

HTTP:

هو اختصار ل Hyper Text Transfer Protocol

وهو Tcp Protocol يستخدم للاتصال ونقل المعلومات والبيانات الخاصة بالمواقع والتطبيقات والاتصال يتم بين حاجتين وهم browser و Server ويتكون

1-Request

2-Response

ويتم الدخول على التطبيقات عن طريق URL او ربط الخاص ب APP

URL:

هو اختصار ل Uniform Resource locator وهو الرابط الخاص بالصفحة المراد الوصول اليه.

شكل URL القديم

<http://images.google.com:80/Secrets/index.php?id=13378&message=hello#top>

Http:Scheme Protocol

images.google.com :Host

images:SubDomain

google:Second Level Domain

com:TLP

80:port

Secrets:Directory

Index.php:file

id=13378&message=hello:Parameters

top:Fragment

#note:

لا يتم كتم port في بعض احيانا اللي لو في port معين مستخدم لكن لو احنا بنستخدم Http يكون port 80 اما لو Https يبقى 443

شكل URL حصله تغير في مفهوم Directory File ويبقى عندا حاجة جديد اسمه Route وهو عبارة عن رابط الملفات بمسار معين في URL

<http://images.google.com:80/Secrets/index/1/hello/#top>

Secrets/index:Route

طريقة الوصول الي المواقع

1-نكتب URL في Browser او Domain بيحصل رحلة مع DNS وبعده مبيخلص بيعت ip

2- Browser يبدأ يتوصل مع Server عن طريق انه بيعت Request

3- Server بيرد ب Response

#note:

في عيب في بروتوكول HTTP انه بينقل البيانات من غير تشفير ودا بيخلي البيانات تسريب عن طريق حاجة اسمه MITM اختصار ل Man In The Middle ودا بيخلي الهاكر يشوف الدتا المبعوت بين user and Browser والحل هو بروتوكول HTTPS هو النسخة امنه لنقل البيانات من بروتوكول HTTP ويعمل علي Port 443

HTTP يوجد لديه نوعين من Headers :

Request Headers-1

Response Headers -2

Request Headers

Host :

يحتوي علي ال Domain او ip المرسل له Request

User-Agent:

يحتوي علي معلومات الخاصة بالمتصفح او الالة المستخدمة في ارسال Request ودا بتعرف منه هو فتح من موبايل او كمبيوتر .

Connection:

يحتوي علي حالة الاتصال بعد عمل Request وهو يحدد الاتصال هيفضل مفتوح ولا لا بعد Request

Referer:

يحتوي علي العنوان الذي من خلاله ارسال Request

Accept-language:

يستخدم من Client لتعريف السيرفر اللغة المدعومه

Accept-Encoding:

يستخدم من Client لتعريف السيرفر علي نوع الترميز المدعوم

Accept:

يستخدم من Client لتعريف السيرفر نوع الدتا المدعومة

Cookie:

يحتوي علي ملفات الارتباط بين المستخدم و السيرفر.

Content-Length:

يحتوي علي حجم الدتا الموجودة في Body

Content-Type:

يحتوي علي نوع الدتا المرسله من في Request

Response-Headers:

Date:

يحتوي علي الوقت الخاص بالسيرفر في وقت ارسال Request

Locartion:

يحتوي علي العنوان المحول اليه عند تحويل Request الي مسار اخر.

Set-Cookie:

يتم استخدامه لانشاء Cookie جديدة او التعديل عليه .

Server:

يحتوي علي معلومات السيرفر و اصداره

HTTP-Methods:

GET:

تستخدم لطلب الصفحات عن طريق URL وترسل معها مدخلات المستخدم في [user-input] URL

POST:

تستخدم في ارسال البيانات عن طريق Request Body

HEAD:

شبيهة بال GET ولكن Response بيرجع بدون Body

OPTIONS:

تستخدم لمعرفة METHODS المتاحة

PUT:

شبيهة بال POST وتستخدم لانشاء بيانات جديدة

PATCH:

تستخدم لتعديل البيانات

DELETE:

تستخدم لحذف البيانات

#note:

GET و POST يستخدموا بشكل اساس من Web App لكن DELETE او PUT او PATCH اللي لم APP بيكون شغل ب API

Request Codes:

-1XX

اكود خاصة باعطاء معلومات والعمل علي Request

-2XX

اكود تدل علي قبول Request بارجع بالدتا المطلوبة

-3XX

الاكود تدل علي التحويل الي مكان اخر

-4XX

الاكود تدل علي الخطاء الخاصة بال Client

-5XX

الاكود تدل علي الخطاء الخاصة بالسيرفر

-Response Common Codes :

-200 Ok

يدل علي نجاح ال Request ويعود بالدتا المطلوبة .

-302 Found

تحديد Request الي مكان اخر

-400 Bad Request

وجود مشكلة في Request

-403 Forbidden

عدم وجود صلاحية

-404 Not Found

عدم وجود الصفحة او المكان المطلوب

-500 Internal Server Error

وجود خطأ من السيرفر

Scripting Programming Language:

-لغة الكمبيوتر الاساسه هي لغة الاله تتالف من 010101010 ولا تعريف اله غير ذلك لكن صعب جدا التعامل مع هذا اللغة مما ادي الي ظهر لغات عاليه المستوي تسمى High Level Language يمكن قراءة وفهم بشكل اساس اسهل من قبل المطورون لكن اله لا تفهم اللي 0 و 1 لذلك يتم انشاء اليه تسمى Compiler تقوم بتحويل الاكواد الكتوبه باللغات المستوي العالي الي اللغة اله

Compiler:

هو برنامج يقوم بتحويل او ترجمه الاكواد البرمجه الي لغة الة طريقة عمل Compiler ياخذ الكود كلمين بعد انتهاء من كتابيه وتحويله دفعة واحدة الي ملف يستطيع الكمبيوتر فهمه وقراءة وتنفذه.

Scripting language:

هي اللغة النصيه اللي بتحتوي على الاوامر والتوجهات لتنفيذه في البرنامج الذي كتب باستخدام Programming languages و scripting language هو الفئه الفرعيه من اللغات التي يتم استخدامها في توجيه او التحكم في برنامج اخر للغات البرمجه النصيه التي تتمتع بدين مكيه وسهوله اكثر في عملها وايضا في تعلمها مما يميز scripting language هي لا تحتاج إلى compiler ولكن تحتاج إلى interpreter.

Interpreter:

هو برنامج يقوم بوظيفه ترجمه الكود الى تعليمات لتنفيذه في برنامج معين طريقة عمل interpreter تختلف عن compiler حيث انه يقوم بترجمه كل السطر على الفور ويساعده مبرمجين على العثور على الاخطاء بشكل اسرع لانه يقوم بترجمه كل سطر ويؤدي الى التعامل مع الكود المبرمج بشكل اسهل.

Programming language

تحتاج الي compiler الذي يحول الكود الى اللغة الاله واللغات المستخدمه في بناء المشاريع الكبيره التي تعمل بشكل مباشر مع الاله

Scripting language

تحتاج الي interpreter وتكون هذه اللغات ليست مخصصه لكتاب التطبيقات من الصفر ولكن تهدف الى تكمله مكونات في برامج الكمبيوتر وغالبا ما تستخدم لغات البرمجه النصيه لتوسيع مميزات برنامج معين ولكنه نادرا ما تستخدم في خوارزميات معقده و هياكل بيانات.

Ex:

متصفح chrome هو عباره عن برنامج سطح مكتب مبرمج بلغه c++ هدفه اظهار صفحات الانترنت لكن تظن ان انه تحتاج الى متصفح كروم خسر كل صفحه نريد الدخول اليه لذلك يتم تطوير اللغات برمجه نصيه مثل js و php وغيرها لعرض صفحات الانترنت على المتصفح وفي المتصفح واصله بعض المميزات بدون الرجوع الى مؤسسي برنامج Google chrome لكن مؤسسي برنامج Google chrome قام بعمل interpreter واتلقوا عليه اسم اكماسكريبت لتفسير الاكواد المكتوبه بلغه الجافا سكريبت وعرضها على المتصفح لذلك اذا اردنا انشاء صفحه انترنت نكتب سكريبت او كود باستخدام اللغات النصيه وتنفيذه في المتصفح.

#note:

هذا هو الفرق بين الاسماء الصحيحه

Compiled language

Interpreted language