دانشکده مهندسی کامپیوتر

گروه مهندسی نرم افزار

عنوان پروژه:

تعیین سیستم عامل و سرویس دهنده وب از طریق پروتکل های شبکه

پیش نویس اول پروژه کلاسی **شماره ۶** درس کامپایلر پیشرفته

دانشجو:

مرتضی ذاکری

استاد:

دکتر سعید پارسا

پاییز ۱۳۹۵

فهرست مطالب

فهرست شکل ها

۱ مقدمه

مسئله تعیین اجزای شبکه و سیستم های عامل و نرم افزار های نصب شده بر روی آن از مسائل ا صلی در انجام ت ست های نفوذ و امنیت ا ست. اگر سی ستم را شنا سایی کنیم م سیرهای ما متمرکز بر روی آن سیستم خاص خواهد. بنابراین اولین گام تعیین این اجزا است.

در این گزارش به دنبال راه حلی برای یافتن سرویس دهنده وب و سیستم عامل سروری که سرویس دهنده روی آن اجرا می شود، هستیم. برای این منظور بایستی از ساختار پروتکل هایی که روی شبکه ا ستفاده می شوند آگاهی کامل دا شته با شیم. در این جا یک راه حل استفاده از پروتکل انتقال ابر متن یا همان HTTP است که در وب استفاده می شود. بنابراین ابتدا ساختار این پروتکل را شرح می دهیم و سپس برنامه ای را پیاده سازی می کنیم که اطلاعات مورد نیاز را از این پروتکل بیرون می کشد.

۲ پروتکل HTTP

یک پروتکل لایه کاربرد (Application Layer) برای سیستم‌های توزیع شده می‌با شد. پروتکل انتقال ابرمتن یک پروتکل درخوا ست و پا سخ در مدل کلاینت–سرور می‌با شد. برای مثال یک مرورگر وب می تواند یک کلاینت و نرم‌افزار موجود بر روی سرویس‌دهنده وب سایت، یک سرور باشد. شروع این پروتکل از طرف کلاینت است که با ارسال یک درخواست HTTP به سمت سرور گفت وگو را آغاز می‌کند. سرور بر اساس درخواست ارسالی یا منبعی مانند یک فایل را در اختیار کلاینت می‌گذارد و یا عملیات خاصی را انجام می‌دهد. نتیجه این عملِ سرور در بسته پاسخ HTTP برای کلاینت ارسال می‌شود. بسته پاسخ شامل اطلاعات وضعیت و احتمالاً محتویات منبع درخواست شده می‌باشد.

۲ـ۱ جلسه

به دنباله‌ای از درخواست‌ها و پاسخ ها جلسه[۱] گفته می‌شود. کلاینت با ایجاد یک اتصال TCP بر روی یک درگاه از پیش تعیین شـده بر روی سـرور (معمولا درگاه شـماره ۸۰)، جلسـه را آغاز می‌کند. سـرور وب همواره بر روی درگاه در انتظار درخواسـت‌های کلاینت‌ها می‌باشـد. بعد از دریافت درخواست ارسال شده، سرور با ارسال یک خط وضعیت[۲] و بدنه، پاسخ کلاینت را به او بازمی‌گرداند. بدنه بسته پاسخ معمولاً حاوی منبع درخواست شده است؛ با این حال از آن برای ار سال خطا و اطلاعات دیگر نیز ا ستفاده می شود. یک نمونه از خط و ضعیت در پا سخ به یک درخواست مجاز به صورت زیر است:

HTTP/1.1 200 OK

۲ـ۲ روش‌های درخواست

HTTP روش[۳] هایی را برای ایجاد یک درخوا ست در نظر گرفته ا ست که هر کدام از آن‌ها باعث انجام عمل خاص **در سمت سرور** می‌شـوند. نسـخه ۱/۰روش‌های درخواسـت GET، POST و HEAD را دارا بود. در نسخه۱/۱پنج روش جدید افزوده شد OPTIONS، PUT، DELETE، TRACE و CONNECT. از آنجایی که عملکرد این روش‌ها به طور کامل تعریف و شـرح داده شـده اسـت، لذا تمامی مرورگر ها و سرور ها به راحتی می‌توانند این روش‌ها را پیاده‌سازی و استفاده نمایند. اگر روشـی برای سـرور تعریف نشـده باشـد، با آن به عنوان یک روش غیر ایمن برخورد خواهد کرد. در ادامه به طور خلا صه روش های درخوا ست موجود در ا ستاندارد ا صلی پروتکل را شرح می دهیم.

- GET

[۱] Session
[۲] Status Line
[۳] Method

درخواست نمایش منبع درخواست داده شده را می‌دهد (این منبع معمولا یک فایل یا پرونده می‌باشد) این روش فقط اطلاعات را از سرور دریافت[۴] می‌کند و نباید هیچ تاثیری بر روی منابع سرور بگذارد.

- HEAD

این روش دقیقا مانند روش GET عمل می‌کند با این تفاوت که بدنه پاسخ را نمی‌خواهد. از این روش برای به‌دست‌آوردن فرا داده های موجود در سرآیند[۵] استفاده می‌شود. یکی از استفاده های رایج این نوع درخواست، بررسی تغییر یافتن یک منبع است.

- POST

در این روش به همراه بسته درخواست اطلاعاتی نیز فرستاده می‌شود. سرور با توجه به نشانی وب (URL) درخواست شده و اطلاعات ارسال شده، منبع مورد نظر را در بسته پاسخ باز می‌گرداند. این اطلاعات ارسالی می‌تواند نام کاربری و کلمه عبور، یک دیدگاه بر روی یک مطلب و یا اطلاعات هر فرم دیگری که توسط کاربر وارد شده است، باشد.

- PUT

در این روش منبعی به همراه بسته درخواست ارسال شده و از سرور تقاضا می شود که این منبع را در آدرس موجود در بسته بارگذاری کند. اگر در محل درخواست شده قبلا منبع دیگری قرار داشته باشد، منبع جدید جایگزین خواهد شد.

- DELETE

از سرور درخواست می‌کند که آدرس فرستاده شده را حذف نماید.

- TRACE

در این روش سرور اطلاعات ارسال شده را عیناً به کلاینت باز می‌گرداند. این روش برای بررسی تغییراتی که واسطهای شبکه بر روی بسته می‌گذارند، استفاده می‌شود.

- OPTIONS

---

[۴] Get

[۵] Header

از سرور تقا ضا می‌کند تا روش‌های درخوا ست موجود[۶] برای ن شانی فر ستاده شده را اعلام نماید. برای گرفتن تمامی روش‌های درخواســت قابل اجرا بر روی ســرور می‌توان از نشــانی '*' استفاده کرد.

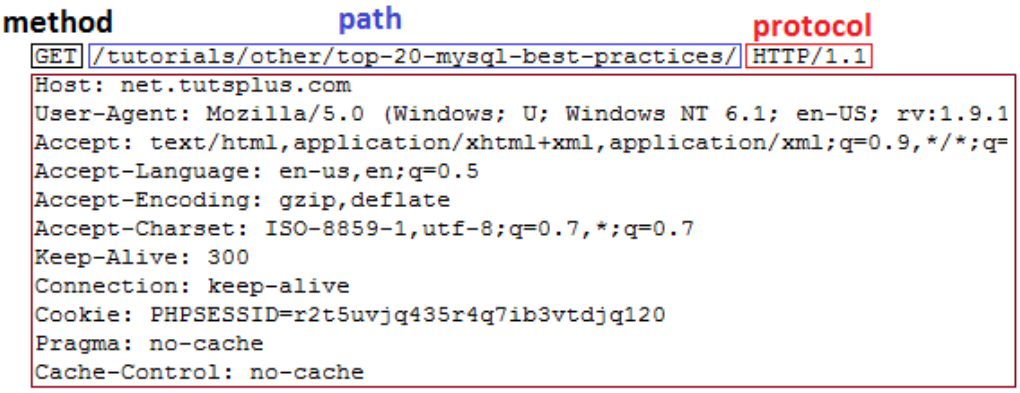
- CONNECT

بســته پروتکل ابرمتن را به یک تونل TCP/IP تبدیل می‌کند. این عمل معمولاً برای برقراری ارتباط امن (HTTPS) بر روی یک پراکسی سرور ناامن استفاده می‌شود.

سرورهای وب موظف هستند حداقل روش‌های GET و HEAD را پیاده‌سازی نمایند.

۲ـ۳ سرآیند HTTP

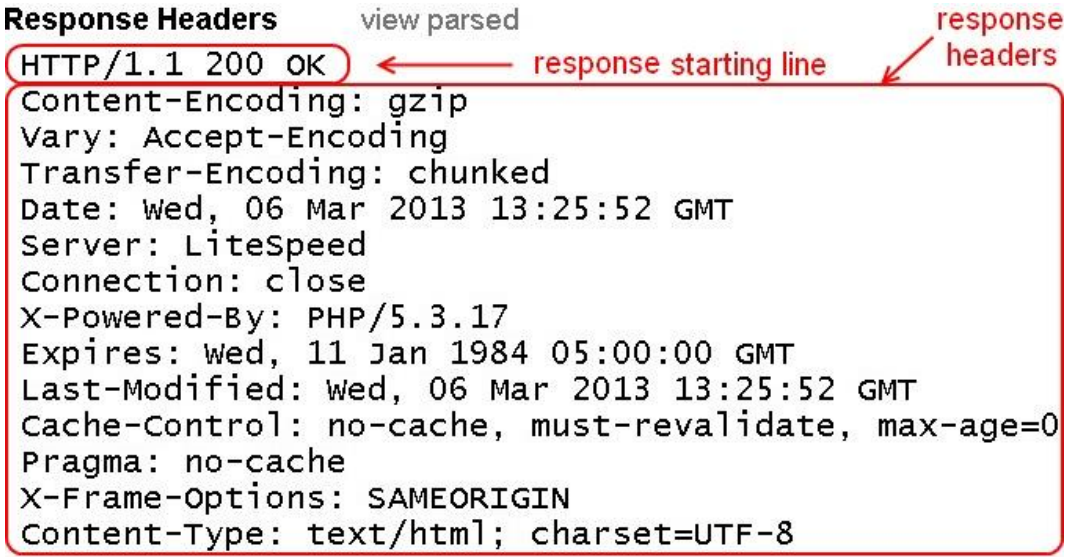
ســرایند حاوی فیلدهایی اســت که پارامترهای یک ارتباط پروتکل را مشــخص و مقداردهی می‌کنند. فیلدهای سرآیند بعد از خط وضعیت (اولین خط هر پیام) ارسال می‌شوند. این فیلدها به صورت متن ساده بوده و دارای یک نام یا کلید و یک یا چند مقدار ه ستند که با علامت دو نقطه (:) از هم جدا می‌شوند. هر خط سرآیند می‌تواند حاوی یک فیلد سرآیند باشد. در واقع در پایان هر فیلد ســرآیند باید حروف CR و LF قرار بگیرند. این حروف از حروف کنترلی در رایانش هستند که باعث رفتن به خط بعد می شوند. شکل ۲-۱نمونه ای از سرایند درخواست HTTP را نشان می دهد.



شکل ۲-۱ نمونه ای از سرایند درخواست HTTP

[۶] Available Request Methods

شکل ۲-۲ نیز نمونه ای از سرایند پاسخ HTTP را نشان می دهد.



شکل ۲-۲ نمونه ای از فیلدهای سرایند پاسخ HTTP

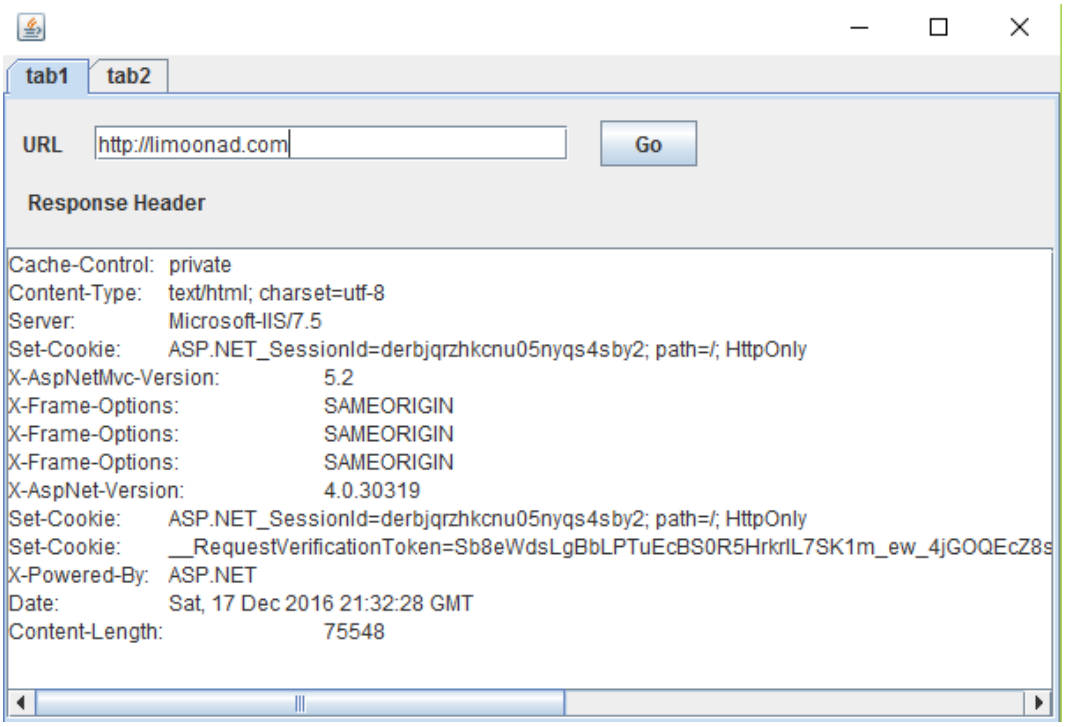
۳ برنامه HTTPTest

می توان از سرایند پاسخ HTTP که در پاسخ به یک Request به کلاینت ارسال می شوند، اطلاعاتی را پیرامون سرویس دهنده وب و نیز سیستم عامل در حال اجرا روی سرور بیرون کشید. این اطلاعات در تست های نفوذ و امنیت به کار می آید. یک روش پویش فایل سرایند توسط یک برنامه نوشته شده است. در برنامه HTTPTest که ما برای این منظور توسعه داده ایم، با استفاده از کتابخانه Apache HTTPClient اقدام به ارسال یک درخواست GET یا Head به یک سرور می کنیم و سپس اطلاعات پاسخ دریافت شده را پویش کرده و به تفکیک نام : مقدار نشان می دهیم. شکل ۳-۱ تصویری از محیط اجرایی برنامه را نشان می دهد.

در این اجرا می خواهیم اطلاعات مربوط به سیستم عامل و سرویس دهنده وب تارنمای http://limoonad.com را به دست آوریم. بدین منظور در قسمت URL این اطلاعات را وارد نموده و سپس دکمه Go را می زنیم. برخی از اطلاعات مفید به دست آمده عبارت اند از:

- سرویس دهنده وب Microsoft-IIS 7.5
- زبان برنامه نویسی وب سایت: ASP.NET

- سیستم عامل میزبان: احتمال ۹۹٪ Windows Server
- و اطلاعات کوکی های موجود.



شکل ۳-۱محیط اجرایی برنامه HTTPTest

۴ کارهای آتی

این برنامه بـ سیار ابتدایی ا ست و باید تو سعه پیدا کند. در نـ سخه های آتی ما باید توانایی های تشخیص اجزای مختلف را به برنامه اضافه کنیم. به علاوه برای برخی از سایت ها مقادیر برخی از فیلدها هنگام ایجاد پا سخ پر نمی شوند که این بـستگی به تنظیمات Web Server دارد. باید به دنبال راه حل هایی برای این مورد و موارد مشابه دیگر باشیم.

۵ منابع و ماخذ