| موضوع | ردیف | شرح ویژگی |
|---|---|---|
| تشخیص و تصدیق هویت | ۱ | برنامه‌ی کاربردی باید قبل از اعطای حق دسترسی به منابع و نقش‌ها، هویت کاربران را تصدیق کند. |
| | ۲ | برنامه‌ی کاربردی باید هویت تمامی برنامه‌ها و موجودیت‌های بیرونی که از طرف کاربر عملی را انجام می‌دهند و با نرم افزار تعامل دارند را بررسی و با حداقل حقوق دسترسی آن را تصدیق کند. |
| | ۳ | قبل از اجازه‌ی دسترسی به منابع، برنامه‌ی کاربردی باید پیام اخطاری را که شامل موارد حساس نیست به کاربر نمایش بدهد؛ مانند:<br>۱) کاربر به سامانه ..... سازمان وارد شده است.<br>۲) سامانه دارای طبقه بندی ..... می‌باشد.<br>۳) اعلان ردگیری و بازبینی فعالیت‌های کاربر وجود دارد.<br>۴) کاربر در قبال اطلاعات حساس مورد دست‌یابی، مسئول می باشد. |
| | ۴ | اطلاعات مشخص شده بر حسب شناسه‌ی کاربری شامل، تاریخ، زمان، آدرس IP و نام دامنه‌ی چند ورود اخیر کاربر و تعداد دفعات تلاش برای ورودهای ناموفق، قبل از آخرین ورود موفق نمایش داده شود. (برای تغییر IP به کاربر پیغام هشدار نشان داده شود) |
| | ۵ | فرایند تشخیص و تصدیق هویت کاربران علاوه بر استفاده از نام‌های کاربران و کلمات عبور باید از توکن‌های سخت‌افزاری استفاده کند. |
| | ۶ | برای برنامه‌های کاربردی با اطلاعات مهم علاوه بر بند ۵ باید یکی از روش‌های زیست سنجی یا زیرساخت کلید عمومی (به صورت دو طرفه) استفاده شود. |
| | ۷ | برنامه‌ی کاربردی باید مطمئن شود که زنجیره‌ای از عامل‌های تصدیق هویت شده بین کارگزار برنامه‌ی کاربردی و سامانه‌های پشت صحنه نظیر سامانه مدیریت پایگاه داده ایجاد شده باشد. |
| | ۸ | تمامی اطلاعات تصدیق هویت کاربران باید از طریق یك مسیر امن ارسال شود. |
| | ۹ | تمامی اطلاعات تصدیق هویت کاربران باید در رسانه‌های ذخیره‌سازی به صورت رمزنگاری شده ذخیره گردد. |
| | ۱۰ | تمامی اطلاعات تصدیق هویت کاربران از طرف کلاینت باید بصورت درهم سازی شده به سرور ارسال گردد. |
| | ۱۱ | تصدیق هویت برنامه‌ی کاربردی نباید به عنوان راهکار جایگزین تصدیق هویت سامانه‌های پشتیبان نظیر سامانه‌های مدیریت پایگاه داده‌ها در نظر گرفته شود. |
| | ۱۲ | برنامه‌ی کاربردی باید با ایجاد واسط‌های کاربری تعداد دفعات تلاش برای ورود ناموفق را، توسط مدیران سامانه تنظیم‌پذیر کند. |
| تشخیص و تصدیق هویت | ۱۳ | برنامه‌ی کاربردی باید با ایجاد سازوکارهایی، کاربرانی را که در یك مدت زمان مشخصی تلاش‌های ناموفقی برای ورود به برنامه‌ی کاربردی داشته‌اند، در زمانی قابل تنظیم، قفل کند. همچنین برنامه کاربردی IP هایی را که برای N بار ارتباط در M دقیقه برای دسترسی خاص تلاش می کند را قفل نماید. |
| | ۱۴ | هیچ نقشی در سیستم نباید بدون اطلاعات تصدیق هویتی باشد. |
| | ۱۵ | برنامه‌ی کاربردی ابتدا باید هویت کاربر را بصورت جداگانه تصدیق کند و سپس صحت ادعای عضویت یك کاربر را برای عضویت او در یك گروه/نقش خاص بررسی کند. |
| | ۱۶ | اطلاعات تصدیق هویتی نباید در کد برنامه ذخیره شود. |
| | ۱۷ | برای جلوگیری از انجام حملات جستجوی حالات ممکن و حمله لغت‌نامه‌ای باید کلمه‌های عبور بر اساس خطمشی انتخاب کلمه‌های عبور انجام شود. از کد امنیتی captcha با الگوی مناسب استفاده شود. |

| | | |
|---|---|---|
| | ۱۸ | مدیریت کلمه‌های عبور در برنامه‌های کاربردی باید دارای خواص زیر باشد:<br>۱) مدیران سامانه باید قادر باشند به کاربران کلمه‌های عبور اختصاص دهند.<br>۲) اجبار کاربران برای تغییر کلمه‌های عبوری که مدیر سامانه به آن‌ها اختصاص داده است بعد از اولین ورود.<br>۳) توانا کردن کاربران به قابلیت تغییر کلمه‌های عبور به صورت دوره‌ای (حداقل دوره ۳ ماهه) بر حسب خط مشی‌های سامانه و یا بر حسب تقاضای کاربر<br>۴) اجبار کاربران به انتخاب کلمه‌های عبور جدید با حداقل ۴ کاراکتر جدید.<br>۵) قابل تنظیم بودن تعداد کلمات عبور قبلی که کاربر نمی‌تواند انتخاب نماید. (حداقل آن ۲ باشد)<br>۶) تعداد کاراکترهای پسورد حداقل ۸ باشد<br>۷) نوع پیچیدگی ۲ نوع و برای هر نوع حداقل ۳ کاراکتر باشد.<br>۸) نمایش گرافیکی قوت و ضعف کلمه عبور. |
| | ۱۹ | برنامه‌ی کاربردی باید تا زمانی که کاربر کلمه‌ی عبور تاریخ مصرف گذشته‌ی خود را عوض نکند، اجازه‌ی ورود به سامانه ندهد. |
| | ۲۰ | برنامه کاربردی باید قابلیت تولید نام کاربری تصادفی و همچنین قابل تولید توسط مدیر را داشته باشد. این قابلیت قابل تنظیم برای مدیر باشد. بصورتی که بتواند هر دو روش را همزمان یا بصورت جداگانه انتخاب نماید. |
| | ۲۱ | برنامه‌ی کاربردی نباید اجازه دهد یك شناسه‌ی کاربر با چند کلمه‌ی عبور انتخاب شده و یا با یك شناسه کاری یکسان امکان ورود چندین کاربر وجود داشته باشد. |
| | ۲۲ | برنامه‌ی کاربردی نباید اطلاعات تصدیق هویت کاربر را در کوکی ها، اسکریپت های سمت کارگزار و یا مشتری ویا دیگر فایل‌هائی که این اطلاعات بتواند از آن بدست آید، ذخیره کند. |
| تشخیص و تصدیق هویت | ۲۳ | برنامه‌ی کاربردی نباید شامل شناسه‌ی کاربری بدون هویت (بدون انتساب به شخص) برای ورود باشد. (سامانه بتواند شناسه کاربری بلااستفاده را بر حسب محدوده زمانی شناسایی و برای مدیر قابل تنظیم باشد- مدیر بتواند کاربرانی که از سازمان خارج شده اند را بدون حذف سوابق آنها غیرفعال نماید) |
| | ۲۴ | اطلاعات برنامه‌ی کاربردی که برای دسترسی به آن‌ها نیازی به راهکارهای تصدیق هویت نیست باید شناسائی شده و از اطلاعات خصوصی که برای دسترسی به آن‌ها نیاز به تشخیص و تصدیق هویت است تفکیك گردد. |
| | ۲۵ | برنامه‌ی کاربردی باید بطور پیش فرض کمترین حق دسترسی را برای شناسه‌ی کاربری در نظر بگیرد. |
| | ۲۶ | رشته‌ی اتصال به پایگاه داده نباید در کد برنامه و یا فایل‌های پیکربندی بصورت شفاف ذخیره گردد. |
| | ۲۷ | برنامه‌ی کاربردی نباید مشخص کند که علت شکست ورود ، کلمه‌ی عبور نادرست بوده است و باید شناسه کاربری را بعد از ورود ناموفق مجددا از کاربر بخواهد. |
| | ۲۸ | برنامه‌ی کاربردی باید تلاش‌های ناموفق برای ورود را برای ممیزی، ردگیری کند. |
| | ۲۹ | اگر کاربر وارد سامانه گردید و سپس برای یک مدت زمان از سامانه استفاده ننمود، جهت استفاده مجدد باید احراز هویت شود. |
| | ۳۰ | جهت انجام هرگونه فعالیت حساس در سامانه احراز هویت مجدد از کاربر باید صورت گیرد. صفحات حساس قابل تنظیم برای مدیر باشد. (برای سامانه‌های مهم شناسه نشست نیز تغییر یابد) |
| تصدیق حقوق دسترسی | ۱ | برنامه کاربردی باید قبل از فراخوانی فرایندها توسط کاربران و یا دسترسی به منابع سامانه ، حقوق دسترسی آنها را بررسی کند و مشخص کند آیا اجازه فراخوانی توابع و یا دسترسی به منبع مربوطه توسط کاربر وجود دارد یا خیر؟ |

| | | |
|---|---|---|
| | ۲ | برنامه‌ی کاربردی باید رابط‌های کاربری لازم را برای ایجاد و مدیریت فهرست کنترل دسترسی ها و سایر اطلاعات حقوق دسترسی تا پایین ترین سطح ( خواندن، نوشتن، ویرایش، تایید، حذف، فایل‌های پیوستی و ... ) را برای هر فعالیت فراهم آورده باشد. |
| | ۳ | حقوق دسترسی اعطا شده به برنامه‌ی کاربردی در هر زمان باید کمترین حق دسترسی مورد نیاز برای انجام فعالیت‌های لازم باشد. |
| | ۴ | سامانه بازه های زمانی دسترسی کاربران به سامانه را مدیریت نماید. ( با استفاده از واسط مدیریتی زمان ( برحسب بازه های زمانی سال، ماه، هفته، روز و ساعت ) دسترسی کاربران/کاربر به سامانه قابل تنظیم باشد ) |
| | ۵ | برنامه کاربردی بتواند مدیریت IP کاربران را انجام دهد و با استفاده از واسط مدیریتی قابل تنظیم باشد بصورتی که مدیر بتواند یکی از تنظیمات زیر را برای کاربر تعریف نماید:<br>۱) کاربر برای بار اول که لاگین می کند IP تنظیم شود و برای مراجعات بعدی فقط از این IP بتواند لاگین کند<br>۲) کاربر از یک رنج IP بتواند لاگین کند.<br>۳) کاربر با رنج های مختلف بتواند لاگین کند.<br>توجه: سرویس عمومی براساس مدیریت IP سامانه صورت گیرد. سرویس‌های مدیریتی علاوه بر سامانه توسط سایر سازوکارهای شبکه نظیر فایروال انجام شود. |
| | ۶ | حقوق دسترسی در برنامه‌ی کاربردی باید به‌صورت نقش‌گرا پیاده‌سازی شود. |
| | ۷ | کاربر انجام وظایف خود را تنها با نگاشت يك نقش بتواند انجام دهد. |
| تصدیق حقوق دسترسی | ۸ | برنامه‌ی کاربردی باید بتواند برچسب‌های محرمانگی و جامعیت مناسب را روی داده اعمال کند و یا کاربران را قادر سازد تا این برچسب‌ها را روی داده‌ها و فراداده‌هایی که ایجاد یا تغییر می‌دهند اعمال کنند. این برچسب‌ها باید بتوانند توسط راهکارهای کنترل دسترسی شناخته شده و استفاده گردند. ملاحظه: برچسب گذاری داده‌ها بر اساس نیاز برای فیلد و رکورد امکان پذیر باشد. |
| | ۹ | اگر کاربری یک آدرس URL نامعتبر را درخواست نماید، برنامه کاربردی نباید فهرست دایرکتوری را به کاربر نشان دهد. |
| | ۱۰ | برنامه‌ی کاربردی نباید به کاربران اجازه دهد با تایپ مستقیم يك URL در خط آدرس مرورگر به صفحاتی که اجازه‌ی دسترسی ندارند، دسترسی پیدا کنند. |
| | ۱۱ | برای امنیت پایگاه داده از رویه‌های ذخیره شده استفاده نموده و کنترل حقوق دسترسی را برای هر رویه‌ی ذخیره شده اعمال نمایید به عنوان مثال ارتباط با پایگاه داده را از طریق رویه انجام دهید. |
| | ۱۲ | داده‌های تولید شده توسط برنامه‌ی کاربردی که به سیستم‌های دیگر انتقال می‌یابند و یا توسط چاپگر چاپ می‌شوند باید برچسب طبقه‌بندی اطلاعات داشته باشند. |
| مدیریت نشست | ۱ | نشست‌ها باید به صورت کد شده باشند. |
| | ۲ | شناسه‌ی نشست‌ها باید بگونه ای باشد که قابل حدس زدن نباشد. |
| | ۳ | برنامه‌ی کاربردی باید راهکار تنظیم زمان غیر فعال بودن کاربر برای منقضی کردن نشست را از طریق واسط‌هایی در اختیار مدیر سامانه قرار دهد. بعد از منقضی شدن نشست، کاربر دوباره باید تصدیق هویت شود. |
| | ۴ | امکان نشست‌های هم‌زمان برای یک کاربر وجود نداشته باشد. |
| | ۵ | اطلاعات حساس نظیر کلمه عبور، آدرس URL نباید در کوکی‌های ماندگار ذخیره شود. |

| | | |
|---|---|---|
| | ۶ | برنامه‌ی کاربردی به هر دلیلی بسته شد باید نشست به اتمام برسد و بتواند فرمان پایان نشست (خروج) را بطور صریح و قابل دسترس در اختیار کاربر قرار دهد. |
| | ۷ | برنامه‌ی کاربردی باید محتویات کوکی های تصدیق هویت را رمز کند. |
| | ۸ | شناسه‌های نشست نباید از طریق Query Stringهای URL انتقال یابد. |
| بررسی صحت ورودی و خروجی | ۱ | تمام ورودی‌ها، خروجی‌ها و ناحیه‌های امن برنامه‌ی کاربردی برای اعمال راه‌کارهای کنترلی باید مشخص شود. |
| | ۲ | داده‌های ورودی باید بر اساس نوع، مقدار، شکل، اندازه و محدوده و نیز پاک‌سازی ورودی بر اساس فهرست کاراکترها و الگوهای بد نیت بررسی و تطبیق داده شوند. |
| | ۳ | بررسی صحت داده‌های ورودی در سطح برنامه‌ی کاربردی باید به صورت متمرکز و با استفاده از توابع و روش های طراحی شده در این راستا انجام شود. |
| | ۴ | سامانه فایل دریافتی اپلود شده را از نظرحجم، تعداد، نوع، طول، محتوا و پسوند اعتبارسنجی کرده و از آپلود فایل‌های مخرب جلوگیری کند. سامانه فایل آپلود شده را ضمن افزودن پسوند مورد انتظار به انتهای نام آن، با نامی هش شده ذخیره کند. |
| | ۵ | داده های خروجی برنامه‌های کاربردی که از طرف کاربر فراهم آورده شده باشند باید براساس نوع، مقدار، شکل، اندازه و محدوده پاکسازی شده و نیز خروجی براساس فهرست کارکترها و الگوهای بد نیت بررسی و تطبیق داده شده تا امکان سوء استفاده نفوذگران از این ناحیه جلوگیری گردد. |
| بررسی صحت ورودی و خروجی | ۶ | در صورتی که برنامه‌ی کاربردی ورودی‌های نامعتبر از کاربر دریافت کرد، ضمن ثبت آن در رویدادنگاری از او درخواست کند مجدداً ورودی را به‌صورت صحیح وارد نماید. در صورت تکرار آن تا ۳ بار فرایند کاربر را خاتمه داده و یك پیام خطای مناسب مبنی بر خاتمه‌ی فرایند به علت ورودی‌های نامعتبر به او نمایش دهد. |
| | ۷ | برنامه کاربردی باید تمامی متغیرها را مقداردهی اولیه نماید. |
| | ۸ | معماری برنامه‌ی کاربردی باید به گونه‌ای باشد که تنها به بررسی صحت ورودی در سمت کاربر اکتفا ننموده و صحت ورودی در سمت سرویس دهنده نیز بررسی شود. |
| دسترس پذیری | ۱ | برنامه‌ی کاربردی نباید به‌گونه‌ای عمل نماید که با تغییر محیط و شرایط فعالیتش، داده‌ها از دسترس خارج شوند. |
| | ۲ | برنامه‌ی کاربردی نباید شامل خطاها و آسیب‌پذیری‌هائی باشد که با سوء استفاده از آن برنامه‌ی کاربردی کارگزار از دسترس خارج شود. |
| | ۳ | برنامه‌ی کاربردی با مدیریت آستانه‌ی بار باید ویژگی در دسترس پذیری را متناسب با سطح طبقه بندی فراهم نماید. |
| مدیریت خطاها و استثنائات | ۱ | در صورت بوجود آمدن خطا در برنامه‌ی کاربردی و باشکست مواجه شدن آن، خطا و یا نقص مربوطه نباید باعث شود برنامه‌ی کاربردی به یك وضعیت ناامن برود. |
| | ۲ | برنامه‌ی کاربردی باید خطاها و استثنائات را با ساختارهای تعریف شده در زبان برنامه‌سازی به شکل مناسب و متمرکز مدیریت کند. |
| | ۳ | برنامه‌ی کاربردی باید به‌گونه‌ای باشد که محتوای خطاهای برگردانده شده به کاربر حاوی اطلاعات حساس نباشد. |
| | ۴ | ساختارهای مدیریت خطاها و استثنائات باید تمامی خطاها واستثنائات بوجود آمده را ثبت کند. |

| | | |
|---|---|---|
| | ۵ | متناسب با سطح طبقه‌بندی و نیاز، برنامه کاربردی باید هر شش سطح خطاء و استثنائات (Info، warn، error، fatal، debug و trace) را ثبت نماید. |
| انکارناپذیری | ۱ | به منظور اثبات اصالت و هویت، برنامه‌ی کاربردی باید قادر باشند تا داده‌های مربوطه را به صورت دیجیتالی امضاء کنند. |
| انکارناپذیری | ۲ | امضاهای دیجیتال استفاده شده در برنامه کاربردی باید با بسترهای امضای دیجیتال فراهم شده در سازمان مربوطه سازگار باشد. |
| انکارناپذیری | ۳ | با استفاده از برچسب های امنیتی نظیر واترمارك، سربرگ و پی نوشت های امنیتی از انکارپذیری در خروجی های برنامه جلوگیری شود. |
| طراحی، معماری و کدنویسی | ۱ | برای استخراج تهدیدات متصور هر برنامه کاربردی ابتدا باید تهدیدات مدل سازی شده و سپس ماتریس تهدیدات ارائه گردد تا بتوان نیازمندی‌های امنیتی آن برنامه کاربردی را متناسب با تهدیدات و سطح طبقه بندی احصاء نمود. |
| طراحی، معماری و کدنویسی | ۲ | با توجه به تهدیدات استخراج شده در بند ۱ سازوکارهای امنیتی متناسب باید طراحی و معماری شود. |
| طراحی، معماری و کدنویسی | ۳ | برای انتقال داده‌ها حتی اگر از پروتکل SSL هم استفاده شده باشد باید به جای متد GET از متد POST استفاده شود. |
| طراحی، معماری و کدنویسی | ۴ | برنامه‌ی کاربردی نباید هیچگونه داده را در اسکریپت ها ذخیره کند. |
| طراحی، معماری و کدنویسی | ۵ | داده‌های دارای طبقه بندی و حساس نظیر کلمات عبور و کلیدهای رمزنگاری نباید در کد برنامه ذخیره گردد. |
| طراحی، معماری و کدنویسی | ۶ | کد اجرائی برنامه‌ی کاربردی از فایل‌هایی که ایجاد و استفاده می‌کند باید مجزا باشد. |
| طراحی، معماری و کدنویسی | ۷ | برنامه‌ی کاربردی نباید داده‌های حساس نظیر کلمات عبور را در نوع‌های تغییر ناپذیر نظیر ثابت‌های زبان Java یا نوع‌هایی که توسط سرویس‌های جمع آوری زباله (Garbage Collected Service ) از حافظه پاك می‌شوند، قرار دهد. |
| طراحی، معماری و کدنویسی | ۸ | برنامه‌ی کاربردی نباید از سرویس‌ها و فناوری‌هایی با مخاطرات امنیتی بالا نظیر Telnet استفاده کند. |
| طراحی، معماری و کدنویسی | ۹ | برنامه‌ی کاربردی نباید شامل توابع و یا متدهای استفاده نشده ای باشند که بطور صریح در برنامه‌ی کاربردی فراخوانی نشده باشند. |
| طراحی، معماری و کدنویسی | ۱۰ | برنامه نباید از مولفه ها و قابلیت های وابسته به سایر بسترهای آماده استفاده نماید و تمامی مولفه ها و قابلیت ها باید طراحی و پیاده سازی شوند، برنامه نباید شامل توابع کتابخانه‌ای غیر ضروری باشد. |
| طراحی، معماری و کدنویسی | ۱۱ | برنامه‌ی کاربردی باید شامل متدهای ساده‌که یك وظیفه‌ی مشخص را انجام می‌دهند باشد و نه یك مولفه که چندین کار پیچیده را انجام می‌دهد و شامل مولفه‌های مستقل باشد تا در صورت وجود آسیب‌پذیری در این مولفه‌ها، تغییر مولفه‌ی مزبور تاثیری در کارکرد سایر مولفه‌ها نداشته باشد. |
| طراحی، معماری و کدنویسی | ۱۲ | برای محافظت از کد برنامه در مقابله با مهندسی معکوس مبهم سازی کد انجام شود، برای سامانه‌های مهم درصورت امکان رمزنگاری انجام شود. |
| طراحی، معماری و کدنویسی | ۱۳ | برنامه‌ی کاربردی نباید بر اساس هیچ پیش فرضی در مورد امنیت سایر اجزاء، طراحی شود. به عبارت دیگر ملاحظات امنیتی باید به صورت مستقل پیاده سازی گردد. |
| طراحی، معماری و کدنویسی | ۱۴ | متناسب با زبان برنامه سازی قواعد کد نویسی امن باید رعایت گردد به گونه ایی که برنامه‌ی کاربردی باید با استفاده از راه کارهای مناسب از آسیب‌پذیری‌های رایج نظیر XSS، CSRF، Injection و Overflow جلوگیری کند. |