University of Warsaw
Interdisciplinary Centre for Mathematical
and Computational Modelling

Jakub Kopeć
Student's book no. 417354

Exploration of cooperation-enabling solutions in HPC

Second cycle degree thesis
field of study COMPUTATIONAL ENGINEERING

The thesis written under the supervision of:
Marek Michalewicz, Ph.D.
Interdisciplinary Centre for Mathematical
and Computational Modelling

Warsaw, March 2020

Oświadczenie kierującego pracą

Oświadczam, że niniejsza praca została przygotowana pod moim kierunkiem i stwierdzam, że spełniła ona warunki do przedstawienia jej w postępowaniu o nadanie tytułu zawodowego.

Data Podpis kierującego pracą

Statement of the Supervisor on Submission of the Thesis

I hereby certify that the thesis submitted has been prepared under my supervision and I declare that it satisfies the requirements of submission in the proceedings for the award of a degree.

Date Supervisor's signature

Oświadczenie autora pracy

Świadom odpowiedzialności prawnej oświadczam, że niniejsza praca dyplomowa została napisana przeze mnie samodzielnie i nie zawiera treści uzyskanych w sposób niezgodny z obowiązującymi przepisami.
Oświadczam również, że przedstawiona praca nie była wcześniej przedmiotem procedur związanych z uzyskaniem tytułu zawodowego w wyższej uczelni.
Oświadczam ponadto, że niniejsza wersja pracy jest identyczna z załączoną wersją elektroniczną.

Data Podpis autora pracy

Statement of the Author's on Submission of the Thesis

Aware of legal liability I certify that the thesis submitted has been prepared by myself and does not include information gathered contrary to the law.
I also declare that the thesis submitted has not been the subject of proceedings resulting in the award of a university degree.
Furthermore I certify that the submitted version of the thesis is identical with its attached electronic version.

Date Author's signature

SAGE2

SAGE stands for Scalable Amplified Group Environment and it is Node.js-based (JavaScript) software that facilitates use of large video-walls that are intended to be used by multiple users at a time. It works as the server's software - all the resource-demanding operations are handled by the server, so the end-user do not need to posses powerful workstation in order to use a video-wall. The special script run on the machine that operate the displays creates a server that provide two services. The first one is Google Chrome-based interface that displays the working environment on the video-wall. The second one is web-accessible portal that allows authenticated users (each SAGE2 session could be password-protected) to control content displayed on the video-wall. When user connects to the server he is presented a simplified schema of the video-wall that shows how the display space is arranged and an intuitive interface that allows to control the contents of the display. [2]

SAGE2 provides user with the modules that may be displayed on the screen. The essential ones, like web browser, google maps, notepad etc. are already implemented by the creators of the SAGE2, but there is special appstore where developers publish their own modules designated to support another software. If one would like to create his own module the developer's guide for making such module is available at SAGE2 project homepage.

During the installation of the SAGE2 in Technology Center of Interdisciplinary Centre for Mathematical and Computational Modelling (ICM) author encountered few problems concerning the network configuration as SAGE2 server required public IP address and DMZ network what could create some complications if the network design had not been adapted to such use. The ICM technicians bypassed these issues with ports forwarding, but they also noticed a security weakness induced by such solution - open unsecured port pose threat of unauthorized access to the network. In order to eliminate such possibility the access to the port used by SAGE2 was secured with username/password authentication.

After the installation author prepared a presentation for ICM's staff about the functionality and instructions on how to use SAGE2 software. The presentation was followed by a discussion on possible appliances of SAGE2 as a tool for cooperation between research facilities. The most repeated remark was that this software allows only cooperation between two SAGE2 sites and that there is practically no support for users that are not present in front of the video-wall. Another issue that was pointed out by the audience was the fact that API for making own SAGE2 module is rather fixed on JavaScript and it would not be easy to create such module for an application that was not designed in advance to support such functionality. The last but not least was the matter of security of SAGE2. The audience noticed that there is no clear declaration about the decryption used by SAGE2 and that SAGE2 protocols could not be sufficiently secure to be used in projects that require confidentiality.

To sum up, the SAGE2 could act as platform for conducting collaborative research, but in present-day form it may be used locally as middleware for large resolution screens rather than for remote collaborative work. Even though the creators of SAGE2 successfully conducted remote collaborative work session [2], in the author's opinion preparing such sessions require significant effort to establish reliable connection between two SAGE2 sites that would be justified only in case of long cooperation between two institutions where multiple SAGE2 sessions would bring noticeable boost in cooperation. Moreover the issues mentioned in the previous paragraph should be addressed beforehand.s On the other side - SAGE2 is perfect tool to facilitate the collaborative effort in case when all users are physically present in front of the video-wall. Perfect example of such use is StickySchedule app that was intended to be launched on SAGE2 site and it's purpose is to ease and precipitate the conferences scheduling [3].

DTP
The second idea was to create from scratch a web portal that would mask IT's expertise-demanding part of big data moving aspects from the end user and simplify such task as much as possible. The draft name for the project was "Data Transfer Portal" (abr. DTP). The main motivation behind the project was fact that software that is used in HPC applications to move large amount of data is rather unfriendly and unintuitive for the user that is not IT-technician responsible for data transfer. Not only is the use of such software complicated, but it is also necessary to test the connection properties between source and destination in order to optimise the transfer. The DTP is intended to handle all this operations and provide the end-user with simple web interface that is easy to use and do not require IT expertise. On the beginning author committed some time to learn how to use django framework with python [1] as it seemed that project would require creating a web portal at some point. Nevertheless, when author started to think on how DTP should look like he encountered a problem trying to answer the question "How DTP server will know that the user that require data transfer is really who he claim that he is and if he is allowed to transfer that data (permissions control)?". At that point author completely focused on research on user authentication and authorization methods. The main issue was the fact that assumedly users would not be the members of one organization and each user should be a member of at least two different parties (one source and one destination).

Authentication methods:[4]

- basic authentication - user and password (may be encrypted)
- SAML - Security Assertion Markup Language[5]
- Oauth2.0[6] with OpenID[7]

SAML - Security Assertion Markup Language SAML is a XML-based standard created and maintained by OASIS (Organization for the Advancement of Structured Information Standards). It's main purpose is to describe how the security information could be exchanged on-line between two separate parties. It is based on the exchange of standardised messages , called SAML assertions, that are created according to the standard's syntax and rules. The framework's assumption is to provide components that could be used in many configuration to meet the user's requirements. Moreover, the SAML specification includes profiles that are predefined to satisfy the most common use-cases.[5]

OpenID Connect OpenID Connect is the authentication standard used on top of the OAuth2.0 authorization protocol. In the previous versions OpenID and OAuth were separate standards. OpenId's purpose was to verify the identity of the user based on the authentication that is performed by OpenID provider[8]. OAuth 2.0 protocol was responsible for verification of the user permissions to the requested assets[6] while OpenID just ensures the service provider that the user is in control of some identifier (e.g. the gmail account) and there was no way of determining if the user name or any other data are valid and real. It is possible to create OpenID provider on one's own "(http://wiki.openid.net/w/page/12995226/Run your own identity server)" and use it to issue conduct completely valid authentication to the service provider using OpenID. In the newest version of the standard - OpenID Connect - these two protocols were connected and now OpenID provides not only the user authentication but it also enables the user authorization.[7]

Difference between SAML and OpenID Before the implementation of the OpenID Connect there was a significant difference between SAML and OpenID. First of all the OpenID2.0 is the authentication protocol while SAML provides the authorization and the authentication as well. Secondly, SAML authorization was based on the thrust relationship and on the beforehand arrangements between partners. The service provider trusted that the identity provider is or was able to authenticate the user real identity. The example of such scenario are e-identity services available on government's and local authorities' sites. When a citizen is to fulfill administrative matters online he may use his bank account to log in to authorities' portal. In this case the user's bank is the identity provider and the government's site is a service provider that trusts that bank's employee verified the user's identity (for example - checked his ID card or passport) when he opened the bank account. SAML standardised the messages used in authentication (and, on the next stages, in authorization) process. As for OpenID such authentication was not possible as identity provider could not guarantee truthfulness of the user's data). After the introduction of the OpenID Connect where authorization was joined with authentication the difference in functions between those two standards blurred. Nevertheless the implementation of OpenID Connect is simpler than the implementation of SAML protocol so OpenID is used in simpler web applications while the SAML is used in large federations (e.g. university or enterprise federations) as it has already been widely adopted in existing federations and is more mature standard than OpenID. On the other hand SAML is restricted to browser use, so in the case of application or device usage OpenID is the obvious choice.[9]

As an introduction to the research on the history and the development of data transmission protocol the author decided to look for the information how much data is created nowadays by humanity and how that amount has changed over the history. According to the academic research conducted in 2007 all the information humanity was able to store by 1986 had the capacity of 2.6 optimally compressed exabytes. In the article authors also describe the methodology that they use to estimate the amount of data and the assumptions that they introduced during their research. The storage capacity estimated by the authors of that study grew exponentially through the years: 15.8 EB in 1993, 54.5 EB in 2000, to overwhelming 200 exabytes in 2007 (when the study was conducted). [11] The information on the global storage capacity is not easy to estimate as well it is not effortless to find it. One of the sources of such information (also mentioned in [11]) is International Data Corporation (IDC) - it is private company that offer market intelligence, advisory services information technology, telecommunications, and consumer technology markets. [12] IDC prepared recent report on world's storage capacity, but due to the company's policy it is not freely available. Nevertheless in one of their press releases IDC passed two numbers that may be interesting: they estimated that current world's storage capacity is 6.8 ZB and they forecast that this number will increase to 8.9 ZB in 2024. [13] The fig. 1 was prepared basing on the data presented in this paragraph and one may easily notice that the world storage capacity is now rising as rapidly as never before.

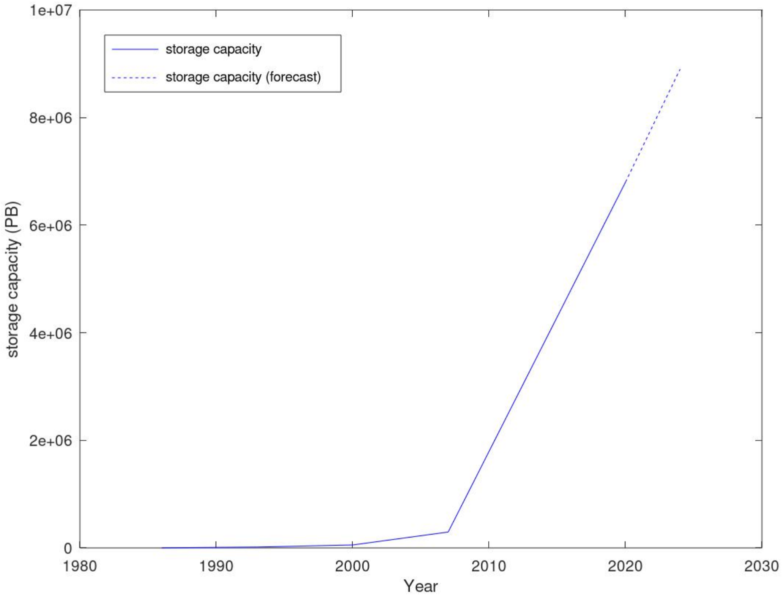

Figure 1: Global storage capacity basing on [11] and [13]

The next aspect that author decided to check was the rate of the internet growth. Again it was surprisingly hard to find any accurate information on that matter, but the author came across Cisco's Annual Internet Reports where the technological giant include estimation of average global Internet traffic. In one entry on Cisco's official blog the member of the team responsible for the Annual Internet Reports aggregate all their historical data on global Internet traffic for the years 1984-2014. [14] For the next years and the forecast up to 2022 year it was necessary to went through Cisco's Annual Internet Reports from years 2016-2019 that was published on Cisco's website. [15] The author aggregated all the results from the reports and presented them on the fig. 2.

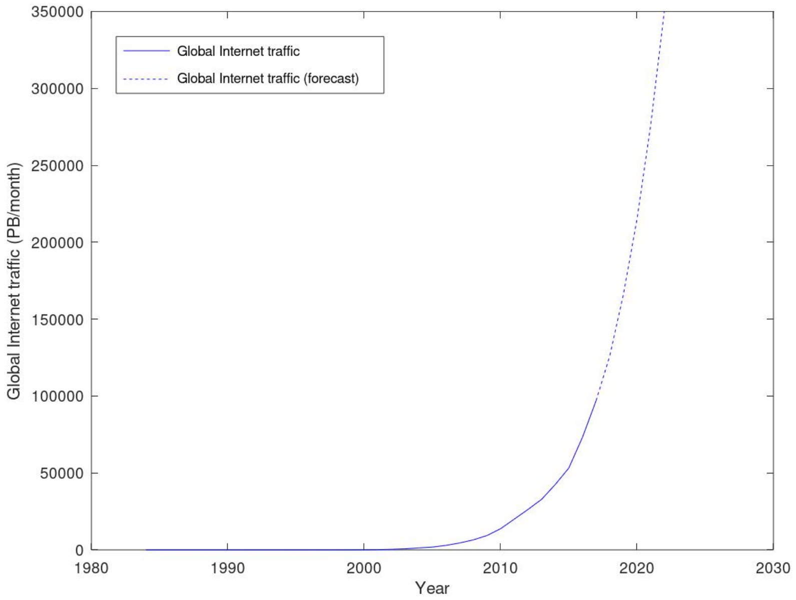

Figure 2: Global Internet traffic in years 1984-2017 with forecast up to 2022

One studying fig. 1 and fig. 2 may easily recognise the exponential growth of the depicted numbers. It is obvious that the growth of Internet traffic and global storage capacity is strictly connected with the new technology development. Over time humanity invented new techniques to transfer and store more and more data.

The beginnings of the computer networking If one was to find the beginnings of computer's networking he would have go back to year 1957 when the Soviet Union (USSR - Union of Soviet Socialist Republics) won the space race during the Cold War and launched Sputnik - the first space satellite. As a result of failure the United States' Department of Defense founded Advanced Research Projects Agency that was responsible for execution of innovative research as well as for providing innovative tools that empower the dynamic science development.[16] One of the tools that were used for science that time were computers that created new possibilities for various researchers. It was noticed at once that computer resources, which were scarce and expensive, need to be shared between multiple scientific projects and researchers. To address this problem computer timesharing was introduced - one computer resources were allocated to multiple users - when some users were idle the others would use the computer. L. Kleinrock noticed that such system could be used to share communication links in order to build efficient communication network. He came with the idea of splitting the messages into smaller parts and to send them independently. It was called packet switching and was essential in building reliable networks used in data transmission. Two other researchers - Paul Baran (RAND Corporation) and Donald Davies (National Physical Laboratory (UK)) came up with the same idea independently. In 1962 J.C.R. Licklinder was appointed as the first director of the Information Processing Techniques Office (IPTO) in ARPA and he was the first one to came up with a vision of "galactic" network of computers where interconnected processing nodes will enable anyone to access data and programs from anywhere. [16] In 1965 ARPA recognized the need to connect their researchers to the few computers that was spread across the United States. Such connection would allow ARPA to share its geologically spread resources in a cost-effective manner. The pursuit to fulfill that need led to the project named ARPANET. It is worth noticing that research on this project was exceptional as it was open to broad community researchers.[17] There was a problem on how different computers (different architecture, different operating system etc.) could communicate each other. In 1968 Bolt, Beranek and Newman (BBN) came with a project of Interface Message Processor (IMP) that was in fact a separate computer that handled switching and communication functions. There was an IMP at each computer's location which served as an interconnection between the computer and the network. In 1969 the ARPANET was launched - it was the first computer network that connected four computers located at the University of California at Los Angeles (UCLA), the University of California at Santa Barbara (UCSB), the Stanford Research Institute (SRI) in California and the University of Utah. ARPANET utilised Network Control Protocol (NCP) that was the first network protocol - a set of signals that was agreed between parties that was used to establish communication channels and enabled data to be passed between the computers. Over time more computers and smaller subnetworks were connected to the ARPANET, the first international links were created. In 1971 Ray Tomlinson from BBN came up with a minor software that allowed the electronic mail exchange between the computers in the network. After few updates it turned out that users were eager to use this form of communication - after few months emails accounted for the majority of traffic in the ARPANET. [17] In 1974 Vin Cerf and Robert Kahn came up with a new protocols that could supersede NCP as it started to be insufficient. ~~The main issue with NCP was the fact that each subnetwork used its own protocol and in order to join to the ARPANET it had to be interconnected via IMP (and the data between subnetworks were exchanged using NCP). - zdanie do wywalenia!!!~~ [17] The new protocols proposed by Cerf and Kahn were the Transmission Control Protocol that includes set of rules for computer on how to use network (how to establish and break connections) and the Internet Protocol (IP) that determined how individual data packets were routed. These protocols were also implementing open architecture philosophy that was revolutionary at the time - so they were freely available for the whole research community. [16] After few years TCP/IP were implemented widely not only in the ARPANET but also in smaller subnetworks what enabled to exchange data between networks (and therefore between computers) freely without any significant barriers. As a reason the number of computers and links in the global network has risen abruptly over upcoming years. The TCP/IP protocol was updated few times since then, but it is still in use in today's Internet. Another invention that is worth mentioning in the aspect of global network development is the World Wide Web proposed by Tim Berners-Lee from CERN in 1989. He came up with the idea of building distributed hypermedia server that will allow network users to create and share electronic documents that may comprise of multiple file types such as text files, sounds, pictures etc.. In the upcoming years he created WWW client, Hypertext Transfer Protocol (HTTP) and HyperText Markup Language (HTML) that accelerated significantly development of the global network and created the Internet in the form that we know today.[16]

The Open Systems Interconnection (OSI) Model ~~OSI model is the reference model on how computers may communicate each other. The main challenge in accomplishing it is the fact that different computer systems may be incompatible - they may despite the fact that there are variety of computer systems.~~

Back in the 1970s the compatibility between different networks and computer systems was a significant challenge. Even though the TCP/IP protocol was already invented and implemented in ARPANET there were no universal standard on how interconnection protocol should look like in general and almost each vendor of computer systems was using it's own protocol. The task to create standard that would satisfy each vendor (different processor architectures/speeds/network interfaces/operating systems/memory etc.) was impracticable, so in 1977 the International Standards Organization (ISO) undertook the task of creating the framework for building open communication standards that was adapted in 1984 as the Open Systems Interconnect (OSI) Model. [18] The model was described in the international standard and it also defines what open system mean: "The fact that a system is open does not imply any particular systems implementation, technology or means of interconnection, but refers to the mutual recognition and support of the applicable standards". [20] So the ISO did not create new protocol nor described every possible network, but it created the template for creating network protocols that would enable different systems to interconnect each other despite the differences between them. It was accomplished by dividing the communication between the systems into several sub-tasks creating a layered structure where given layer provide services for layer that is above using the services provided by the previous layer. It is not necessary to create separate protocols for each layer, but this reference model is helpful in describing the given protocol's role in the communication process.

The OSI model defines 7 layers [18, 19]:

- Physical layer The physical layer of the protocol describes how to transfer bits of information between devices. The main issue is the medium that is used to transfer information - it could be copper wire (or set of wires), optic fiber or radio waves etc. The physical layer of the protocol also defines how 1's and 0's are represented in the medium (e.g. different voltage levels, transition between voltage levels, light impulses, frequency modulation, how long does impulse last) and how wires, plugs and sockets are built.

- Data link layer The main data link layer task is to organise bits of information into frames that are in most cases prefixed with preamble pattern of bits that inform interconnected devices that meaningful information will be broadcast in the medium. This layer defines the format of the frames and may implement flow control of data and error detection over physical medium.

- Network layer The network layer is responsible for passing the data between the source and destination device across the network that may consist of multiple sub-networks. It's main task is to route data between networks via optimal path, but it also may also help to control congestion in sub-nets in order to prevent bottlenecks.

- Transport layer The transport layer's main task is to provide reliability of message transmission between end users. It splits the data into smaller packets that are sent separately over the network and provides mechanisms that allow verification if the whole message was delivered correctly and flow control of data between the source and destination.

- Session Layer This layer is responsible for maintaining connection session (established ongoing communication) between interconnected devices and handles the connection recovery and synchronization in case of unexpected disconnection.

- Presentation Layer The Presentation Layer converts transmitted data into format that is suitable for user's application and reversely - it transform received data (basing on the application's requirements and receiver's system configuration) into format compatible with the application. Data encryption/decryption may be conducted at this stage of communication.

- Application Layer The last layer specify how end user interface with network services - it allows user or given software to utilise network connection or access remote resources.

TCP/IP protocol - the foundation of the Internet The protocols mentioned before - Internet Protocol and Transmission Control Protocol - are the fundamentals of today's Internet so they deserve a chapter in this thesis, even though they may be not the fastest and the most effective protocols that may be used in large datasets transfer or in HPC interconnects. The version of TCP/IP protocol that we use to this very day was established in 1980 and was declared as the standard for ARPANET in 1983.[17] TCP/IP is used in each communication in the Internet (the Internet, as the name suggest, is just a set of smaller interconnected networks that span around the globe), even though TCP/IP packets may be encapsulated by other protocols packet on route between the source and destination, they are still the basis of the Internet that we know and use today. If one compare TCP/IP with the OSI reference model one may say that TCP is responsible for the tasks of transport layer and IP for the tasks of network layer. One of the reasons of TCP/IP popularity is the fact that these standards are open - so they are available to use for anyone as opposed to proprietary protocols. TCP - Transmission Control Protocol TCP is connection-oriented protocol - the whole process of data transmission begins with three-way handshake - side that initiated the connection (client) sends the SYN packet to the server, server responses client with SYN-ACK packet. After receiving SYN-ACK packet the client sends the ACK packet to the server to confirm that SYN-ACK packet has been received - on this stage the connection is established and both ends of the connection are ready to send the data. When TCP obtains the data to send from higher layers it splits them into smaller parts - the packets (packet size depends on system specification and lower layers requirements) - that are send to the destination separately. The packet consist of the header - which contains information on source/destination port, the sequence number, packet size and the checksum, that allows to verify data correctness - and of the actual data that are transmitted by the given packet. The packets may arrive to the receiver out of order or some of them may not arrive at all. The TCP task on the receiver side is to respond for every received packet with ACK packet and to put them in correct order, if sender notice that some sent packets are not responded with ACK it re-transmits them until receiver acknowledge them (or transmission time-out will pass). When the received message is complete and set in correct order the TCP pass the data to higher layers of the receiver. Transferring data using TCP is reliable as the sender is assured that message was received completely and the receiver is able to put the packets in correct order. Once the connection is established data may be sent both ways until it will be terminated with the exchange of appropriate packets (FIN and ACK packets). The main con of using TCP protocol is it's reliability - higher layers' applications do not have to implement any control mechanisms - they may assume that TCP delivered the data (or notify them in case of transmission failure). On the other hand providing reliability in that way may lead to significant overhead in data transmission - first of all, the headers are additional data that need to be transmitted, but more importantly - the packet that arrives the last limits transfer rate - data may not be passed further until the last packet arrives despite the fact that the delivered portion of data may be already processed by an application. [21]

IP - Internet Protocol The Internet Protocol is responsible for passing the data from the source to the destination. It add to the data passed from higher layers (e.g. TCP packet) it's own header that include sender and the receiver internet addresses (IP numbers). The most challenging task of this protocol is routing process - finding the most optimal path for packet from source to destination. Devices that take part in this process and relay the packets are called routers - it may be dedicated hardware or computer may act as one - that switch incoming packets with the use of routing tables. The routing table is a list maintained by each router that specifies where packets should be passed in order to get to their destination. When the packet arrive to the router it checks if the packet's destination address is in it's routing table - if the destination address is found then the router pass the packet to the path specified by the routing table. But if the destination address is not in router's routing table it passes the packet to it's default gateway that is usually the other router that is higher in hierarchy (have larger/wider routing table) and the process is repeated until the destination address is found. [22] The routing tables may be maintained by administrators in smaller networks, but in wide area networks they are maintained and updated autonomously by appropriate algorithms. The routing table may include more than one path for given destination and juxtapose it with metrics that may be updated (e.g. on the base of current path's congestion) and the packets are sent through the most optimal path. As a result the packets that creates one longer message may be sent through different paths as they are sent and routed independently. Due to the fact that nowadays the number of interconnected networks is tremendous (and is still growing) there are multiple paths that could be used for packet routing - that makes the Internet immune to breakdowns. Even if some links or part of networks become corrupted (by natural causes or deliberate actions) the rest of the networks may remain operational as packets may be delivered via alternate paths.

UDP - User Datagram Protocol UDP was created and defined in 1980 and was intended to work on top of Internet Protocol. It was intended to enable datagram transmission in packet-switched networks. It is simple procedure to exchange messages between computers in network. UDP is message-oriented it prepares message with data, adds the header with the source port number, the destination port number, the data length and the checksum (that allow receiver to verify data integrity). There are no acknowledgments nor three-way handshake characteristic to TPC - as a result the transmission via UDP is not reliable, but in return the protocol is a lot simpler and the transmission overhead is significantly smaller. UDP may be used in simple application as a basic network interface (with the awareness that the data delivery is not guaranteed) or in scenarios where the real-time data flow is more important than reliable transfer of all packets. Examples of such cases are video/audio streaming, videotalks via the Internet or online computer games - in case of loss a part of the packets the quality of service may be decreased (e.g. lower sound/video quality), but such applications cannot work if sender keeps retransmitting a lost packet instead of providing new packets that are necessary (while the lost packet may be already useless). UDP may be also used when another network layers provide reliability of transfer - if such solution is implemented in a smart way it may provide the reliability similar to TCP, but with lower overhead or more optimal workflow or data processing. [23]

Ethernet - an example of protocol concerning data link and physical layers The ethernet protocol was designed by Xerox and it was introduced in 1980, after 4 years, in 1984 the transfer of IP datagrams over ethernet network was standardised by Network Working Group in the characteristic form of the Request for Comments. [24] It became a popular standard in LANs and in industrial networks and is still widely used in those fields. The ethernet specifies how multiple hosts in one network share physical medium, the frame format and the physical aspects of the medium (wire/fiber specification, plugs shape and size etc.). Each device that may be connected to the network yields vendor-assigned MAC address (media access control) and it is one of the ethernet's task to enable the hosts to map network's addresses into MAC addresses. It accomplish this task using ARP (Address Resolution Protocol) packets that are broadcast to all host connected to the medium in order to query destination MAC address. The ethernet is also responsible for managing the access to the medium and checking transmitted data integrity (checksums).

ATM - Asynchronus Transfer Mode

Bibliography

[1] https://docs.djangoproject.com/en/3.1/intro/

[2] T. Marrinan, J. Aurisano, A. Nishimoto, K. Bharadwaj, V. Mateevitsi, L. Renambot, L. Long, A. Johnson, and J. Leigh, “SAGE2: A New Approach for Data Intensive Collaboration Using Scalable Resolution Shared Displays” (best paper award), 10th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing. 2014.

[3] Vishal Doshi, Sneha Tuteja, Krishna Bharadwaj, Davide Tantillo, Thomas Marrinan, James Patton, G. Elisabeta Marai, “StickySchedule: an interactive multi-user application for conference scheduling on large-scale shared displays“, Proceedings of the 6th ACM International Symposium on Pervasive Displays (PerDis ’17), Lugano, Switzerland, June 7-9, 2017. http://dx.doi.org/10.1145/3078810.3078817

[4] https://dzone.com/articles/my-security-notes

[5] Security Assertion Markup Language V2.0 Technical Overview Committee Draft 02 25 March 2008 http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0.html

[6] Security Assertion Markup Language V2.0 Technical Overview Committee Draft 02 25 March 2008 http://docs.oasis-open.org/security/saml/Post2.0/sstc-saml-tech-overview-2.0.html

[7] OpenID Connect specifications https://openid.net/specs/openid-connect-core-1 0.html

[8] OpenID2.0 specification https://openid.net/specs/openid-authentication-2 0.html

[9] Ubisecure’s white paper "SAML vs OAuth 2.0 vs OpenID Connect" https://www.ubisecure.com/about/resources/saml-oauth-openid-connect/

[10] Great article about Federated Identity Management - may be useful later Chadwick, David W. (2009) Federated Identity Management. In: Aldini, Alessandro and Barthe,Gilles and Gorrieri, Roberto, eds. FOSAD 2008/2009. LNCS (5705). Springer-Verlag, Berlin,pp. 182-196. ISBN 978-3-642-03828-0. https://kar.kent.ac.uk/30609/1/FederatedIdManChapter.pdf

[11] Martin Hilbert, Priscila López Research Article "The World’s Technological Capacity to Store, Communicate, and Compute Information" Science, 332(6025), 60 –65. doi:10.1126/science.1200970 http://www.uvm.edu/pdodds/files/papers/others/2011/hilbert2011a.pdf (09.01.2021)

[12] International Data Corporation (IDC) about page https://www.idc.com/about (09.01.2021)

[13] IDC Media Center article "IDC's Global StorageSphere Forecast Shows Continued Strong Growth in the World's Installed Base of Storage Capacity" available online at: https://www.idc.com/getdoc.jsp?containerId=prUS46303920 (09.01.2021)

[14] Arielle Sumits, Cisco Blog article "The History and Future of Internet Traffic" https://blogs.cisco.com/sp/the-history-and-future-of-internet-traffic (09.01.2021)

[15] Cisco Annual Internet Report https://www.cisco.com/c/en/us/solutions/executive-perspectives/annual-internet-report/index.html (09.01.2021)

[16] Raphael Cohen-Almagor, Internet History,International Journal of Technoethics Vol. 2(2):45-64 April 2011 DOI: 10.4018/jte.2011040104 https://www.researchgate.net/publication/215660523_Internet_History(11.01.2021)

[17] Leonard Kleinrock, An early history of the internet [History of Communications], IEEE Communications Magazine (Volume: 48, Issue: 8, August 2010) Page(s): 26 - 36 Date of Publication: 03 August 2010 https://ieeexplore.ieee.org/document/5534584 (11.01.2021)

[18] Louis Costa, Research Article Open Systems Interconnect (OSI) Model, JALA: Journal of the Association for Laboratory Automation, Volume: 3 issue: 1, page(s): 28-35 Issue published: March 1, 1998 https://journals.sagepub.com/doi/full/10.1177/221106829800300108 (24.01.2021)

[19] Rachelle Miller, The OSI Model: An Overview, SANS Institute Information Security Reading Room https://www.sans.org/reading-room/whitepapers/standards/osi-model-overview-543 (26.01.2020)

[20] International Organization for Standardization, ISO 7498-1 Standard Information technology - Open Systems Interconnection - Basic Reference Model: The Basic Model https://standards.iso.org/ittf/PubliclyAvailableStandards/s020269_ISO_IEC_7498-1_1994(E).zip (24.01.2021)

[21] Postel J., RFC: 793 - TRANSMISSION CONTROL PROTOCOL,DARPA INTERNET PROGRAM, PROTOCOL SPECIFICATION, Information Sciences Institute, University of Southern California, September 1981 https://tools.ietf.org/html/rfc793(30.01.2021)

[22] Rus Shuler, How Does the Internet Work?, Pomeroy IT Solutions, 2002 https://web.stanford.edu/class/msande91si/www-spr04/readings/week1/InternetWhitepaper.htm (31.01.2021)

[23] J. Postel, RFC 768 - User Datagram Protocol, ISI, August 1980 https://tools.ietf.org/html/rfc768(02.02.2021)

[24] J. Postel, RFC 895 - A Standard for the Transmission of IP Datagrams over Experimental Ethernet Networks, ISI, April 1984