

	گستره: گروه نرم افزاری پیوست
ویرایش: 1	نام: مهمترین و خطرناک ترین حملات سطح وب
تاریخ:	کد: واحد بهبود عملکرد

مقدمه :

امروزه یکی از بزرگترین دغدغه‌های فعالان حوزه آی تی، برقراری امنیت اطلاعات می‌باشد. با پدید آمدن بانک‌های داده‌ای آماری و مالی، حساسیت مسئله صد چندان می‌شود. در ادامه چک لیستی را ارائه می‌نمایم که با کمک آن می‌توانید تا حدود بسیار خوبی امنیت نرم افزار تحت وب خود را برقرار نمایید. در برخی از موارد مثال‌هایی از تکنولوژی میکروسافت آورده شده است که این بدلیل تخصص نویسنده در تکنولوژی‌های میکروسافت می‌باشد. در صورتیکه شما از تکنولوژی‌ها و زبان‌های سورس باز بهره می‌برید، می‌بایست معادل مورد ذکر شده را در زبان مورد استفاده خود بیابید.

ابتدا اجازه دهید مقداری با حملات آشنا شویم و سپس راه مقابله را در کنار هم بررسی نماییم.

مهمترین و خطرناک ترین حملات سطح وب :

حمله XSS

این نوع حملات بدین صورت است که هکر با استفاده از فرم‌های عمومی یا خصوصی (پنل‌های سایت) اقدام به ثبت کدهای مخرب جاوااسکریپت درون دیتابیس شما می‌نماید. همانطور که می‌دانید پایه اصلی سیستم‌های احراز هویت، ساخت فایل کوکی بر روی کامپیوتر کاربر می‌باشد. زمانی که مطلب ثبت شده‌ی هکر برای کاربران شما نمایش داده می‌شود، کدهای جاوا اسکریپت هکر روی مرورگر کاربر، اجرا شده و اطلاعات کوکی‌های کاربر به راحتی برای سایت هکر ارسال می‌شود (معمولا هکر یک صفحه روی وب می‌سازد تا بتواند اطلاعات دریافتی از کدهای جاوا اسکریپت خود را دریافت و در جایی ذخیره کند).

حال هکر به راحتی کوکی را بر روی مرورگر خودش تنظیم می‌کند و بعد وارد سایت شما می‌شود. سیستم شما او را با کاربر شما اشتباه می‌گیرد و به راحتی هکر به اطلاعات پنل کاربری (ان) شما دست پیدا می‌کند.

حمله SQL Injection

این حمله معروفترین حمله است که تقریبا با قدرت می‌توانم بگویم که در تکنولوژی **ASP.Net** با امکانات فوق العاده‌ای که بصورت توکار در دات نت در نظر گرفته شده است، بصورت کامل به فراموشی سپرده شده است. فقط **2** تا نکته‌ی ریز هست که باید در کدهایتان رعایت کنید و تمام.

این حمله بدین صورت است که هکر یک سری دستورات **SQL** را در کوئری استرینگ، به صفحات تزریق می‌کند و بدین صورت می‌تواند در کدهای کوئری **TSQL** شما اختلال ایجاد کند و اطلاعات جداول شما را بدست بیاورد. در این نوع حمله، هکر از طریق باگ سطح کد نویسی کدهای نرم افزار، به دیتابیس حمله می‌کند و اطلاعاتی مثل نام کاربری و کلمه‌ی عبور ادمین یا کاربران را می‌دزد و بعد می‌رود داخل پنل و خرابکاری می‌کند.

حمله CSRF

این حمله یکی از جالب‌ترین و جذاب‌ترین نوع حملات است که هوش بالای دوستان هکر را نشون می‌دهد. عبارت **CSRF**

	گستره: گروه نرم افزاری پیوست
ویرایش: 1	نام: مهمترین و خطرناک ترین حملات سطح وب
تاریخ:	کد: واحد بهبود عملکرد

مخفف Cross Site Request Forgery است (احتمالا دوستان ام وی سی کار، این عبارت برایشان آشناست). در این نوع حمله هکر یک فایل برای کاربر شما از طریق ایمیل یا روش های دیگر ارسال می کند و کاربر را به این سمت سوق می دهد که فایل را باز کند. کاربر یک فایل به ظاهر معمولی مثل عکس یا ... را می بیند و فایل را باز می کند. وقتی فایل باز می شود دیتای خاصی دیده نمی شود و گاهی هم اروری مبنی بر ناقص بودن فایل یا ... به کاربر نمایش داده می شود و کاربر فکر می کند که فایل، ناقص برای ارسال شده...

اما در حقیقت با کلیک بر روی فایل و باز کردن آن یک درخواست **POST** از کامپیوتر کاربر برای سایت شما ارسال می شود و در صورتیکه کاربر در آن زمان در سایت شما لاگین باشد، سایت درخواست را با روی باز می پذیرد و درخواست را اجرا می کند. بدین صورت هکر می تواند درخواست هایی را به سرویس های سایت شما که مثلا برای حذف یک سری داده است، ارسال کند و اطلاعات کاربر را حذف کند.

حمله Brute Force

در این حمله، هکر از یک سری برنامه برای ارسال درخواست های مکرر به فرم های سایت شما استفاده می کند و بدین صورت فرم های عمومی سایت شما مورد هجوم انبوهی از درخواست ها قرار می گیرد که این امر در بهترین حالت موجب ثبت کلی دیتای اسپم در دیتابیس شما و در بدترین حالت موجب داون شدن سایت شما می شود.

حمله DDOS

این نوع حمله مانند حمله **Brute Force** است؛ با این تفاوت که درخواست به همه ی صفحات شما ارسال می شود و معمولاً درخواست ها از چندین سرور مختلف برای سایت شما ارسال می شوند و حجم درخواست ها به قدری زیاد است که عملاً سرور شما هنگ می کند و کاملاً از دسترس خارج می شود. این نوع حمله در سطح کد راه حل زیادی ندارد و در سطح سرور و فایروال باید حل شود و حل آن هم بدین صورت است که درخواست های بیش از حد طبیعی از یک آی پی خاص تشخیص داده شده و به سرعت، آی پی بلاک می شود و از آن به بعد درخواست های آن آی پی در فایروال از بین می رود و دیگه به سرور نمی رسد.

حمله SHELL

شل فایلی است خطرناک که اگر بر روی سرور سایت شما آپلود و اجرا شود، هکر از طریق آن دسترسی کاملی به کل سرور سایت شما خواهد داشت. فایل های دیگری با نام بک دور [1] نیز وجود دارند که نویسنده تمایل دارد آنها را نیز از نوع حمله **SHELL** معرفی نماید. این نوع از فایل ها به مراتب بسیار خطرناک تر از فایل های شل می باشند؛ تا جایی که ممکن است سال ها هکر به سروی دسترسی داشته باشد و مدیر سرور کاملاً از آن بی خبر باشد. اینجاست که باید شدیداً مراقب فایل هایی که روی سایت شما آپلود می شوند باشید. نویسنده به تمامی خوانندگان پیشنهاد می نماید، در صورتیکه نرم افزار حساسی دارند، حتماً از سرور اختصاصی استفاده نمایند؛ چرا که در هاست های اشتراکی که در آنها فضا و امکانات یک سرور بصورت اشتراکی در اختیار چندین سایت قرار می گیرد، وجود باگ امنیتی در سایر سایت های موجود بر روی سرور اشتراکی می تواند امنیت سایت شما را نیز به مخاطره بیاندازد. نویسنده تهیه ی سرور اختصاصی را شدیداً به توسعه دهندگان سایت های دارای تراکنش های بانکی بالا

	گستره: گروه نرم افزاری پیوست
ویرایش: 1	نام: مهمترین و خطرناک ترین حملات سطح وب
تاریخ:	کد: واحد بهبود عملکرد

(داخلی یا خارجی) پیشنهاد می نماید. زیرا درگاه تراکنش های بانکی بر روی آی پی هاست شما قفل می شوند و در صورتیکه سرور بصورت اختصاصی تهیه شده باشد، آی پی سرور شما فقط و فقط در اختیار شماست و هکر نمی تواند با تهیه هاستی بر روی سرور اشتراکی شما، به راحتی آی پی قفل شده در درگاه بانکی شما را در اختیار داشته باشد. بدیهی است تنها در اختیار داشتن آی پی سرور شما جهت انجام خرابکاری در درگاه بانکی شما کافی نیست. ولی به نظر نویسنده این مورد در بدترین حالت ممکن 30٪ کار هکر می باشد. البته بحث حمله شل به سطح مهارت متخصصان سرورها نیز بستگی دارد. نویسنده اظهار می دارد اطلاعات دقیقی از تنظیماتی که بتواند جلوی اجرای انواع شل و یا جلوی دسترسی فایل های شل را بگیرد، ندارد. بنابراین از متخصصان این حوزه دعوت می نماید اطلاعاتی درباره این موضوع ارائه نمایند.

حمله SNIFF

در این نوع حملات، هکر پکت های رد و بدل شده بین کاربران و سرور شما را شنود می نماید و به راحتی می تواند اطلاعات مهمی مثل نام کاربری و رمز عبور کاربران شما را بدست آورد.

پیشنهادهای امنیتی پروژه های نرم افزاری تحت وب

-بررسی کامل ورودی های دریافتی از فرم های سایت؛ هم در سمت کلاینت و هم در سطح سرور.

-در تکنولوژی دات نت به منظور تمیز سازی ورودی ها و حذف تگ های خطرناکی همچون **script**، کتابخانه ای با نام **Microsoft.Security.Application** وجود دارد. کتابخانه های سورس باز دیگری نیز وجود دارند که نمونه آن کتابخانه **AntiXss ()** <https://www.nuget.org/packages/AntiXss/> سایت نوگت **www. Nuget.org** می باشد.

-بررسی کامل ورودی های دریافتی از کوئری استرینگ های **Query String** سایت. اگر از **ASP.Net MVC** استفاده می نمایید، تا حدی زیادی نیاز به نگرانی نخواهد داشت، زیرا تبدیلات **Casting** در سیستم **Model Binding** انجام می پذیرد و این موضوع تا حد زیادی شما را در برابر حملات **SQL Injection** مقاوم می نماید.

-حتما در فرم های عمومی سایتتان از تصویر کپچا با امنیت بالا استفاده نمایید. این موضوع جهت شناخت روبات ها از انسان ها می باشد و شما را در برابر حملات **Brute Force** مقاوم می نماید.

-حتما سیستم شخصی سازی صفحات ارور را فعال نمایید و از نمایش صفحات ارور حاوی اطلاعات مهمی مانند صفحات ارور **ASP.Net** جلوگیری نمایید. این موضوع بسیار حساس می باشد و می تواند نقاط ضعف نرم افزار شما را برای هکر نمایان کند. حتی ممکن است اطلاعات حساسی مانند نام بانک اطلاعاتی، نام کاربری اتصال به بانک اطلاعاتی و نام جداول بانک اطلاعاتی شما را در اختیار هکر قرار دهد.

-استفاده از **ORM** ها یا استفاده از پروسیجرهای پارامتریک. این موضوع کاملاً شما را در برابر حملات **SQL Injection**

	گستره: گروه نرم افزاری پیوست
ویرایش: 1	نام: مهمترین و خطرناک ترین حملات سطح وب
تاریخ:	کد: واحد بهبود عملکرد

مقاوم می‌نماید. کما اینکه **ORM** ها، سطحی از کش را بصورت توکار دارا می‌باشند و این موضوع در سرعت دستیابی به داده‌ها نیز بسیار تاثیر گذار است. از طرف دیگر بانک اطلاعاتی **SQL** نیز امکانات توکاری جهت کش نمودن پرس و جویهای **Procedure** پارامتریک دارد.

-**لاگ کردن** ارورهای سطح کد و سطح روتینگ **Routing**. یکی از مهمترین خصیصه‌های پروژه‌های با کیفیت، لاگ شدن خطاهای سطح کد می‌باشد. این امر شما را با نقاط حساس و ضعف‌های نرم افزار آگاه می‌سازد و به شما اجازه می‌دهد به سرعت در جهت رفع آنها اقدام نمایید. لاگ نمودن خطاهای سطح روتینگ شما را از فعالیت‌های هکرها جهت یافتن صفحات لاگین و صفحات مدیریتی پنل مدیریتی سایت آگاه می‌نماید، همچنین شما را از حملات **SQL Injection** نیز آگاه می‌نماید.

-**جلوگیری از ایندکس شدن** صفحات لاگین پنل مدیریت سایت در موتورهای جستجو. بخش مهمی از عملیات هکرها، قرار دادن روبات‌های تشخیص رمز بر روی صفحات لاگین می‌باشد که به نوعی می‌توان این نوع حملات را در دسته حملات **Brute Force** قرار داد. موتورهای جستجو یکی از ابزارهای مهم هکرها می‌باشد. عملیات‌هایی مانند یافتن صفحات لاگین پنل مدیریتی یکی از کاربردهای موتورهای جستجو برای هکرها می‌باشد.

-**لاگ کردن** ورود و خروج افراد به همراه تاریخ، زمان، آی پی افراد و وضعیت لاگین. با کمک این موضوع شما می‌توانید ورود و خروج کاربران نرم افزار خود را کنترل نمایید و موارد غیر طبیعی و مشکوک را در سریعترین زمان مورد بررسی قرار دهید.

-**استفاده از روال‌های استاندارد جهت بخش "فراموشی کلمه عبور"**. همیشه از استانداردهای نرم افزارهای بزرگ پیروی نمایید. بدیهی است استانداردهای استفاده شده در این نرم افزارها بارها و بارها تست شده و سپس بعنوان یک روال استاندارد در همه نرم افزارهای بزرگ بکار گرفته شده است. استاندارد جهانی بخش "فراموشی کلمه عبور" که در اغلب نرم افزارهای معروف جهان بکار گرفته شده است، عبارت است از دریافت آدرس ایمیل کاربر، احراز هویت ایمیل وارد شده، ارسال یک نامه‌ی الکترونیکی **Email** حاوی نام کاربری و لینک تنظیم کلمه عبور جدید به ایمیل کاربر. بهتر است لینک ارسال شده به ایمیل کاربر بصورت یکبار مصرف باشد. کاربر پس از کلیک بر روی لینک تنظیم کلمه عبور جدید، وارد یکی از صفحات سایت شده و می‌تواند کلمه‌ی عبور جدیدی را برای خود ثبت نماید. در پایان، کاربر به صفحه‌ی ورود سایت هدایت شده و پیامی مبنی بر موفقیت آمیز بودن عملیات تغییر کلمه‌ی عبور به او نمایش داده می‌شود. البته روال ذکر شده حداقل رول استاندارد می‌باشد و می‌توان در کنار آن از روال‌های تکمیل کننده‌ای مانند پرسش‌های امنیتی و غیره نیز استفاده نمود.

-**قراردادن امکاناتی جهت بلاک نمودن آی پی‌ها و غیر فعال نمودن حساب کاربری اعضای سایت**. در نرم افزار باید این امکان وجود داشته باشد که آی پی‌هایی که بصورت غیر طبیعی در سایت فعالیت می‌نمایند و یا مکرراً اقدام به ورود به پنل مدیریتی و پنل کاربران می‌نمایند را بلاک نماییم. همچنین در صورت تخلف کاربران باید بتوان حساب کاربری کاربر خاطی را مسدود نمود. این موضوع می‌تواند بسته به اندازه پروژه و یا سلیقه تیم توسعه بصورت خودکار، دستی و یا هر دو روش در نرم افزار در تعبیه شود.

-**امن سازی سرویس‌های ای جکس و چک کردن ای جکس بودن درخواست ها**. حتما جلوی اجرای سرویس‌های درون نرم افزاری از بیرون از نرم افزار را بگیرید. سرویس‌های ای جکس یکی از این نوع سرویس‌ها می‌باشند که در نرم افزارها جهت استفاده‌های داخلی در نظر گرفته می‌شوند. در این نوع سرویس‌ها حتما نوع درخواست را بررسی نمایید و از پاسخگویی

	گستره: گروه نرم افزاری پیوست
ویرایش: 1	نام: مهمترین و خطرناک ترین حملات سطح وب
تاریخ:	کد: واحد بهبود عملکرد

سرویس‌ها به درخواست‌های غیر ای جکسی جلوگیری نمایند. در **ASP.Net MVC** این امر توسط متد **Request.IsAjaxRequest** انجام می پذیرد.

- محدود کردن سرویس‌های حساس به درخواست‌های **POST**. حتما از دسترسی به سرویس هایی از نوع **Delete, Insert, Update** و **POST** از طریق فعل **GET** جلوگیری نمایید. در **ASP.Net MVC** این سرویس ها را به فعل **ASP.Net Web API** این سرویس ها را به افعال **POST, PUT** و **DELETE** محدود نمایید.

- عدم استفاده از آی دی در پنل‌های کاربران بالاخص در آدرس صفحات (کوئری استرینگ) و استفاده از کد غیر قابل پیش بینی مثل **GUID** به جای آن. حتی الامکان بررسی مالکیت داده‌ها در همه بخش‌های پنل‌های کاربری سایت را جهت محکم کاری بیشتر انجام دهید تا خدای نکرده کاربر با تغییر اطلاعات کوئری استرینگ صفحات نتوانند به داده‌های یک کاربر دیگه دسترسی داشته باشند.

- حتی الامکان پنل مدیران را از کاربران بصورت فیزیکی جدا نمایید. این مورد جهت جلوگیری از خطاهایی است که ممکن است توسط توسعه دهنده در سطح سیستم مدیریت نقش رخ دهد و موجب دسترسی داشتن کاربران به بخش هایی از پنل مدیریتی شود.

- استفاده از الگوریتم‌های کدگذاری ترکیبی و کد کردن اطلاعات حساس قبل از ذخیره سازی در بانک اطلاعاتی. اطلاعات حساسی مانند کلمات عبور را حتما توسط چند الگوریتم کدگذاری، کدگذاری نمایید و سپس درون بانک اطلاعاتی ذخیره نمایید.

- تنظیمات حساس نرم افزار را درون فایل **web.config** قرار دهید و حتی الامکان آنها را نیز کدگذاری نمایید. بصورتی که اطلاعات قابلیت دیکد شدن را داشته باشند.

- ساخت پروژه بصورت چند لایه. این موضوع جهت جلوگیری از دستیابی هکر به ساختار لایه‌های پروژه‌های شما می‌باشد. به بیان دیگر اگر نهایتا هکر بتواند به اطلاعات **FTP** هاست شما دست یابد، استفاده از تکنولوژی چند لایه در بدترین حالت هکر را از دستیابی به اطلاعات لایه‌های زیرین نرم افزار باز می‌دارد. البته این کار برای هکرها غیر ممکن نیست، اما بسیار سخت و زمان بر می‌باشد.

- اشتراک گذاری اینترفیس در سرویس‌های خارج برنامه ای و عدم اشتراک گذاری کلاس اصلی. این موضوع از دستیابی هکر به بدنه سرویس ها و پیاده سازی‌های آنها جلوگیری می‌نماید.

- استفاده از تکنیک‌های مقابله با **CSRF** در همه سرویس‌های **POST**. در **ASP.NET MVC** اتریبیوتی با نام **AntiForgery** جهت مقاوم سازی سرویس‌ها از حملات **CSRF** وجود دارد. مکانیزم بدین صورت است که در تمامی فرم‌های سایت یک کد منحصر به فرد تولید می‌گردد که همراه درخواست **GET** به کامپیوتر کاربر ارسال می‌شود و در هنگام ارسال درخواست **POST** به سرور، صحت کد مورد نظر بررسی شده و در صورت صحت، اجازه‌ی اجرای سرویس به درخواست داده می‌شود. بدین صورت وقتی کاربر سایت شما فایل آلوده‌ای را باز می‌نماید، در خواست ارسالی هکر که توسط فایل باز شده، به سرور سایت ما ارسال می‌گردد، فاقد کد منحصر به فرد بوده و از اجرای سرویس جلو گیری می‌شود.

	گستره: گروه نرم افزاری پیوست
ویرایش: 1	نام: مهمترین و خطرناک ترین حملات سطح وب
تاریخ:	کد: واحد بهبود عملکرد

- استفاده از سیستم های مدیریت نقش امن مانند **IDENTITY** در **ASP.Net MVC** و یا استفاده از امکانات توکار دات نت در سیستم های مدیریت نقش شخصی سازی شده **Custom Role Provider**. بدیهی است امنیت این سیستم ها بارها و بارها تست شده است.

- **بررسی فرمت و پسوند** فایل های آپلود شده. توجه نمایید که بررسی پسوند فایل ها کافی نبوده و فرمت فایل ها نیز می بایست بررسی شود. حتی نویسنده پیشنهاد می نماید فایل ها را به نوع های مرتبطشان تبدیل **Cast** نماید. در حوزه هک باید نمودن انواع ویروس، تروجان، شل و بک دور **Back Door** به فایل های تصویری و متنی یک امر بسیار رایج است. بنابراین حساسیت زیادی روی این موضوع قرار دهید. نویسنده توصیه می نماید کتابخانه های کاملی برای این موضوع تدارک ببینید تا در تمامی پروژه ها نیاز به ایجاد مجدد آنها نداشته باشید و سعی نمایید در هر پروژه این کتابخانه ها را تکمیل تر و بهتر نمایید.

- **تنظیم IIS** جهت جلوگیری از اجرای فایل های اجرایی در مسیر آپلود فایل ها. شاید جمله بیان شده به نظر ترسناک و یا سخت برسد، اما این کار با نوشتن چند تگ ساده در فایل **Web.Config** به راحتی قابل انجام است و نیاز به هیچ نوع کدنویسی ندارد.

- آپلود فایل ها در پوشه **App_Data** و دسترسی به فایل ها از طریق سرویس های خود شما. پوشه **App_Data** پوشه ای امن است و دسترسی مستقیم از طریق آدرس بار مرورگر به فایل های درون آن توسط **IIS** داده نمی شود و افراد فقط از طریق سرویس های خود شما می توانند به فایل های داخل این پوشه دسترسی داشته باشند. بدین صورت در سرویس های خود می توانید با تبدیل نمودن **Cast** فایل ها به نوع خودشان (تصویر. پی دی اف یا ...) هکر را نا امید نمایید. این موضوع شما را در مقابل حملات **SHELL** مقاوم می نماید.

- استفاده از تکنیک های لاگین چند سطحی برای پنل ادمین. در این روش شما حتی با داشتن نام کاربری و کلمه ی عبور ادمین، قادر نخواهید بود وارد پنل ادمین شوید. نویسنده ابزار می دارد که این روش، یک روش ابداعی می باشد که از ترکیبی از احراز هویت ساده توسط نام کاربری و کلمه ی عبور به همراه تکنیک های احراز هویت ایمیل و موبایل مدیریت سایت می باشد.

- استفاده از **SSL** بسیار اهمیت دارد. بالاخص اگر نرم افزار شما **Service Oriented** باشد و نرم افزار شما سرویس هایی جهت اتصال به اپلیکیشن های خارجی مثل اپلیکیشن اندروید دارد. این مورد در صفحات لاگین نیز بسیار مهم است و موجب می شود نام کاربری و کلمه عبور کاربران شما بصورت هش شده بین کامپیوتر کاربر و سرور شما رد و بدل شود و عملا شنود پکت ها فایده ای برای هکر نخواهد داشت، زیرا داده ها توسط الگوریتم های امنیتی که بین سرور و مرورگر کاربران توافق می شود کدگذاری شده و سپس رد و بدل می شوند.

- [1] Back Door
- [2] <https://www.nuget.org/packages/AntiXss/>
- [3] [www. Nuget.org](http://www.Nuget.org)
- [4] Query String
- [5] Casting
- [6] Procedure
- [7] Routing

	گستره: گروه نرم افزاری پیوست
ویرایش: 1	نام: مهمترین و خطرناک ترین حملات سطح وب
تاریخ:	کد: واحد بهبود عملکرد

- [8] Email
- [9] Custom Role Provider
- [10] Cast
- [11] Back Door
- [12] Cast