دوفصل‌نامه علمی منادی امنیت فضای تولید و تبادل اطلاعات (افتا)
سال ۱۴۰۱، جلد ۲۰، شماره ۲ (۵۲–۶۰)
https://monadi.isc.org.ir

ارائه‌شده در نوزدهمین کنفرانس بین‌المللی انجمن رمز ایران در دانشگاه گیلان، ۹ و ۱۰ شهریور ۱۴۰۱

مدل‌سازی سطح توانایی مهاجم با تمرکز بر حملات تزریق روی برنامه‌های کاربردی وب★

سارا مقیمی* و محمدعلی هادوی

دانشکده مهندسی برق و کامپیوتر، دانشگاه صنعتی مالک اشتر، تهران، ایران

اطلاعات مقاله

کلمات کلیدی:
سطح توان مهاجم
تزریق SQL
نوع تزریق
تکنیک تزریق
نقطه‌ی ورود تزریق
امنیت سامانه

dor: 20.1001.1.24763047.1401.11.2.6.5

نوع مقاله: پژوهشی

چکیده

چگونگی سوءاستفاده از آسیب‌پذیری‌ها و اثرات آن در کنار الگوهای شناخته شده، متأثر از توانمندی مهاجم‌ها می‌باشد. هرچه مهاجم توانمندتر باشد، مخاطره تهدیدها و آسیب‌پذیری‌ها افزایش پیدا می‌کند. بنابراین، تحلیل و ارزیابی امنیتی سامانه‌ها وابسته به توانمندی مهاجم است. علاوه بر این، اطلاع از سطح توانمندی مهاجم ارتباط مستقیمی با هزینه مورد نیاز برای امنیت و بکارگیری کنترل‌ها و اقدامات امنیتی متناسب با توان مهاجم دارد. بر این اساس، این مقاله مدل‌سازی توانمندی مهاجم را هدف‌گذاری کرده است. ما در این مقاله با تکیه بر تزریق پیلودهایی که مهاجم برای سوءاستفاده از آسیب‌پذیری‌های تزریق استفاده می‌کند، توانمندی مهاجم را با سه‌گانه‌ی $\langle \texttt{Type}, \texttt{Technique}, \texttt{Entry_Point} \rangle$ مدل می‌کنیم. مؤلفه‌ی Type بیانگر نوع تزریق می‌باشد که شامل مجموعه‌ای شناخته‌شده از انواع حمله تزریق است که مهاجم در طول حمله به کار برده است. مؤلفه‌ی Technique بیانگر تکنیک‌هایی است که مهاجم در طول حمله به کار برده است، و مؤلفه‌ی Entry_Point نشان‌دهنده‌ی مجموعه‌ای از نقاط شناخته‌شده تزریق پیلود است. از این مدل هم برای سطح‌بندی و مقایسه توانمندی مهاجم و هم برای سطح‌بندی امنیت یک سامانه با توجه به سطح توان مهاجمی که می‌تواند امنیت آن را به خطر بیاندازد استفاده می‌شود. نتایج ارزیابیِ تجربیِ انجام شده نشان می‌دهد که مدل ارائه شده برای تعیینِ سطحِ توانمندیِ مهاجم قابل استفاده است. با این‌که مدل ارائه شده با تمرکز بر حملات تزریق SQL است، اما قابل توسعه به بسیاری از حملات دیگر می‌باشد.

© ۱۴۰۱ انجمن رمز ایران

۱ مقدمه

توسعه‌ی یک سامانه‌ی امن، نیازمند ارزیابی و کاهش مخاطره‌ی تهدیدهای شناسایی شده است. با این حال سازمان‌ها متناسب با هزینه‌ی قابل تحمل، برای هر یک از تهدیدها اقدام‌های بازدارنده‌ای انجام می‌دهند. آنها ممکن است با محدودیت‌های بودجه‌ای که دارند تنها به زیرمجموعه‌ای از تهدیدها بپردازند. آن‌ها میزان خطر یک تهدید را برآورد می‌کنند (تابعی از احتمال تهدید و شدت اثرات آن) و از آن برای اولویت‌بندی مقابله با تهدیدها استفاده می‌کنند. دقت برآوردهای مخاطره، منجر به اولویت‌بندی واقعی‌تر تهدیدها و بنابراین بازگرداندن بهتر سرمایه می‌شود. یک روش قدیمی در برآورد مخاطره، شناسایی همه‌ی سناریوهای حمله‌ی احتمالی و برآورد مخاطره آن‌ها است. این رویکرد، بسیار پرهزینه است و در پروژه‌های تجاری، ترجیح داده نمی‌شود [۱، ۲]. رویکردهای دیگری وجود دارد که برای برآورد مخاطره، مجموعه‌ای از عوامل را استفاده می‌کنند؛ مانند OCTAVE که در آن زیرمجموعه‌ای از عوامل مانند دانش، تخصص و در دسترس‌بودن منابع در نظر گرفته شده است [۱، ۲]. از طرف دیگر

★از کمیته علمی نوزدهمین کنفرانس بین‌المللی انجمن رمز ایران برای داوری این مقاله تشکر می‌شود.
*نویسنده مسئول
آدرس‌های رایانامه: saramoqimi@mut.ac.ir (سارا مقیمی)، hadavi@mut.ac.ir (محمدعلی هادوی)
© ۱۴۰۱ تمامی حقوق متعلق به انجمن رمز ایران است.

NIST از عواملی مانند قابلیت‌های مهاجم و قصد مهاجم استفاده می‌کند [۱، ۲]. نقطه ضعف این روش‌ها تفاوت زیاد بین مقدار تخمینی و واقعی مخاطره است. در این روش‌ها، فرض بر این است که مهاجم توانایی انجام حمله را به شرط داشتن دانش کامل از سامانه و در دسترس‌بودن منابع دارد.

روش‌های برآورد مخاطره‌ی موجود، معمولاً توانایی‌های مهاجم را به‌عنوان منابع، دانش و تخصص که می‌تواند برای ایجاد تهدید استفاده شود، در نظر می‌گیرند [۱]. با این حال، تهدیدها معمولاً فقط در صورت برقرار بودن شرایط خاص، قابل استفاده هستند. این شرایط شامل اِعمال موفقیت‌آمیز تهدیدهای خاص (به عنوان مثال، یک عامل داخلی برای همکاری)، پیکربندی خاص سیستم و دسترسی به منابعی که می‌توانند در سناریوهای حمله استفاده شوند، است.

توانایی مهاجم، رکن اصلی برآورد مخاطره است که در بسیاری از پژوهش‌ها مورد غفلت قرار گرفته است. توانایی مهاجم هم بر احتمال وقوع تهدید یا سوءاستفاده از آسیب‌پذیری اثر می‌گذارد و هم بر شدت اثر آن. هرچه مهاجم توانمندتر باشد، احتمال بروز سوءاستفاده افزایش می‌یابد. علاوه بر این توانمندی مهاجم می‌تواند اثرات ناشی از سوءاستفاده از آسیب‌پذیری را تشدید نماید.

هدف اصلی این مقاله ارائه‌ی مدلی برای سنجش سطح توانایی مهاجم است. این پژوهش به ویژه در برآورد مخاطره‌ی تهدیدها بسیار کاربرد دارد و کیفیت برآورد مخاطره را با کاهش عدم اطمینان در برآورد مخاطره بهبود می‌بخشد. افزون بر آن، در انتخاب مناسب مکانیزم‌های مقابله‌ی تهدیدها مفید است. همین امر به ما کمک می‌کند تا سطح مناسبی از امنیت را در سامانه لحاظ کنیم و هزینه‌ی لازم برای ایمن کردن سامانه را به میزان قابل قبولی برسانیم. همچنین برای تحلیل سطح دانش یک جامعه آماری خاص در زمینه نفوذ به سامانه‌ها می‌تواند کاربرد داشته باشد.

تنوع حمله‌ها روی سامانه‌های نرم‌افزاری از حیث جنبه‌هایی چون دانش مورد نیاز مهاجم بسیار زیاد است و زمینه‌های دانشی متعددی مانند مهندسی معکوس، شبکه‌های کامپیوتری، پروتکل‌های امنیتی، کدنویسی، رمزنگاری و پایگاه داده را در بر می‌گیرد. از این رو پرداختن به موضوع خِبرگی مهاجم در این دامنه‌ی وسیع باعث پیچیدگی مسئله و دشواری حل آن می‌شود. بر این اساس، تمرکز ما در این مقاله به حملات تزریق SQL محدود بوده است. تزریق SQL نوعی تزریق کد اجرایی در برنامه است که در آن مهاجم برای دسترسی غیرمجاز، به روش‌های مختلفی کد مخرب به زبان SQL را به کد کاربر اضافه می‌کند. به این کدهای مخرب، پیلود گفته می‌شود. معیار انتخاب حملات تزریق، توجه به این نکته است که آسیب‌پذیری‌های تزریق همواره جزو مهم‌ترین آسیب‌پذیری‌ها در گزارش‌های مراجع معتبر مانند OWASP ذکر شده‌اند [۳].

رویکرد ارائه شده در این مقاله یافتن مؤلفه‌هایی در تزریق‌های انجام شده است که می‌توان از آن‌ها برای تمایز توان مهاجم‌ها بهره گرفت. بررسی پیلودها نشان می‌دهد که عناصری در آن‌ها وجود دارد که بیانگر نوع حمله تزریق و تکنیک‌های به کار رفته در حمله می‌باشد. همچنین از درخواست آلوده‌ی ارسال شده به کارپذیر، نقطه ورود تزریق قابل استخراج است. این سه مورد سه‌گانه اصلی مدل توان مهاجم را تشکیل می‌دهد.

در ادامه‌ی مقاله، در بخش ۲ کارهایی که در حوزه‌ی توان مهاجم انجام شده است را مرور کرده‌ایم. در این کارها توان مهاجم از جنبه‌های مختلفی چون زمان به خطر انداختن سامانه، ویژگی‌های نقض‌شده‌ی امنیتی و حالات مختلف دسترسی به سامانه مورد بررسی قرار گرفته است. در بخش ۳، روش پیشنهادی را تشریح کرده‌ایم و بر اساس این روش توانستیم توان مهاجم را بر اساس سه مؤلفه‌ی نوع تزریق، تکنیک‌های تزریق و نقطه‌ی تزریق مدل کنیم. همچنین مدل به‌دست آمده را برای محاسبه‌ی توان مهاجم و سطح‌بندی آن، به‌کارگیری کرده‌ایم. در بخش ۴، روش ارائه شده را با استفاده از چند آزمون‌گر و یک سامانه واقعی مورد ارزیابی قرار می‌دهیم. در پایان، در بخش ۵ نیز نتیجه‌گیری و جمع‌بندی را ارائه کرده‌ایم.

۲ کارهای مرتبط

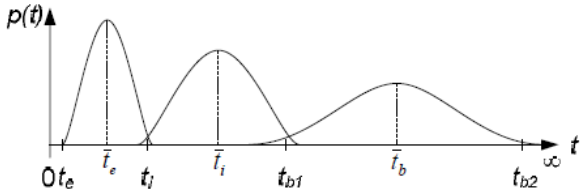
مسئله‌ی ارزیابی توانایی مهاجم در کارهای مروری از جنبه‌های مختلفی چون زمان به خطر انداختن سامانه، ویژگی‌های نقض‌شده‌ی امنیتی و حالات مختلف دسترسی به سامانه مورد بررسی قرارگرفته است. در [۴، ۵] با اندازه‌گیری زمان به خطر انداختن سامانه توسط مهاجم به این موضوع پرداخته شده است. در واقع، مدت زمان انجام حمله موفق به یک سامانه به عنوان معیار اصلی برای تحلیل توان مهاجم در نظر گرفته شده است. نویسندگان در [۶] معتقدند که قابلیت مهاجم با ویژگی‌های نقض شده‌ی امنیتی در ارتباط است. هرچه توان مهاجم بیشتر باشد، ویژگی‌های امنیتی بیشتری نقض می‌شوند. در [۷] بر اساس داده‌های تجربی جمع‌آوری شده از آزمایش‌های نفوذ، یك فرضیه در مورد رفتار رایج مهاجم ارائه شده است. این فرضیه نشان می‌دهد که در طی فرآیند حمله، وقوع نقض‌ها بر اساس رفتار مهاجم می‌تواند به سه مرحله تقسیم شود: مرحله‌ی یادگیری، مرحله‌ی حمله استاندارد و مرحله‌ی حمله ابتکاری.

۱.۲ قابلیت مهاجم بر اساس زمان به خطر انداختن سامانه

McQueen و همکارانش [۴] برای ارزیابی کمی امنیت سامانه‌ی کامپیوتری، استفاده از زمان به خطر انداختن سامانه را پیشنهاد کردند. زمان به خطر انداختن، معیار سنجش تلاشی است که توسط یک مهاجم برای یک حمله‌ی موفق هزینه می‌شود. آن‌ها معتقدند با افزایش این زمان، احتمال حمله‌ی موفقیت‌آمیز و در نتیجه مخاطره، کاهش می‌یابد. این زمان به آسیب‌پذیری و توان مهاجم وابسته است و به عنوان یک فرآیند تصادفی از سه فرآیند مهاجم تشکیل شده است

(۱) فرآیند اول برای مواردی است که حداقل یک آسیب‌پذیری، شناخته شده است و مهاجم حداقل یک سوءاستفاده‌ی کاربردی برای آن دارد.

(۲) فرآیند دوم برای مواردی است که حداقل یک آسیب‌پذیری، شناخته



شکل ۱. احتمال به خطر انداختن سامانه توسط سطوح مختلف مهاجم در واحد زمان [۵]

شده است اما سوءاستفاده‌ی آن در دسترس مهاجم نیست.

(۳) فرآیند سوم شناسایی آسیب‌پذیری‌ها و سوءاستفاده‌های جدید است. این فرآیند موازی با فرآیندهای اول و دوم است.

این فرآیندها، توزیع احتمال شکست متفاوتی دارند. زمان به خطر انداختن سامانه، مجموع مقادیرِ زمانِ اتمامِ مورد انتظارِ سه فرآیند نام‌برده است.

در سال ۲۰۰۸ مقاله‌ای منتشر شد که با استفاده از معیار میانگین زمان به خطر انداختن سامانه (MTTC) امنیت سامانه‌ی کامپیوتری، ارزیابی شده است [۵]. نتیجه‌ی حمله خیلی وابسته به سطح توانایی مهاجم است. مهاجم‌های زیادی وجود دارند اما چون کسب مهارت در این زمینه، نیازمند صرف زمان زیادی است پس بیشتر این مهاجم‌ها از ایجاد تهدیدهای جدی، فاصله دارند. مهاجم‌های تازه‌کار از سوءاستفاده‌های آماده استفاده می‌کنند که حدود ۹۰ درصد فعالیت‌های مهاجم در اینترنت را تشکیل می‌دهد. به همین دلیل ارزیابی سطح توان مهاجم مهم است. هر مهاجمی یک مقدار معین مهارت و تجربه دارد؛ مبتدیان می‌توانند سوءاستفاده‌های موجود را روی آسیب‌پذیری‌های شناخته شده استفاده کنند، مهاجم‌های کمی ماهرتر می‌توانند کد سوءاستفاده‌ها را اصلاح و از آسیب‌پذیری شناخته شده سوءاستفاده کنند و مهاجم‌های ماهر می‌توانند آسیب‌پذیری‌های ناشناخته را پیدا و سوءاستفاده‌ی جدید ایجاد کنند. برای هر گروه مهاجم بازه‌ای بین ۰ تا ۱ اختصاص داده شده است (مبتدی [۰, ۰٫۶]، نیمه ماهر [۰٫۶, ۰٫۸]، ماهر [۰٫۸, ۱]) که توان‌های مهاجم به‌صورت نرمال در این بازه‌ها توزیع شده است. کران بالای هر بازه بالاترین توان گروه محسوب شده است. یک مهاجم ماهر هنگامی که مقدار سطح توان ۱ است برای به خطر انداختن سامانه به زمان t_e نیاز دارد، مهاجم نیمه ماهر با سطح توان ۰٫۸ به زمان t_i نیاز دارد و مهاجم مبتدی زمانی که مقادیر سطح توان ۰٫۶ و ۰٫۱ باشد به زمان‌های t_{b1} و t_{b2} نیاز دارد. $\bar{t}_e$ و $\bar{t}_i$ و $\bar{t}_b$ مقادیر متوسط این فواصل هستند (شکل ۱).

برای ارزیابیِ MTTC در گروهِ سطح توان مهاجم باید از توزیع نرمال به جای مقادیر بالای بازه استفاده شود، زیرا افراد مبتدی، تعداد بیشتری از حمله‌ها را آغاز می‌کنند. محاسبه‌ی MTTC به دلیل کمبود داده‌های آماریِ قابل اعتماد، پیچیده است [۵].

در این رویکرد با اندازه‌گیری زمان به خطر انداختن سیستم، مهاجم‌ها در سه دسته طبقه‌بندی شده‌اند که با توجه به پیچیدگی و چندلایه بودن حمله‌ها، محاسبه‌ی این زمان کار دشوار و پرهزینه‌ای خواهد بود. همچنین با توجه به اینکه تنها معیار زمان در طبقه‌بندی آن‌ها در نظر گرفته شده است پس از دقت کافی برخوردار نیست.

به نظر می‌رسد در این کارها شرط اندازه‌گیری توان مهاجم به طور ضمنی بهره‌برداری از سامانه در نظر گرفته شده است در صورتی که با توجه به سطح امنیتی سامانه مهاجم‌هایی وجود دارند که علی‌رغم توانمندی بالا، موفق به بهره‌برداری از سامانه نمی‌شوند و در دسته‌بندی این تحقیقات در نظر گرفته نمی‌شوند. همچنین این کارها براساس آسیب‌پذیری‌های موجود و سوءاستفاده‌های در دسترس آن‌ها انجام شده است. در پژوهش ما تمامی مهاجم‌ها در هر سطحی مورد توجه هستند و توان مهاجم بر اساس میزان دانش و خِبرگی مورد ارزیابی قرار گرفته است.

۲.۲ توان مهاجم بر اساس نقض ویژگی‌های امنیتی

درک درست از توانایی‌های مهاجم به ما کمک می‌کند تا تصمیم بگیریم چگونه از منابع امنیتی خود به بهترین وجه استفاده کنیم. Basin و Cremers [۶] مهاجم‌ها را بر اساس اینکه «آیا هر مهاجمی می‌تواند یک ویژگی امنیتی خاص را با استفاده از اقدام‌های خود نقض کند» طبقه‌بندی می‌کنند [۶]. دیدگاه Eric و همکارانش [۶] این است که توان مهاجم‌های قدرتمند با الزامات امنیتی نقض شده در ارتباط است. به این معنا که با توجه به مجموعه‌ای از الزامات امنیتی اساسی می‌توان مهاجم‌ها را بر اساس تعداد خصوصیات مختل شده، طبقه‌بندی کرد. بنابراین مدل بررسی خواص چندگانه را توسط مدل بررسی coalgebraic با استفاده از تکنیک‌های همرفتی پیشرفته ارائه می‌کند.

این رویکرد مهاجم‌هایی که موفق به نقض ویژگی‌های امنیتی سامانه نمی‌شوند ولی در جهت به خطر انداختن سامانه تلاش می‌کنند را نادیده می‌گیرد. در پژوهش ما مهاجم‌ها بر اساس دانشی که دارند ارزیابی می‌شوند که عموم کاربران با هر سطحی از دانش را شامل می‌شود.

۳.۲ توان مهاجم بر اساس حالات دسترسی به سامانه

در [۱] برای سامانه‌ی اطلاعاتیِ معین، تعریفِ توان مهاجم نیازمند تعریفِ همه‌ی حالات دسترسی به سامانه دانسته می‌شود و توان مهاجم با تاپل $\langle O, A\rangle$ فرموله شده است که O مجموعه‌ای از اشیا و A مجموعه‌ای از اقدام‌ها است. همه‌ی مهاجم‌ها بالقوه فرض شده‌اند. برآورد احتمال اینکه مهاجم قابلیت c را دارد می‌تواند با استفاده از تاریخچه‌ی داده اگر در دسترس باشد محاسبه شود.

در رویکرد دیگری شانس وجود یك قابلیت در مهاجم را برآورد کرده‌اند. مثلاً به‌ازای هر مهارت یك شماره در مقیاس ۰ تا ۴ به مهاجم اختصاص داده‌اند که شماره ۴ نشان‌دهنده‌ی این است که مهاجم به طور قطع قابلیت c را دارد، شماره ۳ نشان‌دهنده‌ی این است که مهاجم ممکن است قابلیت c را داشته باشد. شماره ۲ نشان‌دهنده‌ی این است که مهاجم می‌تواند قابلیت c را داشته باشد، شماره‌ی ۱ نشان‌دهنده‌ی این است که مهاجم بعید است قابلیت c را داشته باشد، شماره ۰ نشان‌دهنده‌ی این است که مهاجم خیلی بعید است قابلیت c را داشته باشد.

برای برخی از حمله‌ها، یک مهاجم بالقوه به l قابلیت از مجموعه‌ی

قابلیت‌هایش نیاز دارد که در [۸] با استفاده از معادله (۱) فرموله شده است که احتمال قابلیت C را از احتمال‌های l قابلیت محاسبه می‌کند:

$$C = \prod_{k=1}^{l} c_k \qquad (۱)$$

در [۶] قابلیت مهاجم، اطلاعات مرتبط با حمله و شرایطی که حمله را امکان‌پذیر می‌کند در نظر گرفته شده است. برای تعیین قابلیت مهاجم مجموعه‌ای از اشیاء (O_i) و مجموعه‌ای از اقدام‌ها (A_i) را در نظر گرفته و قابلیت را از معادله (۲) محاسبه می‌کند:

$$C_i = \sum A_i O_i \qquad (۲)$$

تعیین احتمال قابلیت به این معناست که میزان یا اندازه‌ی توانایی یک مهاجم برای حمله چه مقدار است که می‌تواند مانند یک امتیاز یا یک مقدار در نظر گرفته شود که می‌گوید مهاجم براساس مقدار قابلیت چقدر مؤثر است. مجموع همه‌ی قابلیت‌ها، توان یک مهاجم را ارائه می‌کند:

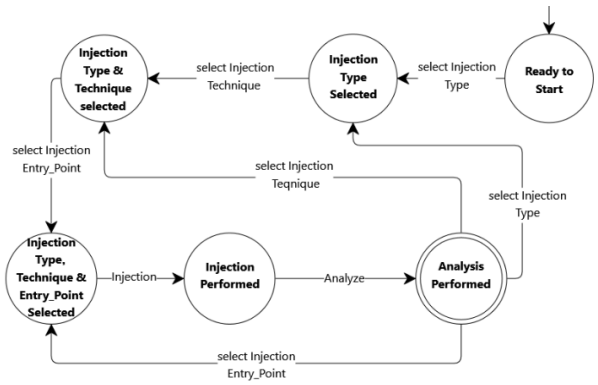
$$CA = \sum C_i \qquad (۳)$$

این رویکرد توجه به قابلیت‌های مهاجم به کاهش عدم قطعیت در مقدار واقعی و تخمین مخاطره کمک می‌کند. این رویکرد توان مهاجم را به‌ازای سامانه‌های مختلف ارزیابی می‌کند. به این معنا که مهاجم X نسبت به سامانه‌ی Y دارای توان CA است. در حالی که در پژوهش ما نگاه بنیادی‌تری به توان مهاجم وجود دارد و ما عناصر تشکیل‌دهنده‌ی توان یک مهاجم را مورد توجه قرار داده‌ایم.

در بیشتر مقالات، ارزیابی امنیت سامانه بیشتر از این جنبه که سامانه چه آسیب‌پذیری‌هایی دارد و چه حمله‌هایی از طریق این آسیب‌پذیری‌ها قابل انجام است به موضوع نگاه شده است؛ ولی نگاه ما بیشتر به خود مهاجم است. ما دانش یك مهاجم برای انجام یك حمله را بر اساس معیارهایی چون سختی پیلودها و قدرت مکانیزم‌های دفاعی موجود، اندازه‌گیری کرده‌ایم. ما با استفاده از کارهای مرور شده و دیگر مطالعات انجام گرفته پیرامون حمله‌های تزریق SQL، مدلی برای توان مهاجم ارائه کرده‌ایم و به دنبال آن هستیم که بتوانیم براساس این مدل، مهاجم‌ها را سطح‌بندی کرده و امنیت سامانه را بر اساس سطح توان مهاجم بیان کنیم.

۳ روش پیشنهادی

در بررسی دانش مهاجم این‌گونه فرض می‌کنیم که دانش مهاجم بازتاب شده در پیلودهایی است که در نقاط مختلف سامانه تزریق می‌کند. بنابراین تلاش کرده‌ایم آن چیزی که برای انجام حمله در ذهن مهاجم شکل می‌گیرد را مدل‌سازی کنیم. با توجه به رفتار مهاجم فرایند عمومی حمله‌ی SQLI را به‌صورت یک ماشین حالت متناهی (FSM) ارائه کرده‌ایم که در شکل ۲ قابل مشاهده است. این فرایند مبنای مدل‌سازی توان مهاجم در روش ما بوده است.



شکل ۲. فرایند عمومی SQLI

این FSM از شش وضعیت و هشت گذر تشکیل شده است. طبق این FSM، برای هر حمله‌ی تزریق باید نوع حمله، روش حمله و نقطه ورود حمله مشخص شود. سپس حمله اجرا شده و نتایج تحلیل حمله (اعم از موفقیت یا عدم موفقیت) می‌تواند منجر به تغییر در هر یک از مولفه‌های نوع، روش، و نقطه ورود حمله گردد. پیلودهایی که توسط مهاجم در اجرای هر حمله وارد سیستم می‌شود، تعیین‌کننده‌ی نوع حمله، روش حمله و نقاط ورود حمله می‌باشد که برای سطح‌بندی توان مهاجم مورد استفاده قرار می‌گیرد. به‌عبارت دیگر توان هر مهاجم بر اساس سه‌گانه $\langle \texttt{Type}, \texttt{Technique}, \texttt{Entry_Point} \rangle$ مدل می‌شود.

۱.۳ مؤلفه‌های مدل توان مهاجم

بر اساس آنچه پیش‌تر بیان شد ما سه مؤلفه‌ی نوع تزریق، تکنیک‌های تزریق و نقطه‌ی تزریق را به عنوانِ مؤلفه‌های محوریِ تعیین‌کننده‌ی توانِ مهاجم در نظر گرفتیم. این مؤلفه‌ها در اجرای حمله بر اساس شکل ۲ مستقل از هم در نظر گرفته شده‌اند. برای هر یک از این مؤلفه‌ها، رایج‌ترین مقادیر آن‌ها را بر اساس مجموعه پیلودهای گردآوری شده در وب‌گاه‌های متعدد، منابع علمی و آموزشی موجود و تجربه‌ی متخصص‌ها در جداولی گردآوری کرده‌ایم که مبنای سطح‌بندی توان مهاجم را تشکیل می‌دهند. از جمله این منابع می‌توان به گیت‌هاب، portSwwigger، OWASP، exploit-db، netsparker، WAPTX، mySql و ... اشاره کرد. گفتنی است با گذشت زمان و تغییرات احتمالی در انواع، روش‌ها، و نقاط ورود حمله، مقادیر در نظر گرفته شده برای هریک از این مولفه‌ها ممکن است نیاز به به‌روزرسانی داشته باشد؛ با این حال مدل توان مهاجم همان سه‌تایی $\langle \texttt{Type}, \texttt{Technique}, \texttt{Entry_Point} \rangle$ خواهد بود.

نوع حمله: Type معرّف نوع تزریق می‌باشد که شامل مجموعه‌ای از انواع حملات تزریق است که مهاجم در طول حمله به کار برده است. بر این مبنا انواع تزریق را به سه دسته‌ی درون-محدوده، خارج-از-محدوده و استنتاج تفکیک کرده‌ایم. بنابراین مؤلفه‌ی Types طبق معادله (۴) مجموعه‌ای از اجتماع این سه مجموعه است که هر مجموعه شامل انواع مختلفی از روش‌های تزریق است. جدول ۱ انواع روش‌های تزریق را در دسته‌های مذکور نشان می‌دهد.

جدول ۱. معرفی انواع حمله

دسته	نوع حمله
درون-محدوده (IN_Band) (سطح E) مهاجم جهت مشاهده‌ی نتیجه‌ی بهره‌برداری از همان کانالِ استفاده شده برای تزریقِ عبارتِ مخرب استفاده می‌کند.	Error_based (سطح E) از همان کانال، پیغام خطایی ایجاد می‌شود که اطلاعاتی را در اختیار مهاجم قرار می‌دهد.
	Union_based (سطح I) نتیجه‌ی پرس‌وجوی اول با پرس‌وجوی تزریقی ادغام می‌شود و در اختیار مهاجم قرار می‌گیرد.
استنتاج (Inference) (سطح I) انواع SQLI این دسته بر تکنیک‌های استنتاج استوار هستند.	Boolean_based_Blind (سطح E) نمی‌توان نتایج پرس‌وجو را مشاهده کرد اما پاسخ را بر اساس محتوای صفحه می‌توان تشخیص داد.
	Time_based_Blind (سطح I) نمی‌توان نتایج پرس‌وجو را مشاهده کرد و راهی برای تشخیص پاسخ پرسش بر اساس محتوای صفحه وجود ندارد. اما می‌توان از میزان تاخیر در رسیدن پاسخ استنتاج کرد.
خارج-از-محدوده (OOB) Out_Of_Band (سطح A) از کانال(های) جایگزین برای استخراجِ داده‌ها از کارپذیر استفاده می‌کنند.	HTTP (سطح E) نتیجه‌ی پرس‌وجوی SQL از طریق GET، به سمت کارپذیر HTTP کنترل‌شده توسط مهاجم ارسال می‌شود.
	System_File (سطح E) از سیستم فایل اشتراک‌گذاری شده به‌عنوان کانال جایگزین برای استخراج اطلاعات استفاده می‌شود.
	Email (سطح I) نتیجه‌ی پرس‌وجوی SQL به ایمیل مهاجم ارسال می‌شود.
	DNS (سطح I) هنگامی که کارخواه نیاز به جستجوی نام شبکه مورد استفاده در برنامه دارد، از کارپذیرهای DNS درخواست می‌کند. نتیجه‌ی درخواست‌های DNS توسط کارپذیر کنترل‌شده توسط مهاجم رهگیری شده و داده‌های ارزشمند استخراج می‌شود.

$$\text{Types} = \text{IN_Band} \cup \text{Inference} \cup \text{OOB} \qquad (۴)$$

به ازای هر پیلود می‌توان گفت که این پیلود کدام زیرمجموعه‌ی غیرتهی از مجموعه‌ی Types را شامل می‌شود. مؤلفه نوع حمله در مدل توان مهاجم بر اساس این پیلود به صورت مجموعه‌ای از زوج‌های دسته-مقدار نمایش داده می‌شود. به طور مثال برای پیلود شکل ۳ طبق جدول ۱ $\text{Types} = \{(\text{OOB}, \text{HTTP})\}$ است.

روش حمله: Technique شامل مجموعه تکنیک‌هایی است که مهاجم در طول حمله به کار برده است. توان مهاجم بر اساس مؤلفه‌ی Technique، متأثر از دانش مهاجم در زمینه‌ی سمپاد، کارپذیر-وب و

```
SELECT LOAD_FILE(CONCAT('\\\\',(SELECT
password FROM mysql.user WHERE
user='root' LIMIT
1),'.attacker.com\\foobar'));
```

شکل ۳. تزریق پیلود از نوع خارج-از-محدوده

مبهم‌سازی عبارات می‌باشد (جدول ۲). مجموعه Techniques طبق معادله (۵) اجتماعی از تکنیک‌ها در سه مجموعه‌ی نام‌برده است.

$$\text{Techniques} = \text{DBMS} \cup \text{Obfuscation} \cup \text{Web_Server} \qquad (۵)$$

به ازای هر پیلود می‌توان گفت که این پیلود کدام زیرمجموعه‌ی غیرتهی از مجموعه‌ی Techniques را شامل می‌شود. مؤلفه روش حمله در مدل توان مهاجم بر اساس این پیلود به صورت مجموعه‌ای از زوج‌های دسته-مقدار نمایش داده می‌شود. به طور مثال برای پیلود «' Order by 1#»، مجموعه‌ی تکنیک عبارت است از:

$$\text{Techniques} = \{(\text{DBMS}, \text{special characters}), (\text{DBMS}, \text{order by}), (\text{obfuscation}, \text{comment})\}$$

نقطه ورود: Entry_Point معرّف مؤلفه‌ی نقطه‌ی تزریق می‌باشد که شامل مجموعه‌ای از نقاط تزریق است که مهاجم در طول حمله به کار برده است. توان مهاجم با توجه به مؤلفه‌ی نقطه‌ی تزریق برآمده از این تفکر است که چه محلی از سامانه برای بازیابی و دستکاری داده‌های پایگاه داده مورد استفاده قرار می‌گیرد. نقطه‌ی تزریق را می‌توان با بررسی درخواست‌های ارسالی به کارپذیر بدست آورد. در این مقاله نقاط تزریق را به سه دسته‌ی ورودی‌های کاربر، متغیرهای HTTP و تزریق دو مرحله‌ای تفکیک کرده‌ایم (جدول ۳) بنابراین مجموعه‌ی Entry_Points طبق معادله (۶) اجتماع سه مجموعه‌ی نام‌برده می‌باشد:

$$\begin{aligned}\text{Entry_Points} = &\text{User_Input} \\ &\cup \text{HTTP_Variables} \\ &\cup \text{Second_Order_SQLI}\end{aligned} \qquad (۶)$$

به ازای هر پیلود با توجه به محتویات درخواست ارسال شده به کارپذیر می‌توان گفت که این پیلود کدام زیرمجموعه‌ی غیرتهی از مجموعه‌ی Entry_Point را شامل می‌شود. مؤلفه روش حمله در مدل توان مهاجم بر اساس این پیلود به صورت مجموعه‌ای از زوج‌های دسته-مقدار نمایش داده می‌شود. به طور مثال اگر پیلود از طریق کوکی ارسال شود داریم:

$$\text{Entry_Point} = \{(\text{Http_Variables}, \text{Cookie})\}$$

۲.۳ مدل توان مهاجم

هر کدام از مؤلفه‌های Type، Technique و Entry_Point شامل مجموعه‌ای از زوج مرتّب‌های (دسته، مقدار) می‌باشد که در بخش قبل شرح آن گفته شد. مهاجم برای رسیدن به هدف مطلوب خود ممکن است از چندین پیلود استفاده کند یا به عبارت دیگر چندین بار فرایند حمله

جدول ۲. معرفی روش‌های حمله

دسته	تکنیک
دانش سمپاد (DBMS) (سطح E)	شناخت دستورات SQL، شناخت ساختار سمپاد، توابع و رویه‌های ذخیره‌شده‌ی موجود در پایگاه‌داده مانند SELECT، FROM، Insert، Delete، CONCAT()، CHAR()، SLEEP()، INFORMATION_SCHEMA.SCHEMA و ...
دانش مبهم‌سازی (Obfuscation) (سطح I)	شناخت تکنیک‌هایی برای دورزدن روش‌های ایمن‌سازیِ سامانه مانند Encoding، Comment، Keywords Replace، Changing Case، Query Complex، ...
دانش کارپذیر-وب (Web_Server) (سطح A)	شناخت نحوه‌ی برخورد کارپذیر-وب با اعمال تغییرات در پارامترها مانند Pollution Parameter Http و Contamination Parameter Http

* لازم به ذکر است به علت زیاد بودن تعداد تکنیک‌ها از نمایش تمام تکنیک‌ها و سطح‌بندی آن‌ها خودداری کرده‌ایم.

را با پیلودهای متفاوت تکرار کند. بنابراین توانِ مهاجم از مجموعه‌ای از تلاش‌های وی برای حمله تشکیل شده است.

برای یک مهاجم که مجموعه پیلودهای $P = \{p_1, p_2, \ldots, p_n\}$ را برای حمله به سیستم تزریق می‌کند مدل توان مهاجم با

$$\langle \mathtt{Type}_P, \mathtt{Technique}_P, \mathtt{Entry_Point}_P \rangle$$

مشخص می‌شود که در آن $\mathtt{Type}_P$ بیانگر انواع پیلودهای به کار رفته در مجموعه پیلود P، $\mathtt{Technique}_P$ بیانگر انواع تکنیک‌های به کار رفته در مجموعه پیلود P و $\mathtt{Entry_Point}_P$ بیانگر مجموعه نقاط تزریق به کار رفته برای تزریق پیلودهای P است که

$$\begin{aligned}
\mathtt{Type}_P &= \mathtt{Type}_1 \cup \mathtt{Type}_2 \cup \cdots \cup \mathtt{Type}_n, \\
\mathtt{Technique}_P &= \mathtt{Technique}_1 \cup \mathtt{Technique}_2 \cup \cdots \\
&\quad \cup \mathtt{Technique}_n, \\
\mathtt{Entry_Point}_P &= \mathtt{Entry_Point}_1 \cup \mathtt{Entry_Point}_2 \\
&\quad \cup \cdots \cup \mathtt{Entry_Point}_n
\end{aligned}$$

به طور مثال برای مجموعه پیلودهای تزریق شده $P = \{p_1, p_2, p_3\}$ داریم:

$$\begin{aligned}
p_1 &= \mathtt{1'And\ sleep(5)\#} \\
p_2 &= \mathtt{1'And\ substr(database(),1,1)='d'\#} \\
p_3 &= \mathtt{1\ order\ by\ 3}
\end{aligned}$$

جدول ۳. معرفی نقاط تزریق

دسته	نقطه‌ی تزریق
ورودی‌های کاربر User_Input (سطح E)	GET method (سطح E)
	POST method (سطح I)
متغیرهای HTTP Http_Variables (سطح I)	User_Agent (سطح E)
	Referer (سطح I)
	Cookie (سطح A)
تزریق دومرحله‌ای Second_Order (سطح A)	Frequenc_based_Primary_Application (E)
	Frequency_based_Secondary_Application (I)
	Secondary_Support_Application (I)
	Cascaded_Submission_Application (A)

$$\begin{aligned}
&\mathtt{Type}_{p_1} = \{(\mathtt{Inference}, \mathtt{Time_based_blind})\} \\
&\mathtt{Technique}_{p_1} = \{(\mathtt{DBMS}, \mathtt{special\ char}), (\mathtt{DBMS}, \mathtt{And}), \\
&(\mathtt{DBMS}, \mathtt{sleep()}), (\mathtt{obfuscation}, \mathtt{comment})\} \\
&\mathtt{Entry_Point}_{p_1} = \{(\mathtt{Input_User}, \mathtt{Get})\} \\
&\mathtt{Type}_{p_2} = \{(\mathtt{Inference}, \mathtt{Boolean_based_blind})\} \\
&\mathtt{Technique}_{p_2} = \{(\mathtt{DBMS}, \mathtt{special\ char}), (\mathtt{DBMS}, \mathtt{And}), \\
&(\mathtt{DBMS}, \mathtt{substr()}), (\mathtt{DBMS}, \mathtt{database()}), \\
&(\mathtt{obfuscation}, \mathtt{comment})\} \\
&\mathtt{Entry_Point}_{p_2} = \{(\mathtt{Input_User}, \mathtt{Get})\} \\
&\mathtt{Type}_{p_3} = \{(\mathtt{IN_Band}, \mathtt{Error_Based\backslash verb})\} \\
&\mathtt{Technique}_{p_3} = \{(\mathtt{DBMS}, \mathtt{order\ by}), \\
&(\mathtt{obfuscation}, \mathtt{comment})\} \\
&\mathtt{Entry_Point}_{p_3} = \{(\mathtt{Input_User}, \mathtt{Get})\} \\
&\mathtt{Type}_P = \{(\mathtt{Inference}, \mathtt{Time_based_blind}), \\
&(\mathtt{Inference}, \mathtt{Boolean_based_blind}), \\
&(\mathtt{IN_Band}, \mathtt{Error_Based})\} \\
&\mathtt{Technique}_P = \{(\mathtt{DBMS}, \mathtt{special\ char}), (\mathtt{DBMS}, \mathtt{And}), \\
&(\mathtt{DBMS}, \mathtt{substr()}), (\mathtt{DBMS}, \mathtt{database()}), \\
&(\mathtt{obfuscation}, \mathtt{comment}), (\mathtt{DBMS}, \mathtt{sleep()}), \\
&(\mathtt{DBMS}, \mathtt{order\ by})\} \\
&\mathtt{Entry_Point}_P = \{(\mathtt{Input_User}, \mathtt{Get})\}
\end{aligned}$$

۳.۳ سطح‌بندی توان مهاجم

همانطور که در مؤلفه‌های مدل توان مهاجم بیان شد توان مهاجم برآمده از سه مؤلفه‌ی Type، Technique و Entry_Point است که هر کدام به سه دسته تفکیک شده است. هر کدام از این دسته‌ها بر اساس

تجربه و منابع موجود دارای اهمیت متفاوتی هستند و می‌توانند سطح متفاوتی از توان مهاجم را نشان دهند. به همین دلیل ما برای بیان سطح توان مهاجم این دسته‌ها را با E (Elementary)، I (Intermediate) و A (Advanced) سطح‌بندی کرده‌ایم. اگر سطح دسته را با l_c و سطح مقدار را با l_v نشان دهیم سطح دسته–مقدار یعنی (c, v) را با $l_c l_v$ نمایش می‌دهیم. بر این اساس، رابطه برتری بین دوسطح به شکل زیر تعریف می شود.

رابطه برتری سطح بین دو دسته–مقدار: سطح دو دسته–مقدار (c, v) و (c', v') را با هم این گونه مقایسه می‌کنیم که سطح (c, v) یعنی $l_c l_v$ از سطح (c', v') یعنی $l_{c'} l_{v'}$ بالاتر است اگر سطح دسته c بزرگتر مساوی سطح دسته c' باشد و یا با سطوح دسته یکسان، سطح مقدار v بزرگ‌تر از سطح مقدار v' باشد. رابطه (۷) مقایسه سطح دو دسته–مقدار (c, v) و (c', v') را نشان می دهد.

$$l_c l_v \geq l_{c'} l_{v'} \Rightarrow (l_c \geq l_{c'}) \vee (l_c = l_{c'} \wedge l_v \geq l_{v'}) \quad (۷)$$

اگر سطح مؤلفه‌ی m را با l_m نشان دهیم، l_m بیشینه سطح دسته–مقدارهای تشکیل دهنده آن مؤلفه خواهد بود:

$$l_m = \max\{(l_c l_v)_۱, (l_c l_v)_۲, \ldots, (l_c l_v)_z\} \quad (۸)$$

به طور مثال برای مجموعه پیلود P داریم:

$$\begin{aligned} l_{\texttt{Type}_P} &= \max\{(II), (IE), (EE)\} = II \\ l_{\texttt{Technique}_P} &= \max\{(EE), (EE), (EI), (EI), \\ &\quad (IE), (EI), (EE)\} = IE \\ l_{\texttt{Entry_Point}_P} &= \max(EE) = EE \end{aligned}$$

با توجه به اینکه مجموعه‌ی L دارای سه عضو است برای l_m نُه حالت متصور است. مجموعه‌ی L' حالت‌های l_m را در بر دارد که رابطه برتری بین آن‌ها برقرار است:

$$\begin{aligned} L' &= \{EE, EI, EA, IE, II, IA, AE, AI, AA\} \\ &\xrightarrow{\text{such that}} EE < EI < EA < IE \\ &< II < IA < AE < AI < AA \end{aligned} \quad (۹)$$

با در اختیار داشتن مدل توان مهاجم، سطح توان مهاجم (L_A) با استفاده از معادله (۱۰) محاسبه می‌شود.

$$L_A = \langle l_{\texttt{Type}}, l_{\texttt{Technique}}, l_{\texttt{Entry_Point}} \rangle \quad (۱۰)$$

به طور مثال برای مهاجم با مجموعه P داریم:

$$L_A = \langle II, IE, EE \rangle \quad (۱۱)$$

۴.۳ سطح‌بندی امنیت سامانه با استفاده از سطح توان مهاجم

طبق آنچه پیش‌تر گفته شد سطح توان مهاجم A را از سطح توان مهاجم B بزرگ‌تر گوییم اگر مؤلفه‌های L_{CA} مهاجم A نظیر به نظیر از مؤلفه‌های L_{CA} مهاجم B بزرگ‌تر باشد.

بنابراین با رتبه‌بندی L_{CA}ها می‌توان گفت اگر سامانه به وسیله‌ی مهاجم با بیشترین L_{CA} آسیب‌پذیر نباشد سطح امنیت سامانه حداقل به اندازه‌ی L_{CA} این مهاجم است و توسط هر مهاجمی که L_{CA} آن کمتر از این باشد نیز آسیب‌پذیر نمی‌باشد.

بر این اساس امنیت یک سامانه را معادل کمترین سطح مهاجمی که بتواند حمله موفق به آن داشته باشد تعریف کنیم. برای مقایسه امنیت دو سامانه $S۱$ و $S۲$ می‌گوییم $S۱$ امن‌تر است اگر سطح توان مهاجم متناظر با آن بیشتر از سطح توان مهاجم متناظر با سامانه $S۲$ باشد. به طور مشابه اگر سامانه به وسیله‌ی مهاجم با کمترین L_{CA} آسیب‌پذیر باشد سطح امنیت سامانه حداقل به اندازه‌ی L_{CA} این مهاجم است و توسط هر مهاجمی که L_{CA} آن بیشتر از این باشد نیز آسیب‌پذیر است.

۴ ارزیابی تجربی مدل توان مهاجم

برای ارزیابی روش پیشنهادی از چند مهاجم با توان متفاوت استفاده شده است. به این صورت که یک سامانه برای آزمون در اختیار این مهاجم‌ها قرار گرفته و پیلودهایی که هرکدام استفاده کردند جمع‌آوری شد. سپس، طبق روش ارائه شده، مدل توان مهاجم برای هر یک، تولید و سطح‌بندی شد. به‌ازای هر مهاجم می‌توان سطح توان به‌دست آمده را با آنچه قبل از ارزیابی درباره سطح توان وی تصور می‌شد مقایسه کرد و تحلیلی بر اساس نتیجه‌ی مقایسه درباره‌ی توان مهاجم ارائه کرد. همچنین می‌توان کمترین سطح توان مهاجم که موفق به کشف آسیب‌پذیری در سامانه شده است را به‌عنوان سطح امنیت سامانه در نظر گرفت.

در آزمایش انجام شده یک برنامه تحت وب که در آن از کتابخانه react برای واسط کاربری و چارچوب Net Core 5. برای Backend استفاده شده بود، انتخاب گردید. در این سامانه پایگاه داده MS SQL Server و Neo4j استفاده شده بود.

در این آزمون سه مهاجم شرکت کردند. هر مهاجم به مدت یک ساعت روی سامانه بدون استفاده از ابزارهای تزریق SQL تزریق پیلود را انجام دادند. همه درخواست‌ها و پاسخ‌ها لاگ‌برداری شد و پیلودهای هر یک استخراج شد و مورد تجزیه و تحلیل قرار گرفت. سطح توان مهاجم‌ها بر اساس پیلودهای زده شده طبق معادله (۱۲) استخراج گردید:

$$\begin{aligned} l_{CA_۱} &= \langle II, EI, EE \rangle \\ l_{CA_۲} &= \langle II, EI, IE \rangle \\ l_{CA_۳} &= \langle II, EI, IE \rangle \end{aligned} \quad (۱۲)$$

با مقایسه‌ی سطح توان‌های به‌دست آمده، توانایی مهاجم‌ها به صورت معادله (۱۳) رده‌بندی شد:

$$(CA)_۱ < (CA)_۲ < (CA)_۳ \quad (۱۳)$$

نتایج به دست آمده در مورد توان مهاجمین در این مدل، با انتظار و شناخت قبلی از آن‌ها براساس تجربیات کاری و بازخوردهای افراد خبره در مورد توان آن‌ها است.

۵ نتیجه‌گیری

در این مقاله، فرایند عمومی حمله‌ی تزریق SQL را با استفاده از ماشین حالت قطعی نشان دادیم و با استخراج مؤلفه‌های اصلیِ شکل‌گیریِ یک حمله، مدل توان مهاجم را ارائه کردیم. این مدل شامل سه تایی $\langle \mathtt{Type}, \mathtt{Technique}, \mathtt{Entry_Point} \rangle$ است. هر مؤلفه از این سه‌تایی، مجموعه‌ای از زوج مرتب‌های (دسته–مقدار) است. با تکیه بر منابع موجود و تجربیات مهاجمین، دسته–مقدارها در قالب یکی از سطوح مقدماتی (E)، متوسط (I) یا پیشرفته (A) سطح‌بندی شدند که مبنای سطح‌بندی مهاجم قرار می‌گیرد.

هر چند که پژوهش انجام شده به صورت محدود و با تعداد مهاجمین کم مورد ارزیابی تجربی قرار گرفت، ارزیابی انجام شده می‌تواند روی جامعه بزرگتری از مهاجمین انجام شود. با رتبه‌بندی سطح توان مهاجم‌هایی که یک سامانه را مورد آزمون نفوذ قرار می‌دهند می‌توانیم درباره سطح امنیت سامانه اظهار نظر کنیم و یا امنیت سامانه‌های مختلف را در شرایط یکسان مقایسه نماییم. اگر سامانه به‌وسیله‌ی مهاجم با بیشترین سطح توان آسیب‌پذیر نباشد، سطح امنیت سامانه حداقل به اندازه‌ی سطح توان این مهاجم است و توسط هر مهاجمی که سطح توانش کمتر از این مهاجم باشد نیز آسیب‌پذیر نمی‌باشد.

یکی از نکاتی که در ادامه این پژوهش قابل انجام است تعیین اثر فاکتورهایی مانند زمان است که در مدل ارائه شده مدنظر نبوده است. این که مهاجم با تلاش کمتر و یا با تعداد پیلودهای کمتر و یا در زمان کمتر بتواند امنیت سامانه را نقض کند، می‌تواند به عنوان یک عامل موثر در بهبود مدل‌سازی انجام شده برای سطح توان مهاجم مدنظر قرار گیرد. در نظر گرفتن فاکتور زمان به تنهایی در ارزیابی توانِ طیف‌های مختلفِ مهاجم باعث افزایش مثبت اشتباه و منفی اشتباه خواهد شد و دقت ارزیابی را کاهش می‌دهد. به طور مثال دو مهاجم را در نظر بگیریم: یک مهاجم مبتدی با استفاده از یک مجموعه پیلود آماده یا ابزار، تزریق‌های کورکورانه‌ای روی سامانه انجام می‌دهد و یک مهاجم نیمه‌ماهر سلسه پیلودهای خود را بر اساس نتایجی که از هر تزریق به‌دست می‌آورد انتخاب و تزریق می‌کند. بدیهی است که مهاجم اول از نظر زمانی پیلودهای بیشتری را تزریق می‌کند ولی درک کاملی از تزریق خود ندارد در صورتی که مهاجم دوم پیلودهای کمتری را در واحد زمان تزریق می‌کند ولی طبق استدلال خاصی تزریق خود را انجام می‌دهد. بنابراین بهتر است با استفاده از رفتارشناسی مهاجم‌های مختلف درک درستی از تحلیل مهاجم‌ها از نتیجه‌ی حمله‌ی قبلی و انتخاب حمله بعدی به‌دست آورد تا به ارزیابی دقیق‌تری دست یافت.

مراجع

[1] ben Othmane, Lotfi, Ranchal, Rohit, Fernando, Ruchith, Bhargava, Bharat, and Bodden, Eric. Incorporating attacker capabilities in risk estimation and mitigation. *Computers & Security*, 51:41–61, 2015.

[2] Chhabra, Deepshikha and Sharma, Isha. Role of attacker capabilities in risk estimation and mitigation. in *Analyzing the Role of Risk Mitigation and Monitoring in Software Development*, pp. 244–255. IGI Global, 2018.

[3] https://owasp.org/www-project-top-ten/. [Online].

[4] McQueen, Miles A, Boyer, Wayne F, Flynn, Mark A, and Beitel, George A. Time-to-compromise model for cyber risk reduction estimation. in *Quality of Protection: Security Measurements and Metrics*, pp. 49–64. Springer, 2006.

[5] Paulauskas, N and Garsva, E. Attacker skill level distribution estimation in the system mean time-to-compromise. in *2008 1st International Conference on Information Technology*, pp. 1–4. IEEE, 2008.

[6] Rothstein-Morris, Eric and Jun, Sun. Quantifying attacker capability via model checking multiple properties (extended version). *arXiv preprint arXiv:1811.10400*, 2018.

[7] Jonsson, Erland and Olovsson, Tomas. A quantitative model of the security intrusion process based on attacker behavior. *IEEE Transactions on Software Engineering*, 23(4):235–245, 1997.

[8] ben Othmane, Lotfi, Weffers, Harold, and Klabbers, Martijn. Using attacker capabilities and motivations in estimating security risk. in *Workshop on risk perception in it security and privacy, Newcastle, UK*. Citeseer, 2013.

MonadiAfta-Bianual Scientific Journal for Cyberspace Security
2022, Volume 20, Number 2 (pp. 52–60)
https://monadi.isc.org.ir

Presented at the ISCISC 2022 in University of Giulan, Rasht, Iran

Modeling Attacker's Capability Level Focusing on Injection Attacks on Web Applications★

Sara Moqimi * and Mohammad Ali Hadavi

Faculty of Electrical and Computer Engineering, Malek Ashtar University of Technology, Tehran, Iran

ARTICLE INFO.

Keywords:
attacker's capability
SQL injection
type of injection
injection technique
entry_point
system security

dor: 20.1001.1.24763047.1401.11.2.6.5

Type: research paper

ABSTRACT

How to exploit vulnerabilities and their damage potentials are mainly affected by the capability of attackers. The more powerful the attacker, the greater risk of threats and vulnerabilities. Therefore, the security analysis of a web application and choosing risk mitigation countermeasures depend on the ability of the attackers threaten the application. Focusing on SQL injection attacks, this paper is aimed at modeling the attacker's capability to be further used for appropriate security evaluation and choosing cost-effective security controls. We model the attacker's capability with the triple ⟨`Type`, `Technique`, `Entry_Point`⟩. The value in each component of the triple is obtained from the payloads through which the attacker tries to exploit the injection vulnerabilities. The `Type` represents the injection type, including a known set of injection attack types namely, `Error_based`, `Union_based`, `Boolean_based_Blind` and etc. The `Technique` represents the techniques, which are used by the attacker during the attack, e.g. using `Special` Character, using `UNION`, using `Complex` Query, using `Encoding` and etc. Finally, the `Entry_Point` represents the set of known injection entry points including `GET/POST` method, `Http_Variables`, `Frequenc_based_Primary_Application` and etc. This model is used for leveling and comparing the attacker's capabilities as well as for leveling the security of a web application with respect to the level of the attacker who is able to compromise the web application. The results of the experimental evaluation show that the proposed model can be used to determine the attacker's capability level. The model can be simply extended to adopt other security vulnerabilities attacks.

© 2022 ISC

★ The ISCISC 2022 Program Committee effort is highly acknowledged for reviewing this paper.
* Corresponding author
Email addresses: saramoqimi@mut.ac.ir (Sara Moqimi), hadavi@mut.ac.ir (Mohammad Ali Hadavi)
© 2022 ISC. All rights reserved.