هر چند سال یک مرتبه سازمان OWASP لیستی از شاخص ترین آسیب پذیری های متداول در نرم افزار ها و سرویس های ارائه شده تحت وب در سر تاسر جهان را از طریق مستندی ارائه می دهد که این لیست مبنای امنیتی نرم افزارهای تحت وب به شمار می رود. خلاصه آخرین آسیب پذیری های منتشر شده در سال ۲۰۱۳ توسط این سازمان (**List ۱۰OWASP Top ۲۰۱۳**) شامل:

۱- تزریق (Injection) :

آسیب پذیری تزریق کد همانند تزریق SQL، OS و LDAP زمانی رخ می دهد که داده های نامعتبر به یک مترجم (Compiler or Interpreter) بجای دستور و یا query ارسال می گردند. هکر از طریق داده های نامعتبر قادر به فریب مترجم شده و امکان اجرای دستورات غیر قانونی و یا روئیت اطلاعات حیاتی بدون مجوز دسترسی برای او فراهم می شود.

روشهای اجتناب:

i. بررسی و اعتبارسنجی ورودی ها
ii. پارامتریک کردن Queryها و Spها
iii. Named SQL parameters

۲- تاییدیه شکسته شده و مدیریت جلسه (Broken Authentication and Session Management) :

فانکشنهای نرم افزارهای کاربردی مرتبط با اعطای مجوز دسترسی و مدیریت Session گاها به درستی پیاده سازی نشده و این امکان را به هکرها می دهد تا به اطلاعات حیاتی همانند رمز های عبور، کلید ها، Session Token در جهت سوء استفاده و جعل هویت دسترسی پیدا کنند.

روشهای اجتناب:

i. استفاده از Membership و Role Provider
ii. حتما از Cookie Session استفاده شود
iii. استفاده از Session Expiration بصورت خودکار و دستی
iv. هر گونه اطلاعات و کلیدهای جداول در صفحات رمز گذاری گردد
v. کنترل رمز گذاری های کاربران جهت انتخاب رمز قوی توسط آنها
vi. امکان reset کردن رمزهای عبور کاربران
vii. امکان بخاطر سپاری کاربران فقط درصورتی که کاربر آن را بخواهد

۳- اسکریپت کراس سایت (Cross-Site Scripting) :

این آسیب پذیری زمانی رخ می دهد که نرم افزار کاربردی، داده های نا امن را بدون اعتبار سنجی برای کاوشگر وب ارسال نماید. هکر توسط این آسیب پذیری قادر به اجرای اسکریپت بر روی کاوشگر قربانی، دزدیدن session و یا تغییر مسیر قربانی به وب سایت های مخرب (malicious sites) خواهد بود.

روشهای اجتناب:

i. بررسی تمامی ورودی ها با توجه به لیست سفید
ii. همیشه باید Requestهای کاربر به سرور اعتبار سنجی شود(URLیا Element Addressing)
iii. استفاده از HTML Encoding در زمان نمایش خروجی های مشکوک
iv. استفاده از کامپوننتهای Anti.XSS
v. استفاده از URL Encoding

۴- ارجاع نا امن به اشیاء داخلی برنامه (Insecure Direct Object References):

این اسیب پذیری زمانی رخ می دهد که برنامه نویس دسترسی ارجاع یک منبع به اشیاء داخلی برنامه را باز گذاشته باشد (همانند فایل، دایرکتوری و یا بانک اطلاعاتی). بدون کنترل دسترسی به این اشیاء هکر قادر به دستکاری منابع در جهت دسترسی به اطلاعات حیاتی خواهد بود.

روشهای اجتناب:

i. پیاده سازی Access Control
ii. استفاده از نگاشت غیر مستقیم به منبع
iii. عدم استفاده از منابع قابل شناسایی

۵- پیکربندی امنیتی اشتباه (Security Misconfiguration) :

امنیت مناسب نیازمند تعریف و استقرار پیکربندی مناسب برای نرم افزار، قالب کاری، وب سرور، بانک اطلاعاتی و سیستم عامل می باشد. تنظیمات امن می بایستی تعریف، پیاده سازی و نگهداری شوند که البته تنظیمات پیش فرض بسیار نا امن می باشند. همچنین می بایستی همیشه نرم افزارها بروز نگهداشته شوند.

روشهای اجتناب:

i. بروزنگهداشتن فریم ورک مورد استفاده
ii. سفارشی سازی خطاهای سیستمی در برنامه
iii. استفاده از tracer
iv. غیر فعال سازی Debug

v. اعتبارسنجی Requestها، شبکه شما را امن نگه میدارد
vi. رمزگذاری داده های حساس
vii. اختصاص حداقل دسترسی به کاربران دیتابیس

۶- افشای اطلاعات حساس (Sensitive Data Exposure) :

بسیاری از نرم افزارهای کاربردی تحت وب بدرستی از اطلاعات محرمانه خود (همانند اطلاعات اعتبار سنجی کاربران و اطلاعات کارت بانکی) محافظت نمی کنند. هکر با دزدیدن این اطلاعات قادر به سوء استفاده از آنها و ایجاد خرابکاری خواهد بود. اطلاعات محرمانه و حیاتی نیازمند محافظت ویژه ای می باشند که از آن جمله می توان به رمز نگاری اطلاعات در زمان تبادل اطلاعات با کاوشگر اشاره نمود.

روشهای اجتناب:

i. کشف داده های حساس و رمزگذاری آنها
ii. استفاده از روش های رمزگذاری متفاوت Hashing, Salting, Encryption
iii. استفاده از Membership Provider
iv. استفاده از روشهای امن مدیریت کلیدهای رمزگذاری

۷- عدم سطح دسترسی مناسب برای دسترسی به فانکشن (Missing Function Level Access Control) :

بسیاری از نرم افزارها قبل از اجرای فانکشن و نمایش خروجی در میانای کاربر (UI)، حق دسترسی را بررسی می نمایند. در نظر داشته باشید که نرم افزار همان سطح دسترسی را می بایستی درسمت سرور بررسی کند. در صورتی که در خواست اعتبار سنجی نگردد، هکر قادر به جعل درخواست در جهت دسترسی به فانکشن ها خواهد بود.

روشهای اجتناب:

i. ماژول مرکزی برای کنترل سطح دسترسی به متد و تابع
ii. عدم اجازه دسترسی در حالت پیش فرض
iii. حفاظت فقط با عدم نمایش دکمه های عملیاتی در لایه نمایش برقرار نمیشود و در سمت سرور نیز سطح دسترسی باید بررسی گردد.

۸- جعل درخواست (Cross-Site Request Forgery) :

این آسیب پذیری، کاوشگر قربانی وارد شده به نرم افزار را مجبور می کند که درخواست HTTP جعل شده را به همراه session's cookie قربانی و سایر اطلاعات مورد نیاز اعتبار سنجی شده را به برنامه کاربردی ارسال نماید. هکر توسط این حمله قادر به جعل هویت کاربر و جرای دستورات مخرب بر روی حساب آن خواهد بود.

روشهای اجتناب:

i. استفاده توکن الگوی همزمان ساز
ii. غیر فعال سازی HTTP Get در صفحات آسیب پذیر

۹- استفاده از کامپوننت ها با آسیب پذیری های شناخته شده (Using Components with Known Vulnerability) :

کامپوننت ها همانند کتابخانه ها، قالب های کاری و سایر ماژول های نرم افزار معمولا با دسترسی کامل اجرا می گردند. در صورتی که آسیب پذیری کامپوننتی افشا گردد، تخریب اطلاعات و دسترسی به سرور امکان پذیر خواهد بود. نرم افزار هایی که از کامپوننت هایی با آسیب پذیری های شناخته شده استفاده می کنند، امکان انواع حمله را برای هکر فراهم می سازند.

روشهای اجتناب:

i. در صورت وجود حذف این کامپوننتها از برنامه و جایگزین کردن آن
ii. پیاده سازی کامپوننت جایگزین
iii. بروز کردن کامپوننت های مورد استفاده

۱۰ - تغییر مسیر های نامعتبر (Unvalidated Redirects and Forwards) :

نرم افزارهای کاربردی دائما در حال تغییر مسیر کاربران به صفحات دیگر می باشند و از داده های نا امن برای تشخیص صفحات مقصد استفاده می کنند. بدون استفاده از اعتبارسنجی مناسب، هکر قادر به هدایت قربانی به وب سایت های مخرب و فیشینگ خواهد بود.

روشهای اجتناب:

i. Implementing referrer checking
ii. مبهم سازی URL و URL Naming
iii. غیر قانونی کردن تغییر مسیرهای نامعتبر