تست نرم افزار

پروژه امنیت نرم‌افزار تحت وب باز(OWASP)

آشنایی با پروژه امنیت نرم‌افزار تحت وب باز(owasp)

- تست نفوذ روشی برای ارزیابی یک شبکه یا سیستم توسط شبیه‌سازی حمله به آن‌هاست. تست نفوذ برنامه‌های وب تنها بر روی امنیت برنامه‌های کاربردی تمرکز دارد. Owasp تست نفوذ برنامه‌های وب است که از تست جعبه سیاه استفاده می‌کند. به عبارت دیگر ارزیاب هیچ اطلاعی در مورد برنامه کاربردی مورد تست ندارد.

آشنایی با پروژه امنیت نرم‌افزار تحت وب باز(owasp)

- بنیاد owasp از سال ۲۰۰۱ به عنوان یک موسسه غیر انتفاعی در آمریکا تاسیس شد، owasp یک شبکه اجتماعی است که منحصرا فعالیت خود را متمرکز کرده تا به سازمان‌ها بگوید چگونه می‌توان یک نرم افزار تحت وب امن ایجاد کرد و از آن پشتیبانی کرد . فعالیت‌های owasp منحصرا در حوزه امنیت نرم افزارهای تحت وب است.

آشنایی با پروژه امنیت نرم‌افزار تحت وب باز(owasp)

- Owasp یک پروژه متن باز است که توضیح واضحی از مشکلات امنیتی، ابزارها، الگوها و مراحل کاربردی به شما ارایه می‌دهد و شما می‌توانید از آن‌ها برای حل یا کشف مشکلات امنیتی نرم افزار خود استفاده کنید.
- توصیه‌های owasp بسیار کوتاه و کاربردی هستند و به شما کمک می‌کنند تا به درک درستی از مشکلات خاص و تکنیکی دست پیدا کنید و روش‌های مقابله با آن‌ها را فرا بگیرید.

آشنایی با پروژه امنیت نرم‌افزار تحت وب باز(owasp)

- پروژه امنیت نرم افزار تحت وب باز یک انجمن متن باز است که هدف خود را انحصارا ممکن ساختن توسعه، خرید و نگهداری نرم افزارهای قابل اطمینان و تکیه برای سازمان‌ها قرار داده است. ما از رویکردی که به امنیت به عنوان یک مشکل مردمی، فرآیندی و تکنولوژیکی نگاه می‌کند حمایت می‌کنیم، زیرا موثرترین رویکرد به امنیت نرم افزاری، پیشرفت‌هایی در تمامی این زمینه‌ها را در بر می‌گیرد.

آشنایی با پروژه امنیت نرم‌افزار تحت وب باز(owasp)

در owasp موارد زیر را می‌توان رایگان دریافت کرد:

- استانداردها و ابزارهای مرتبط با امنیت برنامه‌های کاربردی
- اسناد کاملی در مورد امنیت آزمایی برنامه‌های کاربردی، پدیدآوری کدهای ایمن، بازبینی امنیت کدها
- کتابخانه‌ها و کنترل های امنیتی استاندارد

آشنایی با پروژه امنیت نرم‌افزار تحت وب باز(owasp)

- هدف اصلی پروژه استاندارد بررسی امنیتی نرم افزار owasp نرمالسازی دامنه و سطح دشواری بررسی امنیتی یک نرم افزار در بازار بر اساس راهکارهای تجاری است. این استاندارد یک مبنا جهت تست تخصصی مکانیزم‌های امنیتی نرم افزار و همچنین تمام مکانیزم‌های امنیتی تخصصی محیط است که برای رخنه هایی مانند اسکریپت بین سایتی و تزریقات پایگاه داده بر آنها تکیه می‌شود. استفاده از این استاندارد می‌تواند یک سطح اعتماد معقول در امنیت نرم افزارهای وبی ایجاد نماید.

چارچوب فعالیت‌های owasp

این چارچوب دارای فعالیت‌هایی است که باید در مواقع زیر انجام گیرند:

- قبل از شروع ایجاد
- در حین تعریف و طراحی
- در حین ایجاد
- در حین به کار گیری
- نگه داری و عملیات

قبل از شروع ایجاد

قبل از شروع اجرای تست نفوذ باید مراحل زیر انجام گیرد:

- اطمینان از وجود SDLC در جایی که امنیت لازم است: این یکی از بهترین راه‌های جلوگیری از خطاهای امنیتی در تولید برنامه‌های کاربردی بهبود بخش توسط افزودن امنیت به تمامی مراحل آن است. SDLC ساختاری برای تولید نرم افزار است.
- اطمینان از اینکه سیاست و استانداردهای مناسبی برای گروه کاری وجود دارند: در واقع باید معلوم شود که گروه مجری تست از چه روشی برای اجرای تست استفاده می‌کند.
- توسعه معیارها و شرایط سنجش: در مورد معیارهای بررسی کیفیت انجام تست نیز باید شرکت مجری تست و شرکت مورد تست به توافق برسند.

در حین تعریف و طراحی

برای طراحی یک روش مناسب و درست برای اجرای تست نفوذ باید گام‌های زیر انجام شود:

- مرور نیازمندی‌های امنیتی: نیازمندی‌های امنیتی، نحوه کار برنامه کاربردی را از دید امنیتی مشخص می‌کنند. ضروری است که نیازمندی‌های امنیتی بررسی شوند. در واقع منظور از بررسی نیازمندی‌های امنیتی، بررسی فرضیاتی است که باعث به وجود آمدن این نیازها می‌شود.
- مرور طراحی و معماری: برنامه‌های کاربردی‌ای بهترند که گزارشی در مورد طراحی و معماری داشته باشند. با بررسی این گزارش می‌توان اطمینان حاصل کرد که از سطح امنیتی مناسبی برخوردار هستند. شناسایی ضعف‌های امنیتی در این مرحله نه تنها از لحاظ اقتصادی به صرفه است بلکه زمان مناسبی برای انجام تغییرات است.

در حین تعریف و طراحی

- ایجاد و مرور UML: پس از تکمیل طراحی و معماری، مدل UML ایجاد می‌شود. مدل UML در حقیقت نحوه کار برنامه کاربردی را توصیف می‌کند. در صورتی که ضعف‌هایی شناسایی شد، باید راه حل‌هایی برای آن‌ها پیدا کرد.
- ایجاد و مرور مدل‌های تهدید: پس از ایجاد طرح و معماری و ایجاد مدل UML، باید مدلی برای تهدیدات ایجاد کرد و سناریوهای تهدید واقعی ایجاد کرد. در واقع با این تهدیدات اطمینان حاصل می‌شود که معماری و طرح مورد نظر از ایمنی کاملی برخوردار است.

در حین ایجاد

پس از این که یک روش مناسب برای تست طراحی شد اقدام‌های زیر به بهبود کار کمک می‌کنند:

- بررسی خطاهای کد: در این قسمت تیم امنیتی، کد را از لحاظ داشتن خطا بررسی می‌کند. هدف از این مرحله فهم سطح بالای روند انجام کار و ساختار کد تشکیل دهنده برنامه کاربردی است.
- مرور کد: با کمک فهم ساختار کد در قسمت قبل، ارزیاب در این قسمت می‌تواند کد واقعی را از لحاظ ضعف‌های امنیتی بررسی کند.

در حین به کار گیری

در هنگام اجرای تست نفوذ باید تست را به دو قسمت تست نفوذ برنامه کاربردی و تست مدیریت پیکربندی تقسیم شود:

- تست نفوذ برنامه کاربردی: انجام تست نفوذ بر روی برنامه کاربردی در این مرحله این اطمینان را می‌دهد که تمامی مراحل امنیتی به درستی انجام شده است.
- تست مدیریت پیکربندی: تست نفوذ برنامه کاربردی، نحوه کار و ایمنی بودن آن را چک می‌کند.

نگه داری و عملیات

شرکت‌های مجری تست نفوذ نیز باید تدابیری پس از انجام تست انجام دهند که عبارتند از:

- مرور مدیریت عملیات: نیاز به فرآیندی است که نحوه مدیریت برنامه‌ها و ساختارها را از لحاظ امنیتی بررسی کند.
- انجام بررسی‌های دوره‌ای امنیتی: بررسی‌های امنیتی به صورت ماهانه و یا فصلی برای اطمینان از این که خطر جدیدی به وجود نیامده است لازم است.
- اطمینان از تایید تغییرات: پس از انجام تغییرات اطمینان حاصل کنید که این تغییرات سطح امنیتی را تحت تاثیر خود قرار نداده است.

مخاطره‌های مهم در مورد امنیت برنامه‌های کاربردی تحت وب

- تزریق(injection): خطر تزریق زمانی رخ می‌دهد که داده‌های نامطمن به عنوان بخشی از یک دستور یا پرسمان به یک مفسر ارسال می‌شود. داده‌های مهاجمان، می‌تواند مفسر را به اجرای دستورات ناخواسته و یا دسترسی غیر مجاز به اطلاعات وادار می‌کند.
- مدیریت نشست و اعتبارسنجی فروشکسته: در بیشتر مواقع بخشی از عملیات برنامه کاربردی که به احراز هویت و مدیریت نشست ارتباطی مرتبط است، به طرز صحیحی پیاده سازی نمی‌شود. این مسیله به مهاجمان اجازه حمله به گذر واژه‌ها، نشان واره و به کارگیری هویت سایرین را می‌دهد.

مخاطره‌های مهم در مورد امنیت برنامه‌های کاربردی تحت وب

- پیکربندی نادرست امنیت: برای تامین امنیت نیازمند تنظیمات دقیق امنیتی برای برنامه‌های کاربردی، چارچوب‌ها، خدمت دهنده‌ها وب، خدمت دهنده‌های نرم افزار، سرور پایگاه داده، پلت فرم و ... هستیم. لازم است تا تمامی این تنظیمات به درستی تعریف، اجرا و نگهداری شود زیرا بسیاری از آنها با پیش فرض‌های امن عرضه نشده است. همچنین نگهداری به روز تمام نرم افزارها را نیز شامل می‌شود.
- نمایش داده‌های حساس: بسیاری از برنامه های کاربردی از داده‌های حساس به درستی محافظت نمی‌کنند. مانند کارت‌های اعتباری، شناسه مالیاتی و اعتبارنامه‌های احراز هویت. مهاجمان ممکن است از این حفاظت ضعیف اطلاعات برای انجام سرقت استفاده کنند. داده‌های حساس نیازمند حفاظت بیشتری نظیر رمزنگاری در همه حالات چه در حال سکون و چه در حال انتقال هستند. همانند اقدامات احتیاطی ویژه زمانی که اطلاعات با مرورگر ردوبدل می‌شود.

مخاطره‌های مهم در مورد امنیت برنامه‌های کاربردی تحت وب

- کنترل دستیابی سطح عملکرد مفقوده: تقریبا تمام برنامه‌های کاربردی وب عملکرد قوانین سطح دسترسی را قبل از اینکه در رابط کاربر قابل مشاهده باشد بازبینی می‌کنند. با این حال برنامه‌های کاربردی زمانی که هر تابعی مورد دسترسی قرار می‌گیرد، نیاز به چک کردن همان کنترل دسترسی بر روی سرور دارند. اگر درخواست‌ها بازبینی نشوند، مهاجمان قادر خواهند بود درخواست‌ها را به منظور دسترسی به عملکردهای غیر مجاز جعل کنند.

- تغییر مسیر دهی و انتقال‌های نامعتبر: برنامه‌های کاربردی تحت وب به طور مرتب کاربران را به صفحات وبگاه‌های دیگر هدایت می‌کنند و از داده‌های غیر ایمن برای تعیین صفحات مقصد استفاده می‌کنند. بدون اعتبارسنجی مناسب، مهاجمان می‌توانند قربانیان را به وبگاه‌های مخرب یا صفحات غیر مجاز هدایت کنند.

استانداردی برای تایید امنیتی برنامه‌های کاربردی تحت وب

- Owasp استانداردی برای تایید امنیتی برنامه‌های کاربردی مبتنی بر وب ارایه نموده است. ASVS(application security verification standard) یک استاندارد برای وارسی امنیت برنامه‌های کاربردی است و می‌تواند برای ایجاد سطحی از اطمینان در امنیت برنامه‌های کاربردی تحت وب استفاده شود.

سطوح وارسی امنیت یک برنامه کاربردی

- ASVS چهار سطح از وارسی را تعریف می‌کند که با افزایش سطح وسعت و عمق‌شان بیشتر می‌شود. وسعت هر سطح با مجموعه‌ای از نیازمندی‌های امنیتی که باید در آن سطح برآورده شوند تعریف می‌شود و عمق هر وارسی با رویکرد و درجه دقتی که لازمه وارسی هر نیازمندی امنیتی است، تعریف می‌شود.
- اگر برنامه کاربردی تمام نیازمندی‌های یک سطح را براورده کند به عنوان برنامه کاربردی سطح N در OWASP ASVS شناخته می‌شود. N نشان دهنده سطحی است که برنامه کاربردی با آن مطابقت دارد.

سطح۱- وارسی خودکار

- سطح ۱، برای برنامه‌های کاربردی مناسب است که نیاز به اطمینان در استفاده از کنترل‌های امنیتی دارند. تهدید کنندگان امنیت معمولا در این سطح شامل ویروس‌ها و کرم‌ها می‌شوند. محدوده‌ی وارسی شامل کدی است که برای ساختن برنامه کاربردی ایجاد و یا اصلاح شده است. در سطح ۱، وارسی شامل استفاده از ابزارهای خودکاری می‌شود که به وسیله‌ی وارسی دستی تکمیل شده‌اند. این سطح تنها شامل بخشی از وارسی امنیت برنامه کاربردی می شود.توجه کنید که وارسی دستی، وارسی برنامه‌های کاربردی را در این سطح کامل نمی‌کند بلکه درستی وارسی خودکار را تحقیق می‌کند.
- سطح ۱ از دو مولفه تشکیل شده است. سطح 1A برای استفاده از ابزارهای خودکار برای پویش آسیب پذیری برنامه کاربردی به کار می‌رود(تحلیل پویا). سطح 1B به منظور استفاده از ابزارهای خودکار برای پویش کدهای منبع به کار می‌رود(تحلیل ایستا). برای وارسی ممکن است از هر کدام از این مولفه‌ها به صورت مستقل یا ترکیبی از هر دو برای رسیدن به درجه کامل سطح ۱ استفاده شود.

سطح۱- وارسی خودکار

- در سطح ۱ مولفه‌های برنامه کاربردی ممکن است به صورت انفرادی یا گروهی از فایل‌های منبع، کتابخانه‌ها و یا فایل‌های اجرایی تعریف شوند. در سطح۱ فهرست گزارش نیازی به مرتب شدن یا سازماندهی ندارد، فقط باید مشخص شود که کدام مولفه جزیی از برنامه کاربردی و کدام جزیی از محیط IT است. با برنامه کاربردی می‌توان به عنوان گروهی از مولفه‌ها با یک موجودیت واحد و یکپارچه رفتار کرد. در این سطح مسیر یا مسیرهایی که درخواست کاربر نهایی از آن‌ها به برنامه کاربردی برده می‌شود، نیاز به مستند سازی یا شناسایی ندارند.

سطح 1A- پویش پویا(وارسی خودکار جزیی)

نیازمندی‌های وارسی کنترل امنیت در پویش پویا

- پویش پویا که با نام پویش آسیب پذیری برنامه کاربردی نیز شناخته می‌شود هنگام اجرای برنامه کاربردی شامل استفاده از ابزارهای خودکار برای دسترسی به واسطهای برنامه کاربردی می‌شود تا آسیب پذیری را در کنترل‌های امنیتی برنامه کاربردی شناسایی کند. به یاد داشته باشید که پویش پویا برای وارسی درستی طراحی، پیاده سازی و استفاده از کنترل امنیت کافی نیست، اما برای وارسی در سطح ۱ قابل قبول است. محدوده‌ی وارسی به وسیله نیازمندی‌های معماری امن این سطح تعریف می‌شود.

سطح 1B- پویش کد منبع(وارسی خودکار جزیی)

نیازمندی‌های وارسی کنترل امنیت در پویش کد منبع

- پویش کد منبع که با نام تحلیل ایستا نیز شناخته می‌شود شامل استفاده از ابزارهای خودکار برای جست و جوی کد منبع در برنامه کاربردی می‌شود تا الگوهایی را پیدا کند که آسیب پذیری را نشان می‌دهند. به یاد داشته باشید که تحلیل ایستا برای وارسی درستی طراحی، پیاده سازی، و استفاده از کنترل امنیت کافی نیست اما برای وارسی در سطح ۱ قابل قبول است. محدوده‌ی وارسی یه وسیله نیازمندی‌های معماری امن این سطح تعریف می شود.

سطح ۲ وارسی دستی

- معمولا برای برنامه‌های کاربردی مناسب است که وظیفه مدیریت تراکنش‌های شخصی، اداره تراکنش های تجاری-تجاری، پردازش اطلاعات کارت اعتباری یا پردازش اطلاعات شخصی را به عهده دارند.سطح ۲ اطمینان می‌دهد که کنترل‌های امنیتی نه تنها درست کار می‌کنند بلکه به درستی نیز استفاده شده‌اند.
- تهدید کنندگان امنیت در این سطح ویروس‌ها، کرم‌ها و حمله‌کنندگان با ابزارهای حرفه‌ای و منبع باز هستند. محدوده وارسی شامل تمام کد تولید یا اصلاح شده برای برنامه کاربردی، همچنین آزمایش امنیت برای تمام مولفه‌های شخص ثالث فراهم کننده‌ی امنیت برنامه کاربردی می شود. سطح دو از دو مولفه سازنده تشکیل شده است.

سطح ۲ وارسی دستی

- اگر برنامه کاربردی نیازمندی‌های یکی از سطوح 2A یا 2B را برآورده نمی‌کند، نمی‌توانیم آن را به عنوان یکی از برنامه‌های کاربردی سطح ۲ به حساب آوریم. به علاوه به دلیل بالاتر بودن سطح ۲ از سطح ۱، نیازی به اجرای ابزارهای خودکار برای برآوردن نیازمندی‌های سطح ۲ نیست. وارسی کننده فقط اجازه دارد از تکنیک‌های دستی برای برآوردن نیازمندی‌ها استفاده کند. البته اگر نتایج ابزارهای خودکار موجود باشند، وارسی کننده ممکن است از آن‌ها برای تایید تحلیل‌ها استفاده کند.
- توجه کنید که برآوردن نیازمندی‌های سطح ۱ که مشابه آن در سطح ۲ نیز وجود دارد به معنی این نیست که این نیازمندی‌ها در سطح ۲ نیز برآورده شده‌اند.

سطح ۲ وارسی دستی

- محدوده وارسی:
- شامل تمام کدی می‌شود که برای ایجاد برنامه کاربردی تولید یا اصلاح شده است. این نیازمندی در سطح ۱ معرفی شده است.
- محدوده وارسی شامل تمام کد برای قالب‌های کاری شخص ثالث، کتابخانه و سرویس امنیتی می‌شود که توسط امنیت برنامه کاربردی فراخوانی شده یا امنیت برنامه کاربردی را پشتیبانی می‌کند. این نیازمندی‌ها در سطح ۲ جدید است.
- وارسی کننده ممکن است به عنوان قسمتی از وارسی از پویش خودکار یا تحلیل کد استفاده کند، اما وارسی خودکار در این سطح نمی‌تواند جایگزین بازبینی دستی کد که لازمه‌ی وارسی هر نیازمندی در سطح ۲ است، شود. اگر نتایج پویش کمکی به وارسی کننده در جهت سریع‌تر شدن کارها یا تکمیل شدن نتایج وارسی کند، بی شک می‌تواند در برآوردن نیازمندی‌های سطح ۲ استفاده شود.

سطح 2A- آزمون امنیت(وارسی دستی جزیی)

نیازمندی‌های وارسی کنترل امنیت برای آزمون نفوذپذیری دستی برای برنامه کاربردی

- آزمون دستی برای امنیت برنامه کاربردی شامل ایجاد آزمون‌های پویا برای وارسی مناسب بودن طراحی، پیاده‌سازی و استفاده از کنترل های امنیتی می‌شود. محدوده وارسی به وسیله نیازمندی‌های معماری امن این سطح تعریف می‌شود.
- در جای مناسب وارسی کننده از نمونه سازی برای تصدیق استفاده موثر از کنترل امنیت استفاده می‌کند. وارسی کننده ممکن است یک الگوی آسیب پذیری مستند کند تا تولید کنندگان برنامه‌های کاربردی بتوانند از آن استفاده کنند.

سطح 2B- بازبینی کد(وارسی دستی جزیی)

نیازمندی‌های وارسی کنترل امنیت به صورت بازبینی دستی کد

- بازبینی دستی کد، شامل تحلیل کد منبع برنامه کاربردی به منظور وارسی طراحی، پیاده سازی و استفاده درست از کنترل‌های امنیتی می شود. برای چنین تحلیلی می‌توان از ابزارهای ساده‌ای مثل ویرایش‌گرها یا IDE استفاده کرد. محدوده وارسی به وسیله نیازمندی‌های معماری امن این سطح تعریف می‌شود.
- در جای مناسب، وارسی کننده از نمونه سازی برای تصدیق استفاده موثر از کنترل امنیت استفاده می‌کند. وارسی کننده ممکن است یک الگوی آسیب پذیری مستند کند تا تولید کنندگان برنامه‌های کاربردی بتوانند از آن استفاده کنند.

سطح ۳- وارسی طراحی

- معمولا برای برنامه‌های کاربردی مناسب است که تراکنش‌ها تجاری-تجاری مهم را مدیریت می‌کنند. از جمله‌ی این تراکنش‌ها می‌توان پردازش اطلاعات بهداشتی درمانی، پیاده سازی عملیات بحرانی یا حساس تجاری را نام برد. تهدیدهای امنیتی در این سطح توسط ویروس‌ها، کرم‌ها و حمله‌کنندگان به وجود می‌آید. محدوده وارسی شامل تمام کد تولید یا اصلاح شده برای برنامه کاربردی، همچنین آزمایش امنیت برای تمام مولفه‌های شخص ثالثی می‌شود که امنیت را برای برنامه کاربردی فراهم می‌کنند. سطح ۳ اطمینان می‌دهد که کنترل‌های امنیتی به درستی کار می‌کنند و در هر جایی از برنامه کاربردی برای اجرای سیاست‌های خاص برنامه استفاده می‌شوند. سطح ۳ به چند مولفه شکسته نمی‌شود.

سطح ۳- وارسی طراحی

- محدوده وارسی: محدوده وارسی شامل تمام کدی می‌شود که برای ساختن برنامه کاربردی تولید یا تصحیح شده است. این نیاز در سطح ۱ معرفی شده است.
- محدوده وارسی شامل تمام کد برای قالب‌های کاری شخص ثالث، کتابخانه و سرویس امنیتی شود که توسط امنیت برنامه کاربردی فراخوانی می‌شود یا امنیت برنامه کاربردی را پشتیبانی می‌کند. این نیاز در سطح ۲ معرفی شده است.
- محدوده وارسی شامل کد برای تمام کتابخانه‌ها و سرویس‌های مرتبط با برنامه کاربردی می‌شود. این نیاز در سطح ۳ جدید است.

سطح ۴- وارسی داخلی

- سطح ۴ معمولا برای برنامه‌های کاربردی مناسب است که از زندگی و ایمنی، زیر ساخت‌های بحرانی یا عملکردهای دفاعی پشتیبانی می‌کنند. همچنین، این سطح ممکن است برای برنامه‌های کاربردی که دارایی‌های حساس را پردازش می‌کنند نیز مناسب باشد. سطح ۴ اطمینان می‌دهد که کنترل‌های امنیتی به درستی کار می‌کنند و در هر جایی از برنامه کاربردی که نیاز به اجرای سیاست‌های خاص برنامه کاربردی است، استفاده می‌شوند. در این سطح، تهدیدهای امنیتی از طرف حمله کنندگان انجام می‌شود و محدوده وارسی نسبت به سطح ۳ گسترش می‌یابد تا تمام کد استفاده شده به وسیله برنامه کاربردی را شامل شود.

سطح ۴- وارسی داخلی

محدوده وارسی

- محدوده وارسی شامل تمام کدی می‌شود که برای ساختن برنامه کاربردی تولید یا تصحیح شده است. این نیازمندی در سطح ۱ معرفی شده است.
- محدوده وارسی شامل کدی می‌شود برای تمام قالب کاری شخص ثالث، کتابخانه و سرویس امنیتی که توسط امنیت برنامه کاربردی فراخوانی می‌شود یا امنیت برنامه کاربردی را پشتیبانی می‌کند. این نیازمندی در سطح ۲ معرفی شده است.
- محدوده وارسی شامل کد برای تمام کتابخانه‌ها و سرویس‌های مرتبط با برنامه کاربردی می‌شود. این نیازمندی در سطح ۳ معرفی شده است.
- محدوده وارسی شامل تمام کد باقی مانده مرتبط با برنامه کاربردی می‌شود. برای مثال می‌توان چارچوب‌ها، کتابخانه‌ها، محیط‌های زمان اجرا، ابزارهای توسعه و ابزارهای ساخت را نام برد.

سطح ۴- وارسی داخلی

- همانند سطح ۳ در سطح ۴ نیز معماری برنامه کاربردی باید ثبت شود. به علاوه در این سطح، تمام کد برنامه کاربردی، کد شامل شده نه فقط کدی که صریحا آزمایش شده، باید به عنوان قسمتی از تعریف برنامه کاربردی مشخص شود. این کد باید شامل تمام کتابخانه‌ها، قالب‌های کاری و کد حمایت شده‌ای که مورد اعتماد برنامه کاربردی است باشد. وارسی قبلی این مولفه‌ها می‌تواند دوباره به عنوان قسمتی از یک وارسی دیگر مورد استفاده قرار گیرد. از جمله سکوهایی که کدهای آن‌ها شامل ارزیابی در سطح ۴ نمی‌شوند می‌توان سیستم عامل، ماشین مجازی یا کتابخانه‌های اصلی به وجود آمده از محیط ماشین مجازی، سرویس دهنده‌ی وب یا سرویس دهنده برنامه کاربردی را نام برد.

جزییات وارسی نیازمندی‌ها

- این بخش از OWASP ASVS، جزییات وارسی نیازمندی‌هایی را تعریف می‌کند که از نیازمندی‌های سطح بالای موجود در هر کدام از سطوح وارسی تعریف شده در این استاندارد، به دست آمده‌اند.

نیازمندی‌های مستند سازی معماری امن

- برای تمام سطوح ASVS مستند سازی اطلاعات مربوط به معماری امنیتی برای اطمینان تکمیل و صحت وارسی امنیتی انجام شده بر روی نرم افزار، امری ضروری است. تحلیل‌ها می‌توانند مورد توجه قرار گیرند و نتایج آنها در معماری امنیتی سطح بالای نرم افزار بازخورد پیدا کنند. این نیازمندی‌ها با یک سطح پایه از جزییات معماری امنیتی که باید حاصل شود آغاز می‌شوند. این سطح از جزییات با هر سطح گسترش می‌یابد.

نیازمندی‌های وارسی مدیریت نشست

- نیازمندی‌های مربوط به مدیریت نشست به صورت مجموعه‌ای از نیازمندی‌ها برای استفاده‌ی امن از درخواست‌ها و پاسخ‌های HTTP، نشست‌ها، کوکی ها و سرآیندها تعریف می‌شود.

نیازمندی‌های وارسی کنترل دسترسی

- کنترل دسترسی برای urlها، کارکردهای تجاری، داده‌ها، سرویس‌ها و فایل‌ها انجام می‌گیرد. در اکثر نرم‌افزارها کنترل دسترسی باید در چندین مکان متفاوت در میان لایه‌های مختلف نرم افزار اجرا شود.

نیازمندی‌های وارسی اعتبارسنجی

- نیازمندی‌های اعتبار سنجی مجموعه نیازمندی‌هایی را برای معتبر ساختن ورودی تعریف می‌کند که در نتیجه آن استفاده از آن‌ها توسط نرم‌افزار، امن خواهد بود.

نیازمندی‌های وارسی کدگذاری خروجی

- نیازمندی‌های کدگذاری خروجی‌ها به این منظور است که از کدگذاری درست خروجی مطمن شویم.

نیازمندی‌ها(ادامه)

نیازمندی‌های وارسی رمزنگاری

- نیازمندی‌های لازم برای رمز نگاری به صورت مجموعه‌ای از نیازمندی‌ها شامل بررسی رمزگذاری برنامه‌ی کاربردی، مدیریت کلیدها، اعداد تصادفی و عملیات درهم سازی می‌باشد. برنامه‌ی کاربردی باید همیشه از ماژول‌های پنهان سازی معتبر استفاده کند.

نیازمندی‌های وارسی ثبت و کنترل خطا

- نیازمندی‌های لازم برای ثبت و کنترل خطا به منظور پیگیری حوادث امنیتی و شناسایی رفتار حمله می‌باشد.

نیازمندی‌ها(ادامه)

نیازمندی‌های وارسی حفاظت داده‌ها

- معمولا حفاظت برای داده‌های حساس مانند شماره‌ی کارت اعتباری، شماره‌ی گذرنامه و اطلاعات شخصی انجام می‌گیرد.

نیازمندی‌های وارسی امنیت ارتباطات

- نیازمندی‌های وارسی امنیت ارتباطات، مجموعه نیازمندی‌هایی را که می‌توانند برای وارسی اینکه تمام ارتباطات نرم افزار به طور مناسب امن شده باشند مورد استفاده قرار بگیرند، تعیین می‌کنند.

نیازمندی‌ها(ادامه)

نیازمندی‌های وارسی امنیت HTTP

- مجموعه نیازمندی‌هایی را که می‌توانند برای بررسی امنیت درخواست‌ها، پاسخ‌ها، نشست‌ها، کوکی‌ها، هدرها و گزارش گیری‌های HTTP مورد استفاده قرار بگیرند، تعیین می‌کنند.

نیازمندی‌های وارسی پیکربندی امنیتی

- نیازمندی‌های وارسی پیکربندی امنیتی مجموعه نیازمندی‌هایی را تعیین می‌کنند که می‌توانند برای بررسی نگهداری ایمن از تمام اطلاعات پیکربندی که رفتار امنیتی نرم افزار را هدف قرار می‌دهند، مورد استفاده قرار بگیرند. حفاظت این اطلاعات پیکربندی در اجرای امن نرم افزار حیاتی است. این نیازمندی‌ها در مکان مخصوص به خود ذخیره شده و روی رفتار برنامه‌ی کاربردی تاثیر می‌گذارد.

نیازمندی‌ها(ادامه)

- نیازمندی‌های وارسی جستجوی کدهای مخرب
- برای سطح ۴، جستجو برای کد مخرب در هر کدی که پس از وارسی سطح ۳ هنوز وارسی نشده است، لازم است.

- تیازمندی‌های وارسی امنیت داخلی
- هدف نیازمندی‌های امنیت داخلی، حفاظت برنامه‌ی کاربردی از خودش می‌باشد تا در برابر جریان‌های پیاده‌سازی ایستادگی کند.

نیازمندی‌های گزارش وارسی

- یک گزارش OWASP ASVS شامل توصیفی از برنامه‌ی کاربردی است که در آن نیازمندی‌های ASVS برای هر سطح تحلیل شده است.

مقدمه‌ی گزارش:

- در مقدمه‌ی گزارش باید اطلاعات کافی فراهم شود تا گزارش و برنامه‌ی کاربردی که گزارش برای آن نوشته شده است، مشخص باشند.
- در مقدمه‌ی گزارش باید میزان اطمینان از برنامه‌ی کاربردی به صورت خلاصه آورده شود.
- در مقدمه‌ی گزارش باید کلید ریسک‌های تجاری که با عامل برنامه‌ی کاربردی مشارکت داشتند، مشخص شود.
- در مقدمه‌ی گزارش باید کلید ریسک‌های تجاری که با عامل برنامه‌ی کاربردی مشارکت داشتند، مشخص شود.
- در مقدمه‌ی گزارش باید قوانین درگیری و مشکلاتی که در رابطه با بررسی موجود بوده و نیز حوزه بررسی بیان شود.

نیازمندی‌های گزارش وارسی

- توصیف برنامه‌ی کاربردی
- در توصیف برنامه‌ی کاربردی باید شرح کافی از برنامه‌ی کاربردی فراهم شود، به گونه‌ای که به فهم عملیات و آشنایی با محیط عمل برنامه کمک کند.

- معماری امنیتی برنامه‌ی کاربردی
- در قسمت معماری امنیتی برنامه‌ی کاربردی باید جزییاتی که اطمینان برنامه‌ی کاربردی را فراهم می‌کند، آورده شود. این قسمت گزارش، زمینه‌ی تحلیل را فراهم می‌آورد و جزییات متفاوت هر سطح را بر اساس استاندارد بیان می‌کند.

- نتایج وارسی
- در این قسمت نتایج تحلیل بر اساس نیازمندی‌های ASVS بیان می‌شود.