrate,sizeof(unsigned long)))
perror(«udtp_err: setsockopt error bitrate");
…
```

Type = 2 – Connection



Type = 3 – data transfer



UDTP transport protocol with guaranteed data delivery between devices

1. udtp on demand of a network application to each Traffic
   - ✓ allocates bandwidth;
   - ✓ guarantees latency level;
   - ✓ normalizes latency fluctuation
2. udtp excludes interference between traffic
3. udtp combines under a single channel "real time" and noncritical to latencies traffic
4. udtp excludes traffic management add-ons

Transport protocol **udtp** provides:

1. Data transfer quality through:
   - ✓ Allocating a given transfer rate;
   - ✓ Support for the quality of traffic transfer with arbitrary latencies and losses;
   - ✓ Lack of interference between traffics.
2. Security of information transferred is ensured by:
   - ✓ Establishing connection between authenticated participants only;
   - ✓ Maintaining session history and connections between participants;
3. Profitability of data transfer is achieved by:
   - ✓ Reduced management costs of the Internet Provider;
   - ✓ Reduced costs for programmer(s) training;

Comparative characteristics of transport protocols

	TCP	SCTP	FASP/udt	QUIC	UDTP
Handshake	3-way	4-way	4-way	3-way|1-way	2x2-way
Connection			based		
Connection ID	IP/port	IP/port	IP/port	ID (32..144)	GFL-64
Data transmission	Byte oriented	Byte & Message oriented	Byte oriented	String oriented	Byte oriented
Acknowledgment Number	Byte-oriented	Chunk-oriented	Byte-oriented	String oriented	Byte-oriented
Multiple streams	-	Multi Stream	n/a	Multi Streams	Multi Streams
Stream ID	-	32-bit	31-bit	62-bit	64-bits
Congestion Control	Loss, Delay, (AIDM)	Loss, Delay	Delay	Delay	Bitrate
Data flow control	Window-based	Window-based	Buffer-based	Credit-based	Credit-based, Predict-based
Prioritization	In packet	In packet	In packet	Application	Application
Loss Detection	Checksum	Crc32	Checksum	Crc32	Crc32
Transport security	(ext. TLS)	(ext. TLS)	(ext. TLS)	int	int
Security Considerations	no	no	no	yes	yes
Packet chain	-	-	-	-	yes
DoS	flood SYN	No	flood/UDP	No	No
Header	20	12+16 (Data Chunk)	8+16	32 (variable-length)	24

Transport protocol (UDTP ver. 1.0)

Developed:

1. **Driver**:
   - ✓ Linux kernel 3.0 ÷ 5.4 (udtp.ko)
   - ✓ Distributives:
     - ➢ Ubuntu 14, 16, 18, 20;
     - ➢ OpenSUSE 15;
     - ➢ Debian 9, 10;
     - ➢ CentOS 7,8;
     - ➢ Astra Linux SE 1.3, 1.4, 1.5, 1.6, CE 2.12;

2. Supported architecures:

- ✓ x86(i386), x86_64(amd64)
- ✓ Arm32(AArch32), Arm64(AArch64)
- ✓ Elbrus

3. Applications:

- ✓ udtp_sendf – file transfer analog ftp, scp
- ✓ udtp_proxy – proxy server for data transmission between applications;
- ✓ udtp_tunnel – ethernet tunnel for local networks;
- ✓ udtp_test – test program for network experiments analog of iperf3.

4. Service

- ✓ Network ID

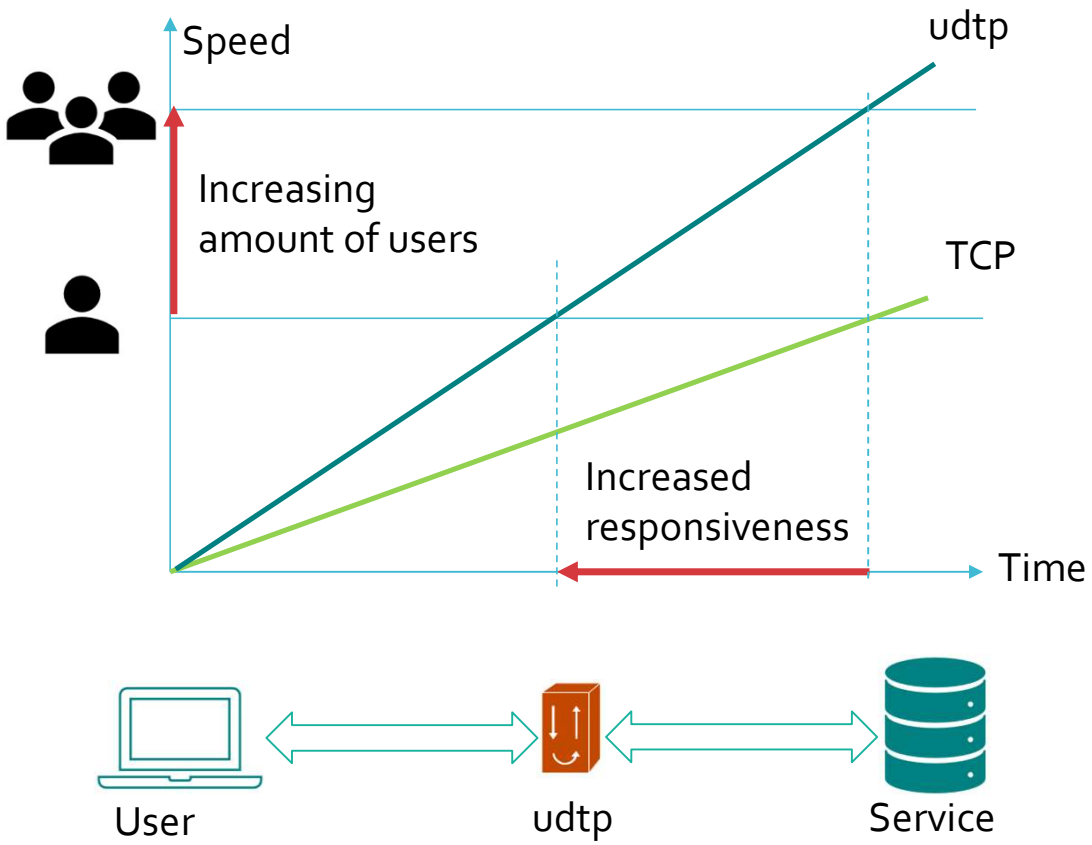
Services acceleration

Maintenance:

1. At the speed of interface
2. Guaranteed bandwidth

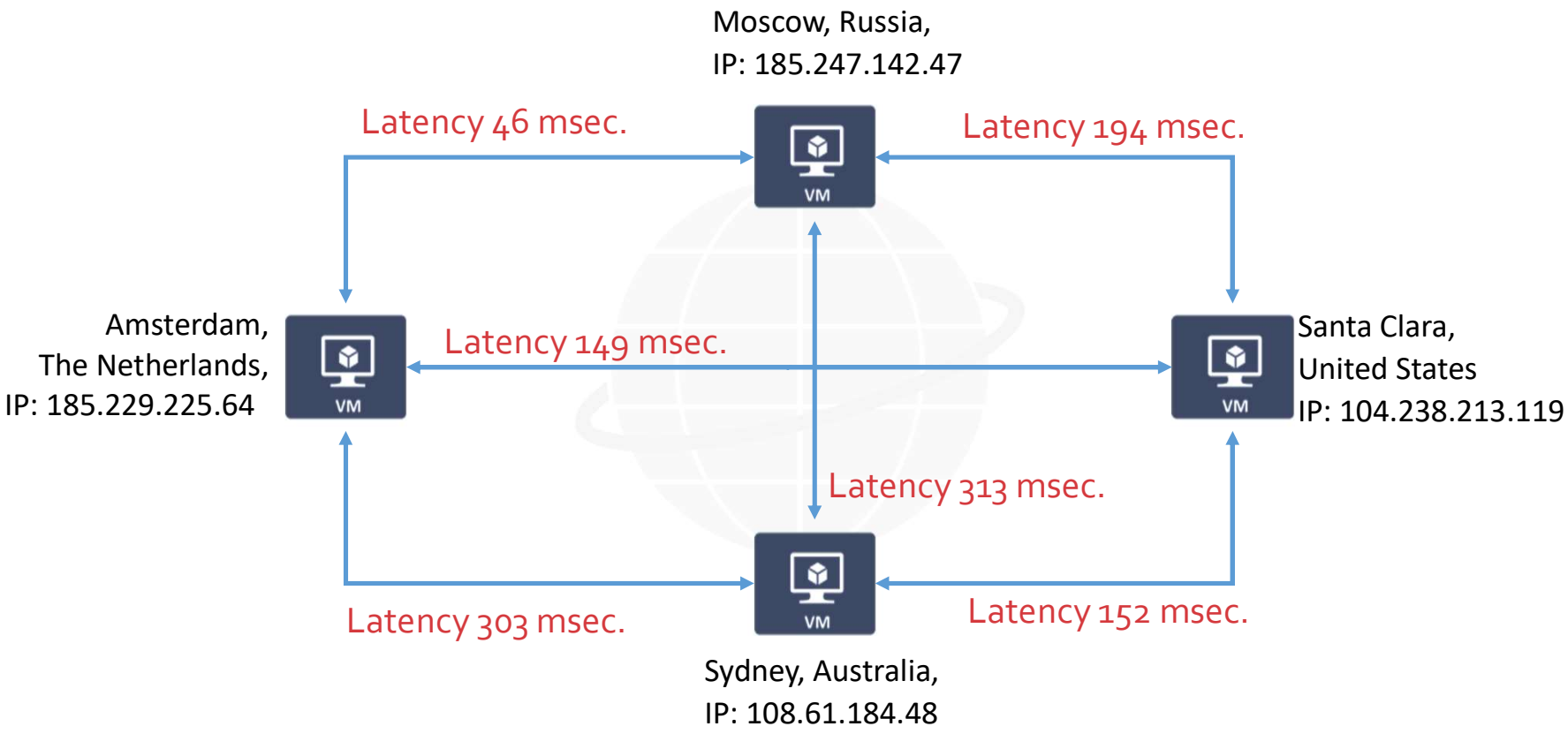
Advantages:

1. Increased responsiveness
2. Increased operation speed
3. Cyber attack resistance



Testing

Pilot zone scheme - Test 1

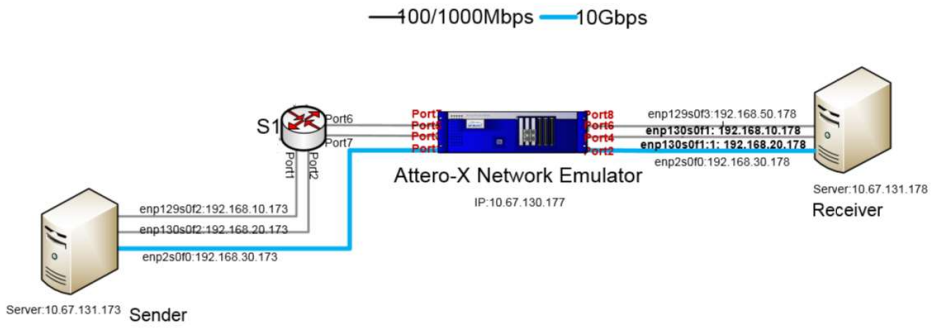


UDTP

Test results

Direction	Transmission Time	Bitrate, Mbytes/s	Average value	Transmission Time	Bitrate, Mbytes/s	Average value	Increase +X
	Standard transmission (scp/tcp/ip)			New transmission (sendf /udtp/ip)			
Moscow - Santa Clara (latency 194 ms)	00:25 00:26 00:27 00:26 00:26	10.2 9.8 9.4 9.8 9.8	9.8 MB/s	00:01.959 00:02.779 00:02.472 00:02.261 00:02.143	130.1 91.7 103.0 112.7 118.9	111,28 MB/s	+11X
Moscow – Amsterdam (latency 46 ms)	00:07 00:07 00:07 00:07 00:07	36.4 36.4 36.4 36.4 36.4	36.4 MB/s	00:01.964 00:02.212 00:01.845 00:02.308 00:03.016	129.7 115.2 138.1 110.3 84.4	115,54 MB/s	+3X
Moscow – Sydney (latency 313 ms)	02:55 02:07 01:49 02:00	1.5 2.0 2.3 2.1	1,975 MB/s	00:02.738 00:02.509 00:02.365 00:02.522 00:02.691	93.1 101.6 107.7 101.0 94.7	99,62 MB/s	+50X

Test results



Network Scenarios		Bandwidth usage		
Latency	Packet loss rate	udtp	TCP (bbr for congestion control)	TCP (congestion control is cubic)
Delay 22 ms Max.Delay 200 ms Min.Delay 2 ms	0%	**92.35%**	0.27%	22.61%
	0.1%	92.30%	0.28%	2.35%
	1%	90.46%	0.27%	0.58%
	10%	72.70%	0.22%	0.13%
Delay 44 ms Max.Delay 80 ms MinDelay 40 ms	0%	**92.35%**	36.33%	38.35%
	0.1%	92.31%	33.27%	3.01%
	1%	91.90%	27.74%	0.74%
	10%	87.69%	13.65%	0.14%
Delay 204 ms Max.Delay 600 ms MinDelay 160 ms	0%	**92.35%**	5.29%	5.99%
	0.1%	62.36%	4.79%	1.50%
	1%	58.57%	3.67%	0.28%
	10%	38.60%	2.37%	0.05%

Test results

Conclusions from test results:

1. Udtp protocol is working on existing network
2. Udtp protocol is not blocked by Internet infrastructure .
3. Transmission rate increase recorded

OS	Architecture							
	x86 (i386)	x86_64 (amd64)	arm32 (AArch32)	arm64 (AArch64)	mips64	ppc32, ppc64le	RISC-V RISCV64	Other
Linux								
Kernel 3.x	1,2,3,6	1,2,3,6	1,2,3,6	1,2,3,6	6,9,10	7,11	x	x
Kernel 4.x	1,2,3,6	1,2,3,6	1,2,3,6	1,2,3,6	6,9,10	7,11	10,x	x
Kernel 5.x	1,2,3,6,7	1,2,3,6,7	1,2,3,6,7	1,2,3,6,7	6,9,10	7,11	10,x	x
Window								
10, 8, 7	1,3,4,5,9	1,3,4,5,9						
2012 R2, 2016, 2019		4,5						
Android								
5,6,7,8,9,10,11			1,2,3,5,8,9	1,2,3,5,8,9	1,2,3,5,8,9			
macOS								
10.13, 10.14, 10.15	1,3,4,5,9	1,3,4,5,9						
iOS								
12, 13, 14			1,2,3,5,8,9	1,2,3,5,8,9				
RT OS (QNX)								
	x	x	10,11,x	10,11,x	10,x		x	x