Journal Of the Iraqia University (76-3) June (2026)

ISSN(Print): 1813-4521 OnLine ISSN:2663-7502
Journal Of the Iraqia University
available online at
https://iasj.rdd.edu.iq/journals/journal/view/247

مقال مراجعة بروتوكول نقل النص التشعبي HTTP

م.د. آلاء أمير يوسف
كلية الآداب الجامعة المستنصرية

A Review Article on the Hypertext Transfer Protocol (HTTP)

Dr. Alaa Ameer Yousif
Mustansiriyah University/ College of Arts
alaa.a85@uomustansiriyah.edu.iq

الملخص:

يعد بروتوكول نقل النص التشعبي HTTP الركيزة الأساسية لتبادل البيانات عبر شبكة الويب العالمية منذ بدايات الانترنت وحتى الوقت الحاضر، ومع التطور المستمر في بنية الويب وزيادة احتياجات المستخدمين لاداء عالٍ وامان اكبر خضع البروتوكول لسلسلة من التحسينات الجوهرية من النسخة الاولى (HTTP/09) الى احدث اصداراته (HTTP/3). يهدف المقال الى استعراض التطور التاريخي والتقني لبرتوكول نقل النص التشعبي HTTP منذ نشأته، ومراجعة تحليلية لأجياله وأدائه وأمنه في بيئة الويب الحديثة، مع التركيز على التطورات التي حسنت من اداء الاتصال بالشبكات، والتغييرات في تصميم البنية التحتية لنقل البيانات، والتحديات الأمنية في سياق استخدام HTTPS,TLS، وتبرز اهمية المقال كونه يجمع بين الجانب التاريخي والتقني والتحليلي لبروتوكول HTTP.

الكلمات المفتاحية: بروتوكول نقل النص التشعبي، HTTP/1.1، HTTP/2، HTTP/3، HTTPS.

Abstract:

The Hypertext Transfer Protocol (HTTP) has been the fundamental backbone for data exchange across the World Wide Web since the early beginnings of the internet up to the present time. With the continuous evolution of web architecture and the increasing demand for higher performance and stronger security, the protocol has undergone a series of substantial improvements, from its first version (HTTP/09) to its latest release (HTTP/3). This article aims to review the historical and technical development of HTTP since its inception, providing an analytical examination of its generations, performance, and security within the modern web environment. It focuses on advancements that have enhanced network communication performance, the changes in data transport infrastructure design, and the security challenges in the associated with the use HTTPS and TLS. The importance of this article lies in its integration of historical, technical, and analytical perspectives on the HTTP protocol.

Keywords: Hypertext Transfer Protocol, HTTP/1.1, HTTP/2, HTTP/3, HTTPS.

المقدمة:

يعد بروتكول نقل النص التشعبي HTTP العمود الفقري الأساسي للانترنت اذ يحدد الية تبادل المعلومات وجلب موارد بين منصات واجهزة اتصال، وهو ما يتيح للمستخدمين تحميل مواقع الويب، وقد كان واحدا من ثلاث تقنيات رئيسية عرفها تيم بيرنرز لي عند اختراعه للويب، الى جانب المعرفات الفريدة للموارد والتي اشتقت منها محددات مواقع الموارد الموحدة URLs ولغة ترميز النص التشعبي HTML، اما بقية اجزاء الانترنت فلها بروتوكولاتها ومعاييرها الخاصة التي تنظم الية عملها وكيفية توجيه الرسائل عبر الشبكة، مثل البريد الالكتروني الذي يعتمد على بروتوكولات SMTP, IMAP, POP. يُستخدم بروتوكول HTTP من قِبل مبادرة المعلومات العالمية للويب منذ عام ١٩٩٠، وشهد بروتوكول HTTP العديد من التغيرات السريعة حيث تم تطوير العديد من الاصدارات ساهمت في تحسين السرعة والأداء والأمان، وهذه الإصدارات هي:

HTTP/0.9 حيث بدأ مع ظهور الشبكة العالمية (WWW)، حيث استخدم بروتوكول HTTP كنظام جديد لتتبع المعلومات المستخدمة او التي يتم انشاؤها، وكان اول اصدار تم تطويره هو HTTP/0.9 عام ١٩٩١، وهو بروتوكول بسيط جدا يفتقر الى الرؤوس، ويعاني من العديد من

القيود، اذ كان يدعم طريقة واحدة فقط هي GET وقد صممت هذه الطريقة لجلب بيانات HTML فقط دون الصور، وقد تم تدارك هذا القصور في الجيل التالي من بروتوكول HTTP وهو HTTP/1.0.

HTTP/1.0 صدر عام ١٩٩٦ ادى الى ادخال عدد هائل من التعديلات على ذلك البروتوكول البسيط الذي كان قد اطلق ثورة في عالم الشبكات، فقد تضمن العديد من الخصائص الجديدة مثل الرؤوس Headers ومعالجة الاخطاء، والطلبات الشرطية، وترميز المحتوى، ورموز الاستجابة، وقد جعلت هذه الاضافات بروتوكول HTTP اكثر تطورا واتساعا مما كان عليه، وكانت رموز الحالة ووصفها مفاهيم جديدة في HTTP/1.0، اذ لم يكن في HTTP/0.9 مفهوم لرموز الاستجابة وكانت الاخطاء تعرض فقط ضمن محتوى HTML المعاد. وعلى الرغم من هذه التغييرات فقد عانى من بعض اوجه القصور، فقد واجه مشكلة عدم القدرة على ابقاء الاتصال مفتوحا حتى يتم تنفيذ الطلب بالكامل، كما كان يفتقر الى وجود راس Host الالزامية، والى الخيار الاساسي الخاص بالتخزين المؤقت Caching.

HTTP/1.1 حيث صدر عام ١٩٩٧ يتضمن هذا البروتوكول متطلبات أكثر صرامة من بروتوكول HTTP/1.0 يعتمد على نظام المرجعية الذي يوفره مُعرّف الموارد الموحد URI كموقع URL أو اسم URN للإشارة إلى المورد الذي تُطبّق عليه الطريقة، وتمرر الرسائل بتنسيق مشابه للتنسيق المُستخدم في بريد الإنترنت كما هو مُعرّف في امتدادات بريد الإنترنت متعددة الأغراض MIME. يستخدم بروتوكول HTTP/1.1 أيضًا كبروتوكول عام للتواصل بين وكلاء المستخدمين والخوادم الوسيطة/البوابات، وقادر على التكامل مع أنظمة إنترنت أخرى تعتمد بروتوكولات مختلفة مثل SMTP وNNTP وFTP وGopher وWAIS، وبهذه الطريقة يتيح HTTP/1.1 الوصول الأساسي إلى الوسائط المتعددة للموارد المتاحة من تطبيقات متنوعة. وتتضمن كل استجابة HTTP/1.1 رمز حالة يشير الى حالة الطلب، رمز الحالة هو عدد صحيح مكون من ثلاثة أرقام ووصف نصي لهذا الرمز، يُشير إلى نتيجة محاولة فهم الطلب وتلبيته، يحدد الرقم الأول من رمز الحالة فئة الاستجابة، ولا يستخدم الرقمان الأخيران في أي تصنيف. أما الرقم الأول، فله خمس قيم:

١xx Informational (معلوماتي): تم استلام الطلب، جارٍ متابعة العملية.

٢xx Successful (ناجح): تم استلام الطلب وفهمه وقبوله بنجاح.

٣xx Redirection (إعادة توجيه): يلزم اتخاذ إجراء إضافي لإكمال الطلب.

٤xx Client Error (خطأ من جانب العميل): يحتوي الطلب على صيغة غير صحيحة أو لا يمكن تنفيذه

٥xx Server Error (خطأ من جانب الخادم): فشل الخادم في تنفيذ طلب يبدو صحيحًا.

:HTTP/2

في عام ٢٠١٥ تم تطوير نسخة جديدة من بروتوكول HTTP يعرف HTTP/2، وضع بروتوكول SPDY الاساس لبروتوكول HTTP/2، واستخدم لاختبار واثبات عدد من خصائصه الأساسية، ويقوم المبدا الجوهري لـ HTTP/2 على تحسين اداء طبقة النقل، الى جانب تمكين معدلات اعلى مع زمن تاخير اقل، ويعد HTTP/2 بروتوكول على مستوى التطبيق موجهاً بالاتصال، ويعمل فوق اتصال TCP حيث يكون العميل هو الجهة التي تبدأ اتصال TCP. يكمن الدافع في تطوير بروتوكول HTTP/2 في اوجه القصور في بروتوكول HTTP/1.1 بالاضافة الى الاحتياجات المتطورة في عصر تقنيات الويب الحديثة، وتتلخص العوامل الرئيسية فيما يلي:

١. يعد بروتوكول HTTP/1.1 بروتوكولا جيدا، الا ان أداءه قد تراجع في السنوات الاخيرة نتيجة التطورات السريعة في عالم الانترنت، فعند تطويره كان الهدف الرئيسي للويب هو عرض صفحات ويب ثابتة Static تحتوي على صور فقط، ومع مرور الوقت اصبح تحميل صفحة ويب مهمة تستهلك الكثير من الموارد، نظرا لان معظم صفحات الويب اصبحت ديناميكية بطبيعتها، مع وجود العديد من الصور والسكربتات التي تتحكم في سلوك الصفحة اثناء التشغيل، ومن هنا ظهرت الحاجة الى اعادة النظر في اداء البروتوكول وتحسينه.

٢. اضافة اي ميزة جديدة الى HTTP/1.1 لمواكبة احتياجات الويب المتزايدة جعلت البروتوكول اكثر تعقيدا، وهذا بدوره جعل تطبيقاته وبرمجياته اكثر عرضة للاخطاء والمشاكل البرمجية.

٣. يعتمد بروتوكول HTTP/1.1 على ASCII، ويشير الى American Standard COD for Information Interchange او الرموز المعيارية الامريكية لتبادل المعلومات، وهو طريقة لترميز الاحرف والنصوص في الحاسوب بحيث يمثل كل حرف او رقم او رمز برقم محدد، فعلى سبيل المثال الحرف A يمثل بالرقم ٦٥. وفي سياق HTTP/1.1 الرسائل مثل الطلبات والاستجابات ترسل كنص مكتوب بصيغة ASCII

وهذا يعني ان كل حرف يرسل بشكل منفصل كرمز وليس كبيانات ثنائية مضغوطة، لذلك عندما يرسل البروتوكول صفحات ويب تحتوي على الكثير من البيانات، يزيد حجم البيانات المرسلة Bandwidth ويزيد جهد المعالج لمعالجة هذا النص.

٤. لا توجد طريقة لجلب الموارد في وقت واحد بسبب الطبيعة التسلسلية الموروثة للبروتوكول.

واهم المزايا والاضافات والتحسينات في بروتوكول HTTP/2 هي:

١. البنية الثنائية Binary Property: تم تحسين اداء بروتوكول HTTP/2 عن طريق جعله بروتوكولا ثنائياً وهذا من شأنه ان يقلل من حجم البيانات لكنه يظل قويا بما يكفي لتحليلها، كما يجعل البرتوكول اقل عرضه للأخطاء واقل تعقيدًا.

٢. ضغط الرؤوس Header Compression: يستخدم بروتوكول HTTP/2 خاصية ضغط الرؤوس، لتقليل استخدام النطاق الترددي Bandwidth، كما تقلل هذه الخاصية من الحقول المكررة في الرؤوس، ويستخدم خوارزمية HPACK لعملية الضغط، والتي تعتمد على ترميز هوفمان Huffman Coding توفر هذه الميزة في HTTP/2 طبقة اضافية من التشفير نتيجة الضغط، بينما يعمل الاستخدام الالزامي لبرتوكول TLS كطبقة اولى من التشفير.

٣. تعدد الارسال Multiplexing: بموجب هذه الخاصية يمكن ارسال عدة طلبات على التوالي عبر نفس اتصال TCP، ويمكن استلام الاستجابات بأي ترتيب. تتيح هذه التقنية التخلص من الحاجة الى فتح عدة اتصالات بين العميل والخادم، كما يمكن ارسال الطلبات دون الانتظار لاستجابات الطلبات السابقة، ويساهم تعدد الارسال ايضا في تقليل الاعباء المتعلقة بالاتصال حيث يبقى اتصال TCP مفتوحا لنقل عدة طلبات.

٤. الدفع من الخادم Server Push: تستخدم هذه الخاصية لدفع الموارد الاضافية مع استجابات الطلبات التي تم تقديمها، فهي تتوقع ان المستخدم قد يحتاج الى معلومات اضافية تتعلق بالطلب الذي ارسله، وتقوم بارسالها بشكل استباقي مع الاستجابة. إحدى الفوائد المباشرة هي تحسين الأداء حيث إذا طلب العميل ملف index.html الذي يحتوي على روابط تشعبية لمجموعة من الموارد الأخرى، مثل الصور وملفات JavaScript وCSS، فيمكن للخادم أن يتوقع بشكل منطقي أن هذه الموارد ستُطلب لاحقًا، وبالتالي إرسالها إلى العميل مسبقًا.

٥. الأولوية Prioritization: تتيح هذه الخاصية للعميل تحديد اي الموارد اهم من غيرها، بحيث يتم التعامل معها وفق الترتيب المحدد. تؤثر أولوية التحميل على سرعة تحميل صفحة الويب، فعلى سبيل المثال، قد تعيق بعض الموارد، كملفات جافا سكريبت الكبيرة تحميل باقي الصفحة إذا تم تحميلها أولاً، بينما يمكن تحميل المزيد من أجزاء الصفحة دفعة واحدة إذا تم تحميل هذه الموارد التي تعيق التحميل في النهاية، بالإضافة إلى ذلك، يؤثر ترتيب تحميل موارد الصفحة على كيفية إدراك المستخدم لوقت تحميلها، فإذا تم تحميل المحتوى الخفي مثل ملف CSS او المحتوى الذي لا يراه المستخدم مباشرةً (مثل إعلانات البانر أسفل الصفحة) أولاً فسوف يعتقد المستخدم أن الصفحة لا تحمل على الإطلاق، أما إذا تم تحميل المحتوى الأكثر أهمية للمستخدم أولاً، مثل الصورة في أعلى الصفحة، فسوف يدرك المستخدم أن الصفحة تحمل بشكل أسرع.

٦. التحكم في التدفق Flow Control: تتيح هذه الخاصية الاستخدام المتوازن لجميع التدفقات streams مما يساهم في زيادة الاداء، ويهدف هذا التحكم الى تحسين استخدام الموارد، بحيث لا يحرم اي تدفق محدد من الموارد، مع ادارة العلاقات الصاعدة والهابطة وبشكل فعال ومرن.

:HTTP/3

يعد HTTP/3 احدث اصدار رئيسي من بروتوكول HTTP تم اصداره عام ٢٠٢٢ ويمكن لمتصفحات الويب والخوادم استخدامه لتحقيق تحسينات كبيرة في تجربة المستخدم من حيث الاداء والموثوقية والامان، ومن ابرز الفروق في HTTP/3 ان يعمل فوق بروتوكول نقل جديد يعرف باسم QUIC، والذي صمم ليكون سريعا ويدعم الانتقال السريع بين الشبكات المحتلفة، ويعتمد بروتوكول QUIC على بروتوكول مخططات بيانات المستخدم UDP بدلا من بروتوكول التحكم في الارسال TCP، وفضلا عن ذلك يجعل بروتوكول QUIC اتصال النقل في الطبقة الرابعة مستقلا عن تدفق عناوين IP في الطبقة الثالثة، مما يتيح الانتقال بين شبكات مختلفة دون حدوث انقطاع في الاتصال. ساعد هذا البروتوكول في معالجة اوجه القصور في HTTP/2 وذلك من خلال: تقليل تأثير فقدان الحزم، فعند عدم وصول احدى حزم البيانات الى وجهتها لن يؤدي ذلك بعد الان الى حجب جميع تدفقات المعلومات وهي المشكلة المعروفة حجب راس السطر Head-of-Line Blocking في TCP، وتسريع انشاء الاتصال، وتشفير اكثر شمولا حيث يعمل QUIC بتشفير افتراضي مما يجعل HTTP/3 اكثر امانا من HTTP/2، معالجة بطء الاداء عند انتقال الهواتف الذكية بين الشبكات، مثل التحول من شبكة Wi-Fi الى بيانات الهاتف الخلوي عند مغادرة المنزل او مكان العمل.

امن البروتوكول HTTPS/ TLS:

كان بروتوكول HTTP في الاصل بروتوكول يعتمد على النص العادي Plain-Text حيث ترسل رسائل HTTP عبر الانترنت بدون تشفير، وبالتالي يمكن لاي طرف يرى الرسالة اثناء مرورها ان يقرا محتواها، كما يشير اسم الانترنت فهو شبكة من اجهزة الحاسوب، وليس نظاما نقطة بنقطة، ولا يوفر الانترنت اي سيطرة على كيفية توجيه الرسائل وكمستخدم للانترنت لا يمكنك معرفة عدد الاطراف الاخرى التي ستتمكن من رؤية رسائلك اثناء انتقالها من مزود خدمة الانترنت (ISP) الى شركات الاتصالات والاطراف الاخرى، ونظرا لان HTTP يعتمد على النص العادي يمكن اعتراض الرسائل وقراءتها وحتى تعديلها اثناء مرورها. اما HTTPS فهو النسخة الامنة من HTTP حيث يقوم بتشفير الرسائل اثناء النقل باستخدام بروتوكول امان طبقة النقل Transport Layer Security (TLS)، والذي كان يعرف سابقا Secure Sockets Layer (SSL). ويضيف HTTPS ثلاثة مفاهيم مهمة على رسائل HTTP:

١. التشفير Encryption: تمنع قراءة الرسائل من قبل اطراف ثالثة اثناء النقل.

٢. النزاهة Integrity: يضمن عدم تعديل الرسائل اثناء النقل، حيث تُوقع الرسالة المشفرة بالكامل رقميا، ويتم التحقق من التوقيع بشكل تشفيري قبل فك التشفير.

٣. المصادقة Authentication: يضمن ان الخادم هو نفسه الذي ترغب بالتواصل معه.

تم بناء HTTPS على اساس بروتوكول HTTP وهو متوافقا معه تماما، يتم استضافته على منفذ مختلف افتراضيا (٤٤٣ بدل من المنفذ ٨٠ الخاص ببروتوكول HTTP القياسي)، كما انه يستخدم نظام عناوين URL مختلف (https:// بدلًا من http://)، ومع ذلك فانه لا يغير بشكل جوهري طريقة استخدام HTTP من حيث الصياغة او تنسيق الرسائل، باستثناء عملية التشفير وفك التشفير نفسه.

الخاتمة:

من خلال مراجعتنا لبروتوكول نقل النص التشعبي HTTP نستنتج انه البروتوكول قد مر بمراحل تطوير متعاقبة ادت الى ادخال تحسينات جوهرية في مجالات الاداء والسرعة والامان مما جعله اكثر قدرة على التكيف مع طبيعة الشبكات الحديثة ومتطلبات التطبيقات المتقدمة، وقد سعى كل اصدار جديد من بروتوكول HTTP الى معالجة اوجه القصور والمشكلات القائمة في الاصدارات السابقة، الامر الذي اسهم في توفير تجربة تصفح اكثر مرونة وكفاءة وموثوقية، كما اظهرت هذه التطورات الدور المحوري لبروتوكول HTTP في دعم التطبيقات الحديثة مثل الهواتف الذكية فضلا عن قدرته على التكامل مع بروتوكولات وتقنيات اخرى، ويعكس هذا المسار التطوري التزام المجتمع التقني المستمر بتحسين منظومة الاتصال عبر الانترنت بما يضمن اتصالات اسرع واكثر امانا واستقرارا، ويسهم في بناء بيئة رقمية ذكية قادرة على مواكبة التحديات المستقبلية.

المصادر:

1. Meenakshi Suresh, P. P. Amritha, Ashok Kumar Mohan. An Investigation on HTTP/2 Security, Journal of Cyber Security and Mobility, Vol. 7 1, 2018, pp. 161–180, P 163,167.
2. Naveen Tiwari, Security Analysis of HTTP/2 Protocol, Master of Science, Arizona State University, 2017, p4.
3. Parry Pollard, HTTP/2 in Action, New York, Manning Publications Co. 2019 P28, 30.
4. Roy Fielding, James Gettys, Jeffrey C. Mogul, Henrik F. Nielsen, Larry Masinter, Paul Leach, Timothy Berners-Lee, Hypertext Transfer Protocol -- HTTP/1.1 (RFC 2616), Network Working Group, 1999.
5. Thomson, Ed, Benfield, Ed, HTTP/2, Internet Engineering Task Force (IETF), 2022, P. 6
6. What is HTTP/2, access on https://bunny.net/academy/http/what-is-http2-protocol-and-how-does-it-work/
7. https://www.cloudflare.com/learning/performance/http2-vs-http1.1/