در شکارگاه چه می‌گذرد؟

کتابچه‌ی راهنمای برنامه‌نویسی امن

Secure Programming HandBook

Ravro

برای کدهایی امن و جهانی امن‌تر

به مناسبت ۲۵۶مین روز سال، تصمیم گرفتیم بخشی از حاصل بررسی‌هایمان برروی حدود ۱۰۰۰ گزارش آسیب‌پذیری دریافت‌شده در دو سال اخیر در پلتفرم باگ‌بانتی راورو را، با شما به اشتراک بگذاریم. در این PDF گفته‌ایم که بر اساس یافته‌هایمان، در نقاط مختلف سامانه، چه آسیب‌پذیری‌هایی رایج‌تر و پرتکرارترند؟ به عبارتی دیگر، هر نقطه از سامانه، مخفی‌گاه رایج چه باگ‌ها و ایمپاسترهایی است؟

۷ کُلنی باگ‌ها و آسیب‌پذیری‌ها که ما در این گزارش به ارائه‌ی لیستی از آسیب‌پذیری‌های رایج در هر یک می‌پردازیم، عبارتند از:

۱. فرم لاگین به سایت
۲. فرم فراموشی رمزعبور
۳. فرم آپلود عکس / CSV
۴. فرم ثبت اطلاعات در سایت
۵. دامنه‌ها و ویژگی‌های سایت
۶. اتصال به درگاه
۷. فرم مشخصات کاربران

برای امن‌ترسازی هر بخش نیز، چند توصیه‌ی کوتاه امنیتی برایتان داریم.

تقدیم به:

تمام شب‌بیدارهایی که قهوه را به کد تبدیل می‌کنند ؛)

با تشکر از:

تمامی شـــکارچیانی کـه در کشـــف این بـاگ‌هـا مشـــارکـت داشـــتـنـد و بـا گزارش‌هـای خود، رای بـه خروج و حـذف ایمپـاســـترهـا دادنـد. دانش، تخصـــص و مهـارت خود را برای تحقق جهـانی امن‌تر بـه کـار بســـتـنـد.

و تمام میدان‌هایی که به باگ‌بانتی اعتماد کردند و در جریان‌گرفتن این دانش، سهیم بودند.

بازی Among Us ، بازی‌ای شبیه به بازی مافیاست. بازیکنان در دو گروه Crewmates و Impostors قرار می‌گیرند. Crewmates باید با تلاش خود، Impostor ها را تشخیص دهند و رای به خروج آن‌ها از بازی دهند. اما شناخت Impostor ها آسان نیست، و لازم است که بازیکنان هوش خود را در بررسی رفتار آن‌ها به کار گیرند...

۱ آسیب‌پذیری‌های رایج در فرم لاگین به سایت:

آسیب‌پذیری‌های رایج در فرم لاگین به سایت، به ترتیب فراوانی عبارتند از :

۱. آسیب‌پذیربودن نسبت به کشف اطلاعات کاربر در سامانه؛ User Enumeration

هنگامی که برای لاگین به سایتی مراجعه می‌کنیم، با ورود نام کاربری و پسورد منتظر پاسخ می‌مانیم .پاسخ سرور م ی‌تواند برملاکننده‌ی برخی اطلاعات باشد .یکی از راه‌ها و شگردهایی که مهاجمین برای نفوذ از آن استفاده می‌کنند، بررسی وجود یک کاربر در یک سامانه است. مهاجم با سوءاستفاده از این آسیب‌پذیری، در فرم‌های ثبت نام و ورود، به طور شانسی یا برحسب بعضی اطلاعات و محاسبات خویش، یک نام کاربری را وارد می‌کند .و بر اساس پاسخ‌هایی که سرور به او می‌دهد، متوجه وجود/عدم وجود چنین کاربری در آن سایت می‌شود . سپس بقیه‌ی اطلاعات کاربر را حدس می‌زند و به حساب کاربری او وارد شود .

۲. آسیب‌پذیربودن نسبت به ارسال سیلی از پیامک؛ آسیب‌پذیری SMS Flooding

در برخی صفحات، هنگامی که برای لاگین اطلاعات خود را وارد فرم می‌کنیم، سرور کد یک‌بارمصرفی را در قالب SMSبرای ما ارسال می‌کند تا با ورود آن، از هویت ما اطمینان پیدا کند .مهاجم با سوءاستفاده از این آسیب‌پذیری، با لاگین‌های متعدد درخواست ارسال سیلی از پیامک را به صورت پیاپی از سرویس می‌کند .به این ترتیب، با ایجاد وضعیت منع سرویس می‌تواند موجب اختلال در سامانه شود. با ایجاد اختلال در سامانه، مهاجم امکان ارسال درخواست‌های مخرب خود را لابه‌لای سایر درخواست‌ها خواهد داشت .سامانه‌ی مختل‌شده نیز آن‌ها را به‌اشتباه خواهد پذیرفت و مهاجم خواهد توانست که به صورت مجاز اطلاعاتی مخرب را به سامانه تزریق کند.

۳. آسیب‌پذیر بودن از لحاظ منقضی‌نشدن کد پیامک ارسالی به کاربر

در برخی صفحات، هنگامی که برای لاگین اطلاعات خود را وارد فرم می‌کنیم، سرور کد یک‌بارمصرفی را در قالب SMSبرای ما ارسال می‌کند تا با ورود آن، از هویت ما اطمینان پیدا کند. در صورتی که SMSهای ارسالی شامل کدها و اطلاعات، پس از گذشت مدتی، منقضی نشوند. مهاجم فرصت بیش‌تری دارد تا از طریق راه‌های مختلف به کدها دسترسی یابد و به حساب کاربری دیگران نفوذ کند.

۴. آسیب‌پذیربودن از لحاظ تعداد کاراکتر کد تایید ارسال‌شده در پیامک به کاربر

این آسیب‌پذیری زمانی رخ می‌دهد که کد یک‌بارمصرف ارسالی سامانه به کاربر، از طریق پیامک، طول مشخصی نداشته باشد. و هنگام ورود کد در سامانه توسط کاربر نیز تعداد کاراکتر کد دریافتی بررسی نشود. مهاجم با سوءاستفاده از این آسیب‌پذیری می‌تواند مقادیری را از طریق فیلد کد تزریق کند و آسیب‌پذیری‌های ثانویه‌ای مانند SQL Injection را سبب شود.

۵. آسیب‌پذیربودن نسبت به ارسال سیلی از درخواست به سرور؛ آسیب‌پذیری HTTP Flooding

آسیب‌پذیری HTTP Flooding زمانی رخ می‌دهد که برای تعداد درخوا ست ار سالی از سمت کاربر به سمت سرور محدودیتی وجود نداشته باشد. مهاجم با سوءاستفاده از این آسیب‌پذیری، سیلی از درخواست‌ها را به سمت سرور ارسال می‌کند .این‌گونه، سامانه دچار DoS یا منع سرویس و مختل می‌شود .مهاجم می‌تواند در این اختلال از تزریق داده‌های خود به سامانه سو استفاده کند و با داده‌های جعلی تزریقی خود به سامانه نفوذ کند .

۶. آسیب‌پذیربودن از لحاظ عدم پیکربندی صحیح OAuth

روش «OAuth» روش احراز هویت امن‌تری نسبت به استفاده از روش سنتی «نام کاربری و گذرواژه «است. روش OAuth مانند نمکی ضدعفونی‌کننده، نسبت به ورودی‌ها عمل می‌کند، اما وای از آن روز که بگندد نمک... اگر پیکربندی آن به طور صحیح و ایمن صورت نگرفته باشد، مهاجم می‌تواند با سوءاستفاده از این آسیب‌پذیری با ورود پارامترهای مختلف، سبب اختلال در این روش شود و به اطلاعات کاربری کاربران دسترسی یابد .

۷. آسیب‌پذیربودن نسبت به هدایت کاربر به صفحات مخرب و فیشینگ؛ آسیب‌پذیری Open Redirect

گاهی در سامانه‌های وب، پارامتری در URL آدرس صفحه‌ای که کاربر باید به آن ریدایرکت شود قرار می‌گیرد. مهاجم با سوءاستفاده از این آسیب‌پذیری می‌تواند با تغییردادن مقدار آدرس آن پارامتر، به آدرس سایت خود، اطلاعات کاربری را سرقت کند.

۸. آسیب‌پذیربودن سرور نسبت به ورودی کاربر؛ آسیب‌پذیری Mass Assignment

فریمورک‌های نرم‌افزاری در مواقعی به برنامه‌نویس این امکان را می‌دهند که پارامترهای درخواست HTTP را مستقیم و به صورت خودکار به متغیرها و آبجکت‌های درون برنامه‌ی خود متصل کنند. اما این روند می‌تواند موجب بروز آسیب‌پذیری Mass Assignment شود. مهاجم با سوءاستفاده از از این آسیب‌پذیری می‌تواند پارامترهای جدیدی ایجاد کند که برنامه‌نویس نیازی ندارد. این پارامترهای ا ضافی در درخوا ست HTTP می‌توانند سبب تغییر یا ایجاد مقادیری جدید شوند و اگر آن مقادیر مخرب باشند سبب آسیب‌پذیری‌های از نوع Stored می‌شوند و امکان اجرای اسکریپت مخرب و حتی نفوذ به پایگاه داده شوند.

۹. آسیب‌پذیربودن نسبت به تزریق دستورات SQL؛ آسیب‌پذیری SQL Injection

آسیب‌پذیری SQL Injection، از خانواده‌ی آسیب‌پذیری‌های تزریقی است. مهاجم با سوءاستفاده از این آسیب‌پذیری، دستورات SQLرا به صورت بدخواهانه در قالب دستور مجاز SQLای دیگر در فیلدهای فرم تزریق می‌کند .و این‌گونه، از طریق پرسش‌های متعدد به سیستم مدیریت پایگاه داده دسترسی می‌یابد.

۱۰. آسیب‌پذیربودن نسبت به تزریق اسکریپت مخرب به فرم‌ها؛ آسیب‌پذیری XSS

آسیب‌پذیری Cross Site Scripting یا XSS از رایج‌ترین آسیب‌پذیری‌های موجود در وب به شمار می‌رود .مهاجم با سوءاستفاده از این آسیب‌پذیری، اسکریپت‌های مخرب را در ساختار صفحات وب به ویژه در فرم‌ها، تزریق می‌کند .این‌گونه اطلاعات کاربری مانند Session ها و Token های کاربران دیگر را به دست می‌آورد و با استفاده از آن‌ها وارد حساب کاربری آن‌ها می‌شود.

۱۱. آسیب‌پذیربودن نسبت به تزریق داده‌های مخرب به XML؛ آسیب‌پذیری XXE

باز هم از خانواده آسیب‌پذیری‌های تزریقی !آسیب‌پذیری XXEیا XML External Entity، بر پایه‌ی ارسال و دریافت داده‌ها از طریق XML است. مهاجم با سوءاستفاده از این آسیب‌پذیری داده‌ها، موجودیت‌ها یا Entityهای بدخواهانه از جمله اسکریپت مخرب را در قالب یک XML تزریق و ارسال می‌کند .ماشین دریافت‌کننده پس از خواندن XML دریافت‌شده، به طور خودکار به پیلود ارسالی مهاجم آلوده می‌شود.

پس برای داشتن **فرم لاگین** امن‌تر:

- در مواردی قصد پیغام خطای مناسب برای تکراری‌بودن مقادیر فیلدهای ورودی‌ها نمایش دهید.
- تعداد دفعات محدودی را برای درخواست ارسال پیامک از سامانه، تعیین کنید.
- زمان انقضای مشخصی را برای هر کد ارسالی در پیامک ارسال‌شده از سوی سامانه به کاربر، تعیین کنید.
- طول ثابتی برای کد یک‌بارمصرف ارسالی با SMS به کاربر، تعیین کنید.
- سقف مشخصی برای تعداد دفعات درخواست‌های ارسالی از سمت کاربر تعیین کنید و حساب کاربری متخلف را مسدود کنید.
- پیکربندی OAuth را مطابق با آخرین استانداردهای امنیتی به‌روز کنید.
- پارامترهای ورودی در URL دارای آدرس، برای ریدایرکت کاربر را کنترل کنید.
- پارامترهای اضافی موجود در درخواست‌های HTTP را، درنظر نگیرید.
- مقادیر ورودی کاربر را اعتبارسنجی و کلمات کلیدی زبان SQL را در ورودی‌ها فیلتر کنید.
- مقادیر ورودی کاربر را اعتبارسنجی و برخی کلمات کلیدی زبان Javascript که نشان‌دهنده‌ی مقادیر هستند، را در ورودی‌ها فیلتر کنید.
- پشتیبانی سرور از موجودیت‌های خارجی را لغو کنید و XInclude را غیرفعال کنید.




۲ آسیب‌پذیری‌های رایج در فرآیند فراموشی رمزعبور:

آسیب‌پذیری‌های رایج در فرآیند تغییر یا فراموشی رمز عبور، به ترتیب فراوانی عبارتند از:

۱. آسیب‌پذیربودن از لحاظ انقضای Token

بعد از هر ورود به سامانه، برای شناسایی کاربر در سمت سرور، رشته‌ی ناخوانای توکن تولید و به سمت کاربر ارسال می‌شود تا برای هر درخواست در آینده از همان توکن برای تعامل با سمت سرور استفاده شود. در صورتی که توکن پس از مدت معقولی منقضی نشود، مهاجم با سوءاستفاده از آن می‌تواند با توکن کاربر دیگر به اکانت همان کاربر نفوذ کند .

۲. آسیب‌پذیربودن نسبت به ارسال سیلی از درخواست به سرور؛ آسیب‌پذیری HTTP Flooding

آسیب‌پذیری HTTP Flooding زمانی رخ می‌دهد که برای تعداد درخواست ارسالی از سمت کاربر به سمت سرور محدودیتی وجود ندا شته با شد. مهاجم با سوءا ستفاده از این آ سیب‌پذیری، سیلی از درخوا ست‌ها را به سمت سرور ارسال می‌کند .این‌گونه، سامانه دچار DoS یا منع سرویس و مختل می‌شود .مهاجم می‌تواند در این اختلال از تزریق داده‌های خود به سامانه سو استفاده کند و با داده‌های جعلی تزریقی خود به سامانه نفوذ کند .

۳. آسیب‌پذیری تصاحب اکانت به وسیله‌ی ایمیل؛ آسیب‌پذیری Account Takeover

این آسیب‌پذیری به همراه ایجاد اختلال در روند خدمات‌دهی سامانه، صورت می‌گیرد .در زمانی که سامانه از نظر پردازشی مختل می‌شود مهاجم با درخواست تغییر آدرس ایمیل به آدرس ایمیل کاربر دیگر می‌تواند با جعل اطلاعات او به حساب آن کاربر نفوذ کند .

۴. آسیب‌پذیر بودن از لحاظ نحوه‌ی تولید کدهای یک‌بارمصرف (OTP)

نحوه‌ی تولید کد یک‌بار مصرف باید کاملا تصادفی باشد. در صورتی که سامانه برای تولید کد از یک منطق و الگوی مشخص استفاده کند، مهاجم می‌تواند با بررسی کدهای پیشین، الگو را حدس بزند و با سوءاستفاده از این آسیب‌پذیری، پیش از کاربر کد را وارد کند و وارد شود .

پس برای داشتن **فرم فراموشی رمز عبور** امن‌تر:

- برای توکن‌های تولیدشده زمان انقضای مشخص و محدودی تعیین کنید.
- سقف مشخصی برای تعداد دفعات درخواست‌های ارسالی از سمت کاربر تعیین کنید و حساب کاربری متخلف را مسدود کنید.
- سقف مشخصی برای تعداد دفعات درخواست‌های ارسالی از سمت کاربر تعیین کنید و آدرس مبدأ متخلف را مسدود کنید.
- برای تولید کدهای یک‌بارمصرف، از کتابخانه‌های معتبری که کدهای تصادفی و باالگوی غیرقابل‌کشف تولید می‌کنند، استفاده کنید.

۳ آسیب‌پذیری‌های رایج در فرم آپلود فایل :

آسیب‌پذیری‌های رایج در فرم‌های آپلود فایل به ویژه عکس و CSV ، به ترتیب فراوانی عبارتند از :

۱. **آسیب‌پذیر بودن نسبت به تعدد پارامترهای URL؛ آسیب‌پذیری IDOR**

در آسیب‌پذیری IDOR یا Insecure Direct Object Reference پارامتری در URLمقدار id مربوط به یک دسته اطلاعات را از سوی کاربر دریافت می‌کند و بر اساس آن، اطلاعات مربوط را در صفحه‌ای که مختص کاربر است، نمایش می‌دهد. عدم اعتبارسنجی ورودی کاربر، امکان دسترسی مهاجم به اطلاعات صفحه‌ی هر کاربر را فراهم می‌کند.

۲. **آسیب‌پذیربودن از لحاظ آپلود فایل‌های مخرب با پسوند دل‌خواه**

مهاجم با سوءا ستفاده از این آ سیب‌پذیری می‌تواند فایل‌های مخرب خود را با پ سوندهای موردنظر خود، آپلود و به این وسیله به سامانه نفوذ کند.

۳. **آسیب‌پذیر بودن نسبت به تزریق اسکریپت مخرب به فرم‌ها از طریق آپلود فایل SVG؛ آسیب‌پذیری XSS Stored**

آسیب‌پذیری XSS Stored یا Cross Site Scripting Stored یکی از گونه‌های آسیب‌پذیری تزریقی XSS و از رایج‌ترین آسیب‌پذیری‌های موجود در وب به شمار می‌رود .مهاجم با سوءاستفاده از این آسیب‌پذیری، با تزریق اسکریپت‌های مخرب در ساختار صفحات وب) به‌ویژه در فرم‌ها (اطلاعات کاربری کاربران دیگر مانند Sessionها و Tokenها را به دست می‌آورد و با استفاده از آن‌ها می‌توانند وارد حساب کاربری آن‌ها شود .زمانی که فایل SVG آپلود می‌شود، در زمان نمایش به کاربر در صفحه، در زبان HTML تبدیل به کد می‌شود .اگر در درون کد فایل SVG مقادیر مخربی مانند آدرس یک اسکریپت مخرب، تزریق‌شده باشد، امکان اجرای آن کد در صفحه به وجود می‌آید.

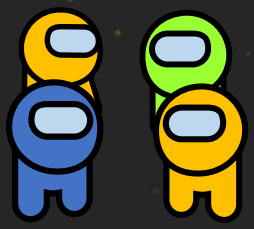
۴. آسیب‌پذیر بودن نسبت به جعل هویت در ارسال درخواست؛ آسیب‌پذیری SSRF

این آسیب‌پذیری به مهاجم این امکان را می‌دهد که از طریق اتصال به یک سرویس سامانه به سرویس‌های دیگر در داخل سامانه درخواست ارسال کند. ارسال درخواست از طریق سامانه سبب می‌شود که درخواست مجاز به نظر برسد. مهاجم با سوءاستفاده از این آسیب‌پذیری می‌تواند درخواست‌های مخرب خود را در قالب مجاز ارسال و به سرویس‌های دیگر سامانه نفوذ پیدا کند.

۵. آسیب‌پذیربودن نسبت به تزریق کد مخرب از طریق آپلود CSV؛ آسیب‌پذیری CSV Injection

این آسیب‌پذیری با نام دیگر Formula Injection نیز شناخته می‌شود و مربوط به جاسازی ورودی نامطمئن در فایل‌های CSV است. وقتی این فایل دارای ورودی نامطمئن، با برنامه‌ای مانند اکسل باز شود، هر cell که با کاراکتر = شروع می‌شود، به عنوان فرمول شناخته می‌شود .مهاجم می‌تواند با قرار دادن کد مخرب خود بعد از کاراکتر = خروجی موردنظر خود را دریافت کند .

۶. آسیب‌پذیربودن از لحاظ آپلود عکس‌ها در سایز دل‌خواه



مهاجم با سوءاستفاده از این آسیب‌پذیری می‌تواند فایل‌های حجیمی را آپلود کند و از این طریق سامانه را، از نظر پرداز شی و از نظر کمبود فضای ذخیره سازی، دچار اختلال کند .

۷. آسیب‌پذیربودن نسبت به تزریق داده‌های مخرب به XML؛ آسیب‌پذیری XXE

آسیب‌پذیری XXE یا XML External Entity، بر پایه‌ی ارسال و دریافت داده‌ها از طریق XML است. مهاجم با سوءاستفاده از این آسیب‌پذیری داده‌ها، موجودیت‌ها یا Entityهای بدخواهانه از جمله اسکریپت مخرب را در قالب یک XML تزریق و ارسال می‌کند. ماشین دریافت‌کننده پس از خواندن XML دریافت‌شده، به طور خودکار به پیلود ارسالی مهاجم آلوده می‌شود.

پس برای داشتن **فرم آپلود** امن‌تر:

- داده‌های ارسالی از سوی کاربر را در سمت کاربر و در سمت سرور اعتبارسنجی کنید.
- در قسمت آپلود فایل‌ها در سامانه توسط کاربر، پسوندهای محدودی را به عنوان پسوندهای مجاز مشخص کنید.
- مقادیر ورودی کاربر را اعتبارسنجی و برخی کلمات کلیدی زبان Javascript که نشان‌دهنده‌ی مقادیر هستند، را در ورودی‌ها فیلتر کنید.
- آدرس‌های مجاز برای دریافت درخواست HTTP در سرور، را با لیست مجاز یا White list تعریف کنید.
- در قسمت آپلود فایل‌ها در سامانه توسط کاربر،محدوده‌ی سایز مجازی را به‌عنوان سایز مجاز مشخص کنید.
- پشتیبانی سرور از موجودیت‌های خارجی(Entities External) را لغو کنید و XInclude را غیرفعال کنید.

۴ آسیب‌پذیری‌های رایج در فرم‌های ثبتی سایت:

آسیب‌پذیری‌های رایج در فرم‌های ثبتی سایت (ثبت‌نام یا خرید) به ترتیب فراوانی عبارتند از :

۱. **آسیب‌پذیربودن از لحاظ عدم تایید ایمیل و شماره تلفن برای ثبت‌نام**

در عملیات‌هایی مانند ثبت‌نام، پس از ثبت شماره‌تلفن و یا ایمیل در فرم، پیامی برای تایید به کاربر ارسال می‌شود . تا سامانه از صحت آدرس ایمیل و شماره‌تلفن واردشده، آگاه شود. در صورتی که سامانه الزامی برای تایید ایمیل و یا شماره تلفن نداشته باشد و از این مکانیزم استفاده نکند، مهاجم با سوءاستفاده از این آسیب‌پذیری می‌تواند تعداد بسیار زیادی حساب کاربری با اطلاعات غلط بسازد و یا از همین طریق درخواست‌های بی‌شماری به سامانه ارسال کند .تا عملکرد سامانه را مختل کند.

۲. **آسیب‌پذیربودن نسبت به تزریق اسکریپت مخرب به فرم‌ها؛ آسیب‌پذیری XSS**

آسیب‌پذیری Cross Site Scripting یا XSS از رایج‌ترین آسیب‌پذیری‌های موجود در وب به شمار می‌رود .مهاجم با سوءاستفاده از این آسیب‌پذیری، اسکریپت‌های مخرب را در ساختار صفحات وب به ویژه در فرم‌ها، تزریق می‌کند .این‌گونه اطلاعات کاربری مانند Sessionها و Tokenهای کاربران دیگر را به دست می‌آورد و با استفاده از آن‌ها وارد حساب کاربری آن‌ها می‌شود.

۳. **آسیب‌پذیربودن از لحاظ عدم پیکربندی صحیح** Captcha

از مکانیزم کپچا در فرم‌ها برای درامان‌نگه داشتن سامانه از مواجهه با سیلی از درخواست‌ها و حمله‌ی بات‌های رایانه‌ای استفاده می‌شود .درصورتی که کپچا طبق الگویی مشخص، نمایان شود و برای کاربر غیرانسان قابل‌کشف باشد، مهاجم می‌تواند به سوءاستفاده از آن بپردازد.

۴. **آسیب‌پذیربودن به ثبت یک موجودیت از سوی حداقل دو مبدا؛ آسیب‌پذیری Race Condition**

اگر دو درخواست هم‌زمان برای یک موجودیت (Entity) ارسال شوند، آسیب‌پذیری Race Condition رخ می‌دهد . به عنوان مثال اگر دو مسافر بلیط آنلاین خود را برای یک صندلی در یک لحظه ثبت کنند، اگر اعتبارسنجی در سمت سرور به درستی صورت نگیرد، دو بلیط برای یک صندلی صادر می‌شود .مهاجم می‌تواند با سوءاستفاده از این آسیب‌پذیری، داده‌های مدنظر خود را وارد پایگاه داده کند.

پس برای داشتن فرم ثبت امن‌تر:

- هنگام ثبت‌نام کاربران، آدرس ایمیل و شماره‌تلفن واردشده در سامانه را صحت‌سنجی کنید.
- مقادیر ورودی کاربر را اعتبارسنجی و برخی کلمات کلیدی زبان Javascript که نشان‌دهنده‌ی مقادیر هستند، را در ورودی‌ها فیلتر کنید.
- از قابل کشـــف نبودن الگوهای کپچای خود، اطمینان حاصـــل کنید. از الگوهای غیرقابل‌کشـــف‌تر منابع معتبر که ترکیبی از حروف و اعداد هستند، استفاده کنید. (مانند reCaptcha گوگل)
- در فرآیند ثبت اطلاعات در پایگاه داده، از توابع مربوط به Race Condition استفاده کنید.

۵ آسیب‌پذیری‌های رایج در دامنه‌ها و ویژگی‌های سایت:

آسیب‌پذیری‌های رایج در دامنه‌ها و ویژگی‌های سامانه، به ترتیب فراوانی عبارتند از:

(این آسیب‌پذیری‌ها کم‌تر با کاربر عادی مرتبط هستند و بیش‌تر به سمت سرور ارتباط دارند.)

۱. آسیب‌پذیر بودن نسبت به تعدد پارامترهای URL؛ آسیب‌پذیری IDOR

در آسیب‌پذیری IDOR یا Insecure Direct Object Reference پارامتری در URL مقدار id مربوط به یک دسته اطلاعات را از سوی کاربر دریافت می‌کند و بر اساس آن، اطلاعات مربوط را در صفحه‌ای که مختص کاربر است، نمایش می‌دهد. عدم اعتبارسنجی ورودی کاربر، امکان دسترسی مهاجم به اطلاعات صفحه‌ی هر کاربر را فراهم می‌کند .

۲. آسیب‌پذیربودن از لحاظ آشکاربودن آدرس آی‌پی اصلی پشت CDN

سامانه‌ها (به ویژه وب‌سایت‌ها) را برای افزایش سرعت بارگذاری و در امان‌ماندن از حملاتی نظیر DDoS و ... که منجر به اختلال سامانه می‌شوند، پشت CDN قرار می‌دهند، تا آی پی سرور اصلی مخفی بماند. در صورتی که پیکربندی به‌طور صحیح صورت نگیرد، آی پی اصلی سامانه از پشت CDN آشکار می‌شود و مهاجم می‌تواند مستقیما با سرور اصلی در تعامل باشد.

۳. آسیب‌پذیربودن از لحاظ عدم پیکربندی صحیح CORS؛ آسیب‌پذیری CORS Misconfiguration

تنظیمات CORS برای جلوگیری از دریافت درخواست‌های HTTP از مبدا غیرمجاز صورت می‌گیرد. در صورتی که پیکربندی این تنظیمات به‌درستی انجام نشود یا به عبارت دیگر موارد زیر تنظیم شده باشد، مهاجم با سوءاستفاده از این آسیب‌پذیری می‌تواند کوکی‌ها و اطلاعات کاربران یک سامانه را به سرقت ببرد و به اکانت کاربر نفوذ کند:

- Access-Control-Allow-Origin: *
- Access-Control-Allow-Credentials: true

۴. **آسیب‌پذیربودن نسبت به تصاحب زیردامنه؛ آسیب‌پذیری Subdomai n Takeover**

در مواقعی که زیردامنه ثبت شده باشد، اما هنوز هیچ آدرس به عنوان host برای آن تعیین نشده باشد، به دست آوردن کنترل این زیردامنه ممکن می‌شود. مهاجم با سوءاستفاده از این آسیب‌پذیری می‌تواند کنترل یک زیردامنه از سامانه‌ی هدف را به دست آورد.

۵. **آسیب‌پذیربودن نسبت به تزریق هدر به درخواست HTTP؛ آسیب‌پذیری** Header Injection

در صورتی که در هدر درخواست HTTP ، بر اساس ورودی کاربر و به صورت پویا ایجاد شود، مهاجم با سوءاستفاده از این آسیب‌پذیری می‌تواند مقادیر مدنظر خود را در هدر تزریق کند. به این وسیله، می‌تواند مسیر درخواست‌ها را به وب‌سایت خود منحرف و اطلاعات کاربر را سرقت کند.

۶. **آسیب‌پذیربودن نسبت به تکرار یک پارامتر در درخواست HTTP؛ آسیب‌پذیری** HTTP Parameter Pollution

این آسیب‌پذیری بر اساس ایجاد اختلال در روند اعتبارسنجی پارامترهای ورودی در درخواست HTTP بروز پیدا می‌کند. مهاجم با سوءاستفاده از این آسیب‌پذیری، یک پارامتر را چند بار در یک درخواست قرار می‌دهد و با این کار سبب می‌شود که در فرآیند اعتبارسنجی، مقادیر این پارامتر به طور صحیح اعتبارسنجی نشوند و از ایست بازرسی اعتبارسنجی به راحتی عبور کند.

۷. آسیب‌پذیربودن نسبت به عدم تشخیص کاراکتر CRLF؛ آسیب‌پذیری CRLF Injection

این آسیب‌پذیری از نوع تزریقی است و زمانی اتفاق می‌افتد که مهاجم یک کاراکتر CRLF را در درخواست HTTP در جایی دور از انتظار استفاده کند. در فرآیند جداسازی هدر و بدنه‌ی درخواست، اختلال ایجاد شود. مهاجم با سوءاستفاده از این آسیب‌پذیری، هدرهای مدنظر خود را تزریق می‌کند تا موجب اختلال شود.

پس برای داشتن **دامنه و ویژگی‌های** امن‌تر:

- داده‌های ارسالی از سوی کاربر را در سمت کاربر و در سمت سرور اعتبارسنجی کنید.
- لیست مجاز یا White list برای آدرس‌های لبه‌ی CDNدر نظر بگیرید.
- لیست مجاز یا White list برای آدرس‌های CORS در نظر بگیرید و ترجیحا دامنه‌ی همان سامانه را برای CORS قرار دهید.
- لازم است در دوره‌های منظم درخواست‌های سامانه به زیردامنه‌ها را بررسی کنید و در صورت کشف زیردامنه‌ی استفاده نشده آن را غیرفعال کنید.
- پارامترهای ارسالی از سوی کاربر را در سمت کاربر و در سمت سرور اعتبارسنجی کنید.
- در درخواست HTTP کاراکترهای مجاز را محدود کنید.

۶ آسیب‌پذیری‌های رایج در اتصال به درگاه:

آسیب پذیری های رایج مربوط به زمانی که کاربر از درون سامانه به درگاه آنلاین بانکی هدایت می‌شود، به ترتیب فراوانی عبارتند از :

۱. **آسیب‌پذیربودن از لحاظ عدم عملکرد صحیح درگاه آنلاین در تراکنش ناموفق و بررسی سایت**

در یک عملیات آنلاین بانکی زمانی که کاربر از سامانه وارد درگاه می‌شود، سامانه تا زمان پایان عملیات بانکی صبر می‌کند تا پارامترهای بازگشتی از درگاه را دریافت کند تا بروز مشکل در پرداخت و یا موفقیت پرداخت مشخص شود. اما اگر این پیکربندی به درستی صورت نگیرد، کاربر می‌تواند زمانی که در درگاه قرار دارد، دکمه‌ی انصراف را بزند اما سامانه، عملیات را به‌طورثابت موفقیت آمیز و یا به‌طورثابت دارای مشکل در نظر بگیرد.

۲. **آسیب‌پذیربودن از لحاظ عدم تشخیص درست درخواست‌ها؛ آسیب‌پذیری** Double-spending

این آسیب‌پذیری بیش‌تر از ارسال درخواست‌های پیاپی و ایجاد منع سرویس ناشی می‌شود. مهاجم با سوءاستفاده از این آسیب‌پذیری، سیلی از درخواست را به سامانه ارسال می‌کند تا سبب گیج‌شدن سامانه شود. سامانه در اثر اختلال، چندین درخواست را به اشتباه، موفق در نظر می‌گیرد.

پس برای داشتن اتصال امن‌تر به درگاه آنلاین:

- ترتیبی دهید که سامانه پس از دریافت نتیجه‌ی فرآیند پرداخت، بر آن اساس پیغام مناسب را به کاربر نمایش دهد؛ در صورت موفقیت‌آمیز نبودن فرآیند، عملیات را موفق در نظر نگیرد و بالعکس.
- سقف مشخصی برای تعداد دفعات درخواست‌های ارسالی از سمت کاربر تعیین کنید.

۷ آسیب‌پذیری‌های رایج در فرم مشخصات کاربران:

آسیب‌پذیری‌های رایج درفرم‌ها به ویژه فرم‌های مربوط به مشخصات و اطلاعات کاربران سامانه، به ترتیب فراوانی عبارتند از:

۱. **آسیب‌پذیربودن نسبت به ارسال تعداد بی‌شمار درخواست به سرور؛ آسیب‌پذیری HTTP Flooding**

آسیب‌پذیری HTTP Flooding زمانی رخ می‌دهد که برای تعداد درخواست ارسالی از سمت کاربر به سمت سرور محدودیتی وجود نداشته باشد. مهاجم با سوءاستفاده از این آسیب‌پذیری، سیلی از درخواست‌ها را به سمت سرور ارسال می‌کند. این‌گونه، سامانه دچار DoS یا منع سرویس و مختل می‌شود. مهاجم می‌تواند در این اختلال از تزریق داده‌های خود به سامانه سو استفاده کند و با داده‌های جعلی تزریقی خود به سامانه نفوذ کند.

۲. **آسیب‌پذیر بودن از لحاظ عدم اعتبارسنجی تعدد پارامترهای URL؛ آسیب‌پذیری IDOR**

در آسیب‌پذیری IDOR پارامتری در URL مقدار id مربوط به یک دسته اطلاعات را از سوی کاربر دریافت می‌کند و بر اساس آن اطلاعات مربوط را در صفحه، نمایش می‌دهد که مختص کاربر است. عدم اعتبارسنجی درست کاربر، امکان سوءاستفاده‌ی مهاجم از این آسیب‌پذیری و دسترسی به اطلاعات آن صفحه را برای هر کاربر فراهم می‌کند.

۳. **آسیب‌پذیربودن نسبت به تزریق دستورات SQL؛ آسیب‌پذیری SQL Injection**

مهاجم با سوءاستفاده از آسیب‌پذیری SQL Injection، دستورات SQL را به صورت بدخواهانه در قالب دستور مجاز SQLای دیگر در فیلدهای فرم تزریق می‌کند. و این‌گونه، از طریق پرسش‌های متعدد و به سیستم مدیریت پایگاه داده دسترسی می‌یابد و به اطلاعات پایگاه داده‌ی سامانه دست می‌یابد.

۴. **آسیب‌پذیربودن نسبت به تزریق اسکریپت مخرب به فرم‌ها؛ آسیب‌پذیری XSS**

آسیب‌پذیری XSS یا Cross Site Scripting از رایج‌ترین آسیب‌پذیری‌های موجود در وب به شمار می‌رود. مهاجم با سوءاستفاده از این آسیب‌پذیری، اسکریپت‌های مخرب را در ساختار صفحات وب به ویژه در فرم‌ها، تزریق می‌کند. این‌گونه اطلاعات کاربری مانند Sessionها و Tokenهای کاربران دیگر را به دست می‌آورد و با استفاده از آن‌ها وارد حساب کاربری آن‌ها می‌شود.

پس برای داشتن فرم مشخصات امن‌تر:

- سقف مشخصی برای تعداد دفعات درخواست‌های ارسالی از سمت کاربر تعیین کنید.
- پارامترهای ارسالی از سوی کاربر را در سمت کاربر و در سمت سرور اعتبارسنجی کنید.
- مقادیر ورودی کاربر را اعتبارسنجی و کلمات کلیدی زبان SQL را در ورودی‌ها فیلتر کنید.
- مقادیر ورودی کاربر را اعتبارسنجی و برخی کلمات کلیدی زبان Javascript که نشان‌دهنده‌ی مقادیر هستند، را در ورودی‌ها فیلتر کنید.

سخن آخر:

یک مهاجم، می‌تواند اهداف مختلفی مانند از کار انداختن سامانه، نفوذ به اکانت کاربران، دیفیس سایت و ... را داشته باشد. گاهی مهاجم، برای دستیابی به این هدف، با قدم‌های کوچک و پیوسته پیش می‌رود؛ از آسیب‌پذیری‌های مختلف، در راستای رسیدن به هدف کلی خود، استفاده کند. برخی آسیب‌پذیری‌ها ممکن است، به تنهایی خطر زیادی نداشته باشند و آورده‌ی چندانی برای مهاجم به همراه نیاورند، مانند؛ آلوده کردن سامانه، دست‌یابی به اکانت یکی از کاربران عادی سامانه، ایجاد اختلال در سامانه و ... اما مهاجم با ترکیب آورده‌های خود که حاصل آسیب‌پذیری‌های مختلف هستند، می‌تواند به اهداف بزرگ‌تری دست یابد.

روز و روزگارتون امن ؛)