مجله محاسبات نرم، جلد ۱۰، شماره ۲ (پاییز و زمستان ۱۴۰۰)، صفحه ۷۲ـ۸۳
https://doi.org/10.22052/scj.2022.242857.0

مجله محاسبات نرم
SOFT COMPUTING JOURNAL
تارنمای مجله: scj.kashanu.ac.ir

دانشگاه کاشان
University of Kashan

مروری بر نیازمندی‌های امنیتی در فرآیند تولید نرم‌افزار✧

حانیه قربانی‌مقدم[۱]، دانشجوی کارشناسی، بهناز جاماسب[۲]، دانشجوی دکتری، حامد دهدشتی جهرمی[۱*]، دانشیار

[۱] دانشکده فنی و مهندسی، دانشگاه جهرم، جهرم، ایران.

[۲] دانشکده مهندسی کامپیوتر و فناوری اطلاعات، دانشگاه صنعتی شیراز، شیراز، ایران.

اطلاعات مقاله

تاریخچه مقاله:
دریافت ۰۲ خرداد ماه ۱۴۰۰
پذیرش ۲۵ تیر ماه ۱۴۰۱

کلمات کلیدی:
سیستم‌های ایمن
نیازمندی‌های امنیتی
تولید نرم‌افزار
روش‌های نیازمندی‌های امنیتی
فاکتورهای امنیتی

چکیده

در سال‌های اخیر مساله افزایش امنیت نرم‌افزار موجب تحقیقاتی گسترده در فرآیند تولید نرم‌افزار شده است. یکی از جنبه‌های اصلی مهندسی سیستم‌های ایمن، شناسایی نیازمندی‌های امنیتی برای محافظت از دارایی‌ها است. جنبه هزینه، زمان و کارایی هر یک از روش‌های شناسایی نیازمندی‌های امنیتی، موجب ارائه روش‌های جدید پیاده‌سازی امنیت در سیستم‌های نرم‌افزاری شده است. این روش‌های امنیتی به منظور افزایش اطمینان از سیستم‌ها و محافظت از دارایی‌های سازمان در چرخه‌های تولید نرم‌افزار پیاده‌سازی می‌شود. مهندسان باید اطمینان حاصل کنند که نرم‌افزاهای تولید شده در مقابل تهدیدهای بالقوه و حملات مهاجمان ایمن است. شناسایی، طبقه‌بندی، اولویت‌بندی و اندازه‌گیری نیازمندی‌های امنیتی از نظر امنیت حریم‌خصوصی، ساختار نرم‌افزار و دیگر موارد بخشی از نگرانی‌های توسعه‌دهندگان نرم‌افزار می‌باشد. در این مقاله گزیده‌ای از روش‌های نوین نیازمندی‌های امنیتی و فاکتورهای امنیتی آن، از سال ۲۰۱۷ تا ۲۰۲۲ جمع آوری شده و به مقایسه فاکتورهای هر یک از روش‌ها برحسب بیشترین تکرار آنها پرداخته شده است. این تحقیق با معرفی چگونگی کارکرد روش‌های نوین به توسعه‌دهندگان، موجب انتخاب بهتر و گزینش مناسب‌ترین روش می‌شود. همچنین به افرادی که به دنبال پژوهش‌های اخیر در حوضه نیازمندی‌های امنیتی هستند، کمک شایانی می‌کند.

۱. مقدمه

رشد چشم‌گیر نرم‌افزارهای در حال استفاده موجب نگاهی تخصصی‌تر به تولید نرم‌افزار شده است. از فرآیند SDLC[۱] در توسعه نرم‌افزار استفاده می‌شود که در مهندسی نرم‌افزار و رشته‌های مرتبط با موضوع سیستم‌های اطلاعاتی مورد استفاده قرار می‌گیرد و موضوع آن تشریح فرآیندهای مرتبط با برنامه‌ریزی، هدف‌گذاری، تحلیل، تولید، آزمایش، استقرار و نگهداری سیستم‌های نرم‌افزاری است که اغلب در یک چرخه دوام و تکرار پیدا می‌کنند. این چرخه همچنین می‌تواند مشتمل بر فرآیندهای مربوط به تامین نرم‌افزارها دانسته شود [۱-۳]. فرآیند مهندسی سیستم و نرم‌افزار تمام فعالیت‌های مربوط به کشف، مستندسازی و نگهداری مجموعه نیازمندی‌های یک سیستم را دربرمی‌گیرد. نیازمندی‌های سیستم، توصیفی از

✧ نوع مقاله: مروری
* نویسنده مسئول
پست(های) الکترونیک: haniyeghm.1996@gmail.com (قربانی‌مقدم)
b.jamasb@sutech.ac.ir (جاماسب)
jahromi@jahromu.ac.ir (دهدشتی جهرمی)

[1] System Development Life Cycle

نحوه ارجاع به مقاله: قربانی‌مقدم، حانیه، جاماسب، بهناز، دهدشتی جهرمی، حامد، «مروری بر نیازمندی‌های امنیتی در فرآیند تولید نرم‌افزار»، مجله محاسبات نرم، جلد ۱۰، شماره ۲، ص۷۲ـ۸۳، پاییز و زمستان ۱۴۰۰.

کارهایی است که یک سیستم باید انجام دهد. این نیازمندی‌ها، خواسته‌های مشتریان از سیستم را مشخص می‌کند که می‌تواند کنترل کردن یک دستگاه، سفارش کالا یا جستجوی اطلاعات باشد. فرآیند یافتن، تحلیل، مستندسازی و بررسی کردن این خواسته‌ها و محدودیت‌ها، مهندسی نیازمندی (RE)[1] نام دارد. از آنجا که امنیت سیستم در کشورهای در حال توسعه مساله بسیار مهمی است از این رو ویژگی‌های دسترس‌پذیری، امنیت، قابلیت اطمینان، ایمنی و انعطاف‌پذیری باید لحاظ شود. عدم توجه به امنیت دنیای نرم‌افزار، سواستفاده، صدمات و خسارت‌های جبران‌ناپذیری را به دنبال می‌آورد که موجب نگرانی کاربران و طراحان نرم‌افزار شده است. این قضیه به نوبه خود باعث توجه به مهندسی نیازمندی‌های امنیتی شده است. مهندسی نیازمندی‌های امنیتی وسیله‌ای مناسب برای آشکار شدن و الگوسازی در زمینه سواستفاده‌ها و صدمات دنیای نرم‌افزار است. بسیاری از طراحان نرم‌افزار بر این باورند که پیاده‌سازی امنیت امری دست و پا گیر، دارای هزینه‌های اضافه برای سازمان و اجرا کردن آن غیرمتعارف است؛ ولی عدم توجه به این نکته ممکن است موجب خسارت‌های سنگین و جبران‌ناپذیری شود. طراحان باید توجه داشته باشند که پیاده‌سازی امنیت زمانی ممکن است که از همان مراحل ابتدایی تولید نرم‌افزار، موارد امنیتی مورد توجه قرار گیرد. در مهندسی نیازمندی‌های امنیتی، روش‌ها و الگوهایی ارائه شده است که با توجه به سطح خواسته‌های کاربران و طراحان از این الگوها استفاده می‌شود. هدف از نوشتن این مقاله، مرور، جمع‌آوری و بررسی گزیده‌ای از روش‌های نوین در ۵ سال اخیر می‌باشد و می‌تواند به افرادی که به دنبال پژوهش‌های اخیر در حوضه نیازمندی‌های امنیتی هستند، بسیار کمک کند. از این رو هدف ما پاسخ به سوالات تحقیق زیر است:

- در پنج سال اخیر چه مقالاتی در رابطه با مهندسی نیازمندی‌های امنیتی ارائه شده است؟
- در مقالات به دست آمده از چه فاکتورهای امنیتی استفاده شده است؟
- کدام یک از مقالات دنباله‌روی بیشتری از فاکتورهای امنیتی بیان شده دارد؟

با توجه به موارد ذکر شده، بررسی نیازمندی‌های امنیتی، شامل فاکتورهایی است که در جدول (۱) آورده شده است.

جدول (۱) – فاکتورهای نیازمندی‌های امنیتی [۵]

فاکتورهای امنیتی		توضیح فاکتور
نیازمندی‌های	شناسایی[2]	فاکتورهای مورد نیاز شناسایی افراد به منظور ورود به سیستم می‌باشد.
	احراز هویت[3]	سیستم بایستی تمام کاربرانی که سعی در دسترسی به سیستم دارند را شناسایی و تایید کند.
	مجوز[4]	به فرآیندی که در آن به تایید اینکه شما به چه چیزهایی دسترسی دارید، می‌پردازد.
	مصونیت[5]	مصونیت، نسبت به مقابله با خطا برای خدمات ضروری سیستم‌ها ارائه می‌شود.
	یکپارچگی[6]	یکپارچگی اطلاعات به این معنا است که تنها سیستم‌ها و افرادی مجاز می‌توانند در داده‌ها تغییر ایجاد کنند.
	تشخیص نفوذ[7]	سیستم باید سیستم تشخیص نفوذ را برای نظارت بر هر نوع دسترسی غیرمجاز پیکربندی کند.
	شناسایی عدم انکار[8]	طرفین درگیر یک نزاع حقوقی، نتوانند صحت و اعتبار یک قرارداد یا متنی که پیش از این طراحی شده است را انکار نمایند.
	حریم خصوصی[9]	سیستم بایستی قابلیت حفظ و جلوگیری از افشای اطلاعات به افراد غیرمجاز را داشته باشد.
	دسترس‌پذیری[10]	اطلاعات توسط کاربران مجاز قابل دسترسی و استفاده است.
	محرمانگی[11]	اطلاعات برای اشخاص یا پروسه‌های غیرمجاز مورد دسترسی یا افشا قرار نمی‌گیرد.

۲. مروری بر کارهای پیشین

در این بخش، به بررسی روش‌های موجود در چند مقاله مرتبط می‌پردازیم. در مرجع [۴] روش نقشه‌برداری سیستماتیک امنیت چابک در مهندسی نیازمندی‌ها ارائه شده است که این مطالعه از

[1] Requirement Engineering

[2] Identification Requirements
[3] Authentication Requirements
[4] Authorization Requirements
[5] Immunity Requirements
[6] Integrity Requirements
[7] Intrusion detection Requirements
[8] None-repudiation Requirements
[9] Privacy
[10] Availability
[11] Integrity

جنبه روش‌های چابک، مراحل RE، نوع حل مساله، نوع تحقیق و ارزیابی تجربی انجام شده است. برای مقایسه، در ابتدا شش سوال زیر مطرح شده است:

۱. کدام یک از روش‌های چابک ارائه شده است؟
۲. رویکردهای ارائه‌شده، به کدام مراحل RE پاسخ می‌دهند؟
۳. نیازمندی‌های امنیتی در هر یک از مراحل، چگونه اعمال می‌شود؟
۴. نوع جنبه‌های تحقیق چیست؟
۵. کدام نوع ارزیابی تجربی بوده است؟
۶. رویکردهای شناسایی شده با چه محدودیت‌هایی روبه‌رو است؟

سپس مراحل زیر شرح داده شده است:

أ) اهداف و سوالات پژوهش
ب) در استراتژی جستجو از یک جستجوی ترکیبی پایگاه داده، مبتنی بر رشته در کتابخانه دیجیتالی Scopus استفاده شده و سپس مجموعه مقالات شناسایی شده است.
ج) انتخاب مطالعه
۱. اولین فیلتر با تعریف کردن معیارهایی برای خروجی در جدول، دسته‌بندی شده است.
۲. دومین فیلتر با خواندن عناوین و خلاصه مقالات، حذف شده است.
۳. دستورالعمل snowballing guidelines دنبال شده است [۶، ۷].
د) مطابق با سوالات مشخص شده در ابتدای کار، مقالات استخراج و طبقه‌بندی شده است.

در مرجع [۸]، طی بررسی تحقیقات ۲۰ سال اخیر، تصفیه‌های تکراری در این مقاله منجر به ۵ نوع اصلی دانش شده که توسط روش‌های SRE استفاده شده‌اند و شامل (۱) الگوهای امنیتی، (۲) طبقه‌بندی و هستی‌شناسی، (۳) الگوها و پروفایل‌ها، (۴) کاتولوگ‌ها و مدل‌های عمومی و (۵) مخلوط از آنها می‌باشند. همچنین در این مرجع چارچوبی برای تجزیه و تحلیل و مقایسه روش‌های SRE پیشنهاد شده است. به هر حال، مراحل اصلی مطالعه نقشه‌برداری سیستماتیک شامل تعریف سوالات تحقیق، جستجوی مقالات مربوطه، غربالگری مقالات بر اساس چکیده آنها و طبقه‌بندی است که باید بترتیب عملیات تعریف سوالات تحقیق، بررسی دامنه، انجام جستجو، غربالگری مقالات، یافتن مقالات مرتبط، بررسی کلمات کلیدی استفاده شده، طبقه‌بندی طرح، فرآیند استخراج و ترسیم نقشه و نقشه سیستماتیک انجام شود. هدف اصلی تهیه چنین چارچوبی، مراجعه دانشجویان و محققان برای یافتن پاسخ سوالات مختلف در مقام مقایسه است. در مرجع [۹]، برخی از مدل‌های SRE معرفی و از دو روش برای مقایسه آنها استفاده شده است که شامل (۱) انجام تجزیه و تحلیلی برای انتخاب سازگاری روش‌ها با RA[1] و MDE[2] و (۲) انجام مقایسه برحسب آیتم‌های ریسک، تهدید، ذینفعان و CIA[3] است. در کاری مشابه، مرجع [۱۰] به معرفی برخی از مدل‌های SRE پرداخته و سپس هر یک از آنها را برحسب CIA مقایسه نموده است.

مرجع [۱۱]، به جمع‌آوری مقالات برای تعیین نیازمندهای امنیتی رسمی پرداخته است. پرسش‌های اولیه این مقاله عبارتند از: (۱) به چه موضوعات تحقیقاتی پرداخته شده است، (۲) مشخصات نیازمندی‌های امنیتی با روش‌های رسمی چه مواردی هستند و (۳) چالش‌های تحقیقاتی در این زمینه چیست. طبق معیارهای ورودی، خروجی و ارزیابی کیفیت، جدول (۲) مراحل بررسی مشخصات نیازمندی‌های امنیتی توسط روش‌های رسمی را نشان می‌دهد.

۳. معرفی روش‌های بررسی شده

روش نیازمندی‌های امنیتی، مبتنی بر الگویی رویکردی برای نوشتن کامل نیازمندی‌های امنیتی: در مرجع [۱۲]، یک ابزار نمونه اولیه به نام SecureMEReq با استفاده از زبان PHP و MVC (که یک برنامه تعاملی را به سه جزء مدل، دید و کنترلر تقسیم می‌کند)، برای بهبود استخراج نیازمندی‌های امنیتی ایجاد

[1] Risk Analysis
[2] Model-driven Engineering
[3] Confidentiality,Integrity,Availability

شده است که دارای چهار ویژگی کلیدی: (۱) استخراج نیازهای امنیتی ذینفعان – مشتری، (۲) ارزیابی نیازمندی‌های امنیتی احتمالات انبوه و نگارش انبوه، (۳) بررسی نیازمندی‌های امنیتی و ساختار کلیدی اجزا و (۴) ارزیابی کامل اولویت‌بندی می‌باشد. در این مرجع، از کتابخانه‌های secLib (که کتابخانه‌ای استاتیک برای کنترل جریان اطلاعات است) و SRCLib (که کتابخانه‌ای برای قابلیت هک شدن است)، به منظور پشتیبانی از روند توسعه سیستم اتوماسیون، به ویژه نوشتن نیازمندی‌های امنیتی استفاده شده است. این مقاله طبق سوال «روش مبتنی بر الگو چگونه به شما در استخراج نیازمندی‌های امنیتی کمک می‌کند؟» پیش رفته است. مراحل این روش در جدول (۳) بیان شده است.

جدول (۲): مراحل بررسی مشخصات نیازمندی‌های امنیتی توسط روش‌های رسمی

ردیف	مرحله بررسی مشخصات
۱	مقالات تحقیقاتی که در مورد مشخصات نیازمندی‌های امنیتی، اطلاعاتی را ارائه می‌دهند.
۲	مقالات پژوهشی که قبل از سال ۲۰۰۰ منتشر شده‌اند.
۳	مقالاتی که معیارهای ورود را نداشته کنار گذاشته شده است.
۴	آیا مقاله یافته‌های معتبری با داده‌های پشتیبانی گزارش می‌دهد؟
۵	آیا نویسندگان دیگر، از این مقالات استفاده کرده‌اند.
۶	آیا معیارهای خروج و ورود به درستی توضیح داده شده‌اند؟
۷	آیا جستجوی این مقاله، همه مطالعات مرتبط را پوشش داده است؟

جدول(۳): توضیح مراحل رویکرد مبتنی بر الگو

ردیف	مراحل رویکرد مبتنی بر الگو
۱	مرحله اول شامل سه فرآیند که اجزای مورد نیاز امنیتی به دست آمده را RE در متنی طبیعی استخراج می‌کند.
۲	مرحله دوم بررسی می‌کند که به ارزیابی نیازمندی‌های امنیتی احتمالات انبوه و نگارش انبوه در فرآیند نیاز است.
۳	مرحله سه به بررسی نیازمندی‌های امنیتی و ساختار کلیدی اجزا می‌پردازد.
۴	در مرحله چهارم، نتایج ارزیابی کامل اولویت‌بندی، سطح کامل بودن نیازمندی‌های امنیتی را نشان می‌دهد.

معماری سطح بالای این ابزار شامل سه لایه، ارائه، پردازش تجارب و لایه داده‌های مدیریتی می‌باشد. در این طرح، معماری لایه‌ها از یکدیگر جدا بوده و این استقلال سبب بهبود عملکرد، نگهداری آسان و مقیاس‌پذیری بیشتر می‌گردد. در لایه اول (لایه ارائه)، تعامل بین کاربران و سیستم کنترل می‌شود. در واقع در این لایه، کاربر از طریق این ابزار، با کنترل‌کننده جزء تعامل پیدا می‌کند. کنترل‌کننده حاوی فیلمنامه‌ای از سمت مشتری است که درخواست HTTP و منطق تجاری این ابزار را کنترل کرده و ورودی را به عنوان یک رویداد دریافت و آن را بسته به درخواست خدمات برای MODEL یا VIEW، ترجمه می‌کند. زمانی که کاربری به ابزار دسترسی پیدا می‌کند، اسکریپت‌ها در کنترلر، نوع مرورگر و دستگاه مورد استفاده توسط وی را تعیین کرده و سپس نمایشی صحیح از مولفه‌های VIEW را درخواست می‌کنند. در لایه پردازش تجارب، سرور apache میزبان پیاده‌سازی php برای رویدادهای اصلی این ابزار می‌باشد که شامل عناصر اصلی استخراج اجزای مورد نیاز نیازمندی‌های امنیتی از متن است. در لایه مدیریت داده، سرور پایگاه داده MySQL حاوی کتابخانه‌های نیازمندی‌های امنیتی و کتابخانه تراکم می‌باشد. در این روش از فاکتورهای امنیتی شامل احراز هویت، شرایط مجوز و نفوذ استفاده می‌شود.

روش تحلیل تهدید برای استخراج نیازمندی‌های امنیتی در ماشین یادگیری مبتنی بر سیستم: مرجع [۱۳]، به کاربرد مرسوم مدل‌سازی تهدید کتابخانه‌های حمله و آدرس‌دهی امنیت، ماشین یادگیری مبتنی بر سیستم (MLBS)[1] در مرحله نیازمندی‌ها پرداخته است. همچنین در مراجع [۱۴–۱۶]، مطالعاتی در رابطه با MLBS انجام شده و تمامی حالت‌ها مورد بررسی قرار گرفته است. در این مراجع از DFD[2] به عنوان یکی از روش‌های اصلی برای تعریف و تحلیل سیستم‌های مبتنی بر داده و از STRIDE[3] به عنوان یک مدل طبقه‌بندی که تهدیدها را بر اساس دسته‌های کلاهبرداری، دستکاری، انکار، افشای اطلاعات، انکار خدمات و ارتقای مزایا طبقه‌بندی می‌کند، نام برده شده است. در این روش اصطلاح ALM[4] به حملاتی اطلاق می‌شود که فرض می‌کنند راهکاری مبتنی بر هوش مصنوعی در سیستم دفاعی مورد استفاده قرار گرفته‌اند. به هر حال، دیدگاه کلی روش تحلیل

[1] Machine Learning Based System
[2] Data Flow Diagrams
[3] Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service
[4] Adversarial Machine Learning

تهدید برای استخراج نیازمندی‌های امنیتی در MLBS به موارد زیر دسته‌بندی می‌شود.

۱. شناسایی تهدیدهای مربوط به MLBS با استفاده از نمودار جریان داده (DFD) و STRIDE شامل:

۱,۱. ایجاد یک مدل معماری نرم‌افزاری برای MLB با استفاده از DFD.

۱,۲. تدوین طبقه‌بندی تهدید ALM بر اساس نقص داده‌ها، استخراج مدل مهندسی معکوس از رفتار یا خروجی سیستم و نیز استخراج داده‌ها- بازیگران تهدید، تهیه نقشه طبقه‌بندی تهدید ALM برای DFD و بیان شکاف‌های مفهومی بین تهدیدهای ALM و STRIDE.

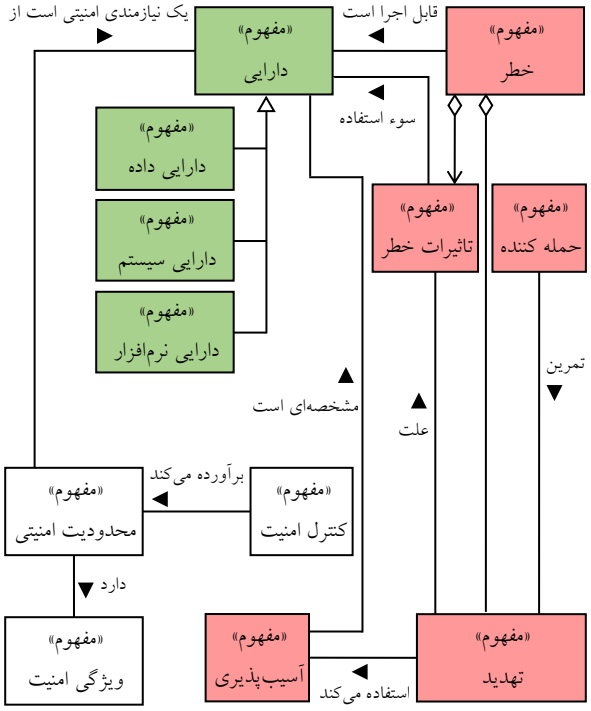
۱,۳. استفاده از STRIDE برای شناسایی تهدیدهای ALM مربوط به MLBS.

۲. استفاده از نوار اشکال Microsoft AL/ML در رتبه‌بندی تاثیرات تهدیدهای شناسایی شده.

۳. استخراج کاهش تهدیدات ALM با استفاده از کتابخانه حمله Microsoft AI/ML.

در نهایت باید اشاره کرد که در این روش، از فاکتورهای امنیتی شامل عدم کلاهبرداری، دستکاری، انکار و جلوگیری از افشای اطلاعات استفاده شده است.

روش ادغام نیازمندی‌های امنیتی با MBSE[1]: در مرجع [۱۷]، مشخصات امنیتی MBSE، یک تیم چندوجهی را قادر به انجام تجزیه و تحلیل امنیت به موازات فرآیند مهندسی سیستم در یک پروژه MBSE می‌کند. در این روش، از مدل‌سازی چارچوب معماری یکپارچه (UAF)[2]، ارزیابی ترکیبی آسیب و ایمنی برای سیستم‌های اطلاعاتی (CHASSIS)[3]، ارزیابی امنیت با زبان مدل‌سازی سیستم (SysML Sec)[4] و زبان مدل‌سازی یکپارچه (UML Sec)[5]، به منظور تجزیه و تحلیل امنیت استفاده شده است. در ادامه، مدل دامنه مشخص شده با ابزار مدل‌سازی MagicDraw در نمودار کلاس UML، مفاهیم امنیتی و روابط آنها را توصیف می‌کند. سه گروه در مدل دامنه امنیتی متمایز هستند که در شکل (۱) نشان داده شده‌اند. آنها شامل (۱) مفاهیم تضمین امنیت (گره‌های سفید)، (۲) مواردی که باید مورد حفاظت قرار گیرد (گره‌های سبز) و (۳) نقاط ضعف احتمالی سیستم (گره‌های قرمز) هستند. در واقع، مدل دامنه امنیتی اجازه می‌دهد که خطرهای مختلف با توجه به شرایط و ایجاد روابط منطقی موجود، طبقه‌بندی شوند.



شکل (۱): مدل دامنه امنیتی [۱۷]

از قابلیت پروفایل داخلی UML 2.5 به منظور تغییر مفاهیم امنیتی مشخص شده در مدل دامنه SysML Sec و از ISO 27001 به منظور مبنای نمایه امنیتی استفاده شده است. در واقع، ISO 27001 در رابطه با شناسایی عناصر زیر صحبت می‌کند.

۱. دارایی‌های موجود در محدوده ISMS[6] و صاحبان آنها

[1] Model-Based System Engineering
[2] Unified Architecture Framework
[3] The combined harm assessment of safety and security for information systems
[4] Systems Modeling Language
[5] Unified Modeling Language
[6] Information Security Management System

۲. تهدیدات این دارایی‌ها
۳. آسیب‌پذیری‌هایی که ممکن است مورد استفاده تهدیدها قرار گیرد
۴. تاثیراتی که ممکن است به وجود بیاید و موجب از دست دادن محرمانگی، یکپارچگی اطلاعات و در دسترس بودن دارایی‌ها شود. به همین دلیل برای پشتیبانی از این مرحله کلیشه‌های خطر، دارایی (شامل دارایی داده، دارایی سیستم و دارایی نرم‌افزار)، تهدید، آسیب‌پذیری و تاثیرات خطر ایجاد شده است.

به هر حال در روش ادغام نیازمندی‌های امنیتی با MBSE، از فاکتورهای امنیتی شامل دسترس‌پذیری، محرمانگی و یکپارچگی استفاده شده است.

چارچوب مبتنی بر هدف با اتخاذ فرآیند SQUARE[1] برای حریم خصوصی و نیازمندی‌های امنیتی: در مرجع [۱۸]، این چارچوب ارائه شده است. این چارچوب مبتنی بر شناسایی اهداف امنیتی مربوط به دارایی‌های سازمان با اتخاذ فرآیند SQUARE است. در واقع، این چارچوب توسط دارایی‌های سیستم و اقدامات امنیتی پشتیبانی می‌شود. برای پیاده‌سازی این چارچوب باید به ترتیب، شناسایی عناصر مبتنی بر هدف (شامل دارایی‌ها، خواص امنیتی، اقدامات و اقدامات امنیتی)، موافقت با تعاریف، شناسایی دارایی‌ها و اهداف امنیتی، الگوی هدف پیشنهادی، توسعه و طراحی مصنوع، تجزیه و تحلیل خطرات، انتخاب روش استخراج، تجزیه و دسته‌بندی نیازمندی‌ها و بازرسی انجام شود. الگوی هدفی که در این مرجع ذکر شده، در جدول (۴) و مراحل پیاده‌سازی آن در جدول (۵) قابل مشاهده است. در کل، در این روش از فاکتورهای امنیتی شامل حریم خصوصی[2]، مسئولیت[3]، محرمانه بودن[4]، یکپارچگی[5] و احراز هویت[6] استفاده شده است.

[1] Security quality requirement engineering
[2] Privacy
[3] Accountability
[4] Confidentiality
[5] Integrity
[6] Authentication

جدول (۴): الگوی هدف مرجع [۱۸]

افدامات امنیتی	خواص امنیتی	دارایی	عامل	نوع اقدام
پیشگیری، کشف و پاسخ به یک رخنه	حریم خصوصی	از دارایی‌ها	چه زمانی عامل کار را انجام می‌دهد	خواندن
	مسئولیت			ایجاد، خواندن، حذف، به‌روزرسانی
	محرمانه بودن			خواندن، ذخیره‌سازی
	یکپارچگی			ایجاد، به‌روزرسانی، حذف
	احراز هویت			ایجاد، خواندن، حذف، به‌روزرسانی، جستجو

جدول(۵): مراحل پیاده‌سازی الگوی مبتنی بر هدف

ردیف	مراحل پیاده‌سازی الگوی مبتنی بر هدف
۱	کلیه‌ی دارایی‌های سازمان شناسایی شود.
۲	در مورد تعریف خصوصیات امنیتی، شناسایی و توافق شود.
۳	کلیه اقدامات برای مدیریت سازمان شناسایی شود.
۴	اهداف مربوط به خواص امنیتی مختلف بر پایه اعمالی انجام شده در دارایی‌ها، شناسایی شود.
۵	اهداف مربوط به اقدامات امنیتی شناسایی شود.
۶	الگوی هدف بر روی دارایی تنظیم شود.
۷	بر اساس مرحله ۶ عملکرد جدیدی شناسایی شود.
۸	دارایی‌های جدید را که ممکن است بر اساس مرحله‌های ۶ و ۷ در سیستم ایجاد شده باشند، شناسایی شوند.

رویکرد مبتنی بر هستی‌شناسی برای طبقه‌بندی خودکار نیازمندی‌های امنیتی: در مرجع [۱۹]، رویکردی شامل یک لایه مفهومی و یک زبان ارائه شده است که نیازهای امنیتی را نه تنها بر اساس شواهد لغوی، بلکه در دانش حوزه مفهومی نیز درک می‌کند. به طور خاص در این روش، یک رویکرد سیستماتیک برای شناسایی ویژگی‌های زبانی نیازمندی‌های امنیتی، لایه مفهومی را به لایه زبانی متصل کرده است. که این ویژگی‌ها بر پایه یک نیازمندی امنیتی گسترده، هستی‌شناسی و دانش زبانی می‌باشند. در ادامه، از چنین ویژگی‌های زبانی برای آموزش طبقه‌بندی نیازهای امنیتی مستقل از دامنه با استفاده از فناوری‌های یادگیری ماشین استفاده می‌شود. به منظور تنظیم ماشین یادگیری باید موارد بیان شده در جدول (۶) تهیه شوند.

جدول (۶): نیازمندی‌های روش پیشنهادی

ردیف	نیازمندی‌های روش پیشنهادی
۱	یک هستی‌شناسی نیازمندی های امنیتی که شامل محدودیت، اطمینان، اجازه و انطباق می‌باشد و به عنوان یک مفهوم اساسی تعریف می‌شوند.
۲	یک فرآیند سیستماتیک تحت عنوان قوانین اتمی و قوانین ترکیبی، برای به دست آوردن ویژگی‌های زبانی نیازمندی‌های امنیتی بر اساس هستی‌شناسی.
۳	استخراج کلید واژه نیازمندی‌های امنیتی که ترکیبی از واژه‌کاوی مبتنی بر فرکانس و استخراج مبتنی بر نگارش و سپس بررسی این نتایج به صورت دستی تا مجموعه نهایی کلمات کلیدی مورد نیاز امنیت به دست آید.
۴	استفاده از از TF-IDF در استخراج کلمات مبتنی بر فرکانس که یک آمار عددی است و می‌تواند اهمیت یک کلمه را برای یک سند در مجموعه اسناد نشان دهد.
۵	استخراج کلمات مبتنی بر نگارش
۶	آموزش طبقه‌بندی نیازمندی‌های امنیتی
۷	پردازش متن برای ایجاد مجموعه داده‌های آموزشی که دارای سه مرحله تولید درختان تجزیه، تطبیق کلمات کلیدی و تطبیق قوانین زبانی است.

در تنظیم یادگیری ماشین، ۳۵ قانون زبان و ۱۴۰ کلمه کلیدی، به عنوان ویژگی‌های زبانی نیازمندی‌های امنیتی در نظرگرفته شده است که همه این ویژگی‌ها به عنوان ویژگی‌های Boolean تعریف می‌شوند. با توجه به مورد ۷ در جدول (۶)، یک ابزار نمونه اولیه برای پشتیبانی از روش کار، توسعه داده شده و با استفاده از API[1] ارائه گردیده است. این API، ویژگی‌های مختلف پردازش زبان طبیعی را پیاده‌سازی کرده و می‌تواند به طور خودکار مقادیر ویژگی‌های زبانی را بیش از نیازهای متنی تعیین کند. دقت کنید که داده‌های اصلی پردازش شده، شامل لیستی از نیازمندی‌ها است که هر یک جمله‌ای جداگانه هستند، در حالی که خروجی، یک مجموعه داده برای آموزش ویژه است. به هر حال، ابزارها و API های مختلفی وجود دارند که طیف گسترده‌ای از الگوریتم‌های یادگیری ماشین را پوشش می‌دهند و می‌توانند به راحتی با ابزار نمونه اولیه ادغام و استفاده شوند. به طور خاص، برخی از فیلترهای داده می‌توانند برای پیش‌پردازش داده‌ها اعمال شده و سپس الگوریتم‌های طبقه‌بندی مختلف برای آموزش طبقه‌بندی نیازهای امنیتی استفاده شوند و در آخر، API های اعتبارسنجی برای ارزیابی به کار گرفته شوند. در انتها لازم به ذکر است که در این مرجع از فاکتورهای امنیتی شامل دسترس‌پذیری، محرمانگی و یکپارچگی استفاده شده است.

استفاده از فیزیک نت‌گذاری برای ارزیابی امنیت و حریم خصوصی مهندسی نیازمندی‌ها: این روش در مرجع [۲۰] و به صورت آنچه در ادامه ذکر می‌شود ارائه شده است. در ابتدا، روش امنیتی TROPS که روشی برای توسعه سیستم‌های نرم‌افزاری است مورد بررسی قرار گرفته و سپس به منظور ارائه رویکردی موثر برای ارزیابی روش فیزیک نت‌گذاری، اصول نت‌نویسی با استفاده از نظریه نشانه‌گذاری و یک نمادگذاری بصری گرافیکی، شامل قوانین ترکیب و تعریف معانی هر نماد بیان شده است. در این روش، ۹ اصل برای طراحی از نقطه نظر شناخت موثر، در نظر گرفته شده‌اند که شامل: (۱) شفافیت شناخت نشانه، (۲) اصل تبعیض‌آمیز ادراکی[2]، (۳) شفافیت معنایی، (۴) مدیریت پیچیدگی، (۵) ادغام شناختی، (۶) بیان بصری، (۷) کدگذاری دوگانه، (۸) اقتصاد گرافیکی و (۹) تناسب شناختی هستند. طبق این ۹ اصل، روش مورد نظر، مورد ارزیابی گرافیکی قرار گرفته است. به هر حال، زبان مدل‌سازی در دو نسخه پیشنهاد شده است اما چندین نسخه باید بسته به سطح تخصص کاربران ارائه شود. از تجزیه و تحلیل انجام شده، مشخص شد که زبان مدل‌سازی secure trops چهار مورد از ۹ اصل را به طور کامل برآورده کرده، چهار اصل دیگر را تا حد راضی‌کننده‌ای ارضا کرده و برای یک اصل اصلا راضی‌کننده نیست. این نتایج به منظور بهبود بهتر زبان، با تمرکز در تجدید نظر عناصر خاص به ارتباط کلی زبان کاربران می‌تواند کمک کند. در این روش از حریم خصوصی و آسیب‌پذیری‌ها به عنوان فاکتورهای امنیتی استفاده شده است.

[1] Application Programming Interface

[2] Principle of Perceptual Discriminability

روش یکپارچه‌سازی مقرون به صرفه مهندسی نیازمندی‌های امنیتی در چرخه SDLC با استفاده از FRAM[1]: در مرجع [۵]، روش یکپارچه‌سازی مقرون به صرفه نیازمندی‌های امنیتی برای اجرای کارآمد سیستم‌های ایمن پیشنهاد شده است. هدف از این روش، ادغام چارچوب فرآیندهای مهندسی امنیت[2] و خطر محور[3] به منظور اجرای کارآمد سیستم‌های ایمن در هر مرحله از SDLC به کمک روش FRAM است. در واقع این روش نیاز به اولویت‌بندی نیازمندی‌های امنیتی که مطابق با ارزش دارایی، آسیب‌پذیری، تهدید و کاهش هزینه است، را نشان می‌دهد. لذا این روش چارچوبی است که به تجزیه و تحلیل خطر، به منظور جلوگیری از حادثه و یا به حداقل رساندن آن در سیستم‌های اجتماعی می‌پردازد. از این رو، پیاده‌سازی روش پیشنهادی این مرجع مستلزم انجام مراحل زیر می‌باشد:

۱. تجزیه و تحلیل عملکرد FRAM: این مرحله، به تجزیه و تحلیل خطر سیستم‌های اجتماعی- فنی برای جلوگیری از حادثه یا به حداقل رساندن ضرر به طور موثر پرداخته و طبق جدول (۷) پیاده‌سازی می‌شود.

جدول(۷): مراحل روش پیاده‌سازی FRAM

ردیف	مراحل پیاده‌سازی FRAM
۱	هدف از مدل‌سازی مشخص شود.
۲	عملکردهای اساسی سیستم شناسایی شود.
۳	شواهد وابسته به متن و تنوع پتانسیل‌های توابع سیستم مشخص و نرمال‌ترین و بدترین حالت‌ها در نظر گرفته شود.
۴	از جنبه تبعیت/تنوع عملکرد اتصالات بین توابع، توصیف و شناسایی شود.
۵	مکانیسم کنترل یا موانع تغییرپذیری و نظارت بر عملکرد مورد نیاز، شناسایی و مشخص شود.

دقت داشته باشید که هنگام اجرای فرآیندهای این مرحله باید متغیرهای کنترل، زمان، خروجی، ورودی، شرط مقدمه و منابع لحاظ شوند.

۲. تجزیه فعالیت‌ها/وظایف: روش تجزیه و تحلیل مهندسی امنیت، مطابق جدول (۸) و روش تجزیه و تحلیل خطرات کمی، مطابق با جدول (۹) است.

۳. اتصال بین فعالیت/وظایف شناسایی شود.

جدول(۸): مراحل روش تجزیه و تحلیل مهندسی امنیت

ردیف	مراحل روش تجزیه و تحلیل مهندسی امنیت
۱	دارایی‌ها شناسایی شود.
۲	ارزش دارایی و افشاء آن ارزیابی شود.
۳	تهدید و ارزیابی حمله شناسایی شود.
۴	شناسایی، کنترل و امکان سنجی ارزیابی شود.
۵	نیازمندی‌های امنیتی تعریف شود.

جدول(۹): مراحل روش تجزیه و تحلیل خطرات کمی

ردیف	مراحل روش تجزیه و تحلیل خطرات کمی
۱	ارزیابی خطر و مطالعه آسیب‌پذیری انجام شود.
۲	هزینه دارایی‌های مشهود یا نامشهود برآورد شود.
۳	عامل احتمالی افشاء (EF)[4] تخمین زده شود.
۴	تنها ضرر و زیان‌های مورد انتظار (SLE)[5] محاسبه شود.
۵	میزان وقوع سالانه (ARO)[6] تخمین زده شود.
۶	انتظار از دست دادن سالانه (ALE)[7] محاسبه شود.
۷	هزینه/ سود تجزیه و تحلیل شود.

در انتها لازم به ذکر است که در این روش از فاکتورهای امنیتی حریم خصوصی، عدم انکار، تشخیص نفوذ، تمامیت، مصونیت، مجوز و احراز هویت استفاده شده است.

مشخصات، تشخیص و درمان تهدیدات STRIDE برای اجزاء نرم‌افزار: روش پیشنهادی مرجع [۲۱]، برای اعتبارسنجی شناسایی تهدید از روش‌های مدل‌سازی تهدید مدرن پشتیبانی می‌کند. در واقع هدف این روش، رسمی‌کردن نیازمندی‌های امنیتی سیستم است که از طریق مدل‌سازی تهدید و خطر تعیین می‌شود. این مرجع معماران و طراحان را قادر می‌سازد تا امنیت را پیدا و قبل از نوشتن هر خط کد، اقدامات اصلاحی را انجام دهند. روش‌های تجزیه و تحلیل خطر مراحلی را ارائه می‌دهند

[1] Functional Resonance Analysis Method
[2] Security engineering process
[3] Risk-driven process

[4] Exposure Factor
[5] Single Loss Expectancy
[6] Annualized Rate of Occurrence
[7] Annualized Loss of Expectancy

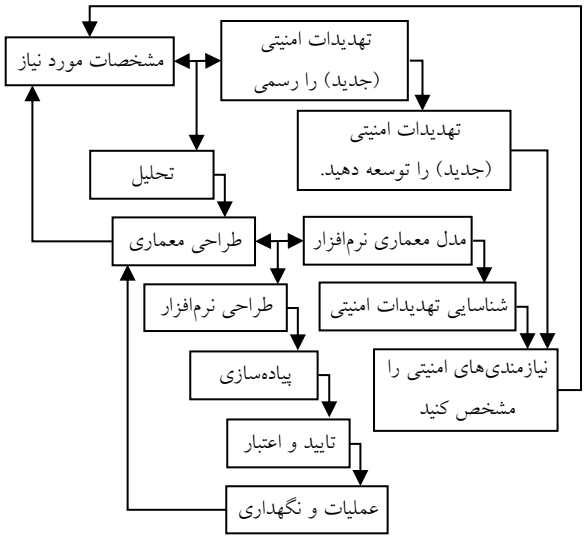
که باید به منظور شناسایی، ارزیابی و کاهش خطرات احتمالی دنبال شوند. دامنه‌ای که دسته‌های تهدید را مشخص می‌کند به صورت انتزاعی نگه داشته شده و مستقل از هر فناوری خاص یا اجرای مدل، همسو با روش پیشنهادی باقی می‌ماند. این می‌تواند اجازه دهد تا مشخصات رسمی بر روی پیاده‌سازی‌های مختلف اعمال شده یا معماری‌ها (به شرطی آن که از مفهوم ارسال پیام پشتیبانی کنند)، به عنوان یک عارضه جانبی، برای هر نماینده تهدید STRIDE، یک ویژگی مناسب نیازمندی‌های امنیتی را کدگذاری کرده و در برابر تهدید محافظت و تایید کنند. رویکرد پیشنهادی مرجع [۲۱]، در جدول (۱۰) نشان داده شده است.

جدول (۱۰): مراحل رویکرد مدل پیشنهادی

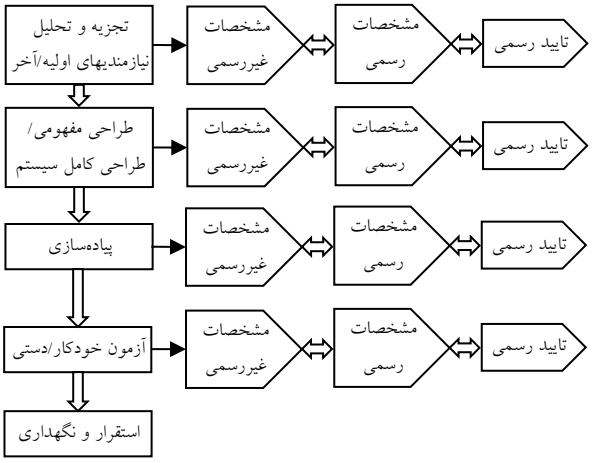
ردیف	مراحل رویکرد مدل پیشنهادی
۱	تهدیدها به عنوان ویژگی‌های یک سیستم مدل شده در یک مشخصه مستقل از فناوری، مشخص شود.
۲	برای تشخیص تهدید از طریق تایید مدل، شرایطی بیان شود که این تهدیدها به زبانی مناسب با پشتیبانی ابزار خودکار، آشکار شود.
۳	مجموعه‌ای از نیازمندی‌های امنیتی برای محافظت در برابر تهدیدات شناسایی شده، پیشنهاد شود.

همان‌طور که ذکر شد، هدف رویکرد پیشنهادی در این مرجع، رسمی کردن نیازمندی‌های امنیتی سیستم است که از طریق مدل‌سازی تهدید و خطر تعیین و تجزیه و تحلیل شده و به عنوان ویژگی‌های مورد نظر از معماری سیستم، تایید می‌گردند. برای سادگی، این رویکرد به صورت یک مدل آبشاری مطابق با شکل (۲) نشان داده شده است. در ضمن در این رویکرد، از فاکتورهای امنیتی محرمانگی، یکپارچگی، دسترس‌پذیری، احراز هویت، مجوز و عدم انکار استفاده شده است.

رسمی‌کردن نیازمندی‌های امنیتی- یک مطالعه موردی بر روی یک برنامه مبتنی بر وب: هدف از روش ارائه شده در مرجع [۲۲]، ایجاد یک چارچوب تجویزی است که کارشناسان را به مشخصه‌های امنیتی رسمی رهنمون کرده و آنها را قادر سازد تا از آغاز آسیب‌پذیری‌های امنیتی اجتناب کنند. با توجه به رویکرد پیشنهادی، رسمی‌سازی نیازمندی‌های امنیتی در تمام مراحل SDLC باید به صورت استراتژیک انجام شود. این رویکرد که در شکل (۳) نشان داده شده، از فاکتورهای امنیتی شامل احراز هویت، مجوز، محرمانه بودن، یکپارچگی، رازداری و صداقت استفاده می‌کند.



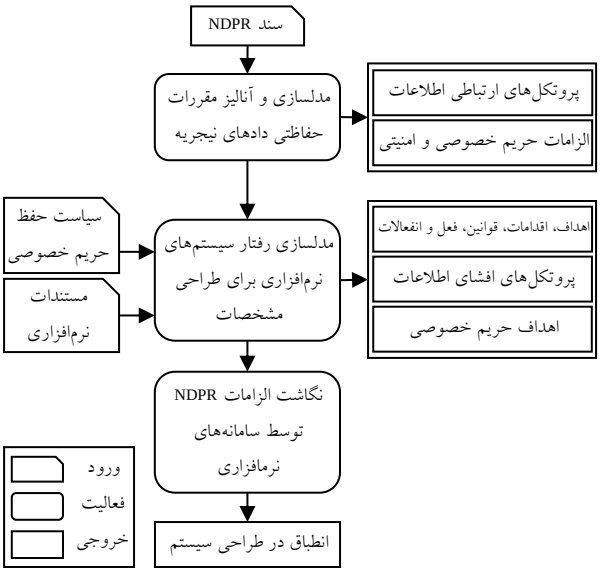
شکل (۲): مدل فرآیند آبشار [۲۱]



شکل (۳): رسمی‌کردن چرخه عمر توسعه نرم‌افزار [۲۲]

روش ارزیابی چارچوب‌ها برای استخراج نیازمندی‌های حریم خصوصی و امنیتی از قوانین و مقررات: در مرجع [۲۳]، یک روش جدید برای پیاده‌سازی نیازمندی‌های حریم خصوصی، امنیت و پروتکل‌های اطلاعاتی (دسترسی و افشا) روی داده‌های مربوط به کشور نیجریه اجرا گردید. در واقع این روش در پاسخ به نیاز به تعمیم کاربرد روش‌هایی که امکان استخراج و مدیریت

نیازمندی‌های حفظ حریم خصوصی و امنیت را با تمرکز بر مقررات مرتبط به امنیت اطلاعات خارج از اتحادیه اروپا و ایالات متحده فراهم می‌کنند، ایجاد شد. این روش در شکل (۴) نشان داده شده است.



شکل (۴): چارچوبی برای تایید رعایت حریم خصوصی در سیستم نرم‌افزاری با استفاده از اطلاعات از NDR[1] [۲۳]

این چارچوب به منظور راستی‌آزمایی انطباق با حریم خصوصی برای مقررات حفاظت از داده‌های نیجریه اعمال می‌شود. مراحل و اجزای این چارچوب به گونه‌ای طراحی شده‌اند که می‌توانند برای سایر مقررات حفظ حریم خصوصی و امنیت که در حال ظهور هستند نیز اعمال شوند. به هر حال، در این چارچوب از فاکتورهای امنیتی شامل انطباق قانونی، قابلیت ردیابی، کنترل دسترسی، قابلیت استفاده، تایید محرمانگی، افشای اطلاعات و حریم خصوصی استفاده شده است.

۴. مقایسه روش‌های مهندسی نیازمندی‌های امنیتی

در این بخش، تکرار فاکتورهای امنیتی در هر یک از روش‌ها در جدول (۱۱) آورده شده است. مقایسه نتایج نشان می‌دهد فاکتور محرمانگی، بیشترین تکرار و روش STRIDE بیشترین پیروی از فاکتورهای امنیتی انتخاب شده را دارد.

جدول(۱۱): مقایسه روش‌های مهندسی نیازمندی‌های امنیتی

روش‌ها	دسترس پذیری	احراز هویت	مجوز	نفوذ	محرمانگی	یکپارچگی	حریم خصوصی	عدم انکار
SecureMEReq		✓	✓	✓				
MLBS								✓
MBSE	✓				✓	✓		
SQUARE		✓			✓		✓	
هستی‌شناسی	✓				✓	✓		
نت‌گذاری							✓	
FRAM		✓	✓	✓				✓
STRIDE	✓	✓	✓		✓	✓		✓
رسمی کردن		✓			✓	✓		
ارزیابی					✓	✓	✓	

۵. نتیجه‌گیری

شناسایی و طبقه‌بندی نیازمندی‌های امنیتی از نقطه نظرهای مختلف از نگرانی‌های مهم توسعه‌دهندگان نرم‌افزار است. لذا در این مقاله گزیده‌ای از روش‌های نوین در این حوزه از سال ۲۰۱۷ تا ۲۰۲۲ جمع‌آوری شده و به مقایسه فاکتورهای هر یک از روش‌ها برحسب بیشترین تکرار آنها پرداخته شده است. به طور خلاصه، در روش نیازمندی‌های امنیتی مبتنی بر الگو، برای نوشتن کامل نیازمندی‌های امنیتی ابزار SecureMEReq تهیه و از فاکتورهای امنیتی شامل احراز هویت، شرایط مجوز و نفوذ استفاده شده است. در روش تجزیه و تحلیل تهدید، به منظور استخراج نیازمندی‌های امنیتی درMLBS امکان کمک به اعضای تیم حتی با دامنه دانش خاص برای کاهش تهدیدهای MLBS وجود دارد و از فاکتورهای امنیتی اعم از عدم کلاهبرداری، دستکاری، انکار و جلوگیری از افشای اطلاعات استفاده شده است. روش ادغام نیازمندی‌های امنیتی با MBSE، یک تیم چند رشته‌ای را قادر به انجام تجزیه و تحلیل امنیت به موازات فرآیند مهندسی سیستم در پروژه MBSE می‌کند و در آن از فاکتورهای امنیتی دسترس‌پذیری، محرمانگی و یکپارچگی استفاده شده است. چارچوب مبتنی بر هدف با اتخاذ فرآیند SQUARE، بر پایه شناسایی اهداف امنیتی مربوط به دارایی‌های سازمان با

[1] Nigeria Data Protection Regulation

استفاده از فرآیند SQUARE بنا شده و از فاکتورهای امنیتی حریم خصوصی، مسئولیت، محرمانگی، صداقت و احراز هویت استفاده می‌کند. رویکرد مبتنی بر هستی‌شناسی برای طبقه‌بندی خودکار نیازمندی‌های امنیتی، روشی سیستماتیک برای شناسایی ویژگی‌های زبانی نیازمندی‌های امنیتی است که لایه مفهومی را به لایه زبانی متصل کرده و از فاکتورهای امنیتی دسترس‌پذیری، محرمانگی و یکپارچگی استفاده می‌کند. در روش بهره از فیزیک نت‌گذاری برای ارزیابی امنیت و حریم خصوصی از فاکتورهای امنیتی شامل حریم خصوصی و آسیب‌پذیری استفاده شده است. روش یکپارچه‌سازی مقرون به صرفه مهندسی نیازمندی‌های امنیتی در چرخه SDLC، با استفاده از FRAM برای اجرای کارآمد سیستم‌های ایمن پیشنهاد شده که در آن از فاکتورهای امنیتی شامل حریم خصوصی، عدم انکار، تشخیص نفوذ، تمامیت، مصونیت، مجوز و احراز هویت استفاده شده است. روش STRIDE، برای رسمی‌سازی نیازمندی‌های امنیتی سیستم از طریق مدل‌سازی تهدید و خطر ارائه شده و از فاکتورهای امنیتی محرمانگی، یکپارچگی، دسترس‌پذیری، احراز هویت، مجوز و عدم‌انکار استفاده می‌کند. روش رسمی‌سازی برای ایجاد چارچوبی تجویزی، کارشناسان را قادر به شناسایی مشخصه‌های امنیتی رسمی و اجتناب از آغاز آسیب‌پذیری‌های امنیتی ساخته و از فاکتورهای امنیتی احراز هویت، محرمانگی، یکپارچگی، رازداری و صداقت استفاده می‌کند. در روش ارزیابی برای پیاده‌سازی نیازمندی‌های حریم خصوصی، از فاکتورهای امنیتی شامل انطباق، قابلیت ردیابی، کنترل دسترسی، قابلیت استفاده، محرمانگی، افشای اطلاعات و حریم خصوصی استفاده شده است. برحسب بیشترین تکرار فاکتورهای امنیتی در هر یک از روش‌ها، STRIDE دارای بیشترین فاکتور امنیتی می‌باشد.

تعارض منافع: نویسندگان اعلام می‌کنند که هیچ تعارض منافعی ندارند.

مراجع

[۱] شیخان م.، عباسی ع.، «راهکارهای ترکیبی نوین برای تشخیص نفوذ در شبکه‌های کامپیوتری با استفاده از الگوریتم‌های هوش محاسباتی»، مجله محاسبات نرم، جلد ۶، شماره ۱، ص.۴۸-۶۵، ۱۳۹۶.

[۲] برنگی ح.، راجی ف.، خاصه ع.، «تحلیل تحقیقات امنیت و حریم خصوصی حوزه بلاکچین: یک مقاله علم سنجی»، مجله محاسبات نرم، جلد ۹، شماره ۱، ص.۴۰-۵۵، ۱۳۹۹.

[۳] یداللهی ا.، سلیمی سرتختی ج.، گلی بیدگلی س.، «مدل‌سازی امنیت ماشین‌های مجازی در ابر با استفاده از تئوری بازی تکرار شونده»، مجله محاسبات نرم، جلد ۱۰، شماره ۱، ص.۲-۱۵، ۱۴۰۰.

[4] Villamizar H., Kalinowski M., Viana M., and Fernández D. M., "A systematic mapping study on security in agile requirements engineering," In 2018 44th Euromicro conference on software engineering and advanced applications (SEAA), pp. 454-461, IEEE, 2018.

[5] Hlaing S. Z. and Ochimizu k., "An Integrated Cost-Effective Security Requirement Engineering Process in SDLC Using FRAM," In 2018 International Conference on Computational Science and Computational Intelligence (CSCI), pp. 852-857, IEEE, 2018.

[6] Behutiye W., Karhapää P., López L., Burgués X., Martínez-Fernández S., Vollmer A.M., Rodríguez P., Franch X., and Oivo M., "Management of quality requirements in agile and rapid software development: A systematic mapping study," Information and software technology, 123: 106225, 2020.

[7] Wohlin C., "Guidelines for snowballing in systematic literature studies and a replication in software engineering," In Proceedings of the 18th international conference on evaluation and assessment in software engineering, pp. 1-10. 2014.

[8] Souag A., Mazo R., Salinesi C., and Comyn-Wattiau I., "Reusable knowledge in security requirements engineering: a systematic mapping study," Requirements Engineering, 21(2): 251-283, 2016.

[9] Muñante D., Chiprianov V., Gallon L., and Aniorté P.,

"A review of security requirements engineering methods with respect to risk analysis and model-driven engineering," In International Conference on Availability, Reliability, and Security, pp. 79-93. Springer, Cham, 2014.

[۱۰] ذاکری م.، شمسی م.، «مقایسه روش‌های مهندسی نیازمندی‌های امنیتی» دومین کنفرانس بین‌المللی مدیریت و اقتصاد در قرن ۲۱، ۱۳۹۵.

[11] Mishra A. D. and Mustafa K., "A review on security requirements specification by formal methods," Concurrency and Computation: Practice and Experience, 34(5): e6702, 2021.

[12] Mustafa N., Kamalrudin M., and Sidek S., "Security requirements template-based approach to improve the writing of complete security requirements," Journal of Theoretical and Applied Information Technology 99(1), 2021.

[13] Wilhjelm C. and Younis A. A., "A Threat Analysis Methodology for Security Requirements Elicitation in Machine Learning Based Systems," In 2020 IEEE 20th International Conference on Software Quality, Reliability and Security Companion (QRS-C), pp. 426-433. IEEE, 2020.

[14] Amershi S., Begel A., Bird C., DeLine R., Gall H., Kamar E., Nagappan N., Nushi B., and Zimmermann T., "Software engineering for machine learning: A case study," In 2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice (ICSE-SEIP), pp. 291-300. IEEE, 2019.

[15] Ishikawa F. and Yoshioka N., "How do engineers perceive difficulties in engineering of machine-learning systems?," In 6th International Workshop on Software Engineering Research and Industrial Practice (SER&IP), pp. 2-9. IEEE, 2019.

[16] Wan Z., Xia X., Lo D., and Murphy G. C., "How does machine learning change software development practices?," IEEE Transactions on Software Engineering, 47(9): 1857-1871, 2021.

[17] Mažeika D. and Butleris R., "Integrating security requirements engineering into MBSE: Profile and guidelines," Security and Communication Networks, 2020.

[18] Hayat B., Shakoor R., Mubarak S., and Basharat K., "A goal based framework by adopting square process for privacy and security requirement engineering," International Journal of Computer Applications, 169(11): 31-34, 2017.

[19] Li T. and Chen Z., "An ontology-based learning approach for automatically classifying security requirements," Journal of Systems and Software, 165: 110566, 2020.

[20] Diamantopoulou V. and Mouratidis H. "Applying the physics of notation to the evaluation of a security and privacy requirements engineering methodology," Information and Computer Security, 26(4): 382-400, 2018.

[21] Rouland Q., Hamid B., and Jaskolka J., "Specification, detection, and treatment of STRIDE threats for software components: Modeling, formal methods, and tool support," Journal of Systems Architecture, 117: 102073, 2021.

[22] Mishra A. D. and Mustafa K., "Formalization of Security Requirements-A Case Study on a Web-Based Application," Journal of Scientific Research, 66(2), 2022.

[23] Olukoya O., "Assessing frameworks for eliciting privacy & security requirements from laws and regulations," Computers and Security, 117: 102697, 2022.