تست سرعت و نفوذ وب سایت

nanoclub.ir

شرکت نوآوران و پژوهشگران نادین

تابستان ۱۳۹۹

فهرست مطالب

مقدمه ۵

بخش ۱ - تست سرعت ۶

۱-۱- تست MySQL ۶

۱-۱-۱- سناریوی انجام تست ۶

۱-۱-۲- نتایج تست.....۶

۱-۱-۳- راهکارهای پیشنهادی ۶

۱-۲- تست Apache..... ۷

۱-۲-۱- سناریوی انجام تست ۷

۱-۲-۲- نتایج تست ۷

۱-۲-۳- راهکارهای پیشنهادی..... ۸

۱-۳- تست پهنای باند..... ۸

۱-۳-۱- سناریوی انجام تست..... ۸

۱-۳-۲- نتایج تست..... ۸

۱-۳-۳- راهکارهای پیشنهادی ۸

بخش ۲ - تست نفوذ ۹

۲-۱- تست Server Version Banner (SVB)..... ۹

۲-۱-۱- سناریوی انجام تست ۹

۲-۱-۲- نتایج تست ۹

۲-۱-۳- راهکارهای پیشنهادی.....۱۰

۲-۲- تست PHP ۱۰

۲-۲-۱- سناریوی انجام تست ۱۰

۲-۲-۲- نتایج تست.....۱۰

۲-۲-۳- راهکارهای پیشنهادی ۱۰

۲-۳- تست D/DOS.....۱۰

۱-۳-۲- سناریوی انجام تست ۱۰
۲-۳-۲- نتایج تست ۱۱
۳-۳-۲- راهکارهای پیشنهادی ۱۱
۴-۲- تست HTTP Cookies ۱۱
۱-۴-۲- سناریوی انجام تست ۱۱
۲-۴-۲- نتایج تست ۱۱
۳-۴-۲- راهکارهای پیشنهادی ۱۱
۵-۲- تست شناسایی تکنولوژی استفاده شده در وب سایت ۱۲
۱-۵-۲- سناریوی انجام تست ۱۲
۲-۵-۲- نتایج تست ۱۲
۳-۵-۲- راهکارهای پیشنهادی ۱۲
۶-۲- تست HTTP Security Headers ۱۲
۱-۶-۲- سناریوی انجام تست ۱۲
۲-۶-۲- نتایج تست ۱۲
۳-۶-۲- راهکارهای پیشنهادی ۱۳
۷-۲- تست Robots.txt ۱۳
۱-۷-۲- سناریوی انجام تست ۱۳
۲-۷-۲- نتایج تست ۱۴
۳-۷-۲- راهکارهای پیشنهادی ۱۴
۸-۲- تست SlowLoris ۱۴
۱-۸-۲- سناریوی انجام تست ۱۴
۲-۸-۲- نتایج تست ۱۴
۳-۸-۲- راهکارهای پیشنهادی ۱۴
۹-۲- تست Session ۱۴
۱-۹-۲- سناریوی انجام تست ۱۴

۲-۹-۲- نتایج تست ۱۴
۳-۹-۲- راهکارهای پیشنهادی ۱۵
۱۰-۲- تست HSTS ۱۵
۱-۱۰-۲- سناریوی انجام تست ۱۵
۲-۱۰-۲- نتایج تست ۱۵
۳-۱۰-۲- راهکارهای پیشنهادی ۱۵
۱۱-۲- تست کتابخانههای JS ۱۵
۱-۱۱-۲- سناریوی انجام تست ۱۵
۲-۱۱-۲- نتایج تست ۱۵
۳-۱۱-۲- راهکارهای پیشنهادی ۱۶
۱۲-۲- تست SQL Injection ۱۶
۱-۱۲-۲- سناریوی انجام تست ۱۶
۲-۱۲-۲- نتایج تست ۱۶
۳-۱۲-۲- راهکارهای پیشنهادی ۱۶
۱۳-۲- تست XHR ۱۶
۱-۱۳-۲- سناریوی انجام تست ۱۶
۲-۱۳-۲- نتایج تست ۱۶
۱۴-۱۳-۲- راهکارهای پیشنهادی ۱۶
۱۴-۲- تست خطاهای سایت ۱۷
۱-۱۴-۲- سناریوی انجام تست ۱۷
۲-۱۴-۲- نتایج تست ۱۷
۳-۱۴-۲- راهکارهای پیشنهادی ۱۷
۱۵-۲- تست POST/GET ۱۷
۲-۱۵-۲- نتایج تست ۱۷
۳-۱۵-۲- راهکارهای پیشنهادی ۱۷

۱۶-۲- تست دسترسی به پوشهها ۱۸
۱-۱۶-۲- سناریوی انجام تست ۱۸
۲-۱۶-۲- نتایج تست ۱۸
۳-۱۶-۲- راهکارهای پیشنهادی ۱۸
۱۷-۲- تست دسترسی به MySQL ۱۸
۱-۱۷-۲- سناریوی انجام تست ۱۸
۲-۱۷-۲- نتایج تست ۱۸
۳-۱۷-۲- راهکارهای پیشنهادی ۱۸
بخش ۳ - اولویت پیشنهادی برای بهبود وب سایت ۱۹
۱-۳- اولویتهای پیشنهادی برای بهبود سرعت و امنیت وب سایت ۱۹

مقدمه

با گسترش تکنولوژی‌های حوزه فناوری اطلاعات، ارائه خدمات به کاربران در بستر وب به سرعت رشد یافته است. استفاده از این بستر، موجب سهولت کاربران در دسترسی به خدمات، کاهش مراجعات حضوری و هزینه‌ها می‌شود. با این وجود، همواره کاربرانی وجود دارند که قصد سوء استفاه از خدمات ارایه شده را دارند. بنابراین، توجه به امنیت و سرعت ارایه خدمات، امری بدیهی و مهم تلقی می‌شود.

در این گزارش، وب سایت باشگاه نانو به آدرس nanoclub.ir در دو جنبه سرعت سرویس‌دهی به کاربران و امنیت بررسی شده است. در حوزه سرعت، سه تست سرعت انجام شده که در بخش ۱ توضیح داده شده است. در حوزه امنیت، ۱۷ تست امنیتی مهم انجام شده است. این تست‌ها در بخش ۲ توضیح داده شده است. برای هر تست، سناریوی انجام تست، نتایج تست و راهکارهای پیشنهادی برای بهبود سرعت/امنیت توضیح داده شده است. در نهایت در بخش ۳، اولویت‌های پیشنهادی برای انجام کارها به منظور بهبود سرعت و امنیت وب سایت، آمده است. امید است این گزارش بتواند کمک شایانی به بهبود کیفیت سرویس‌دهی و امنیت این وب سایت کند.

بخش ۱ - تست سرعت

۱-۱- تست MySQL

۱-۱-۱- سناریوی انجام تست

بررسی منابع سیستم، در حالت کار با وب سایت و کدهای سرور و ساختار MySQL

۱-۱-۲- نتایج تست

در ابتدا نیاز است تا وضعیت منابع سرور هنگامی که هیچ کاربری روی سرور فعالیت ندارد، توجه کرد. در این شرایط، بین ۲۰ الی ۴۰ درصد CPU روی سرور به MySQL اختصاص داده شده است. این وضعیت در حالتی که تنها یک کاربر مشغول فعالیت می‌شود، به سرعت افزایش پیدا می‌کند و دیگر هسته‌های پردازنده را نیز درگیر می‌کند. از طرف دیگر ارسال یک داده ساده و یا یک ارتباط به دیتابیس، فرآیندی کُند بوده و منابع را به مرور مشغول‌تر می‌کند. از طریق بررسی منابع سیستم، اینگونه به نظر می‌رسد که فرآیندهایی روی دیتابیس به صورت مداوم در حال اجرا هستند و همچنین بکاپ گیری نامنظمی در حال رخ‌دادن است. این فرآیندها مطمئناً بدون فعالیت کاربر سمت وب‌سایت رخ می‌دهد و درصورتی‌که کاربری فعالیت خود را آغاز کند، وضعیت به گلوگاه‌های CPU نزدیکتر می‌شود. این اتفاق شانس و اولویت پاسخ سریع و تعامل با کاربر را کمتر کرده و سیستم را برای یک حمله D/Dos و از دسترس خارج ساختن خود به خودی سیستم توسط یک هکر آماده می‌کند.

۱-۱-۳- راهکارهای پیشنهادی

در گام اول نیاز است که نحوه ارتباط با دیتابیس و ساختار استفاده در سمت وب سایت را بهینه شود. به دلیل استفاده از فریم ورک لاراول و پکیج‌های پیش‌فرض خود لاراول، این ارتباطات با یک لایه سوم مواجه می‌شوند که فرآیند تبدیل به لایه دوم و بالاخره لایه اصلی ارتباط با دیتابیس را کند می‌کند. پیشنهاد می‌شود در سمت وب‌سایت حداقل در شرایط پیک (آزمون همگانی یا المپیاد که افراد زیادی شرکت می‌کنند)، ارتباطات را به حداقل رسانده و تنها از یک لایه آن هم Stored Procedure ها استفاده شود. برای اطلاع از نحوه ارتباط با این متد در لاراول به این لینک مراجعه کنید.

در گام دوم، نیاز است تا ساختار ساختار جداول در پایگاه داده نرمال سازی شود. این نرمال سازی، بهتر است تا سطح BCNF انجام شود. سپس، با استفاده از ایجاد index بر روی ستون‌هایی از جداول که جستجوی زیادی بر روی آن‌ها انجام می‌شود و همچنین

استفاده از VIEW برای کوئری‌های پرتکرار، می‌توان سرعت دسترسی به اطلاعات را افزایش داد[1]. این روش‌ها زمان دسترسی به داده‌ها را کاهش داده و تعامل کاربران با وب سایت را بهتر و سریع‌تر می‌کند. از طرف دیگر، منابع اشغال شده در حین ارتباط در سمت سرور را کاهش می‌دهد و اطمینان و صحت ارتباطات را حفظ می‌کند. برای اطلاع از این روش‌ها، می‌توانید به این لینک مراجعه کنید.

در گام سوم، نیاز است که فرآیندهایی نظیر Trigger که به صورت اتوماتیک و دوره‌ای روی سرور کار می‌کنند را کاهش دهید. به این منظور بایستی لیست فرآیندهای در حال اجرا توسط MySQL را بررسی کرده و یک پروتکل و سیاست زمانبندی درست برای اعمالی نظیر بکاپ و ... تهیه کنید. حتما یک مانیتورینگ مناسب برای MySQL فراهم شود تا در این راستا به سرعت به اهداف مطلوب دست پیدا کنید. می‌توانید برای مانیتور کردن دیتابیس از لینک۱ و لینک۲ و لینک۳ استفاده کنید.

در گام چهارم به عنوان آخرین گام، می‌توانید منابع CPU و حافظه را بیشتر کنید. توصیه می‌شود مقدار CPU برای آنلاین بودن هم زمان ۱۰۰۰۰ کاربر را به ۲۴ هسته و مقدار حافظه را به ۱۲ تا ۱۸ گیگابایت ارتقا دهید. این ارتقا می‌تواند یک حریم امن برای هر گونه اتفاق پیش بینی نشده باشد. در حال حاضر این مقدار منابع برای هم‌زمانی تنها ۵۰۰ تا ۱۰۰۰ کاربر مناسب است.

۲-۱- تست Apache

۲-۱-۱- سناریوی انجام تست

بررسی منابع سیستم، در حالت کار با وب سایت و تنظیمات Apache

۲-۱-۲- نتایج تست

یک ثبت نام ساده و ارسال فرم (POST) به خصوص زمانی که به یک بخش سوم مانند ارسال پیامک و همچنین MySQL وابستگی دارد، به طور غیرمعمولی زمان‌بر است. زیرا تاخیر پاسخ در بخش سوم، علاوه بر تاخیر در زمان ارتباط در لایه‌ها، زمان پرسش و پاسخ میان سرور و کاربر را افزایش می‌دهد. مصرف آنچنانی در CPU و حافظه دیده نمی‌شود ولی با وجود تاخیرهای متوالی ممکن است وب سایت از دسترس کاربران سایت خارج شود (Timeout در سمت Apache). به نظر می‌رسد در قسمت بارگیری فایل مشکل عمده و تاخیر زیادی وجود دارد. زیرا پس از اتمام فرآیند در سمت کاربر، همچنان این فرایند در سمت سرور قسمت عمده‌ای از CPU را اشغال می‌کند. فرآیندهایی نظیر Cron-Job روی سرور دیده نمی‌شود، ولی در صورت وجود، می‌توان در زمان پیک آن‌ها را غیر فعال کرد. آزمون برای یک کاربر هنگام ثبت سوالات کمی کند است. این کندی می‌تواند بر اثر نقش ارتباطی MySQL باشد (بخش ۱-۱). در باقی صفحات نیز کندی خاصی مشاهده می‌شود. به خصوص در مورد ثبت نام، که حدود ۲۰ ثانیه برای یک کاربر زمان درخواست طول می‌کشد. نرم افزار gzip منبع زیادی را روی سرور اشغال می‌کند. بهتر است در زمان پیک آن را غیر فعال کرد.

[1] درصورت وجود محدودیت زمانی، می‌توانید در این مرحله نرمال سازی، indexing و ایجاد view-ها را بر روی جداولی که دسترسی زیادی به آن‌ها وجود دارد، انجام دهید و سپس در زمان مقتضی فرایندهای فوق را تکمیل نمایید.

۳-۲-۱- راهکارهای پیشنهادی

در زمان آزمون‌های همگانی، اگر قرار است ارسال پاسخ سوالات یکجا باشد، قسمت دیتابیس و Post در سرور اصلاح شود. با توجه به اینکه با مدل MVVM و با درخواست‌های XHR. هم‌زمانی ارسال‌های یکجا، باعث ایجاد گلوگاه می‌شود، یک پیشنهاد، ارسال پاسخ‌ها بصورت مرحله به مرحله برای جلوگیری از ایجاد این گلوگاه است. همچنین طبق گام چهارم در راه کار پیشنهادی بخش ۱-۳-۱ با افزایش منابع، می‌توان از ایجاد این گلوگاه جلوگیری کرد. حتما برای بررسی عملکرد Apache قسمت مانیتورینگی فراهم شود تا موارد بیشتر و مرتبط و پیش‌بینی‌های قبل و بعد و حین گلوگاه تشخیص داده شود. برای اطلاعات بیشتر به لینک۱ و لینک۲ مراجعه کنید.

۳-۱- تست پهنای باند

۳-۱-۱- سناریوی انجام تست

ارسال و دریافت داده‌ها از سمت وب

۳-۱-۲- نتایج تست

ارسال و دریافت فرم‌ها و مطالب چندان از پهنای باند سرور نمی‌کاهد، اما هم‌زمانی در ارتباطات ممکن است به گلوگاه منجر شود. ارسال و دریافت یک فرم ساده در آزمون طبق موارد پیشنهادی در بخش ۳-۲-۱ اگر یکجا باشد، حدود ۱ مگابیت ارسال و ۱ مگابیت دریافت برای هر کاربر نیاز دارد؛ اما بصورت مرحله‌ای این مقدار کمتر می‌شود.

۳-۱-۳- راهکارهای پیشنهادی

میزان دریافت و ارسال داده‌ها برای کاربران به دلیل Text بودن داده‌ها زیاد نیست. با این وجود، پیشنهادات زیر برای اطمینان خاطر در زمان پیک ارائه می‌شود:

۱- دریافت و ارسال سرور را از طریق فایروال به یک عدد خاص محدود کنید.

۲- در صورت مرحله‌ای بودن فرآیند ارسال داده‌ها (نه یکجا، مانند پاسخ‌های آزمون)؛ سرعت دریافت را ۲ مگابیت بر ثانیه و سرعت ارسال را ۱ مگابیت بر ثانیه برای هر کاربر در نظر بگیرید.

۳- حداقل ۲۰۰ مگابیت پهنای باند ارسال و دریافت برای سرور فراهم کنید. این پهنای باند، برای ارتباط هم زمان در یک لحظه ۱۵۰ کاربر کافی است.

۴- بعد از شروع یک هم‌زمانی نظیر المپیاد، دریافت فایل‌های حجیم را در صورت وجود متوقف کنید تا کاربران با دانلود یکجا و یا یک هکر با سوء استفاده از وضعیت امتحانی، پهنای باند را اشغال نکند.

۵- حتما مانیتورینگ مناسب برای پهنای باند نظیر bmon یا غیره روی سرور داشته باشید (برای اطلاع بیشتر به این لینک مراجعه کنید).

بخش ۲ - تست نفوذ

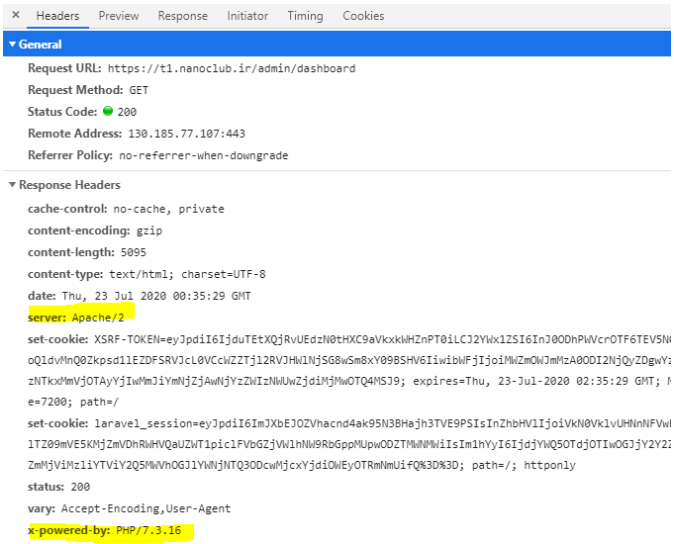
۲-۱- تست Server Version Banner (SVB)

۲-۱-۱- سناریوی انجام تست

اسکن هدر پاسخ‌های یک HttpRequest

۲-۱-۲- نتایج تست

طبق موارد مشخص شده در شکل ۲-۱، موارد هایلایت شده، به هکر اطلاعات کافی درخصوص سرور را می‌دهد و یک نمای انتزاعی از نسخه‌های نرم افزاری مانند PHP را روی سرور برای او ترسیم می‌کند. این جزو اولین مواردی است که برای امنیت سایت باید در نظر گرفته شود، زیرا فاش شدن نسخه‌ها به هکرها برای سرعت بخشیدن به روند شناسایی کمک می‌کند. ممکن است در صورت وجود باگ‌های امنیتی در نسخه مشخص شده و عدم به روز شدن آن توسط مدیر سیستم، هکر با استفاده از حفره‌ها به سیستم نفوذ کند.



شکل ۲-۱: اطلاعات سرور

۳-۱-۲- راهکارهای پیشنهادی

نیاز است که پیکربندی پیش فرض سرور وب (Apache) خود را تغییر دهید. با این روند که به تنظیمات وب سرور خود رجوع کرده و فایل httpd.conf را باز کنید. دو متغیر ServerTokens و ServerSignature را یافته و مقادیر آن را به صورت زیر تغییر دهید:

```
ServerTokens Prod
ServerSignature Off
```

سپس سرور خود را ریستارت کنید. ServerSignature اطلاعات نسخه را از صفحه ایجاد شده توسط Apache حذف می‌کند و ServerTokens هدر را فقط به نوع وب سرور، یعنی Apache، تغییر می‌دهد.

۲-۲- تست PHP

۱-۲-۲- سناریوی انجام تست

بررسی نسخه نصب شده روی سرور از طریق SSH (البته از طریق اسکن SVB نیز می‌توان نتیجه گرفت)

۲-۲-۲- نتایج تست

نسخه PHP سرور7.3.16 است. طبق CVE-2019-11048 گزارش شده، در نسخه های PHP 7.3.x زیر 7.3.18، هنگامی که آپلود فایل HTTP مجاز است، ایجاد کردن نام‌های نامحدود و نام‌های فیلد بسیار طولانی، می‌تواند موتور PHP را به تلاش برای تخصیص فضای زیاد روی حافظه سوق دهد. این ذخیره سازی، موجب سرریز فضای حافظه و توقف پردازش درخواست می‌شود؛ در حالی‌که فایل‌های موقت ایجاد شده توسط درخواست بارگذاری هنوز پاک نشده‌اند. این به طور بالقوه می‌تواند منجر به تجمع فایل‌های موقتی شده و در نتیجه فضای حافظه به حد خود رسیده و سرور خاموش شود. (طبق گزارش CVE-2020-7067 موارد دیگری نیز از حفره‌های امنیتی این نسخه دیده می‌شود.)

۳-۲-۲- راهکارهای پیشنهادی

سرور را در اولین فرصت به روز و آپگرید کنید. حتما از به روز بودن نسخه PHP اطمینان حاصل کرده و در صورت امکان از محدودیت و وابستگی کدهای سمت سرو به نسخه نرم افزارها جلوگیری کنید.

۳-۲- تست D/DOS

۱-۳-۲- سناریوی انجام تست

مطابق موارد ذکر شده در بخش‌های ۱-۲ و ۲-۲ و همچنین عدم شناسایی فایروال در مسیر انتقال داده‌ها به سرور

۲-۳-۲- نتایج تست

آسیب پذیری‌های موارد قبلی و همچنین عدم وجود فایروال ممکن است سرور را در معرض خطر دسترسی غیر مجاز به داده‌های محرمانه و همچنین احتمال حمله D/DOS قرار دهد. یک مهاجم می‌تواند برای هر یک از این آسیب پذیری‌ها، گزارش حفره‌های امنتی را جستجو کرده و از آن برای حمله استفاده کند.

۲-۳-۳- راهکارهای پیشنهادی

سرور را در اولین فرصت به روز و آپگرید کرده و با نصب و فعال‌سازی ماژول mod_security روی آپاچی و همچنین تعبیه یک فایروال از حملات جلوگیری کنید.

۲-۴- تست HTTP Cookies

۲-۴-۱- سناریوی انجام تست

دزدیدن کوکی‌های کاربران تحت شبکه و همچنین تست هدرهای HttpRequest

۲-۴-۲- نتایج تست

شکل ۲-۲، نشان می‌دهد کوکی‌های ذکر شده حاوی پرچم‌های (Flag) مشخص شده نیستند.

Cookie Name	Flags missing
XSRF-TOKEN	Secure, HttpOnly
_nc	Secure
laravel_session	Secure

شکل ۲-۲: کوکی‌ها و عدم وجود پرچم‌های آن‌ها

از آنجا که پرچم Secure روی کوکی تنظیم نشده است، در صورت وجود چنین درخواستی، مرورگر آن را از طریق کانال بدون رمزگذاری (HTTP ساده) ارسال می‌کند. بنابراین، این خطر وجود دارد که یک مهاجم بتواند ارتباط متنی را بین مرورگر و سرور رهبری کند و کوکی کاربر را بدزدد. اگر این یک کوکی نشست (Session) باشد، مهاجم می‌تواند بدون وجود احراز هویت به نشست وب قربانی دسترسی پیدا کند. عدم وجود پرچم HttpOnly به مرورگر این امکان را می‌دهد تا از طریق اسکریپت‌های سمت کلاینت (از قبیل JavaScript، VBScript و غیره) به کوکی دسترسی پیدا کند. این مورد می‌تواند توسط مهاجم با استفاده از حمله XSS به منظور سرقت کوکی آسیب دیده مورد سوء استفاده قرار بگیرد.

۲-۴-۳- راهکارهای پیشنهادی

در تمامی تنظیمات کوکی وب سایت، این پرچم‌ها را اعمال کنید (برای اطلاع بیشتر به این لینک مراجعه شود).

۲-۵- تست شناسایی تکنولوژی استفاده شده در وب سایت

۲-۵-۱- سناریوی انجام تست

بررسی متاداده‌های HTML و همچنین SVB

۲-۵-۲- نتایج تست

می‌توان مشخصات تکنولوژی‌های سمت سرور و سمت کاربر را به راحتی شناسایی کرد. یک مهاجم می‌تواند از این اطلاعات استفاده کند تا حملات خاصی را در برابر نوع و نسخه نرم افزار شناسایی شده انجام دهد. برای این وب‌سایت تکنولوژی‌ها به صورت نشان داده شده در شکل ۳-۲ هستند.

Software / Version	Category
Apache 2	Web Servers
PHP 7.3.16	Programming Languages
Laravel	Web Frameworks
Twitter Bootstrap	Web Frameworks
Google Analytics UA	Analytics
Google Tag Manager	Tag Managers
Select2	JavaScript Frameworks
SweetAlert2	JavaScript Frameworks
jQuery	JavaScript Frameworks

شکل ۳-۲: تکنولوژی‌های استفاده شده در وب سایت

۲-۵-۳- راهکارهای پیشنهادی

اطلاعاتی که اجازه شناسایی پلتفرم نرم افزار، فناوری، سرور و سیستم عامل را می‌دهد (مانند هدرهای سرور HTTP، اطلاعات متا HTML و غیره) حذف کنید. این کار با استفاده از اطلاعات موجود در این لینک و تنظیم mod_headers می‌تواند انجام شود.

۲-۶- تست HTTP Security Headers

۲-۶-۱- سناریوی انجام تست

اسکن Header پاسخ‌های یک HttpRequest

۲-۶-۲- نتایج تست

وضعیت هدرهای امنیتی مهم به صورت نشان داده شده در شکل ۴-۲ است.

HTTP Security Header	Header Role	Status
X-Frame-Options	Protects against Clickjacking attacks	Not set
X-XSS-Protection	Mitigates Cross-Site Scripting (XSS) attacks	Not set
Strict-Transport-Security	Protects against man-in-the-middle attacks	Not set
X-Content-Type-Options	Prevents possible phishing or XSS attacks	Not set

شکل ۴-۲: وضعیت هدرهای امنیتی

از آنجا که هدر X-Frame-Options توسط سرور ارسال نمی‌شود، یک مهاجم می‌تواند این وب سایت را در یک iframe یک وب سایت شخص ثالث جاسازی کند. با دست‌کاری صفات نمایشگر iframe، مهاجم می‌تواند کاربر را در انجام کلیک‌های ماوس در برنامه فریب دهد. بنابراین فعالیت‌هایی را بدون رضایت کاربر انجام می‌دهد (به عنوان مثال: حذف کاربر، مشترک شدن در خبرنامه و غیره). به این حمله Clickjacking گفته می‌شود و در اینجا با جزئیات توضیح داده شده است.

هدر HTTP-X-Protection به مرورگر دستور می‌دهد که هنگام شناسایی حملات Cross-Site Scripting (XSS)، بارگیری صفحات وب را متوقف کند. در صورت عدم وجود این هدر، کاربران برنامه را در معرض حملات XSS قرار می‌دهد.

هدر HTTP Strict-Transport-Security به مرورگر دستور می‌دهد تا وب سایت را از طریق اتصال HTTP ساده بارگیری نکند، بلکه همیشه از HTTPS استفاده کند. عدم وجود این هدر، کاربران را در معرض خطر سرقت داده‌ها یا ویرایش‌های غیرمجاز قرار می‌دهد و مهاجم حمله Man-In-The-Middle انجام داده و ارتباط بین کاربر و سرور را متوقف می‌کند.

هدر HTTP X-Content-Type-Options به مرورگر Internet Explorer خطاب شده و مانع از تفسیر مجدد محتوای یک صفحه وب (MIME-sniffing) و در نتیجه نادیده گرفتن مقدار هدر Content-Type می‌شود. عدم وجود این هدر می‌تواند منجر به حمله‌هایی مانند Cross-Site Scripting یا فیشینگ شود.

۳-۶-۲- راهکارهای پیشنهادی

- هدر پاسخ صفحه X-Frame-Options را به هر صفحه‌ای که می‌خواهید در برابر حملات Clickjacking محافظت شود، اضافه کنید (اطلاعات بیشتر).
- هدر X-XSS-Protection را روی "X-XSS-Protection: 1؛ mode = block" تنظیم کنید (اطلاعات بیشتر).
- هدرStrict-Transport-Security را تنظیم کنید (اطلاعات بیشتر).
- هدر X-Content-Type-Options را روی "X-Content-Type-Options: nosniff" تنظیم کنید (اطلاعات بیشتر).

۷-۲- تست Robots.txt

۱-۷-۲- سناریوی انجام تست

در خواست صفحه https://nanoclub.ir/robots.txt

۲-۷-۲- نتایج تست

هیچ خطر امنیتی خاصی در داشتن فایل robots.txt وجود ندارد. از آنجایی که این فایل غالباً به منظور مخفی کردن برخی از صفحات وب از موتورهای جستجو استفاده می‌شود، هکرها می‌توانند با استفاده از این فایل به URL صفحاتی مانند پنل ادمین را پیدا کنند. در حال حاضر، در بخش Disallow در این فایل، هیچ URL-ای وجود ندارد. بنابراین وب سایت در حال حاضر، طبق این تست، امن است.

۲-۷-۳- راهکارهای پیشنهادی

با توجه به امن بودن وب سایت بر اساس این تست، در این بخش، صرفا پیشنهاد می‌شود که URL-های بخش‌های حساس مانند پنل‌های مدیریت، فایل‌های تنظیمات و غیره همچنان در فایل robots.txt قرار نگیرد. برای اطلاعات بیشتر به این لینک مراجعه شود.

۲-۸- تست SlowLoris

۲-۸-۱- سناریوی انجام تست

استفاده از Nmap

۲-۸-۲- نتایج تست

سرور کنونی، مستعد پاسخ مثبت به این حمله است. با این حمله دیتابیس به طور محدود قابل دسترس شده و حجم استفاده از CPU چند برابر می‌شود. در نتیجه برای پاسخ به کاربران دیگر اولویتی قرار نمی‌دهد. درصد قابل توجهی از کاربران از دسترس وب خارج می‌شوند و یا در یک زمان بسیار طولانی می‌توانند درخواست‌های خود را ارسال کنند.

۲-۸-۳- راهکارهای پیشنهادی

بایستی ماژول‌های مربوطه را در Apache تنظیم کرد و همچنین Timeout پاسخ را به ۶۰ یا کمتر تغییر داد (برای اطلاعات بیشتر به این لینک مراجعه شود).

۲-۹- تست Session

۲-۹-۱- سناریوی انجام تست

دزدیدن کوکی‌های کاربران و همچنین خواندن کدهای مربوطه روی سرور

۲-۹-۲- نتایج تست

نشست‌های (Session) سایت دارای ۳ مشکل اساسی به صورت زیر است:

۱) رمزنگاری نشده و موجب حمله Session Hijack می‌شود.
۲) نشست‌ها بدون توجه به Agent و یا IP کلاینت ذخیره شده و موجب حمله Session Fixation می‌شوند.

۳) نشست‌ها بصورت فایل در سرور ذخیره می‌شوند. بنابراین در صورت افزایش تعداد کاربران آنلاین، حجم این فایل‌ها افزایش یافته و حافظه و CPU را اشغال می‌کند.

۳-۹-۲- راهکارهای پیشنهادی

Sessionها را در جدولی از دیتابیس ذخیره کرده و از رمزنگاری مطمئنی (تغییر کلید خصوصی هر ۳ ماه یکبار) استفاده کنید. می‌توانید از راهکارهایی که فریم ورک لاراول در اختیار قرار می‌دهد؛ بهره ببرید (برای اطلاع بیشتر به این لینک مراجعه شود).

۱۰-۲- تست HSTS

۱-۱۰-۲- سناریوی انجام تست

درخواست داده‌های سایت بدون استفاده از https

۲-۱۰-۲- نتایج تست

با وجود SSL، سرور به درخواست‌های http پاسخ می‌دهد. HTTP Strict Transport Security (HSTS) یک پیشرفت امنیتی انتخاب کننده است که توسط وب‌سایت، از طریق استفاده از یک هدر پاسخ، مشخص می‌شود. هنگامی که مرورگر این هدر را دریافت کرد، از ارسال هرگونه ارتباط از طریق HTTP به دامنه مشخص، جلوگیری می‌کند و در عوض تمام ارتباطات را از طریق HTTPS ارسال می‌کند. همچنین از کلید HTTPS از طریق پرس و جوهای موجود در مرورگرها جلوگیری می‌کند.

۳-۱۰-۲- راهکارهای پیشنهادی

یک هدر به فایل htaccess. وب‌سایت خود اضافه کنید. هدری که باید اضافه شود به صورت زیر است: (برای اطلاع بیشتر به این لینک مراجعه کنید)

Strict-Transport-Security: max-age=31536000; includeSubDomains

۱۱-۲- تست کتابخانه‌های JS

۱-۱۱-۲- سناریوی انجام تست

بررسی نسخه‌های JS موجود در سایت از طریق مرورگر

۲-۱۱-۲- نتایج تست

کتابخانه‌های اسکریپتی سرور همگی به صورت محلی ذخیره شده و استفاده می‌شوند. این موجب شده است که Popper.js، Twitter-bootstrap، Limonte-sweetalert2، Croppie و jQuery به روز نباشند. به خصوص در مورد jQuery که مطابق گزارش CVE-2020-11022 و CVE-2020-11023 نسخه کنونی دارای حفره است.

۳-۱۱-۲- راهکارهای پیشنهادی

به منظور به روز بودن همیشگی این کتابخانه‌ها، بهتر است به صورت لینکی از CDN همان کتابخانه‌ها در صفحات درج شوند. در اینصورت حتما ویژگی SRI را رعایت کنید.

۱۲-۲- تست SQL Injection

۱-۱۲-۲- سناریوی انجام تست

بررسی کدهای موجود در سایت و نسخه فریم ورک لاراول

۲-۱۲-۲- نتایج تست

لاراول در برابر این نوع حملات مقاوم است. اما فقط در صورتیکه، پکیج‌ها و نسخه آن مداوم به روز باشد. طبق گزارشات اخیر در مواردی، این نوع حمله در نسخه کنونی (۵،۸) دیده شده است. بررسی تمام ارتباطات میان سایت و دیتابیس زمانبر بوده و جهت اطمینان از عملکرد تمامی ارتباطات میان دیتابیس و وب سایت، بهتر است به مستندات خود لاراول در این زمینه مراجعه شود.

۳-۱۲-۲- راهکارهای پیشنهادی

نسخه کنونی را به نسخه‌های جدیدتر (ترجیحاً ۷) آپگرید کنید و یا پکیج‌های مورد استفاده در دیتابیس را به روز کنید (برای اطلاعات بیشتر به این لینک مراجعه کنید).

۱۳-۲- تست XHR

۱-۱۳-۲- سناریوی انجام تست

استفاده از XHRهای موجود در صفحه معلم و دانشجو و viewer

۲-۱۳-۲- نتایج تست

سایت حاضر مبتنی بر فریم ورک لاراول و معماری MVC است. استفاده از XHRهایی که بیشتر در مدل MVVM استفاده می‌شود، در وهله اول کنترل و سیاست‌های اعمالی در رندرکردن صفحات را به هم می‌زند. در وهله بعدی، بررسی تمام سیاست‌ها مانند احراز هویت شدن فرد، دسترسی داشتن فرد و ... به محتوا در XHR پیچیده است. (به عنوان مثال: https://nanoclub.ir/student که بایستی حتما با لاگین وارد این لینک شد). همچنین در تمام صفحات وبسایت یک XHR مربوط به دریافت اطلاعات کارت و حساب (که احتمالا مربوط به کیف پول است) ارسال می‌شود. (https://t1.nanoclub.ir/get-cart-count) این مورد را می‌توانید با ابزار توسعه مرورگر، بخش Networks و قسمت XHR بررسی کنید.

۱۴-۱۳-۲- راهکارهای پیشنهادی

بهتر است تمام درخواست‌هایی از این نوع با یک کنترلر هندل شوند و یا کلا با api های تعریف شده در لاراول جایگزین و تمام درخواست‌های اضافی حذف شوند.

۲-۱۴- تست خطاهای سایت

۲-۱۴-۱- سناریوی انجام تست

ایجاد اختلال و یا درخواست صفحات موجود بدون پارامتر ورودی

۲-۱۴-۲- نتایج تست

حالت دیباگ و توسعه روی وبسایت فعال بوده و از این رو بسیاری از درخواست‌های نابجا یا ناقص را با خطای صریح مواجه می‌سازد. از طرف دیگر لاراول تمام خطاها را در صورت فعال بودن حالت دیباگ و توسعه، بسیار جامع و با جزئیات و حتی کدهای مواجه شده با exception را نشان می‌دهد. این بسیار خطرناک بوده و بایستی غیرفعال شود. (به عنوان مثال: https://nanoclub.ir/student)

۲-۱۴-۳- راهکارهای پیشنهادی

در اولین فرصت با مراجعه به مستندات لاراول این قابلیت را حذف کنید.

۲-۱۵- تست POST/GET

۲-۱۵-۱- سناریوی انجام تست

حدس لینک‌های موجود برای ویرایش و ثبت و حذف

۲-۱۵-۲- نتایج تست

بسیاری از این درخواست‌ها علی رغم عدم نمایش به دانشجو یا کسانی که به آن طبیعتا نباید دسترسی داشته باشند، بدون بررسی قابلیت دسترسی و اتمام انقضا در دسترسی، قابل ویرایش و ثبت است. به عنوان مثال لینک ثبت نام در دوره با وجود اتمام زمان آن دوره، با درخواست https://t1.nanoclub.ir/courses/register/14 (یعنی دوره ۱۴ را ثبت کن) ثبت می‌شود. درخواست‌های دیگر نظیر شروع آزمون بدون شرکت در آن آزمون و ... نیز از این قاعده مستثنی نیستند. برخی از فایل‌های تست شده در زیر آمده است:

- https://t1.nanoclub.ir/olympiad_main_exam/start/1
- https://t1.nanoclub.ir/front_olympiad/2
- https://t1.nanoclub.ir/olympiad_main_exam?olympiad=2
- https://t1.nanoclub.ir/buy_credit/2
- https://t1.nanoclub.ir/course_mock_exam/start/14
- https://t1.nanoclub.ir/exam/7/type/235

۲-۱۵-۳- راهکارهای پیشنهادی

تمام ورودی‌ها از جمله قابلیت دسترسی افراد در هر route بررسی شود. پیشنهاد می‌شود شماره شناسه موجودیت‌ها که ورودی در سمت کاربر است، بصورت نامفهوم بوده و عینا منتقل نشود.

۲-۱۶- تست دسترسی به پوشه‌ها

۲-۱۶-۱- سناریوی انجام تست

درخواست عکس‌ها و فایل‌های کاربران در مرورگر با شناسه‌های مختلف

۲-۱۶-۲- نتایج تست

لینک پروفایل افراد و شاید فایل‌های دیگر در مسیر https://nanoclub.ir/upload/ بدون احراز هویت توسط همه قابل دسترس است. مثلا این لینک عکس آپلود شده یکی از کاربران است:

https://nanoclub.ir/upload/profile/1576498037.png

۲-۱۶-۳- راهکارهای پیشنهادی

از سمت سرور دسترسی به این پوشه را از طریق هر درخواستی غیر فعال کنید. در سمت کاربر متد فراخوانی این لینک توسط یک XHR است. به جای فرخوانی لینک عکس یا فایل‌ها، کنترلری روی سرور تعبیه شود که این فایل را مستقیماً بصورت یک فایل php با هدر مربوطه نمایش دهد (برای اطلاعات بیشتر به این لینک مراجعه شود).

۲-۱۷- تست دسترسی به MySQL

۲-۱۷-۱- سناریوی انجام تست

بررسی کدهای سرور و محتویات MySQL

۲-۱۷-۲- نتایج تست

نوع دسترسی کاربر MySQL در وب سایت به عنوان ادیتور نیز هست. این نوع دسترسی در صورت هک شدن وب سایت و یا به هر صورت دیگر با گرفتن دسترسی فریم ورک لاراول، امکان از بین بردن و ویرایش داده‌ها را میسر می‌سازد.

۲-۱۷-۳- راهکارهای پیشنهادی

بهتر است شیوه و لایه ارتباطی بین دیتابیس و وب سایت را با استفاده از Stored Procedure ها فراهم کرد. از جهت دیگر کاربر MySQL فقط دسترسی فرخوانی این پروسجرها را خواهد داشت. بنابراین در هر صورتی امکان اعمال تغییرات خارج از این دسترسی میسر نمی‌شود (برای اطلاعات بیشتر به این لینک مراجعه کنید).

بخش ۳ - اولویت پیشنهادی برای بهبود وب سایت

۳-۱- اولویت‌های پیشنهادی برای بهبود سرعت و امنیت وب سایت

جهت بهبود وب سایت از نظر سرعت و امنیت، با توجه به میزان آسیب پذیری سیستم از موارد ذکر شده در بخش ۲ و همچنین راهکارهای افزایش سرعت وب سایت در بخش ۱، پیشنهاد می‌شودف تمام موارد ذکر شده در این گزارش، به صورت کامل انجام شوند. با این وجود، با در نظر گرفتن اهمیت و محدودیت‌های زمانی، در جدول ۳-۱، اولویت پیشنهادی برای بهبود وب سایت، بیان شده است.

جدول ۱-۱: اولویت پیشنهادی انجام کارها برای بهبود سرعت و امنیت وب سایت

اولویت (به ترتیب از بیشترین اولویت به کمترین اولویت)	بخش	توضیحات
۱	۲-۱ تا ۲-۱۱	انجام این موارد نیاز به زمان زیادی ندارد و تاثیر مهمی بر امنیت وب سایت خواهد داشت.
۲	۲-۱۳ تا ۲-۱۵	انجام این موارد نیاز به زمان بیشتری نسبت به سایر موارد دارد. اما، انجام این موارد به شدت ضروری است.
۳	۲-۱۲ و ۲-۱۷	انجام این موارد نیز ضروری است. اما از درجه اهمیت کمتری نسبت به موارد موجود در اولویت دوم برخوردار هستند. علاوه بر اینف تست ۲-۱۷ به افزایش سرعت وب سایت نیز کمک می‌کند.
۴	۱-۱ تا ۱-۳	این موارد، به افزایش سرعت وب سایت کمک می‌کنند و انجام آن‌ها بعد از بهبود امنیت سیستم ضروری است.