بسمه تعالی

سازمان فناوری اطلاعات ایران

معاونت امنیت فضای تولید و تبادل اطلاعات

مرکز ماهر

جرم‌شناسی در پایگاه‌داده Cassandra

پیشگفتار

رشد روز افزون حجم اطلاعات از یک سو و پیشرفت فن‌آوری‌های نوین از سوی دیگر سبب شده تا طیف وسیعی از تهدیدها و جرم‌ها در پایگاه‌داد مطرح گردد. وجود این تهدیدها و جرم‌ها سبب شده تا تمهیدات امنیتی چون حفظ محرمانگی، صحت و دسترس‌پذیری در این سامانه‌ها از جایگاه ویژه‌ای برخوارد باشد. مکانیزم‌های امنیتی موجود به سبب حفظ کارایی سامانه نمی‌توانند جلوی تمامی تهدیدها و جرایم اینترنتی در پایگاه‌های داده را بگیرند و لذا امکان وقوع جرم در سامانه امکان‌پذیر است. از این‌رو، به منظور شناسایی شواهد جرم، جمع‌آوری اطلاعات مربوطه، تجزیه‌وتحلیل آن‌ها و در نهایت تهیه مستندات لازم جهت اثبات وجود جرم، شاخه‌ی جرم‌شناسی پایگاه‌داده مطرح می‌گردد. وجود زیرساخت‌های مختلف برای سامانه‌های پایگاه‌داده، طبیعت چند بعدی سامانه‌ها، ابزارهای مختلف جرم‌شناسی و فقدان مدیریت دانش مرتبط با جرم‌شناسی را می‌توان از جمله چالش‌های اصلی در این حوزه دانست. وجود این چالش‌ها سبب شده است تا جرم‌شناسی پایگاه‌داده به عنوان یک موضوع پیچیده و بغرنج معرفی گردد.

در این راستا بر آن شدیم تا جرم‌شناسی در پایگاه‌داده Cassandra را برای برخی از عملکردهای پر کاربرد، مورد بحث و بررسی قرار دهیم. بدین منظور با استفاده از فرآیند استاندارد جرم‌شناسی، مراحل شناسایی، گردآوری اطلاعات، تحلیل، ترمیم و ارائه مستندات کافی برای اثبات جرم را تشریح می‌کنیم. لازم به ذکر است که اگر چه جرم‌شناسی پایگاه‌داده از حدود سال ۲۰۰۴ مطرح است اما تا کنون پیشرفت خاصی در زمینه‌ی ابزارهای تجاری و رایگان برای رسیدگی به رویدادهای غیرمجاز ارائه نشده است.

در ادامه و در فصل اول، مفاهیم و تعاریف اولیه‌ی جرم‌شناسی پایگاه‌داده، چالش‌ها، اهداف و گام‌های اجرایی جرم‌شناسی تشریح می‌شود. در فصل دوم، فرآیند جرم‌شناسی و مدیریت جرم برای هر سامانه پایگاه‌داده (سمپاد) مورد بحث و بررسی قرار می‌گیرد. در فصل‌های سوم تا پنجم، گام‌های شناسایی، جمع‌آوری شواهد و اطلاعات، استخراج و تحلیل اطلاعات، ترمیم و ارائه مستندات مربوط به جرم به ترتیب برای هر یک از رویدادهای درج، حذف، بروزرسانی و مشاهده غیرمجاز محتوای جداول، تغییر غیرمجاز شمای پایگاه‌داده و تلاش برای ورود غیرمجاز به سامانه پایگاه‌داده مطالعه می‌شود. لازم به ذکر است که کلیه پیکربندی‌های ارائه شده در مستند حاضر بر اساس Datastax enterprise 6.0.1 می‌باشد.

فهرست مطالب

۱ جرم شناسی پایگاه‌داده

بسیاری از سازمان‌ها و آژانس‌های دولتی از پایگاه‌داده برای ذخیره و بازیابی اطلاعات استفاده می‌کنند. از آنجاییکه داده‌های سازمان‌ها می‌توانند حاوی اطلاعات حساس و حیاتی باشند، لذا حفاظت از پایگاه‌داده به عنوان یک سامانه ذخیره‌سازی اطلاعات، یک امر حیاتی و مهم به حساب می‌آید. داده‌های ذخیره شده در پایگاه‌داده ممکن است توسط کاربران غیرمجاز (از درون سازمان یا بیرون سازمان) تغییر داده شوند. لازم به ذکر است که بسیاری از مجرمان به دلیل فقدان دلایل کافی برای اثبات جرم، محکوم نمی‌شوند. در این شرایط، جرم‌شناسی نقش مهمی در ارایه روش‌های اثبات شده علمی برای جمع‌آوری اطلاعات، تحلیل و بررسی و نهایتا ارائه‌ی شرح مفصلی از فعالیت‌های مجرمانه‌ی سایبری ایفا می‌کند. جرم‌شناسی پایگاه‌داده، شاخه‌ای از جرم‌شناسی دیجیتال است که در آن پایگاه‌داده و فراداده‌های مرتبط با آن به صورت قانونی مورد مطالعه قرار می‌گیرند. یکی از مهم‌ترین اهدفی که در جرم‌شناسی پایگاه‌داده دنبال می‌شود آن است که تشخیص دهیم چه کسی، چه زمانی، چه داده‌ای را تغییر داده است. لازم به ذکر است که پایگاه‌داده قربانی معمولاً شامل اطلاعاتی است که در طول تحقیقات قانونی به کار می‌آید. در ادامه برخی از تعاریف و مفاهیم کلیدی، چالش‌ها، اهداف و گام‌های اجرایی فرآیند جرم‌شناسی در سامانه پایگاه‌داده، مورد بحث و بررسی قرار می‌گیرد.

۱-۱ تعاریف و مفاهیم

در این بخش، برخی از مهم‌ترین تعاریف و مفاهیم جرم‌شناسی پایگاه‌داده که در سرتاسر مستند مورد استفاده قرار گرفته است، مورد بحث و بررسی قرار می‌گیرد.

سامانه مدیریت پایگاه‌داده: به نرم افزاری اطلاق می‌شود که برای نگهداری و مدیریت حجم وسیعی از اطلاعات طراحی شده و مورد استفاده قرار می‌گیرد [۴].

جرم‌شناسی: مجموعه‌ای از آزمایش‌ها یا روش‌های علمی است که در تحقیقات جنایی مورد استفاده قرار می‌گیرد. امروزه جرم‌شناسی به روشی برای به دست آوردن شواهد جنایی به منظور ارائه در دادگاه اشاره دارد.

تجزیه و تحلیل جرم‌شناسی: به فرآیند بررسی رویدادهای غیرمجاز با در نظر گرفتن خط زمانی[1] موجود از شواهد فیزیکی مربوط به آن، تجزیه و تحلیل جرم‌شناسی اطلاق می‌شود. نتایج حاصل می‌تواند در شناسایی مجرم کمک کند و همچنین به منظور ارایه شواهد برای اثبات جرم، مورد استفاده قرار گیرد [۱].

[1] Timeline

جرم‌شناسی دیجیتال: به فرآیندی که در آن از روش‌های علمی مرسوم و اثبات شده برای: (۱): حفاظت، (۲): جمع‌آوری، (۳): اعتبارسنجی، (۴): شناسایی، (۵): تجزیه‌وتحلیل، (۶): تفسیر، (۷): مستندسازی و ارائه‌ی شواهد دیجیتال به منظور تسهیل در بازسازی رویدادهای شناخته شده جنایی یا پیش‌بینی اقدامات غیرمجاز استفاده می‌شود، جرم شناسی دیجیتال اطلاق می‌گردد [۵].

جرم‌شناسی پایگاه‌داده: جرم‌شناسی پایگاه‌داده فرآیندی است که تلاش می‌کند تا زمان/چگونگی/ چرایی و عامل (های) رویداد غیرمجاز در سامانه را مشخص نماید. لازم به ذکر است که محتوای پایگاه‌داده، فراداده‌ها[۲] (به ویژه فایل‌های رویدادنگاری) و داده‌های موجود در حافظه از جمله مهمترین مولفه‌های تاثیرگذار در این فرآیند به حساب می‌آیند.

ممیزی:[۳] به نظارت و ثبت فعالیت‌های کاربران در پایگاه‌داده، ممیزی اطلاق می‌شود [۶].

رویدادنگاری:[۴] به تاریخچه‌ای از فعالیت‌های اجرا شده توسط سامانه مدیریت پایگاه‌داده اطلاق می‌شود که برای تضمین ویژگی‌های جامعیت (ACID[۵]) به هنگام خرابی سخت افزاری[۶] یا از کار افتادگی ناگهانی[۷] مورد استفاده قرار می‌گیرد [۷].

مصنوعات[۸] پایگاه‌داده: رکوردها یا اطلاعاتی هستند که از پایگاه‌داده قابل استخراج بوده و در تجزیه و تحلیل جرم‌شناسی مفید هستند [۲].

رویداد غیرمجاز: به رویدادی اطلاق می‌شود که به صورت خصمانه یا ناخواسته در روال عادی سیستم تغییر نامطلوبی را ایجاد می‌کند.

۲-۱ چالش‌ها

جرم‌شناسی پایگاه‌داده، چالش‌های زیادی به همراه دارد که آن را تبدیل به یک موضوع پیچیده می‌کند. سامانه‌های پایگاه‌داده، سرویس‌ها و زیرساخت‌های مختلفی دارند که از یک پایگاه‌داده به پایگاه‌داده دیگر، متفاوت هستند. علاوه بر این، پایگاه‌های داده مختلف دارای مصنوعات جرم‌شناسی متفاوت همچون روش‌ها، مدل‌ها، چارچوب‌ها، ابزارها، فعالیت‌ها و خط‌مشی‌های مختلف هستند از سوی دیگر، سامانه‌های پایگاه‌داده

---

[2] Metadata
[3] Auditing
[4] Logging
[5] Atomicity-Consistency-Isolation-Durability
[6] Hardware failure
[7] Crash
[8] Artifact

دارای طبیعت چند بعدی شامل سطح داخلی، سطح مفهومی و سطح خارجی هستند. سطح داخلی شامل فایل فیزیکی است و سطح مفهومی، سطح منطقی است که زیرساخت منطقی شمای پایگاه‌داده ازجمله کاربران، جداول، شاخص‌ها و رویه‌ها را نمایش می‌دهد. سطح خارجی با کاربران واقعی سروکار دارد تا بتوانند داده‌ها را تغییر دهند. بنابراین، ابعاد مختلف پایگاه‌داده در جرم‌شناسی پایگاه‌داده ایفای نقش می‌کنند [۸].

یک چالش مهم دیگر در حوزه جرم‌شناسی پایگاه‌داده، تشخیص وجود نقض[9] امنیتی در سامانه است. در واقع، فرآیند شناسایی و ترمیم به هنگام وقوع جرم، تا زمانی که شخصی فکر کند که نقض امنیتی رخ داده است، به تاخیر می‌افتد. در حالت کلی، شواهد برای وجود نقض امنیتی را می‌توان در سه دسته‌ی زیر خلاصه کرد:

- داده‌های سازمان در خارج از سازمان پیدا می‌شوند که مسلماً عادی و مجاز نیست.
- رویدادی مشاهده می‌شود که غیرمنتظره است؛ مثلاً فرآیندی در زمان اشتباه اجرا شده است یا دسترسی به سامانه، خارج از ساعت‌های اداری و مجاز اتفاق افتاده است.
- نشانه‌هایی از تغییر غیرمجاز داده‌ها وجود داشته باشد.

روش‌هایی برای جمع‌آوری و تجمیع شواهد مجرمانه در پایگاه‌داده وجود دارد. جرم‌شناسی پایگاه‌داده زمانی رخ می‌دهد که از مأمور ممیزی، نحوه وقوع نقض امنیتی و شخص مجرم، درخواست شود. روش‌هایی که برای جرم‌شناسی پایگاه‌داده وجود دارند، اصولاً دارای دو محدودیت زیر هستند:

- کاربر پس از هفته‌ها یا ماه‌ها متوجه نقض امنیتی در پایگاه‌داده می‌شود. در این صورت داده‌های ناپایدار در پایگاه‌داده به عنوان شواهد وجود ندارد.
- ممکن است هیچ ممیزی برای پایگاه‌داده فعال نباشد.

دو عامل فوق‌الذکر سبب می‌شوند که بررسی رویداد مورد تقاضا، زمان آن و نتیجه‌ی حاصل از بررسی در پیچیده‌ترین حالت ممکن قرار گیرد. آنچه جرم‌شناسی پایگاه‌داده را از جرم‌شناسی شواهد فیزیکی متمایز می‌کند، حجم پایگاه‌داده و نیاز به در حال اجرا ماندن آن در محیط عملیاتی است.

۳-۱ اهداف

برخی از مهم‌ترین اهدافی که در جرم‌شناسی پایگاه‌داده به دنبال آن هستیم به قرار زیر است [۸-۹]:

ردگیری[10] اعمال DML و DDL: در چرخه حیات پایگاه‌های داده بارها نیاز می‌شود که تغییرات در داده‌ها همچون درج، بروزرسانی و حذف داده‌ها که از اعمال دستکاری داده (DML) به حساب می‌آیند و تغییرات در

[9] Breach

اشیای پایگاه‌داده همچون ایجاد، تغییر و حذف یک جدول که از اعمال تعریف داده (DDL) به حساب می‌آیند، ردگیری و بررسی شوند. با این کار، رویدادهای غیرمجاز تشخیص داده شده و مجرمان شناسایی می‌شوند.

شناسایی داده‌ها پیش و پس از تراکنش: در طول یک تراکنش، ممکن است داده‌ها دستخوش تغییرات زیادی شوند. گاهاً داده‌های جدیدی ایجاد، داده‌های موجود حذف یا تغییر داده می‌شوند. شناسایی تغییرات اعمال شده بر روی داده‌ها، ما را در تشخیص رویدادهای غیرمجاز کمک خواهد کرد.

بازگشت به عقب اعمال غیرمجاز تغییر داده: در صورتی که داده‌ها طی رویدادهای غیرمجاز تغییر داده شوند، باید بتوان آن‌ها را به وضعیت پیش از رویداد غیرمجاز بازگرداند. همچنین در صورت حذف داده‌ها، نیاز به بازیابی داده‌های حذف شده خواهد بود.

اثبات یا رد وقوع نقض امنیتی: یک چالش مهم در حوزه جرم‌شناسی پایگاه‌داده، تشخیص وجود نقض امنیتی در سامانه است. در واقع فرآیند شناسایی و ترمیم به هنگام وقوع جرم، تا زمان تشخیص نقض امنیتی در سامانه به تاخیر می‌افتد. پس از آن با طی کردن گام‌های مطرح در فرآیند جرم‌شناسی پایگاه‌داده می‌توان ثابت کرد که نقض امنیتی رخ داده است یا خیر.

تعیین محدوده‌ی نفوذ به پایگاه‌داده: هنگامی که به پایگاه‌داده حمله می‌شود، تشخیص حمله‌ی انجام گرفته و محدوده‌ی نفوذ به منظور شناسایی خسارات وارد شده و محدوده‌ی تغییرات غیرمجاز، از اهمیت بالایی برخودار است.

کشف اینکه چه اتفاقی در چه زمانی رخ داده است: با بررسی و تحلیل رویدادهای ثبت شده، اطلاعاتی همچون کاربر اجراکننده رویداد، زمان رخداد، داده متاثر از رویداد، چگونگی تغییر و علت آن مشخص می‌شود. با دانستن این اطلاعات، نه تنها جزییات رویدادهای رخ داده مشخص می‌شوند بلکه می‌توان با توجه به توالی زمانی رویدادها و تجمیع آن‌ها، اطلاعات جدیدی نیز کسب کرد. برخی از داده‌ها به تنهایی دارای ارزش کمی هستند ولی هنگامی که با سایر اطلاعات ترکیب می‌شوند، می‌توانند نقض امنیتی را آشکارتر سازند.

---

[1] Trace

۱-۴ گام‌های اجرایی

گام‌هایی که برای جرم‌شناسی پایگاه‌داده باید برداشته شوند به موقعیت و نوع سامانه مدیریت پایگاه‌داده‌ی وابسته است. در ادامه، برخی از مهم‌ترین گام‌هایی که در این راستا می‌بایست لحاظ شود، آورده شده است.

۱ . **مرحله‌ی شناسایی:** در مرحله‌ی شناسایی، رویداد و نوع آن با توجه به نشانه‌های موجود در سامانه شناسایی می‌شود. این مرحله از آن جهت مهم است که سایر مراحل را تحت تأثیر قرار می‌دهد. برخی از مهمترین اهداف این مرحله، شناسایی مفاهیم مرتبط با بررسی جرم‌شناسی همچون منابع پایگاه‌داده، منابع سیستم‌عامل، منابع شبکه، تیم‌های بررسی، روش‌های بررسی، محیط بررسی، خط‌مشی‌ها، قوانین و مجوزها هستند.

۲ . **تعیین روش جمع‌آوری داده‌های جرم‌شناسی:** به منظور بررسی پایگاه‌داده آسیب‌دیده یا مورد سوءاستفاده، سه روش جمع‌آوری داده به شرح زیر وجود دارد:

- **جمع‌آوری داده‌ها به صورت زنده**[1]: این روش جمع‌آوری داده زمانی اتفاق می‌افتد که سامانه مورد تجزیه و تحلیل به طور همزمان در حال سرویس‌دهی نیز می‌باشد.
- **جمع‌آوری داده‌ها به صورت غیرزنده**[2]: روش جمع‌آوری داده به صورت غیرزنده، شامل نسخه‌برداری داده‌ها از سیستم مورد بررسی است.
- **جمع‌آوری داده‌ها به صورت ترکیبی**[3]: روش جمع‌آوری داده‌ها به صورت ترکیبی با بهره‌گیری از ویژگی‌های کلیدی هر دو روش قبل، ترکیبی از این دو روش را برای جمع‌آوری داده در پیش می‌گیرد.

لازم به ذکر است که صرف‌نظر از روش مورد استفاده می‌بایست این اطمینان حاصل شود که شواهد دیجیتالی حفظ و نگهداری می‌شوند و داده‌ها به صورت ناخواسته تغییر نمی‌کنند یا از بین نمی‌روند.

۳ . **جمع‌آوری مصنوعات ناپایدار[4] و پایدار:** مصنوعات و اطلاعات مختلف را می‌توان از پایگاه‌داده، سیستم‌عامل، سرورهای وب یا فایل‌های رویدادنگاری استخراج کرد. لازم به ذکر است که جمع‌آوری مصنوعات و شواهد در سامانه می‌تواند سبب تغییر در پایگاه‌داده شود. از این‌رو، پیش از استخراج اطلاعات از پایگاه‌داده باید نسبت به این موضوع و پایدار یا ناپایدار بودن اطلاعات، آگاهی پیدا کرد.

---

1 Live acquisition
1 Dead acquisition
1 Hybrid acquisition
1 Volatile artifacts

1
2
3
4

هر پایگاه‌داده شواهد مربوط به اعمال مختلف را در فایل‌های رویدادنگاری مختلفی پایگاه‌داده ذخیره می‌کند. این بدین معناست که برای تجزیه و تحلیل جرم‌شناسی می‌بایست نسبت به چگونگی عملکرد پایگاه‌داده، محل فایل‌ها و مصنوعات مختلف اطلاع داشت. مصنوعات در سطح پایگاه‌داده به دو دسته تقسیم می‌شوند: ۱): داده‌های فرار و ۲): داده‌های غیر فرار که در ادامه به تشریح هر یک از آنها می‌پردازیم.

- **داده‌های فرار:** به برخی از داده‌ها که به منظور افزایش سرعت، قابلیت اطمینان و بهره‌وری پایگاه‌داده درحافظه‌های ناپایدار ذخیره می‌شوند، داده‌های فرار اطلاق می‌شود. به عنوان نمونه، پایگاه‌داده اوراکل اطلاعات زیادی را در SGA[۱۵] به منظور افزایش کارایی ذخیره می‌کند. لازم به ذکر است که به طور معمول این دسته از داده‌ها دائما و با سرعت بالایی در حال تغییر هستند.
- **داده‌های غیرفرار:** به رکوردهای ذخیره شده در پایگاه‌داده، داده‌های غیرفرار یا پایدار اطلاق می‌شود.

ذکر این نکته ضروری است که فرآیند جرم‌شناسی در سامانه پایگاه‌داده نباید تنها بر روی پایگاه‌داده متمرکز باشد. پایگاه‌داده در یک محیط مجزا در حال اجرا و سرویس‌دهی نیست و متکی به زیرساخت مهمی چون سیستم عامل است. بنابراین باید سایر مصنوعات و اطلاعات جانبی از سیستم عامل‌ها و رویدادهای ثبت شده در سرور را نیز جمع‌آوری و آنها را بررسی نمود.

۴. **حفاظت[۶] و احراز اصالت[۷] داده‌های جمع‌آوری شده:** هدف از این مرحله آن است که مقدار شواهدی که مجدداً بر روی آن‌ها اطلاعاتی نوشته می‌شود، کاهش پیدا کند. مراقبت‌های شدیدی برای تضمین عدم تغییر غیرمنتظره داده‌ها باید انجام شود. اگر چه داده‌ها را می‌توان با ارسال پرسمان به پایگاه‌داده‌ی تغییریافته به دست آورد، باید از اجرای هر نوع پرسمانی که باعث حذف اطلاعات از پایگاه‌داده شود، اجتناب نمود. علاوه بر این، در صورت وجود پایگاه‌داده آسیب‌دیده یا مورد سوء استفاده قرار گرفته، صرف‌نظر از روش بدست آوردن داده، هیچ عبارت SQL ای نباید اجرا شود؛ زیرا باعث تغییر داده‌های ذخیره‌شده در حافظه و صفحات داده مربوط به پایگاه‌داده می‌شود. این کار همچنین باعث تقسیم شدن صفحه داده داخلی و محل ذخیره‌سازی داده‌های جدید در

[۱۵] قسمتی از حافظه‌ی سیستم که میان تمامی پردازه‌های مربوط به یک نمونه‌ی واحد از پایگاه‌داده‌ی اوراکل مشترک است.

[1] Preservation
[1] Authentication
[6]
[7]

حافظه پنهان[8]می‌شود و فرآیند بررسی را پیچیده‌تر می‌کند. بنابراین باید پیش از فرآیند جمع‌آوری، نسخه پشتیبان از داده‌ها و فایل‌های مهم تهیه گردد. لازم به ذکر است که روش‌ها و ابزارهای مورد استفاده برای جمع‌آوری اطلاعات و شواهد می‌بایست تا حد امکان قابل اطمینان باشند.

۵ . **تجزیه و تحلیل شواهد و تعیین فعالیت‌های مهاجم:** تجزیه و تحلیل داده‌های جمع‌آوری شده به نوع داده‌ها، سامانه پایگاه‌داده و رویداد خاصی که قرار است مورد بررسی قرار گیرد، بستگی دارد. مرحله‌ی تجزیه و تحلیل می‌بایست ابعاد مربوط به هر رویداد و محلی که اطلاعات مربوطه یافت می‌شود را در نظر بگیرد. بنابراین در مرحله‌ی تجزیه و تحلیل، اطلاعاتی همچون شخص مجرم، زمان ارتکاب جرم، داده هدف، دلایل ارتکاب و نحوه اجرای جرم تعیین می‌گردد. تجمیع داده‌ها در فرآیند جرم‌شناسی پایگاه‌داده از اهمیت بسزایی برخوردار است. برخی از داده‌ها به تنهایی دارای ارزش کمی هستند اما هنگامی که با سایر اطلاعات ترکیب می‌شوند، می‌توانند نقض امنیتی را آشکارتر سازند.

۶ . **بازسازی پایگاه‌داده:** در بررسی پایگاه‌داده آسیب‌دیده یا مورد سوءاستفاده، داده‌هایی که قبلاً حذف شده‌اند، باید بازیابی و اقدامات انجام شده توسط مهاجم شناسایی شوند.

۷ . **ارائه مستندات:** در مرحله آخر، کلیه‌ی بررسی‌های صورت گرفته در یک قالب استاندارد مستند و به مدیر سامانه و دادگاه ارائه می‌شود. لازم به ذکر است که مستندات تهیه شده برای سایر بررسی‌کنندگان که سناریوی مشابه‌ای را تجربه می‌کنند و همچنین برای حفاظت از پیگردهای قانونی بررسی‌کنندگان در آینده مفید خواهد بود.

۱-۵ جمع‌بندی

در این فصل به طور مشروح به معرفی مفاهیم جرم‌شناسی پایگاه‌داده و همچنین بررسی چالش‌ها، اهداف و گام‌های اجرایی در فرآیند جرم‌شناسی پایگاه‌داده پرداخته شد. در این راستا و در ابتدا، وجود تنوع‌های گسترده از سامانه‌ها، سطوح مختلف داخلی، مفهومی و خارجی پایگاه‌داده و نحوه تشخیص وقوع جرم به عنوان مهم‌ترین چالش مطرح در حوزه جرم‌شناسی پایگاه‌داده بررسی گردید. در ادامه، برخی از مهم‌ترین اهداف جرم‌شناسی پایگاه‌داده، تحت عناوینی چون شناسایی و اثبات وقوع جرم، کشف رویداد غیرمجاز و زمان وقوع آن، تعیین محدوده نفوذ و بازگرداندن وضعیت سامانه به وضعیت قبل از وقوع جرم معرفی گردید. در نهایت نیز، گام‌های اجرایی فرآیند جرم‌شناسی پایگاه‌داده از مرحله شناسایی جرم تا ارائه مستندات مربوط به شواهد وقوع جرم، تشریح گردید.

---

[1] Cache [8]

۲ شناسایی و مدیریت جرم

پژوهش‌های انجام شده در زمینه‌ی جرم‌شناسی دیجیتال منجر به توسعه‌ی روش‌ها و مدل‌های فرآیندی مختلفی شده است. بسیاری از این روش‌ها به سبب ویژگی‌های خاص هر یک از سامانه‌های پایگاه‌داده، به طور کامل قابل انطباق با جرم‌شناسی پایگاه‌داده نبوده و می‌بایست در آن‌ها تغییراتی صورت پذیرد. با استفاده از مدل‌های فرآیند جرم‌شناسی پایگاه‌داده می‌توان به صورت ساختاریافته، عملیات مربوط به جرم‌شناسی پایگاه‌داده را پیش برد. در این فصل، ابتدا تمهیدات مورد نیاز برای فراهم کردن بستر مناسب برای جرم‌شناسی در پایگاه‌داده مورد مطالعه قرار می‌گیرد. در ادامه و در بخش دوم نیز به شرح و بررسی فرآیند جرم‌شناسی در سامانه پایگاه‌داده می‌پردازیم. در بخش پایانی نیز ملاحظات و رهنمون‌های مورد نیاز برای رهگیری و انجام مراحل موجود در فرآیند جرم‌شناسی تشریح می‌گردد.

۲-۱ تمهیدات جرم‌شناسی

در این بخش، تمهیدات لازم برای جرم‌شناسی پایگاه‌داده را در طیف وسیعی از نیازمندی‌ها از پیش از وقوع جرم تا نیازمندی‌های نهایی مربوط به ارائه مستندات و اثبات وقوع جرم، مورد بحث و بررسی قرار می‌دهیم. لازم به ذکر است که داشتن طرح و برنامه دقیق، مهمترین گام در فرآیند جرم‌شناسی است. در یک دسته‌بندی کلی می‌توان تمهیدات مورد نیاز برای فرآیند جرم‌شناسی پایگاه‌داده را در موارد زیر خلاصه کرد [۱۵-۱۷]:

۱. **تعیین مدیر فرآیند:** این تمهید پیش از وقوع جرم انجام می‌شود. پیش از آنکه جرمی رخ دهد، شخصی باید به عنوان هماهنگ‌کننده تعیین شود. این شخص باید فرآیند تجزیه و تحلیل جرم‌شناسی را رهبری کند و از مستند تهیه شده از فرآیند به عنوان یک چک لیست[۱] برای اطمینان از اجرای گام به گام فرآیند استفاده نماید. علاوه بر این می‌بایست تیمی تحت رهبری هماهنگ‌کننده نیز وجود داشته باشد که از مهارت‌های امنیتی و مدیریتی در حوزه پایگاه‌داده برخوردار باشند. همچنین وجود یک مجموعه از ابزارها برای استفاده در فرآیند تجزیه و تحلیل بسیار مهم است. هریک از ابزارها و هر آنچه که هر ابزار انجام می‌دهد، می‌بایست مستند شود. بزرگترین چالش در فرآیند تجزیه و تحلیل، عدم وجود ابزارهای استاندارد رایگان به منظور تجزیه و تحلیل فایل‌های ردیابی[۲] و رویدادهای تولید شده توسط پایگاه‌داده است. در صورتی که در طول فرآیند جرم‌شناسی از ابزاری استفاده شود، باید بتوان ثابت کرد که ابزارهای مورد استفاده، شواهد جمع‌آوری

[1] Checklist
[2] Trace files

شده را تغییر نمی‌دهند و حذف نمی‌کنند. در صورتی که ابزار توسط خود افراد ایجاد شده باشد، اثبات عدم تغییر در شواهد، می‌تواند دشوار باشد. بنابراین بهتر است از ابزارهای تجاری از پیش تعیین شده‌ای که در دادگاه‌ها قابل قبول و استناد هستند، استفاده شود. لازم به ذکر است که ابزارهای مورد استفاده نیز می‌بایست از حیث امنیتی قابل اعتماد باشند و اجازه ندهند مهاجم از طریق آن‌ها به شواهد موجود خدشه‌ای وارد کند.

۲. **تعیین مولفه مرکزی برای اطلاع‌رسانی:** تعیین مولفه‌ی مرکزی برای گزارش جرم‌های حادث شده به عنوان یک تمهید پیش از وقوع جرم، از اهمیت بالایی برخوردار است. وجود این مولفه سبب می‌شود تا ذینفعان به سرعت از وقوع نقض امنیتی مطلع و بررسی آن را در دستور کار خود قرار دهند.

۳. **تشخیص وقوع نقض امنیتی:** به محض تشخیص نقض امنیتی در سامانه می‌بایست اطلاعات کافی برای مولفه مرکزی ارسال گردد. مولفه مرکزی نیز اطلاع‌رسانی‌های لازم را در این رابطه برای سایر ذینفعان ارسال می‌کند.

۴. **انتقال کنترل فرآیند به مدیر هماهنگ‌کننده:** به محض تشخیص وقوع نقض امنیتی در سامانه و اعلام آن توسط مولفه مرکزی، کنترل فرآیند به مدیر تجزیه و تحلیل جرم‌شناسی منتقل می‌شود تا این اطمینان حاصل شود که فرآیند به درستی و با دقت توسط تیم دنبال می‌شود.

۵. **پرهیز از قطع اتصال شبکه و خاموش کردن سامانه:** در این گام، به هیچ وجه نباید سامانه پایگاه‌داده خاموش یا اتصال آن به شبکه قطع گردد. توجه به این نکته حائز اهمیت است که داده‌ها و شواهد ناپایدار با خاموش شدن سیستم از بین می‌روند. اینکه مهاجم از ادامه‌ی فعالیت‌های مخرب باز بماند، ایده‌ی خوبی است ولی از دست دادن شواهد ناپایدار، بررسی‌های آتی را ممکن است با مشکل روبرو کند.

۶. **بررسی واقعی بودن حمله:** در این مرحله می‌بایست وجود نقض امنیتی و حمله به سامانه بررسی گردد. لازم به ذکر است که بررسی در دسترس بودن داده‌های حساس به طور ویژه برای مهاجم و در حالت کلی در بستر عمومی وب از اهمیت بالایی در این مرحله برخوردار است.

۷. **جمع‌آوری داده‌های فرار:** در این گام، رسیدگی به نقض امنیتی را آغاز نموده و داده‌های فرار را از روی سرور پایگاه‌داده جمع‌آوری می‌نماییم. از جمله داده‌های فرار می‌توان به اطلاعات مربوط به کاربران وارد شده به سامانه، پردازه‌های در حال اجرا، پورت‌های و فایل‌های باز، اشاره کرد. همچنین تمامی فایل‌های رویدادنگاری پایگاه‌داده، فایل‌های ردیابی و فایل‌های پیکربندی نیز باید جمع‌آوری و از آن‌ها به درستی نسخه‌برداری شود. علاوه بر این، از رویدادهای ثبت شده توسط سرور وب و برنامه‌ی کاربردی و سایر فایل‌های رویدادنگاری مهم نیز باید نسخه‌ای تهیه شود. اطلاعات موجود در حافظه نیز باید تخلیه و ثبت شوند. به عنوان نمونه، در پایگاه‌داده اوراکل اطلاعات موجود در SGA باید تخلیه و ثبت شوند. می‌توان آخرین پرسمان SQL اجرا شده را از SGA به دست آورد. در

صورتی که بررسی جرم‌شناسی خیلی سریع انجام شود، شانس این وجود دارد که عبارت‌های SQL که قسمتی از حمله بوده‌اند، در SGA وجود داشته باشند. همچنین می‌توان نشست‌ها و پردازه‌های موجود در SGA را نیز استخراج و جمع‌آوری کرد. تقریباً هرکاری که در پایگاه‌داده انجام می‌شود، آن را تغییر می‌دهد. بنابراین استخراج شواهد از پایگاه‌داده آسیب‌دیده می‌بایست با دقت و با ترتیب درستی انجام شود. لازم به ذکر است که داده‌هایی که بیشتر در معرض تغییر قرار دارند، باید سریعتر استخراج شوند.

۸. **جمع‌آوری داده‌های غیرفرار:** پس از جمع‌آوری داده‌های فرار که حساسیت بیشتری برای جمع‌آوری اطلاعات نسبت به سایر اطلاعات را دارند، سایر شواهد را نیز از پایگاه‌داده جمع‌آوری می‌کنیم. از جمله این شواهد می‌توان به لیست کاربران، مجوزهای کاربران و عضویت در نقش‌ها اشاره کرد.

۹. **قطع اتصال پایگاه‌داده به شبکه:** پس از جمع‌آوری اطلاعات و شواهد مورد نیاز برای بررسی جرم، به منظور کاهش مخاطرات احتمالی ناشی از آن می‌بایست اتصال سامانه به شبکه را قطع نمود.

۱۰. **تهیه نسخه پشتیبان:** در صورت امکان از کل دیسک سخت افزاری و یا در صورت وجود محدودیت، از شواهد موجود در سرور پایگاه‌داده، نسخه پشتیبان تهیه شود.

۱۱. **انجام تجزیه و تحلیل بر روی داده‌ها:** در این گام، بر روی داده‌ها و شواهد جمع‌آوری شده تجزیه و تحلیل صورت می‌گیرد و سعی می‌شود تا حد امکان زمان شروع و خاتمه‌ی حمله به دست آید. لازم به ذکر است که زمان‌های به دست آمده ممکن است با بررسی‌های بیشتر، تغییر نمایند.

۱۲. **ایجاد جدول زمانی[1] از رویدادها:** جدول زمانی، تمامی اطلاعات مربوط به اعمال انجام شده بر روی پایگاه‌داده را در خود جای داده است. بدین ترتیب می‌توان پی برد که:

- مهاجم چگونه به سیستم دسترسی پیدا کرده است،
- از طریق چه نام کاربری وارد شده است،
- چه اطلاعاتی را مشاهده نموده است،
- چه اعمالی را در پایگاه‌داده انجام داده است،
- با چه مجوزهایی به سیستم وارد شده است،
- و چه کارهای بیشتری را با مهارت بیشتر می‌توانست در سامانه اعمال کند.

۱۳. **خاموش کردن سیستم و بازگرداندن[2] آن به وضعیت پیش از حمله:** در این گام می‌بایست با توجه به میزان اهمیت پایگاه‌داده، برای خاموش کردن آن تصمیم‌گیری شود. پیش از خاموش کردن

[1] Timeline [2]

یا بازگرداندن پایگاه‌داده، باید کاملاً مشخص شود که مهاجم چه کارهایی را انجام داده است. در صورتی که داده‌ها تنها توسط مهاجم خوانده شده باشند و هیچ تغییری در آن‌ها اعمال نشده باشد، نیازی به بازگرداندن پایگاه‌داده به نقطه‌ای پیش از حمله نیست.

۱۴. **تهیه مستند از حمله:** مستندسازی فرآیند و کلیه‌ی شواهد جمع‌آوری شده، بسیار مهم است. همچنین باید یافته‌ها و آنچه با بررسی به دست آمده‌است نیز مستند شود. تمامی اعمال مهاجم به همراه نشانه‌هایی از سرقت داده‌ها باید ثبت شوند. ثبت اطلاعاتی همچون سیستم عامل سِرور و نسخه‌ی آن، نوع و نسخه‌ی پایگاه‌داده، رویداد غیرمجاز، نوع رویداد (خصمانه/ناخواسته)، شیوه ممیزی، منابع رویدادنگاری، شیوه یا ابزار تحلیل و امکان ترمیم در یک قالب استاندارد ضروری است. این مستندات برای شناسایی نقاط ضعف سامانه از اهمیت ویژه‌ای برخوردار است. بنابراین شناسایی نقاط ضعف سامانه و پیشنهادهایی برای حل آن‌ها نیز در این مستند ثبت می‌شوند.

۱۵. **گزارش رویدادها و فرآیند طی شده:** پس از تهیه‌ی مستند از رویدادها و فرآیند طی شده، مجموعه مستند گردآوری شده قابل ارائه به دادگاه یا سایر مقامات قانونی است.

۲-۲ فرآیند جرم‌شناسی

با استفاده از مدل‌های فرآیند جرم‌شناسی پایگاه‌داده می‌توان به صورت ساختاریافته‌ای عملیات مربوط به جرم‌شناسی پایگاه‌داده را پیش برد. در ابتدا و پیش از انجام هر عملی، باید نسبت به وقوع رویداد غیرمجاز اطمینان حاصل کرد و آن رویداد را شناسایی نمود. به عنوان مثال، در صورتی که مدیر پایگاه‌داده نسبت به حذف برخی از نام‌های کاربری از پایگاه‌داده مطلع شود، فرآیند جرم‌شناسی برای یافتن اطلاعاتی پیرامون این رویداد و ترمیم داده‌های حذف شده، آغاز می‌گردد.

پس از شناسایی وقوع رویداد غیرمجاز می‌بایست شواهد و اطلاعات پیرامون رویداد، جمع‌آوری گردد. با توجه به اینکه پایگاه‌داده در یک محیط مجزا نبوده و با سیستم عامل و برنامه‌های کاربردی در ارتباط است، می‌توان شواهد را از منابع مختلف به دست آورد. در جمع‌آوری اطلاعات و شواهد توجه به این نکته حائز اهمیت است که برخی از اطلاعات همچون داده‌های موجود در حافظه، فرار بوده و هر لحظه احتمال از دست رفتن آن‌ها وجود دارد؛ بنابراین، اینگونه از اطلاعات باید هرچه سریعتر و پیش از فرا رسیدن موعد حذف، جمع‌آوری شوند. پس از آن، می‌توان اطلاعات پایدار و غیرفرار را استخراج و در مکانی امن نگهداری کرد.

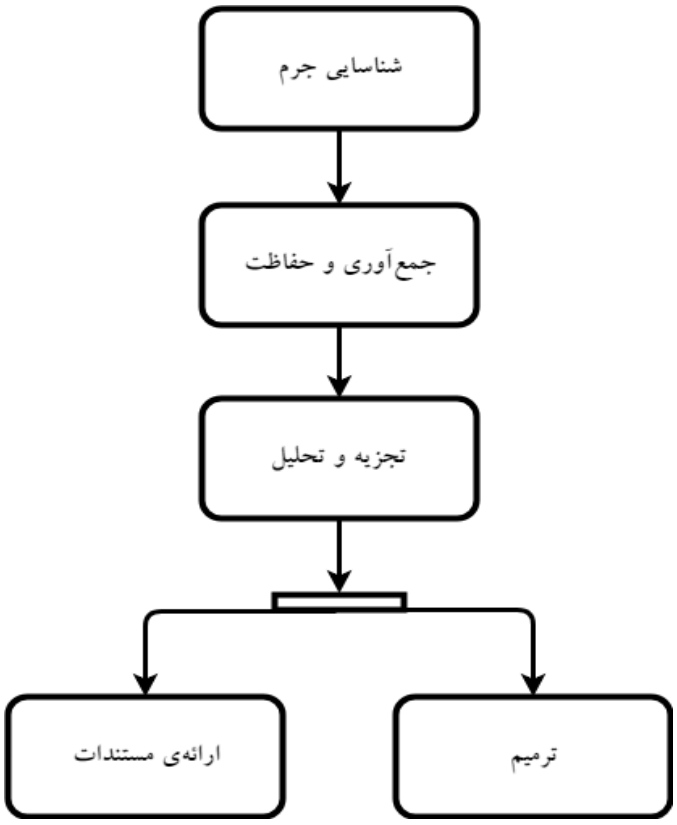
---

[2] Restore [2]

اطلاعات و شواهد جمع‌آوری شده در صورتی برای تجزیه و تحلیل و همچنین ارائه در دادگاه‌های قانونی قابل قبول هستند که به درستی حفاظت شده باشند و بتوان ثابت کرد که در حین فرآیند جرم‌شناسی تغییری در آنها صورت نگرفته است.

در ادامه و در مرحله‌ی تجزیه‌وتحلیل، با بررسی و تحلیل داده‌های جمع‌آوری شده، اطلاعاتی همچون چه‌کسی تغییر را ایجاد کرده، چه‌زمانی تغییر رخ داده، چه داده‌ای تغییر داده شده، چرا و چگونه تغییر رخ داده است، مشخص می‌شود. لازم به ذکر است که تجمیع داده‌ها در فرآیند جرم‌شناسی پایگاه‌داده از اهمیت بالایی برخوردار است. برخی از داده‌ها به تنهایی دارای ارزش کمی هستند ولی هنگامی که با سایر اطلاعات ترکیب می‌شوند، می‌توانند نقض امنیتی را آشکار سازند. تمامی شواهد جمع‌آوری شده و فرآیند طی شده در جرم‌شناسی می‌بایست مستند شود. مستندات و گزارش‌های تهیه شده قابل ارائه به مدیریت سازمان و دادگاه در صورت نیاز خواهند بود.

به سبب گستردگی حوزه جرم‌شناسی پایگاه‌داده، تمرکز ما در فرآیند جرم‌شناسی پایگاه‌داده تنها بر روی اطلاعات ثبت‌شده در پایگاه‌های داده می‌باشد و استفاده از اطلاعات ثبت‌شده بر روی سیستم عامل، داده‌های موجود در حافظه و شبکه نادیده گرفته شده است. لازم به ذکر است که برای داشتن طرحی جامع در این زمینه، در نظر گرفتن کلیه مولفه‌های درگیر ضروری است. شکل ۱، گام‌های فرآیند جرم‌شناسی پایگاه‌داده در رویکرد اتخاذی را به تصویر کشیده است. در ادامه، هر یک از این گام‌ها مورد بحث و بررسی قرار گرفته است.



شکل ۱: فرآیند پیشنهادی جرم‌شناسی پایگاه‌داده منطبق با فرآیند استاندارد

شناسایی جرم:

فرآیند جرم‌شناسی با مشاهده رویدادهای غیرمجاز در سامانه آغاز می‌شود. منظور از رویدادهای غیرمجاز، رویدادهای خصمانه یا ناخواسته‌ای هستند که مدیر پایگاه‌داده انتظار وقوع آن‌ها را در شرایط فعلی ندارد. به عنوان مثال، در صورتی که مدیر پایگاه‌داده کاربرانی را در جدولی از پایگاه‌داده تعریف کرده باشد، درج کاربر جدیدی که توسط مدیر انجام نشده است، یک رویداد غیرمجاز به شمار می‌آید.

برخی از رویدادهای قابل بررسی در فرآیند جرم‌شناسی در سامانه مدیریت پایگاه‌داده عبارتند از:

- **درج، حذف و مشاهده‌ی غیرمجاز محتوای جداول:** رویدادهای غیرمجازی هستند که به صورت خصمانه یا ناخواسته توسط کاربر/برنامه‌کاربردی بر روی پایگاه‌داده اعمال می‌شوند و موجب درج سطر (های) جدید، حذف سطر (های) موجود یا مشاهده تمامی یا بخشی از جدول (ها) در پایگاه‌داده می‌گردند. سه رویداد فوق، از جمله دستورات دستکاری داده (DML)[۳]در پایگاه‌داده به حساب می‌آیند.
- **بروزرسانی غیرمجاز داده‌های جداول**: رویداد غیرمجازی است که به صورت خصمانه یا ناخواسته توسط کاربر/برنامه‌کاربردی بر روی پایگاه‌داده اعمال می‌شود و سبب تغییر سطر (های) موجود می‌گردد. این رویداد نیز از جمله دستورات DML در پایگاه‌داده است. لازم به ذکر است که به سبب اهمیت بروزرسانی داده‌ها و چالش‌هایی که برای شناسایی وقوع جرم وجود دارد، این رویداد در برخی از سمپادها از دیگر رویدادهای مربوط به دستورات دستکاری داده متمایز شده است.
- **تغییر غیرمجاز شمای پایگاه‌داده:** رویداد غیرمجازی است که به صورت خصمانه یا ناخواسته توسط کاربر/برنامه‌کاربردی بر روی پایگاه‌داده اعمال می‌شود و سبب تغییر شمای جدول (ها) در پایگاه‌داده می‌گردد. این رویداد توسط دستورات تعریف داده (DDL)[۴]در پایگاه‌داده قابل اعمال است.
- **تلاش برای ورود غیرمجاز به پایگاه‌داده:** رویداد غیرمجازی است که به صورت خصمانه یا ناخواسته توسط کاربر/برنامه‌کاربردی بر روی پایگاه‌داده اعمال می‌شود و به طور معمول دو هدف زیر را دنبال می‌کند:
  - حدس نام کاربری و کلمه عبور کاربران مجاز جهت ورود به سامانه.

---

[2] Data Manipulation Language [3]
[2] Data Definition Language [4]

- تلاش برای شناسایی شناسه یکتای سامانه[۵](SID) و نهایتا دسترسی به حساب‌های کاربری و اعمال نفوذ در سامانه.

لازم به ذکر است که در برخی سمپادها، این تلاش تنها از طریق حدس نام‌کاربری و کلمه عبور کاربران مجاز جهت ورود به سمپاد امکان‌پذیر است.

- **تغییر غیرمجاز در فایل‌های رویدادنگاری و ممیزی:** رویداد غیرمجازی است که به صورت خصمانه یا ناخواسته توسط کاربر/برنامه‌کاربردی بر روی پایگاه‌داده اعمال می‌شود و هدف آن از بین بردن فایل‌های رویدانگاری به منظور پاک کردن شواهد رویدادهای مجرمانه در پایگاه‌داده است.

جمع‌آوری اطلاعات و شواهد:

مصنوعات و اطلاعات مختلف برای شناسایی رویدادها را می‌توان از پایگاه‌داده، سیستم عامل، سرورهای وب یا فایل‌های رویدادنگاری استخراج کرد. جمع‌آوری مصنوعات و شواهد می‌تواند سبب تغییر در پایگاه‌داده شود. از این‌رو، پیش از استخراج اطلاعات از داخل یا خارج پایگاه‌داده می‌بایست نسبت به پایدار یا ناپایدار بودن اطلاعات آگاهی پیدا کرد. هر پایگاه‌داده شواهد مربوط به اعمال مختلف را در بخش‌ها و فایل‌های رویدادنگاری مختلف ذخیره می‌کند. این بدان معناست که به منظور تجزیه و تحلیل جرم‌شناسی می‌بایست نسبت به چگونگی عملکرد پایگاه‌داده، محل فایل‌ها و مصنوعات مختلف اطلاع داشت. از آنجا که تمرکز ما در فرآیند جرم‌شناسی پایگاه‌داده تنها بر روی اطلاعات حاصل از رویدادنگاری و ممیزی در پایگاه‌داده است، لذا اطلاعات مورد هدف برای گردآوری تنها به پایگاه‌داده محدود شده و شامل اطلاعات موجود در سیستم عامل و حافظه‌ی اصلی نمی‌باشد.

درصورتی که بر روی فایل‌های رویدادنگاری و ممیزی به صورت دوره‌ای اطلاعاتی ثبت شود، به هنگام شناسایی رویداد غیرمجاز می‌بایست ابتدا سرور پایگاه‌داده متوقف شود تا از اجرای پرسمان‌های جدید بر روی سرور خودداری گردد. بدین ترتیب از نگاشته شدن بر روی اطلاعات ثبت شده در فایل‌های رویدادنگاری و ممیزی جلوگیری می‌شود. در نهایت نیز به منظور انجام تحلیل‌های آتی می‌توان از فایل‌های مورد نیاز، یک نسخه‌ی پشتیبان تهیه نمود [۱۲].

مرحله جمع‌آوری اطلاعات و شواهد، شامل دو گام زیر می‌باشد:

۱. **تعیین نحوه‌ی جمع‌آوری اطلاعات:** در صورت مشاهده علائمی از وقوع رویداد غیرمجاز در پایگاه‌داده، با توجه به نوع رویداد می‌توان از منابع مختلفی برای جمع‌آوری اطلاعات استفاده کرد. به

---

[2] System Identifier [5]

عنوان نمونه، برای جمع‌آوری اطلاعات مربوط به برخی از رویدادها می‌بایست از رویدادهای ثبت‌شده در فایل‌های رویدادنگاری و برای برخی دیگر می‌بایست از تعریف خط‌مشی‌هایی ممیزی استفاده کرد.

۲. **بررسی تنظیمات فعلی:** پس از آنکه منابع جمع‌آوری اطلاعات برای یک رویداد ناخواسته مشخص شدند، تنظیمات فعلی سیستم بررسی می‌شود. با بررسی اولیه‌ی تنظیمات مشخص می‌شود که آیا اقدامات اولیه برای ثبت اطلاعات قبل از وقوع رویداد انجام شده‌اند یا خیر. به عنوان نمونه، بررسی می‌شود که آیا در پایگاه‌داده، خط‌مشی‌هایی برای ثبت اطلاعات ممیزی مربوط به جرم تعریف شده است یا خیر.

استخراج و تجزیه و تحلیل اطلاعات:

در این مرحله ابتدا ابزارها و پرسمان‌های هدف به منظور استخراج اطلاعات از منابع تعیین می‌شوند. یک روش محبوب در میان تحلیل‌گران جرم‌شناسی، استخراج عبارات SQL اجرا شده در پایگاه‌داده است. همچنین بی‌شک یکی از المان‌های بسیار مهم در تجزیه و تحلیل جرم‌شناسی، شناسایی هویت مجرم است. هر عملی که ثبت می‌شود را باید بتوان به یک شخص واقعی نسبت داد [۱۵]. سپس اطلاعات هدف از منابع مشخص شده استخراج و مرحله بررسی و تحلیل اطلاعات آغاز می‌گردد. در این مرحله با بررسی و تحلیل داده‌های جمع‌آوری شده، اطلاعاتی همچون عامل وقوع رویداد، زمان وقوع، دلایل وقوع، نوع تغییر حاصل از اجرای رویداد، داده متأثر از تغییر و چگونگی اعمال رویداد غیرمجاز، آشکار می‌شود [۱۰, ۱۱]. لازم به ذکر است که تجمیع داده‌ها در فرآیند جرم‌شناسی پایگاه‌داده بسیار حائز اهمیت است. به منظور سادگی در بررسی هر یک از رویدادهای ناخواسته، بخش استخراج اطلاعات و تجزیه و تحلیل با یکدیگر ادغام شده و تحت عنوان استخراج و تجزیه و تحلیل اطلاعات بیان می‌شود.

ترمیم:

پس از محرز شدن رخداد جرم در سامانه پایگاه‌داده، متناسب با نیاز و شواهد اطلاعاتی موجود، ممکن است ترمیم پایگاه‌داده امکان‌پذیر باشد. در واقع، ترمیم سامانه پایگاه‌داده بدین معناست که وضعیت پایگاه‌داده را به وضعیتی پیش از وقوع رویداد برگردانیم. به عنوان نمونه، در صورتی که داده‌های حساس از یک جدول حذف شده باشند، این مرحله به بازیابی داده‌های حذفی می‌پردازد.

ارائه‌ی مستندات:

در گام نهایی می‌بایست کلیه‌ی بررسی‌های صورت پذیرفته را در یک قالب استاندارد از پیش تعیین شده به نحوی مستند نمود که قابل ارائه به مدیریت سازمانی یا دادگاه قانونی برای طرح دعوی باشد. مستندسازی به سایر بررسی‌کنندگانی که سناریوی مشابه‌ای را تجربه می‌کنند نیز کمک خواهد کرد. به منظور ارائه مستندات مربوط به رویدادهای غیرمجاز کشف‌شده در سامانه، فرم استاندارد زیر ارایه شده است.

جدول ۱: تهیه‌ی مستند از فرآیند جرم‌شناسی پایگاه‌داده

عنوان رویداد			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی			
منابع رویدادنگاری			
شیوه یا ابزار تحلیل			
امکان ترمیم			

۳-۲ رهنمون‌های فرآیند جرم‌شناسی

در این بخش، برخی از مهم‌ترین رهنمون‌های مربوط به مراحل مختلف فرآیند جرم‌شناسی پایگاه‌داده آورده شده است. لازم به ذکر است که ملاحظات ذکر شده در این بخش برای هر سامانه پایگاه‌داده در هر سازمانی حیاتی است. با این وجود، ممکن است متناسب با نوع سازمان و نوع سامانه پایگاه‌داده، ملاحظات دیگری نیز افزوده شود [۱۲-۱۵].

۱. در صورت بروز نقض امنیتی، کل سازمان باید از آن مطلع شده و نشست آموزشی کوتاهی در زمینه رسیدگی به آن با حضور تمامی افراد برگزار شود. در صورتی که اطلاع‌رسانی به بخش‌های مختلف سازمان و تجزیه و تحلیل جرم‌شناسی توسط بخش‌های مختلف، طبق برنامه‌ی مشخصی انجام نشود، بخش‌هایی ممکن است موضوع را نادیده بگیرند و بخش‌های دیگری ممکن است آن را به اطلاع عموم برسانند. بنابراین، هرآنچه که در روند جرم‌شناسی انجام می‌شود می‌بایست طبق برنامه از پیش تدوین شده، صورت پذیرد.

۲. یکی از چالشی‌ترین مسائل در بررسی پایگاه‌های داده آن است که هر چه بیشتر در پایگاه‌داده جستجو و بررسی انجام شود، تغییرات بیشتری در آن رخ می‌دهد. لازم به ذکر است که تقریباً هر عملی که در پایگاه‌داده انجام می‌شود، می‌تواند باعث تغییر در رکوردهای دیکشنری شود.

۳. استفاده از حساب کاربری با مجوزهای بالا می‌تواند خود زمینه ایجاد تغییرات در پایگاه‌داده را فراهم آورد. بنابراین، در نظر گرفتن حساب کاربری با دسترسی‌های محدود فقط خواندنی برای این منظور ایده‌آل است. اگر چنین حساب‌های کاربری قبل از وقوع جرم تعریف نشده باشد، ایجاد حساب کاربری جدید توصیه نمی‌شود؛ چرا که خود موجب تغییرات جدیدی در سامانه می‌شود و با وجود اینکه استفاده از کاربر SYS می‌تواند خطرناک باشد ولی منطقی‌تر است.

۴. برای قابل قبول بودن شواهد در دادگاه قانونی به هنگام طرح دعوی، دو اصل اساسی باید رعایت شود: ۱): اولین اصل آن است که می‌بایست ثابت شود که اطلاعات و شواهد گردآوری شده به صورت

قانونی به دست آمده است و ۲): دومین اصل نیز آن است که می‌بایست ثابت شود که به هنگام جمع‌آوری شواهد و اطلاعات تغییری در آنها اعمال نشده است. برای اثبات اصل اول، پیش از جمع‌آوری شواهد باید قانونی بودن آن‌ها بررسی و تأیید شود. همچنین اصل دوم نیز با مستندسازی شواهد و اعمال اجرا شده بر روی آن‌ها قابل اثبات است. در حقیقت جمع‌آوری شواهد باید طی گام‌های از پیش تدوین شده‌ای صورت گیرد تا اطمینان حاصل شود که شواهد جمع‌آوری شده، تغییر نکرده‌اند. برای اثبات صحت داده‌ها می‌توان از روش مجموع مقابله‌ای[۶] استفاده کرد. این روش می‌تواند بر روی شواهد مختلف از جمله یک فایل یا کل دیسک سخت افزاری اعمال شود. با این روش می‌توان ثابت کرد که شواهد جمع‌آوری شده همان اطلاعاتی است که در ادامه از آن‌ها برای تجزیه و تحلیل استفاده می‌شود و بدون تغییر به دادگاه قابل ارائه است. توجه به این نکته حائز اهمیت است که برای استفاده از روش مجموع مقابله‌ای نباید به راحتی از بسته‌های موجود در پایگاه‌های داده استفاده شود. به عنوان مثال، در پایگاه‌داده اوراکل، نباید از بسته‌ی DBMS_SQLHASH استفاده شود چون ممکن است خود این بسته توسط مهاجم تغییر داده شده باشد. بنابراین در طول بررسی‌های جرم‌شناسی باید بتوان اعتبار ابزارها و عدم تغییر آن‌ها را ثابت کرد. مهاجم همچنین می‌تواند دیدهای موجود در پایگاه‌داده را برای مخفی نگه داشتن اعمال خود تغییر دهد. بنابراین باید یا از جدول‌های پایه در بررسی‌های جرم‌شناسی استفاده شود و یا پیش از استفاده از دیدها از صحت آن‌ها اطمینان حاصل گردد.

۵. یکی از مصنوعات مهم پایگاه‌داده برای تجزیه و تحلیل جرم‌شناسی، رویدادهای ثبت شده توسط ممیزی است. استفاده از ممیزی این اطمینان را ایجاد می‌کند که همیشه شواهدی برای استفاده در تحلیل جرم‌شناسی وجود دارد. می‌توان پایگاه‌داده را برای ثبت رویدادهای مشخص همچون ایجاد یک کاربر، تغییر رمز عبور و دسترسی به محتوای یک جدول تنظیم کرد. در صورتی که ممیزی فعال نباشد می‌توان از سایر قابلیت‌های رویدادنگاری در پایگاه‌های داده برای ثبت تغییرات استفاده کرد. با این وجود، به کارگیری ممیزی با جزئیات کافی و ثبت تمامی اعمال اجرا شده توسط مهاجم، تجزیه و تحلیل جرم‌شناسی را ساده‌تر می‌کند. تنظیمات مربوط به ممیزی می‌بایست بر اساس برنامه‌ای از قبل تعیین شده، مشخص گردد. در واقع بر اساس این برنامه، هرآنچه که نیاز به دانستن است، مشخص می‌گردد. در ادامه، برخی از امکان‌ها برای انجام تنظیمات ممیزی ارایه شده‌اند:
   - افرادی که به پایگاه‌داده وارد یا از آن خارج می‌شوند.
   - افرادی که خود را به جای مدیر پایگاه‌داده نمایش می‌دهند.

---

[2] Checksum [6]

- تلاش برای حمله‌ی تزریق SQL[۲۷]
- تغییر در پروفایل کاربر
- تغییر در ساختار پایگاه‌داده

یک راه حل جامع برای تهیه‌ی ممیزی باید شامل ممیزی از خود ممیزی نیز باشد. در صورتی که مهاجم تلاش به حذف یک رکورد ممیزی کند، باید این عمل ثبت شود. همچنین تلاش برای تغییر تنظیمات ممیزی نیز باید ثبت گردد. در صورتی که ممیزی در پایگاه‌داده فعال باشد، اولین گام شناسایی تنظیمات ممیزی است. پس از آن می‌توان به رویدادهای ممیزی و استفاده از آن‌ها در بررسی‌های جرم‌شناسی پرداخت.

۶. شواهد و اطلاعات مربوط به جرم‌شناسی اغلب در مکان‌های مختلفی از پایگاه‌داده وجود دارند. شناخت شواهد و اطلاعات مهم و اولویت‌بندی آنها از اهمیت بالایی برخوردار است. به غیر از اطلاعاتی که می‌توان از طریق اجرای پرسمان بر روی پایگاه‌داده‌ی تغییریافته به‌دست‌آورد، می‌توان از طریق ابزارهایی که توسط سامانه پایگاه‌داده استفاده می‌شوند؛ همچون نهان‌گاه طرح اجرایی[۲۸] و رویدادنگاری تراکنش‌ها[۲۹] نیز شواهدی را جمع‌آوری کرد. یک طرح اجرایی، کارآمدترین روش اجرای درخواست‌های داده است که در نهان‌گاه طرح اجرایی برای استفاده‌ی مجدد ذخیره می‌شوند. رویدادنگاری تراکنش‌ها در شناخت پرسمان‌های اجرا شده بر روی پایگاه‌داده و برای بررسی‌ها و تحقیقات مختلف، مفید است. سایر منابع شامل فایل‌هایی هستند که تاریخچه‌ی مربوط به پایگاه‌داده را ذخیره می‌کنند. برخی از این فایل‌ها به طور اختصاصی در پایگاه‌داده کاربرد دارند، همچون فایل رویدادنگاری پایگاه‌داده و فایل‌های داده در حالی که سایر فایل‌ها همچون رویدادنگاری سرور وب و رویدادنگاری رویدادهای سیستمی یک سیستم‌عامل به طورخاص به سرور پایگاه‌داده اختصاص داده نمی‌شوند. در هنگام تصمیم‌گیری درمورد اینکه کدام داده اول جمع‌آوری شود، مهم است که سطح بی‌ثباتی یک فایل در نظر گرفته شود.

۷. یکی از بزرگترین مسائل در تجزیه و تحلیل جرم‌شناسی در پایگاه‌داده آن است که به صورت معمول پایگاه‌های داده، رویدادهای مربوط به دسترسی و خواندن محتوای جداول را ثبت نمی‌کنند ولی در تمامی مواقع، تغییرات در پایگاه‌داده همچون بروزرسانی، درج و حذف داده‌ها ثبت می‌شوند. گاهی نیاز است که شواهدی برای بررسی سرقت داده‌ها شناسایی شود. سرقت داده‌ها، ممکن است به

---

2 SQL Injection
2 Execution plan cache
2 Transaction logs
7
8
9

نحوی باشد که داده‌ها را از پایگاه‌داده حذف نکند. بنابراین می‌بایست به دنبال شواهدی برای دسترسی به داده‌ها از طریق پایگاه‌داده بود. دسترسی به این گونه شواهد معمولاً پیچیده است مگر اینکه ممیزی فعال باشد. در حقیقت تنها روشی که می‌توان از آن به طور حتم برای اثبات دسترسی به داده مشخص استفاده کرد، فعال کردن ممیزی روی آن پیش از دسترسی است. به طور طبیعی، ثبت فعالیت خواندن داده‌ها ایده‌آل است هرچند همیشه انجام نمی‌شود. بنابراین باید بتوان در کنار آن، از راه‌های جایگزین نیز استفاده کرد. بدین منظور اصولاً از همبستگی[۳۰]میان شواهد استفاده می‌شود. به عنوان مثال، در پایگاه‌داده اوراکل، ممکن است شواهدی مبنی بر ورود مهاجم و ایجاد اتصال به پایگاه‌داده وجود داشته باشد. مهاجم ممکن است پرسمان SELECT را وارد کرده باشد و بهینه‌ساز[۳۱]در اوراکل وارد عملی برای کامپایل دستور SQL شده باشد. همچنین در صورتی که حمله بلافاصله مورد بررسی قرار گرفته باشد، پرسمان SQL استفاده شده توسط مهاجم ممکن است در SGA وجود داشته باشد. در این شرایط، در صورتی که حمله از طریق سرور وب انجام شده باشد، پرسمان SQL ممکن است در فایل رویدادنگاری مربوط به برنامه‌ی کاربردی تحت وب موجود باشد. بنابراین در صورتی که ممیزی بر روی سیستم فعال نباشد، باید از همبستگی میان شواهد مختلف برای نتیجه‌گیری درمورد سرقت اطلاعات استفاده شود.

۸. برای جلوگیری از حجیم شدن فایل‌های رویدادنگاری و ممیزی می‌توان خط‌مشی‌هایی را برای بازنویسی[۳۲]رویدادها در این گونه از فایل‌ها اعمال کرد. به عنوان مثال می‌توان مشخص کرد که در صورتی که حجم فایل‌های رویدادنگاری به ۱۰۰ مگابایت رسید، اطلاعات جدید از ابتدای فایل بر روی اطلاعات قبلی نوشته شود یا در یک دوره هفت روزه، فایل‌های رویدادنگاری جدید ایجاد شوند. درصورتی که بر روی فایل‌های رویدادنگاری و ممیزی به صورت دوره‌ای اطلاعات ثبت شود، هنگام تشخیص رویداد غیرمجاز، باید ابتدا سرور پایگاه‌داده متوقف شود تا از اجرای اعمال و پرسمان‌های جدید بر روی سرور خودداری شود. سپس می‌توان از فایل‌های مورد نیاز، نسخه‌ی پشتیبان تهیه کرد تا تحلیل‌های آتی بر روی نسخه‌های پشتیبان انجام شود.

۹. در صورتی که خط‌مشی‌هایی برای تهیه‌ی پشتیبان از پایگاه‌داده به صورت دوره‌ای وجود داشته باشد، وضعیت مطلوبی که پایگاه‌داده پیش از این، در آن قرار داشته است در دسترس خواهد بود. در

---

[30] Correlation
[31] Optimizer
[32] Rotate

نتیجه، در صورت وقوع رویداد غیرمجاز با بازگرداندن فایل‌های پشتیبان می‌توان به حالت مطلوب پیش از رویداد غیرمجاز تغییر وضعیت داد.

۱۰. در صورت وجود فایل‌های رویدادنگاری و ممیزی بر روی سیستمی که در آن پایگاه‌داده وجود دارد، مدیر پایگاه‌داده یا کاربر مخرب می‌تواند به طور موقت تهیه‌ی ممیزی و رویدادنگاری را متوقف کرده و اعمال مورد نظر خود را اجرا کند و یا تغییراتی را به صورت دستی در فایل‌های ممیزی و رویدادنگاری ایجاد کند. در صورتی که چندین نسخه از داده‌های حساس در مکان‌های مختلف وجود داشته باشد، امکان از دست رفتن داده‌های حساس در صورت حذف آن‌ها از یک مکان، کاهش می‌یابد. همچنین مطلوب است که تمامی رویدادها به سیستم مرکزی ارسال شده و از آن‌ها به طور مرکزی حفاظت و نگهداری شود.

۱۱. معمولاً مهاجمین پس از حمله به پایگاه‌داده، سعی به حذف ردپای خود در فایل‌های رویدادنگاری و ممیزی می‌کنند. بنابراین محدود کردن دسترسی به این فایل‌ها و کپی‌برداری از آن‌ها و ذخیره‌سازی در سرور مرکزی از اهمیت زیادی برخوردار است. همچنین با تشخیص اعمال تغییر غیرمجاز در فایل‌های رویدادنگاری و ممیزی می‌توان متوجه شد که رویدادهای ثبت شده در این فایل‌ها فاقد اعتبار هستند. به عنوان نمونه، هنگام اجرای پرسمان بر روی فایل‌های رویدادنگاری redo log مربوط به پایگاه‌داده اوراکل، در صورتی که به صورت دستی تغییری در این فایل‌ها اعمال شده باشد، یکی از خطاهای موجود در شکل ۲ و شکل ۳ نشان داده می‌شود که نشان‌دهنده‌ی تغییر غیرمجاز در این فایل‌ها و عدم اعتبار اطلاعات ذخیره شده، است.

```
ORA-00308: cannot open archived log 'C:\Users\Desktop\oracle\oradata\orcl\REDO02.LOG'
ORA-27046: file size is not a multiple of logical block size
OSD-04012: file size mismatch (OS 209715713)
00308. 00000 -  "cannot open archived log '%s'"
*Cause:    The system cannot access a required archived redo log file.
*Action:   Check that the off line log exists, the storage device is
           online, and the archived file is in the correct location.
           Then attempt to continue recovery or restart the recovery
           session.
```

شکل ۲: خطای تغییر غیرمجاز در فایل رویدادنگاری

```
ORA-00368: checksum error in redo log block
ORA-00353: log corruption near block 132010 change 21987320 time 02/02/2018 10:11:30
ORA-00334: archived log: 'C:\USERS\DESKTOP\ORACLE\ORADATA\ORCL\REDO02.LOG'
00368. 00000 -  "checksum error in redo log block"
*Cause:    The redo block indicated by the accompanying error, is not
           vaild. It has a checksum that does not match the block contents.
*Action:   Do recovery with a good version of the log or do time based
           recovery up to the indicated time. If this happens when archiving,
           archiving of the problem log can be skipped by clearing the log
           with the UNARCHIVED option. This must be followed by a backup of
           every datafile to insure recoverability of the database.
*Action:   Restore correct file or reset logs.
```

شکل ۳: خطای تغییر غیرمجاز در فایل رویدادنگاری

۲-۴ جمع‌بندی

در این فصل، ابتدا بستر مورد نیاز برای جرم‌شناسی پایگاه‌داده را تحت عنوان تمهیدات جرم‌شناسی به طور مشروح مورد بحث و بررسی قرار دادیم. سپس فرآیند جرم‌شناسی را از شناسایی جرم تا تهیه و ارائه مستندات به مدیریت سازمان و دادگاه‌قانونی برای طرح دعوی تشریح کردیم. همچنین به منظور ارائه مستندات مربوط به رویدادهای غیرمجاز کشف‌شده در سامانه، فرم استاندارد زیر ارایه گردید.

جدول ۲: تهیه‌ی مستند از فرآیند جرم‌شناسی پایگاه‌داده

عنوان رویداد			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی			
منابع رویدادنگاری			
شیوه یا ابزار تحلیل			
امکان ترمیم			

۳ درج، حذف، بروزرسانی و مشاهده‌ی غیرمجاز محتوای جداول

در این فصل به بحث و بررسی رویدادهای غیرمجازی که به صورت خصمانه یا ناخواسته توسط کاربر/برنامه‌کاربردی بر روی پایگاه‌داده اعمال می‌شوند و موجب درج سطر (های) جدید، حذف و بروزرسانی سطر (های) موجود یا مشاهده تمامی یا بخشی از جدول (ها) در پایگاه‌داده می‌گردند، پرداخته می‌شود. رویدادهای فوق از جمله دستورات دستکاری داده (DML)[۳۳] در پایگاه‌داده به حساب می‌آیند. از آنجاییکه رویدادهای مورد بحث در این فصل از نقطه نظر جرم‌شناسی به هم نزدیک هستند، لذا تنها بر روی رویداد حذف غیرمجاز به عنوان یک نماینده از این گروه رویدادها متمرکز شده و در صورت نیاز تفاوت‌های دیگر رویدادها ذکر می‌گردد.

[3] Data Manipulation Language [3]

۳-۱ شناسایی جرم

فرآیند جرم‌شناسی با مشاهده‌ی رویدادهای غیرمجاز در سامانه آغاز می‌شود. بنابراین در صورتیکه مدیر پایگاه‌داده خود به طور مستقیم یا غیر مستقیم (از طریق کاربران)، نسبت به حذف داده‌ها از جدولی آگاهی پیدا کند که انتظار حذف آنها در شرایط فعلی را نداشته است، فرآیند جرم‌شناسی در سامانه آغاز می‌شود.

۳-۲ جمع‌آوری اطلاعات و شواهد

از آنجاییکه هر پایگاه‌داده شواهد مربوط به اعمال مختلف را در بخش‌ها و فایل‌های رویدادنگاری مختلف ذخیره می‌کند، در این بخش قصد داریم منابع مربوط به شواهد رویداد غیرمجاز حذف را تعیین نماییم. لازم به ذکر است که تمرکز ما در فرآیند جرم‌شناسی پایگاه‌داده تنها بر روی اطلاعات حاصل از رویدادنگاری و ممیزی در پایگاه‌داده است و شامل اطلاعات موجود در سیستم عامل و حافظه‌ی اصلی نمی‌باشد. از این‌رو، در ادامه به معرفی منابع مربوط به شواهد رویداد غیرمجاز حذف و نحوه آگاهی و تنظیم آنها می‌پردازیم.

استفاده از رویدادنگاری پرسمان‌های زمانبر: در پایگاه‌داده‌ی Cassandra می‌توان تنظیم کرد که پرسمان‌ها و رویدادهایی که اجرای آن‌ها کند و زمانبر است، ثبت شوند. از این قابلیت می‌توان برای رویدادنگاری از تمامی پرسمان‌های اجرایی استفاده کرد. تنظیمات مربوط به رویدادنگاری از پرسمان‌های زمانبر، در فایل dse.yaml در قسمت cql_slow_log_options وجود دارد. نمونه‌ای از این تنظیمات در ادامه آورده شده است.

```
cql_slow_log_options:
    enabled: true
#    # When t > 1, log queries taking longer than t milliseconds.
#    #      0 <= t <= 1,  log queries above t percentile
    threshold: 0
#    # Initial number of queries before percentile filter becomes active
    minimum_samples: 1
    ttl_seconds: 259200
#    # Keeps slow queries in-memory only and doesn't write data to the database.
#    # WARNING - if this is set to 'false' then set threshold >= 2000, otherwise there will be a
#    # high load on the database.
    skip_writing_to_db: false
```

مطابق تنظیمات بالا، تمامی پرسمان‌های اجرایی ثبت و به مدت ۲۵۹۲۰۰ ثانیه در پایگاه‌داده باقی می‌مانند.

فعال‌سازی رویدادنگاری پرسمان‌های زمانبر برای رویدادنگاری از تمامی پرسمان‌ها

۱	فعال‌سازی رویدادنگاری از پرسمان‌های زمان‌بر	cql_slow_log_options: enabled: true
۲	رویدادنگاری از تمامی پرسمان‌ها	threshold: 0
۳	ذخیره‌ی رویدادها به مدت ۲۵۹۲۰۰ ثانیه	ttl_seconds: 259200
۴	ذخیره‌ی رویدادها در پایگاه‌داده	skip_writing_to_db: false

استفاده از ممیزی: رویدادهای ممیزی می‌توانند در فایل‌های رویدادنگاری یا در جدولی در پایگاه‌داده‌ی Cassandra ذخیره شوند. هنگامی که ممیزی فعال شود، به صورت پیش‌فرض رویدادها در فایل‌هایی بر روی سیستم‌عامل ذخیره می‌شوند و همچنین در صورتی که اندازه رویدادهای ثبت‌شده افزایش پیدا کند، ذخیره‌ی رویدادها در جدولی در Cassandra عملکرد بهتری خواهد داشت. فایل پیکربندی dse.yaml به منظور فعال‌سازی و تنظیم ممیزی در Cassandra استفاده می‌شود. برای فعال‌سازی ممیزی، در قسمت audit_logging_options، مقدار پارامتر enabled باید true باشد. همچنین سایر تنظیمات مرتبط با ممیزی در فایل dse.yaml در ادامه توضیح داده شده‌اند:

- تنظیم گزینه‌ی logger به منظور ثبت رویدادهای ممیزی در جدولی از Cassandra (مقدار CassandraAuditWriter) یا در فایل (مقدار SLF4JAuditWriter)
- تعیین انواع رویدادهایی که باید از آن‌ها ممیزی تهیه شود، از جمله:
  - **AUTH**: ثبت رویدادهای مربوط به ورود به سیستم
  - **DML**: ثبت رویدادهای درج، به‌روزرسانی، حذف و سایر اعمال مرتبط با تغییر داده‌ها
  - **DDL**: ثبت ایجاد شیئ، تغییر و حذف آن و سایر رویدادهای مرتبط با تعریف داده‌ها
  - **DCL**: ثبت رویدادهای اعطاء و حذف نقش، ایجاد نقش و سایر رویدادهای مرتبط با نقش‌ها
  - **QUERY**: ثبت تمامی پرسمان‌ها

این موارد با پارامترهای include_categories و exclude_categories تنظیم می‌شوند. همچنین اگر مقداری برای این دو پارامتر مشخص نشود، از کلیه‌ی رویدادها ممیزی تهیه می‌شود.

- تنظیم فضای کلیدهای[3] که باید از آن‌ها ممیزی تهیه شود (پارامترهای included_keyspaces و excluded_keyspaces)
- در صورتی که رویدادها در جدولی در Cassandra ثبت می‌شوند، تنظیم زمان نگهداری رویدادهای ثبت‌شده در پارامتر retention_time انجام می‌شود. مقدار پیش‌فرض این پارامتر صفر است و بدین معناست که حفظ تمامی رویدادها به مدت نامحدود است.

[3] KeySpace [4]

```
audit_logging_options:
    enabled: true
    # The logger used for logging audit information
    # Available loggers are:
    #  CassandraAuditWriter - logs audit info to a Cassandra table. This logger can be run synchronously or asynchronously. Audit logs are stored in the dse_audit.audit_log table.
    #  When run synchronously, a query will not execute until it has been written to the audit log table successfully. If a failure occurs before an audit event is written, and it's query is executed, the audit logs might contain queries that were never executed.
    #  SLF4JAuditWriter -     logs audit info to an SLF4J logger. The logger name is `SLF4JAuditWriter`, and can be configured in the logback.xml file.
    logger: CassandraAuditWriter
#     # Comma-separated list of audit event categories to be included or excluded from the audit log.
#     # When not set, the default includes all categories.
#     # Categories are: QUERY, DML, DDL, DCL, AUTH, ADMIN, ERROR.
#     # Specify either included or excluded categories. Specifying both is an error.
    included_categories: DML
#     excluded_categories:
#     # Comma-separated list of keyspaces to be included or excluded from the audit log.
#     # When not set, the default includes all keyspaces.
#     # Specify either included or excluded keyspaces. Specifying both is an error.
#     included_keyspaces:
#     excluded_keyspaces:
#     # Comma separated list of the roles to be audited or not.
#     # Specify either included or excluded roles. Specifying both is an error
#     included_roles:
#     excluded_roles:
    # The amount of time, in hours, audit events are retained by supporting loggers.
    # Only the CassandraAuditWriter supports retention time.
    # Values of 0 or less retain events forever.
    retention_time: 0
```

در صورتی که مقدار پارامتر logger برابر SLF4JAuditWriter باشد، رویدادها در فایل ثبت شوند و می‌توان ادامه‌ی تنظیمات را در فایل پیکربندی logback.xml انجام داد. بدین منظور تنظیمات پیش‌فرضی در این

فایل قرار دارد که می‌توان آن‌ها را پذیرفت و یا تغییر داد. تنظیمات پیش‌فرض مربوط به ممیزی در فایل، در ادامه آورده شده است.

```
<!--audit log-->
<appender name="SLF4JAuditWriterAppender"
class="ch.qos.logback.core.rolling.RollingFileAppender">
  <file>${Cassandra.logdir}/audit/audit.log</file>
  <encoder>
    <pattern>%-5level [%thread] %date{ISO8601} %X{service} %F:%L - %msg%n</pattern>
    <immediateFlush>true</immediateFlush>
  </encoder>
  <rollingPolicy class="ch.qos.logback.core.rolling.FixedWindowRollingPolicy">
    <fileNamePattern>${Cassandra.logdir}/audit/audit.log.%i.zip</fileNamePattern>
    <minIndex>1</minIndex>
    <maxIndex>5</maxIndex>
  </rollingPolicy>
  <triggeringPolicy class="ch.qos.logback.core.rolling.SizeBasedTriggeringPolicy">
    <maxFileSize>200MB</maxFileSize>
  </triggeringPolicy>
</appender>

<logger name="SLF4JAuditWriter" level="INFO" additivity="false">
  <appender-ref ref="SLF4JAuditWriterAppender"/>
</logger>
```

مطابق با این تنظیمات رویدادهای ممیزی در فایل audit.log در مسیر ${Cassandra.logdir}/audit نوشته می‌شوند.

فعال‌سازی ممیزی		
audit_logging_options: enabled: true	فعال‌سازی ممیزی	۱
logger: CassandraAuditWriter logger: SLF4JAuditWriter	ثبت رویدادهای ممیزی در جدول یا فایل	۲
included_categories: DML	تعیین انواع رویدادهای ممیزی (عدم تعیین	۳

	مقدار برای این دو پارامتر = ممیزی از کلیه‌ی رویدادها)	excluded_categories:
۴	تنظیم فضای کلید (عدم تعیین مقدار برای این دو پارامتر = ممیزی از کلیه‌ی فضاهای کلید)	# included_roles: # excluded_roles:
۵	تنظیم زمان نگهداری رویدادهای ثبت‌شده (صفر یعنی حفظ تمامی رویدادها به مدت نامحدود)	retention_time: 0
۶	تنظیم محل فایل ممیزی در فایل پیکربندی logback.xml	<file>${Cassandra.logdir}/audit/audit.log</file>

۳-۳ استخراج و تجزیه و تحلیل اطلاعات

در این بخش نحوه‌ی استخراج اطلاعات برای هر یک از منابع مربوط به شواهد مورد بحث و بررسی قرار می‌گیرد. با استفاده از اطلاعات جمع‌آوری شده، تحلیل‌گر باید داده‌های حذف شده را بررسی و در صورت مشاهده‌ی رویداد حذف غیرمجاز، آن را به عنوان جرم تلقی نماید. در ادامه برای هر یک از منابع حاوی شواهد برای رویدادهای سامانه، نحوه استخراج و تجزیه و تحلیل شواهد تشریح می‌گردد.

استخراج و تحلیل اطلاعات با استفاده از رویدادنگاری از پرسمان‌های زمانبر: برای مشاهده‌ی رویدادهای ثبت شده از دستور زیر استفاده می‌شود:

```
SELECT * FROM dse_perf.node_slow_log;
```

نمونه‌ای از خروجی دستور بالا در شکل ۴ نشان داده شده است.

```
node_ip   | date                            | start_time                           | commands                                                       |
duration | parameters | source_ip | table_names             | tracing_session_id | username
----------+---------------------------------+--------------------------------------+----------------------------------------------------------------+
---------+------------+-----------+-------------------------+--------------------+----------
127.0.0.1 | 2018-10-04 00:00:00.000000+0000 | ffc95800-c795-11e8-b690-495c446d8edb |                                    ['select * from users;'] |
        5 |       null | 127.0.0.1 | {'test_key_space.users'} |               null | cassandra
127.0.0.1 | 2018-10-04 00:00:00.000000+0000 | fed31120-c795-11e8-b690-495c446d8edb |                          ['delete from users where id = 1;'] |
        3 |       null | 127.0.0.1 | {'test_key_space.users'} |               null | cassandra
```

شکل ۴: رویدادنگاری حذف داده

همان‌طور که در شکل ۴ دیده می‌شود، اطلاعاتی همچون زمان اجرای پرسمان، پرسمان اجرایی و کاربر اجراکننده‌ی آن ثبت شده‌اند.

استخراج و تحلیل اطلاعات با استفاده از رویدادنگاری از پرسمان‌های زمانبر

```
SELECT * FROM dse_perf.node_slow_log;
```

استخراج و تحلیل اطلاعات با استفاده از ممیزی : در صورتی کـه رویـدادهای ممیـزی در فایـل ذخیـره شوند، خروجی رویدادها مشابه شکل ۵ خواهد بود.

```
INFO  [CoreThread-2] 2018-07-09 13:54:52,344  SLF4JAuditWriter.java:88 - host:localhost/127.0.0.1|source:/127.0.0.1|
user:cassandra|authenticated:cassandra|timestamp:1531128292340|category:DML|type:CQL_DELETE|ks:test_key_space|cf:use
rs|operation:delete from users where id = 11;|consistency level:ONE
~
~
```

شکل ۵: ممیزی از عمل حذف داده

همچنین می‌توان تنظیمات موجود در فایل dse.yaml را به نحوی انجام داد که رویدادهای ممیزی در جدولی در پایگاه‌داده ذخیره شوند. بدین ترتیب، رویدادهای ممیزی در جدول dse_audit.audit_log نوشته می‌شوند. نمونه‌ای از رویدادهای ثبت‌شده در این جدول در شکل ۶ نشان داده شده‌اند.

```
 date                         | node      | day_partition | event_time                           | authenticated | batch_id | category |
 consistency | keyspace_name  | operation                     | source     | table_name | type       | username
-------------------------------+-----------+---------------+--------------------------------------+---------------+----------+----------+
-------------+----------------+-------------------------------+------------+------------+------------+----------
 2018-07-09 00:00:00.000000+0000 | 127.0.0.1 |         32400 | a15e4271-8358-11e8-bd65-89fd49dcb5c0 |     cassandra |     null |      DML |
       ONE | test_key_space |           use test_key_space; | /127.0.0.1 |       null |     SET_KS | cassandra
 2018-07-09 00:00:00.000000+0000 | 127.0.0.1 |         32400 | a15f53e1-8358-11e8-bd65-89fd49dcb5c0 |     cassandra |     null |      DML |
       ONE | test_key_space |          USE "test_key_space" | /127.0.0.1 |       null |     SET_KS | cassandra
 2018-07-09 00:00:00.000000+0000 | 127.0.0.1 |         32400 | a80c76f1-8358-11e8-bd65-89fd49dcb5c0 |     cassandra |     null |      DML |
       ONE | test_key_space | delete from users where id =10; | /127.0.0.1 |      users | CQL_DELETE | cassandra
```

شکل ۶: ممیزی از عمل حذف داده

همان‌طور که در شکل ۵ و شکل ۶ دیده می‌شود، با استفاده از رویدادهای ممیزی می‌توان اطلاعـاتی در مـورد زمان اجرای پرسمان، پرسمان اجرایی و کاربر اجراکننده‌ی آن به دست آورد.

استخراج و تحلیل اطلاعات ممیزی ثبت شده در جدول
SELECT * FROM dse_audit.audit_log;

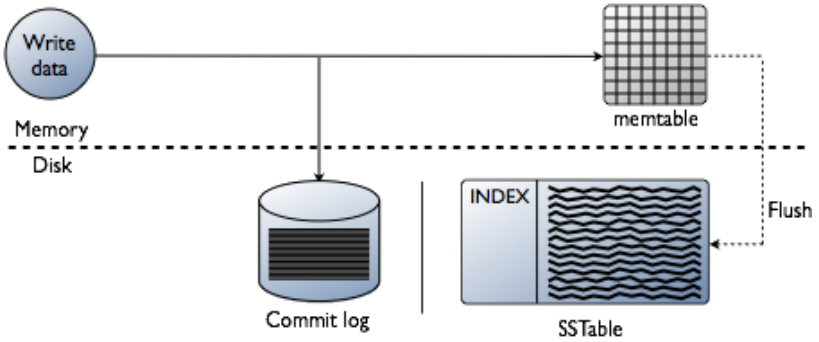
۳-۴ ترمیم

در صورتی که پیش از حذف داده‌ها اقدامات لازم جهت تهیه فایل پشتیبان صورت پذیرفته باشد با بازیابی فایل می‌توان اطلاعات از دست رفته را مجدداً در اختیار گرفت. در این بخش سعی داریم روش‌هایی به غیر از

روش‌های مربوط به تهیه‌ی نسخه‌های پشتیبان را برای بازیابی داده‌های از دست رفته مورد بحث و بررسی قرار دهیم.

یک روش برای بازیابی داده‌های از دست رفته آن است که پرسمان‌های اجرا شده بر روی یک جدول را از زمان ایجاد جدول تا زمان حذف داده از جدول بازیابی کرده و همه‌ی آن‌ها به غیر از دستور حذف را مجددا بر روی جدول اجرا نمود. بدین ترتیب داده‌های حذف شده از جدول، بازیابی می‌شوند. این روش مستلزم آن است که تمامی رویدادهای ثبت شده‌ی مربوط به جدول مورد نظر و داده‌های حذف شده، در دسترس باشند. در ادامه تلاش شده است با استفاده از جدول رشته‌های مرتب شده[3] داده‌های حذف شده بازیابی شوند.

زمانی که عمل نوشتن بر روی دیسک انجام می‌شود، Cassandra داده‌ها را در ساختاری در حافظه به نام memTable ذخیره می‌کند و در عین حال اطلاعات را در Commit log بر روی دیسک می‌نویسد (شکل ۷).



شکل ۷: نحوه‌ی ذخیره‌سازی داده بر روی دیسک

تحت شرایط خاص؛ مانند زمانی که محتوای موجود در memTable از حدآستانه‌ی تنظیم‌شده فراتر برو؛ داده‌های موجود در آن در صفی قرار می‌گیرند تا بر روی دیسک نوشته شوند. داده‌های موجود در Commit log پس از تخلیه‌ی داده‌های memTable بر روی دیسک، پاک می‌شوند.

محتوای memTable تحت شرایط زیر بر روی دیسک نوشته می‌شوند:

- Commit log **به حداکثر اندازه خود رسیده باشد**: هدف اصلی استفاده از Commit log ردگیری تمامی داده‌هایی است که در memTable وجود دارند و هنوز بر روی دیسک نوشته نشده‌اند. در صورتی که Commit log حداکثر فضای مجاز را پر کرده باشد، memTable را مجبور به ذخیره‌ی داده‌ها بر روی دیسک می‌کند.

[3] Sorted Strings Table (SSTable) [5]

- **به صورت دوره‌ای**: با توجه به پارامتر memTable_flush_period_in_ms. این پارامتر که بر روی یک جدول تنظیم می‌شود، دوره زمانی (بر حسب میلی‌ثانیه) را که memTable مربوط به جدول تخلیه[۳۶] می‌شود را مشخص می‌کند.
- **اندازه** memTable **از حد آستانه فراتر رود**

تمامی داده‌های Cassandra در SSTable-ها نوشته و ذخیره می‌شوند. memTable-ها و SSTable-ها برای هر یک از جداول جداگانه ذخیره می‌شوند. Commit log میان تمامی جداول مشترک است. SSTable-ها تغییرناپذیر[۳۷] هستند، بدین معنی که پس از آنکه محتوای memTable بر روی آن‌ها نوشته شد، نمی‌توان مجدداً بر روی آن‌ها داده‌های جدید نوشت. SSTable-ها فایل‌های ذخیره شده بر روی دیسک هستند؛ این فایل‌ها در دایرکتوری data قرار دارند که به صورت پیش‌فرض در لینوکس در مسیر /var/lib/Cassandra/data بوده و پارامتر data_file_directories در فایل پیکربندی Cassandra.yaml نیز این مسیر را تنظیم می‌کند. برای هر یک از فضاهای کلید و برای هر یک از جداول، یک دایرکتوری در دایرکتوری data وجود دارد. به عنوان نمونه، در مسیر /var/lib/Cassandra/data/test_key_space/test-5114ee41836211e897c5db884a5c3753، test_key_space نام فضای جدول است و test-5114ee41836211e897c5db884a5c3753 نام جدول به همراه شناسه‌ی یکتای جدول است. هر تخلیه، یک SSTable جدید بر روی دیسک ایجاد می‌کند. برای هر SSTable، پایگاه‌داده، ساختاری شامل فایل‌های مختلف ایجاد می‌کند که یکی از مهترین این فایل‌ها Data.db است که داده‌های SSTable-ها در آن‌ها ذخیره می‌شوند. به عنوان نمونه، نام فایل داده در یک SSTable به صورت زیر است:

```
aa-1-bti-Data.db
```

این نام شامل چهار قسمت مطابق جدول ۳ است.

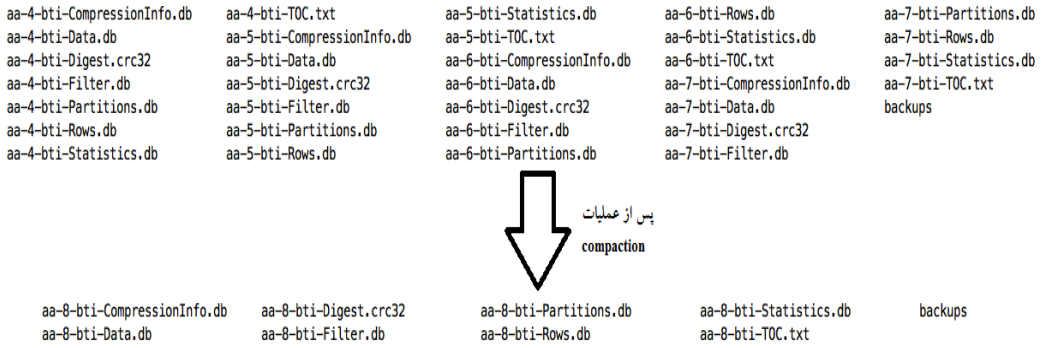
جدول ۳: ساختار نام فایل‌ها در SSTable

قسمتی از نام فایل	معنی
aa	نسخه‌ی فرمت ذخیره‌سازی SSTable
bti	فرمت SSTable
1	به این عدد generation می‌گویند. این عدد یک شماره‌ی شاخص است که با هر بار

[3] Flush
[3] Immutable

	ایجاد SSTable جدید برای یک جدول، یک واحد به آن اضافه می‌شود.
Data	نوع اطلاعاتی که در فایل ذخیره می‌شوند را مشخص می‌کند.

SSTable-های ایجاد شده به صورت دوره‌ای در قالب فرآیندی به نام فشرده‌سازی[3] با یکدیگر ادغام می‌شوند. با تمام شدن فرآیند فشرده‌سازی تنها یک SSTable با عدد شاخص جدید وجود خواهد داشت (شکل ۸).



شکل ۸: عمل فشرده‌سازی در SSTable

استراتژی پیش‌فرض برای عمل فشرده‌سازی، SizeTieredCompactionStrategy (STCS) نام دارد. در این استراتژی در صورتی که تعدادی SSTable با اندازه یکسان بر روی دیسک وجود داشته باشد، عمل فشرده‌سازی انجام می‌شود. تعداد SSTable-هایی که وجود آن‌ها منجر به اجرای فشرده‌سازی می‌شود، با پارامتر min_threshold برای هر جدول مشخص می‌شود که مقدار پیش‌فرض آن برابر ۴ است. برای یک جدول می‌توان مقدار پیش‌فرض پارامتر min_threshold را با استفاده از دستور زیر تغییر داد.

```
ALTER TABLE test
 WITH compaction =
 {'class' : 'SizeTieredCompactionStrategy', 'min_threshold' : 6 }
```

با استفاده از ابزار SSTabledump و مطابق با دستور زیر می‌توان محتوای SSTable را مشاهده کرد.

```
SSTabledump aa-8-bti-Data.db
```

[3] Compaction [8]

نمونه‌ای از خروجی این دستور در شکل ۹ نشان داده شده است.


```
{
  "partition" : {
    "key" : [ "12" ],
    "position" : 138
  },
  "rows" : [
    {
      "type" : "row",
      "position" : 170,
      "liveness_info" : { "tstamp" : "2018-07-27T06:08:17.815056Z" },
      "cells" : [
        { "name" : "note", "value" : "test" }
      ]
    }
  ]
},
{
  "partition" : {
    "key" : [ "3" ],
    "position" : 171,
    "deletion_info" : { "marked_deleted" : "2018-07-27T06:00:37.297778Z", "local_delete_time" : "2018-07-27T06:00:37Z" }
  },
  "rows" : [ ]
}
```


شکل ۹: نمونه‌ای از خروجی SSTabledump

همان‌طور که در شکل ۹ دیده می‌شود، سطری با کلید ۱۲ مقدار ستون note برای آن test است. همچنین سطری با کلید ۳ حذف شده است. حال در صورتی که محتوای SSTable که شامل درج سطر با کلید ۳ بوده است، هنوز طی عملیات فشرده‌سازی پاک نشده باشد، می‌توان محتوای سطر حذف شده را بازیابی کرد (شکل ۱۰).


```
{
  "partition" : {
    "key" : [ "3" ],
    "position" : 19
  },
  "rows" : [
    {
      "type" : "row",
      "position" : 50,
      "liveness_info" : { "tstamp" : "2018-07-09T10:26:08.477897Z" },
      "cells" : [
        { "name" : "note", "value" : "test" }
      ]
    }
  ]
}
```


شکل ۱۰: بازیابی محتوای سطر حذف شده با استفاده از SSTabledump

بنابراین یک راه برای بازیابی داده‌های حذف شده به صورت ناخواسته، خواندن محتوای SSTable-ها با استفاده از ابزار SSTabledump است. البته این هدف در صورتی محقق می‌شود که SSTable مربوط به درج سطر و SSTable مربوط به حذف سطر از یکدیگر مجزا باشند و SSTable مربوط به درج، طی عمل فشرده‌سازی حذف نشده باشد. در صورتی که درج و حذف سطر در یک SSTable ثبت شوند، محتوای SSTable به صورت شکل ۱۱ است و نمی‌توان محتوای سطر را از آن بازیابی کرد.

```
[
  {
    "partition" : {
      "key" : [ "20" ],
      "position" : 0,
      "deletion_info" : { "marked_deleted" : "2018-07-27T12:42:58.585989Z", "local_delete_time" : "2018-07-27T12:42:58Z" }
    },
    "rows" : [ ]
  }
```

شکل ۱۱: محتوای سطر حذف شده در صورت ثبت درج و حذف در یک SSTable

خواندن محتوای SSTable-ها		
SSTabledump <SSTable_FILE_NAME>.db		
برخی دستورات سودمند		
۱	تغییر مقدار پارامتر min_threshold برای یک جدول	ALTER TABLE <TABLE_NAME> WITH compaction = {'class' : 'SizeTieredCompactionStrategy', 'min_threshold' : 6 }

۳-۵ ارائه‌ی مستندات

در این گام، اطلاعات کسب شده در طول فرآیند جرم‌شناسی پایگاه‌داده مستند می‌شود. برای این منظور می‌بایست برای هر یک از رویدادهای درج، حذف و مشاهده جداول در پایگاه‌داده به هنگام بررسی جرم، جداول زیر را به طور دقیق تکمیل و به مدیریت سازمان برای پیگیری‌های لازم و طرح دعوی ارائه کرد.

جدول ۴: تهیه‌ی مستند از فرآیند جرم‌شناسی رویداد درج غیرمجاز در جدول

درج غیرمجاز در جدول			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی	ممیزی ☐		
منابع رویدادنگاری	رویدادنگاری پرسمان‌های زمانبر☐		
شیوه یا ابزار تحلیل	فاقد شیوه یا ابزار تحلیل است.		
امکان ترمیم	حذف سطر(های) درج شده☐		
توضیحات			

جدول ۵: تهیه‌ی مستند از فرآیند جرم‌شناسی رویداد حذف غیرمجاز محتوای جدول

حذف غیرمجاز محتوای جدول			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی	ممیزی ☐		
منابع رویدادنگاری	رویدادنگاری پرسمان‌های زمانبر☐		
شیوه یا ابزار تحلیل	فاقد شیوه یا ابزار تحلیل است.		
امکان ترمیم	استفاده از محتوای SSTable-ها و ابزار SSTabledump☐		
توضیحات			

جدول ۶: تهیه‌ی مستند از فرآیند جرم‌شناسی رویداد مشاهده غیرمجاز محتوای جدول

مشاهده غیرمجاز محتوای جدول			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی	ممیزی ☐		
منابع رویدادنگاری	رویدادنگاری پرسمان‌های زمانبر☐		
شیوه یا ابزار تحلیل	فاقد شیوه یا ابزار تحلیل است.		
امکان ترمیم	فاقد امکان ترمیم است.		
توضیحات			

جدول ۷: تهیه‌ی مستند از فرآیند جرم‌شناسی رویداد بروزرسانی غیرمجاز محتوای جدول

بروزرسانی غیرمجاز محتوای جدول			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی	ممیزی ☐		
منابع رویدادنگاری	رویدادنگاری پرسمان‌های زمانبر☐		
شیوه یا ابزار تحلیل	فاقد شیوه یا ابزار تحلیل است.		
امکان ترمیم	استفاده از محتوای SSTable-ها و ابزار SSTabledump☐		
توضیحات			

۳-۶ جمع‌بندی

در این فصل به بحث و بررسی رویدادهای غیرمجازی که به صورت خصمانه یا ناخواسته توسط کاربر/برنامه‌کاربردی بر روی پایگاه‌داده اعمال می‌شوند و موجب درج سطر (های) جدید، حذف سطر (های) موجود یا مشاهده تمامی یا بخشی از جدول (ها) در پایگاه می‌گردند، پرداختیم. برای این منظور، پس از شناسایی جرم، دو رویکرد به نام‌های رویدادنگاری پرسمان‌های زمانبر و ممیزی برای جمع‌آوری شواهد و اطلاعات مربوط به جرم را معرفی کردیم. برای هر یک از این رویکردها، تنظیمات مربوط به فعال‌سازی، استخراج اطلاعات و شواهد، تحلیل شواهد، ترمیم و در پایان تهیه مستندات را تشریح کردیم.

۴ تغییر غیرمجاز شمای پایگاه‌داده

در این فصل به بحث و بررسی رویدادهای غیرمجازی که به صورت خصمانه یا ناخواسته توسط کاربر/برنامه‌کاربردی بر روی پایگاه‌داده اعمال و موجب تغییر در شمای پایگاه‌داده می‌شود، می‌پردازیم. این دسته از رویدادها، از جمله دستورات تعریف داده (DDL[3])در پایگاه‌داده به حساب می‌آیند. از آنجاییکه رویدادهای مورد بحث در این فصل از نقطه نظر جرم‌شناسی بسیار به هم نزدیک هستند، لذا تنها بر روی رویداد حذف غیرمجاز یک جدول متمرکز می‌شویم.

۴-۱ شناسایی جرم

فرآیند جرم‌شناسی با مشاهده‌ی رویدادهای غیرمجاز در سامانه آغاز می‌شود. بنابراین در صورتیکه مدیر پایگاه‌داده خود به طور مستقیم یا غیرمستقیم (از طریق کاربران)، نسبت به تغییر در شمای پایگاه‌داده همچون حذف یک جدول آگاهی پیدا کند که انتظار تغییر آن در شرایط فعلی را نداشته است، فرآیند جرم‌شناسی در سامانه آغاز می‌شود.

۴-۲ جمع‌آوری اطلاعات و شواهد

از آنجاییکه هر پایگاه‌داده شواهد مربوط به اعمال مختلف را در بخش‌ها و فایل‌های رویدادنگاری مختلف ذخیره می‌کند، در این بخش قصد داریم منابع مربوط به شواهد رویداد حذف غیرمجاز یک جدول را تعیین نماییم. لازم به ذکر است که تمرکز ما در فرآیند جرم‌شناسی پایگاه‌داده تنها بر روی اطلاعات حاصل از رویدادنگاری و ممیزی در پایگاه‌داده است و شامل اطلاعات موجود در سیستم عامل و حافظه‌ی اصلی

[3] Data Definition Language [9]

نمی‌باشد. از این‌رو، در ادامه به معرفی منابع مربوط به شواهد رویداد حذف غیرمجاز یک جدول و نحوه آگاهی و تنظیم آن‌ها می‌پردازیم.

استفاده از رویدادنگاری پرسمان‌های زمانبر: در پایگاه‌داده‌ی Cassandra می‌توان تنظیم کرد که پرسمان‌ها و رویدادهایی که اجرای آن‌ها کند و زمانبر است، ثبت شوند. از این قابلیت می‌توان برای رویدادنگاری از تمامی پرسمان‌های اجرایی استفاده کرد. تنظیمات مربوط به رویدادنگاری از پرسمان‌های زمانبر، در فایل dse.yaml در قسمت cql_slow_log_options وجود دارد. نمونه‌ای از این تنظیمات در ادامه آورده شده است.

```
cql_slow_log_options:
    enabled: true
#    #  When t > 1, log queries taking longer than t milliseconds.
#    #      0 <= t <= 1,  log queries above t percentile
    threshold: 0
#    # Initial number of queries before percentile filter becomes active
    minimum_samples: 1
    ttl_seconds: 259200
#    # Keeps slow queries in-memory only and doesn't write data to the database.
#    # WARNING - if this is set to 'false' then set threshold >= 2000, otherwise there will be a
#    # high load on the database.
    skip_writing_to_db: false
```

مطابق تنظیمات بالا، تمامی پرسمان‌های اجرایی ثبت و به مدت ۲۵۹۲۰۰ ثانیه در پایگاه‌داده باقی می‌مانند. جدول زیر، گام‌های جمع‌آوری اطلاعات و شواهد مربوط به رویداد تغییر غیرمجاز شمای پایگاه‌داده با استفاده از رویدادنگاری پرسمان‌های زمانبر را نشان می‌دهد.

فعال‌سازی رویدادنگاری پرسمان‌های زمانبر برای رویدادنگاری از تمامی پرسمان‌ها		
۱	فعال‌سازی رویدادنگاری از پرسمان‌های زمانبر	cql_slow_log_options: enabled: true
۲	رویدادنگاری از تمامی پرسمان‌ها	threshold: 0
۳	ذخیره‌ی رویدادها به مدت ۲۵۹۲۰۰ ثانیه	ttl_seconds: 259200
۴	ذخیره‌ی رویدادها در پایگاه‌داده	skip_writing_to_db: false

استفاده از ممیزی: رویدادهای ممیزی می‌توانند در فایل‌های رویدادنگاری یا در جدولی در پایگاه‌داده‌ی Cassandra ذخیره شوند. هنگامی که ممیزی فعال شود، به صورت پیش‌فرض رویدادها در فایل‌هایی بر روی سیستم‌عامل ذخیره می‌شوند و همچنین در صورتی که اندازه رویدادهای ثبت‌شده افزایش پیدا کند، ذخیره‌ی رویدادها در جدولی در Cassandra بهتر خواهد بود.

فایل پیکربندی dse.yaml به منظور فعال‌سازی و تنظیم ممیزی در Cassandra استفاده می‌شود. برای فعال‌سازی ممیزی، در قسمت audit_logging_options، مقدار پارامتر enabled باید true باشد. همچنین سایر تنظیمات مرتبط با ممیزی در فایل dse.yaml در ادامه توضیح داده شده‌اند:

- تنظیم گزینه‌ی logger به منظور ثبت رویدادهای ممیزی در جدولی از Cassandra (مقدار CassandraAuditWriter) یا در فایل (مقدار SLF4JAuditWriter)
- تعیین انواع رویدادهایی که باید از آن‌ها ممیزی تهیه شود، همچون AUTH، DML، DDL، DCL و QUERY. این موارد با پارامترهای include_categories و exclude_categories تنظیم می‌شوند. همچنین اگر مقداری برای این دو پارامتر مشخص نشود، از کلیه‌ی رویدادها ممیزی تهیه می‌شود.
- تنظیم فضای کلیدهای[4] که باید از آن‌ها ممیزی تهیه شود (پارامترهای included_keyspaces و excluded_keyspaces)
- در صورتی که رویدادها در جدولی در Cassandra ثبت می‌شوند، تنظیم زمان نگهداری رویدادهای ثبت‌شده در پارامتر retention_time انجام می‌شود. مقدار پیش‌فرض این پارامتر صفر بوده و بدین معناست که حفظ تمامی رویدادها به مدت نامحدود است.

در صورتی که مقدار پارامتر logger برابر SLF4JAuditWriter باشد، رویدادها در فایل ثبت شوند و می‌توان ادامه‌ی تنظیمات را در فایل پیکربندی logback.xml انجام داد. بدین منظور تنظیمات پیش‌فرضی در این فایل قرار دارد که می‌توان آن‌ها را پذیرفت و یا تغییر داد. تنظیمات پیش‌فرض مربوط به ممیزی در فایل، در ادامه آورده شده است.

```
<!--audit log-->
<appender name="SLF4JAuditWriterAppender"
class="ch.qos.logback.core.rolling.RollingFileAppender">
<file>${Cassandra.logdir}/audit/audit.log</file>
<encoder>
```

[4] KeySpace

```
    <pattern>%-5level [%thread] %date{ISO8601} %X{service} %F:%L - %msg%n</pattern>
    <immediateFlush>true</immediateFlush>
  </encoder>
  <rollingPolicy class="ch.qos.logback.core.rolling.FixedWindowRollingPolicy">
    <fileNamePattern>${Cassandra.logdir}/audit/audit.log.%i.zip</fileNamePattern>
    <minIndex>1</minIndex>
    <maxIndex>5</maxIndex>
  </rollingPolicy>
  <triggeringPolicy class="ch.qos.logback.core.rolling.SizeBasedTriggeringPolicy">
    <maxFileSize>200MB</maxFileSize>
  </triggeringPolicy>
</appender>

<logger name="SLF4JAuditWriter" level="INFO" additivity="false">
  <appender-ref ref="SLF4JAuditWriterAppender"/>
</logger>
```

مطابق با این تنظیمات رویدادهای ممیزی در فایل audit.log در مسیر ${Cassandra.logdir}/audit نوشته می‌شوند. جدول زیر، گام‌های جمع‌آوری اطلاعات و شواهد مربوط به رویداد تغییر غیرمجاز شمای پایگـاه‌داده با استفاده از ممیزی را نشان می‌دهد.

فعال‌سازی ممیزی		
۱	فعال‌سازی ممیزی	audit_logging_options: enabled: true
۲	ثبت رویدادهای ممیزی در جدول یا فایل	logger: CassandraAuditWriter logger: SLF4JAuditWriter
۳	تعیین انواع رویدادهای ممیزی (عدم تعیین مقدار برای این دو پارامتر = ممیزی از کلیه‌ی رویدادها)	included_categories: DDL excluded_categories:
۴	تنظیم فضای کلید (عدم تعیین مقدار برای این دو پارامتر = ممیزی از کلیه‌ی فضاهای کلید)	# included_roles: # excluded_roles:
۵	تنظیم زمان نگهداری رویدادهای ثبت‌شده (صفر یعنی حفظ تمامی رویدادها به مدت نامحدود)	retention_time: 0

۶	تنظیم محل فایل ممیزی در فایل پیکربندی logback.xml	<file>${Cassandra.logdir}/audit/audit.log</file>

۴-۳ استخراج و تجزیه و تحلیل اطلاعات

در این بخش نحوه‌ی استخراج اطلاعات از منابع مربوط به شواهد، مورد بحث و بررسی قرار می‌گیرد. با استفاده از اطلاعات جمع‌آوری شده، تحلیل‌گر باید تغییرات در شمای پایگاه‌داده را بررسی و در صورت مشاهده‌ی رویداد غیرمجاز، آن را به عنوان جرم تلقی نماید. در ادامه برای هر یک از منابع حاوی شواهد برای رویدادهای سامانه، نحوه استخراج و تجزیه و تحلیل تشریح می‌گردد.

استخراج و تحلیل اطلاعات با استفاده از رویدادنگاری از پرسمان‌های زمانبر: برای مشاهده‌ی رویدادهای ثبت شده از دستور زیر استفاده می‌شود:

```
SELECT * FROM dse_perf.node_slow_log;
```

نمونه‌ای از خروجی دستور بالا، در شکل ۱۲ نشان داده شده است.

```
 node_ip    | date                          | start_time                   | commands
| duration | parameters | source_ip | table_names           | tracing_session_id | username
-----------+-------------------------------+------------------------------+-----------------------------------------------------------------------------
+----------+------------+-----------+-----------------------+--------------------+----------
 127.0.0.1 | 2018-10-04 00:00:00.000000+0000 | 396d53b0-c799-11e8-b690-495c446d8edb |                                ['drop table users;']
|      977 |       null | 127.0.0.1 | {'test_key_space.users'} |               null | cassandra
```

شکل ۱۲: رویدادنگاری از حذف جدول

همان‌طور که در شکل ۱۲ دیده می‌شود، اطلاعاتی همچون زمان اجرای پرسمان، پرسمان اجرایی و کاربر اجراکننده‌ی آن ثبت شده است.

استخراج و تحلیل اطلاعات با استفاده از رویدادنگاری از پرسمان‌های زمانبر
SELECT * FROM dse_perf.node_slow_log;

استخراج و تحلیل اطلاعات با استفاده از ممیزی: نمونه‌ای از ممیزی تهیه شده حاصل از حذف جدول در ادامه آورده شده است:

```
INFO  [IOThread-1] 2018-07-27 22:03:56,048  SLF4JAuditWriter.java:88 -
      host:localhost/127.0.0.1|source:/127.0.0.1|user:Cassandra|authenticated:Cassandra|timestamp:15327128
      36048|category:DDL|type:DROP_CF|ks:test_key_space|cf:users|operation:drop table users;|consistency
      level:ONE
```

همان‌طور که در خروجی بالا دیده می‌شود، با استفاده از رویدادهای ممیزی می‌توان اطلاعاتی در مورد زمان اجرای پرسمان، پرسمان اجرایی و کاربر اجراکننده‌ی آن به دست آورد.

استخراج و تحلیل اطلاعات ممیزی ثبت شده در جدول
SELECT * FROM dse_audit.audit_log;

۴-۴ ترمیم

در صورتی که پیش از تغییر در شمای پایگاه‌داده و به طور خاص حذف یک جدول، اقدامات لازم جهت تهیه فایل پشتیبان صورت پذیرفته باشد، با بازیابی فایل می‌توان اطلاعات از دست رفته را مجدداً در اختیار گرفت. در این بخش سعی داریم روش‌هایی به غیر از روش‌های مربوط به تهیه‌ی نسخه‌های پشتیبان را برای بازیابی جداول و داده‌های از دست رفته مورد بحث و بررسی قرار دهیم.

به صورت پیش‌فرض پارامتر auto_snapshot در فایل cassandra.yaml مقداری برابر true دارد. این پارامتر با مقدار true سبب می‌شود که هنگام حذف جدول به صورت خودکار از آخرین وضعیت جدول تصویر لحظه‌ای[۴] تهیه شود. با تعریف یک جدول در یک فضای جدول، در مسیر /var/lib/Cassandra/data و در دایرکتوری مربوط به فضای جدول، دایرکتوری با نام و شناسه‌ی یکتای جدول ایجاد می‌شود. با حذف جدول، در دایرکتوری snapshot موجود در دایرکتوری مربوط به جدول، دایرکتوری با فرمت نام dropped-<timestamp>-<tablename> همچون dropped-1532712836088-users ایجاد می‌شود و در آن کلیه‌ی داده‌های مربوط به SSTable از آخرین وضعیت جدول قرار می‌گیرد. با استفاده از فایل schema.cql موجود در این دایرکتوری می‌توان شمای جدول را بازیابی کرد (شکل ۱۳).

```
CREATE TABLE IF NOT EXISTS test_key_space.users (
        id int PRIMARY KEY,
        name text)
        WITH ID = 2b986ce0-91c3-11e8-9e12-77967976d5c5
        AND bloom_filter_fp_chance = 0.01
        AND caching = { 'keys': 'ALL', 'rows_per_partition': 'NONE' }
        AND cdc = false
        AND comment = ''
        AND compaction = { 'max_threshold': '32', 'min_threshold': '4', 'class': 'org.apache.cassandra.
gy' }
        AND compression = { 'chunk_length_in_kb': '64', 'class': 'org.apache.cassandra.io.compress.LZ4(
        AND crc_check_chance = 1.0
        AND dclocal_read_repair_chance = 0.1
        AND default_time_to_live = 0
        AND extensions = {  }
        AND gc_grace_seconds = 864000
        AND max_index_interval = 2048
        AND memtable_flush_period_in_ms = 0
        AND min_index_interval = 128
        AND read_repair_chance = 0.0
        AND speculative_retry = '99PERCENTILE';
~
"users-2b986ce091c311e89e1277967976d5c5/snapshots/dropped-1532712836088-users/schema.cql" 20L, 834C
```

شکل ۱۳: استخراج شمای جدول حذف شده

[4] Snapshot [1]

همچنین فایل Data.db، محتوای جدول پیش از حذف آن را نشان می‌دهد (شکل ۱۴).

```
tools/bin/sstabledump users-2b986ce091c311e89e1277967976d5c5/snapshots/dropped-1532712836088-user
s/aa-1-bti-Data.db
[
  {
    "partition" : {
      "key" : [ "1" ],
      "position" : 0
    },
    "rows" : [
      {
        "type" : "row",
        "position" : 26,
        "liveness_info" : { "tstamp" : "2018-07-27T17:33:34.948159Z" },
        "cells" : [
          { "name" : "name", "value" : "sn" }
        ]
      }
    ]
  },
  {
    "partition" : {
      "key" : [ "2" ],
      "position" : 27
    },
    "rows" : [
      {
        "type" : "row",
        "position" : 56,
        "liveness_info" : { "tstamp" : "2018-07-27T17:33:41.451901Z" },
        "cells" : [
          { "name" : "name", "value" : "dn" }
        ]
      }
    ]
```

شکل ۱۴: استخراج محتوای جدول حذف شده

خواندن محتوای SSTable-ها
SSTabledump <SSTable_FILE_NAME>.db

۴-۵ ارائه‌ی مستندات

در این گام، اطلاعات کسب شده در طول فرآیند جرم‌شناسی پایگاه‌داده مستند می‌شود. برای این منظور می‌بایست به هنگام بررسی جرم، جدول زیر به طور دقیق تکمیل و به مدیریت سازمان برای پیگیری‌های لازم و طرح دعوی ارائه گردد.

جدول ۸: تهیه‌ی مستند از فرآیند جرم‌شناسی رویداد تغییر غیرمجاز شمای پایگاه‌داده

تغییر غیرمجاز شمای پایگاه‌داده			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی	ممیزی ☐		
منابع رویدادنگاری	رویدادنگاری پرسمان‌های زمانبر☐		

شیوه یا ابزار تحلیل	فاقد شیوه یا ابزار تحلیل است.
امکان ترمیم	استفاده از قابلیت auto_snapshot ☐
توضیحات	

۶-۴ جمع‌بندی

در این فصل به بحث و بررسی رویدادهای غیرمجازی که به صورت خصمانه یا ناخواسته توسط کاربر/برنامه‌کاربردی بر روی پایگاه‌داده اعمال و موجب تغییر در شمای پایگاه‌داده می‌گردند، پرداخته شد. برای این منظور، پس از شناسایی جرم، دو رویکرد به نام‌های رویدادنگاری پرسمان‌های زمانبر و ممیزی برای جمع‌آوری شواهد و اطلاعات مربوط به جرم معرفی گردید. برای هر یک از این رویکردها، تنظیمات مربوط به فعال‌سازی، استخراج اطلاعات و شواهد، تحلیل شواهد، ترمیم و در پایان تهیه مستندات تشریح شد.

۵ تلاش برای ورود غیرمجاز به پایگاه‌داده

در این فصل، رویدادهای غیرمجازی که به صورت خصمانه یا ناخواسته توسط کاربر/برنامه‌کاربردی برای ورود به پایگاه‌داده صورت می‌پذیرد را مورد بحث و بررسی قرار می‌دهیم. این رویداد در واقع به هنگام تلاش برای ورود از طریق بدست آوردن نام کاربری و کلمه عبور به سمپاد تحمیل می‌شود. در ادامه، فرآیند جرم‌شناسی مربوط به این رویداد مورد بحث و بررسی قرار می‌گیرد.

۵-۱ شناسایی جرم

در صورتی که تلاش‌های ناموفق برای ورود به سیستم پایگاه‌داده ثبت شوند، با مشاهده‌ی رویدادهای ثبت‌شده می‌توان حمله به پایگاه‌داده برای یافتن نام کاربری یا کلمه‌ی عبور یک نام کاربری را تشخیص داد. در این شرایط فرآیند جرم‌شناسی در سامانه آغاز می‌شود.

۵-۲ جمع‌آوری اطلاعات و شواهد

از آنجاییکه هر پایگاه‌داده شواهد مربوط به اعمال مختلف را در بخش‌ها و فایل‌های رویدادنگاری مختلف ذخیره می‌کند، در این بخش قصد داریم منابع مربوط به شواهد رویداد تلاش برای ورود به پایگاه‌داده را تعیین نماییم. در صورتی که در فایل dse.yaml در قسمت audit_logging_options برای پارامتر included_categories و excluded_categories مقداری تعیین نشود، تمامی رویدادها ثبت می‌شوند. همچنین در صورتی که مقدار این پارامتر برابر AUTH باشد، تنها از رویدادهای مربوط به تلاش برای ورود به پایگاه‌داده (ورود موفقیت‌آمیز و ناموفق)، ممیزی تهیه می‌شود.

فایل پیکربندی dse.yaml به منظور فعال‌سازی و تنظیم ممیزی در Cassandra استفاده می‌شود. برای فعال‌سازی ممیزی، در قسمت audit_logging_options، مقدار پارامتر enabled باید true باشد. همچنین سایر تنظیمات مرتبط با ممیزی در فایل dse.yaml در ادامه توضیح داده شده‌اند:

- تنظیم گزینه‌ی logger به منظور ثبت رویدادهای ممیزی در جدولی از Cassandra (مقدار CassandraAuditWriter) یا در فایل (مقدار SLF4JAuditWriter)
- تعیین انواع رویدادهایی که باید از آن‌ها ممیزی تهیه شود، از جمله:
  - **AUTH**: ثبت رویدادهای مربوط به ورود به سیستم
  - **DML**: ثبت رویدادهای درج، به‌روزرسانی، حذف و سایر اعمال مرتبط با تغییر داده‌ها
  - **DDL**: ثبت ایجاد شیئ، تغییر و حذف آن و سایر رویدادهای مرتبط با تعریف داده‌ها
  - **DCL**: ثبت رویدادهای اعطاء و حذف نقش، ایجاد نقش و سایر رویدادهای مرتبط با نقش‌ها
  - **QUERY**: ثبت تمامی پرسمان‌ها

  این موارد با پارامترهای include_categories و exclude_categories تنظیم می‌شوند. همچنین اگر مقداری برای این دو پارامتر مشخص نشود، از کلیه‌ی رویدادها ممیزی تهیه می‌شود.
- تنظیم فضای کلیدهایی[4] که باید از آن‌ها ممیزی تهیه شود (پارامترهای included_keyspaces و excluded_keyspaces)
- در صورتی که رویدادها در جدولی در Cassandra ثبت می‌شوند، تنظیم زمان نگهداری رویدادهای ثبت‌شده در پارامتر retention_time انجام می‌شود. مقدار پیش‌فرض این پارامتر صفر است بدین معنی که حفظ تمامی رویدادها به مدت نامحدود است.

در صورتی که مقدار پارامتر logger برابر SLF4JAuditWriter باشد، رویدادها در فایل ثبت شوند و می‌توان ادامه‌ی تنظیمات را در فایل پیکربندی logback.xml انجام داد. بدین منظور تنظیمات پیش‌فرضی در این فایل قرار دارد که می‌توان آن‌ها را پذیرفت و یا تغییر داد. تنظیمات پیش‌فرض مربوط به ممیزی در فایل، در ادامه آورده شده است.

```
<!--audit log-->
```

[4] KeySpace [2]

```
  <appender name="SLF4JAuditWriterAppender"
class="ch.qos.logback.core.rolling.RollingFileAppender">
    <file>${Cassandra.logdir}/audit/audit.log</file>
    <encoder>
      <pattern>%-5level [%thread] %date{ISO8601} %X{service} %F:%L - %msg%n</pattern>
      <immediateFlush>true</immediateFlush>
    </encoder>
    <rollingPolicy class="ch.qos.logback.core.rolling.FixedWindowRollingPolicy">
      <fileNamePattern>${Cassandra.logdir}/audit/audit.log.%i.zip</fileNamePattern>
      <minIndex>1</minIndex>
      <maxIndex>5</maxIndex>
    </rollingPolicy>
    <triggeringPolicy class="ch.qos.logback.core.rolling.SizeBasedTriggeringPolicy">
      <maxFileSize>200MB</maxFileSize>
    </triggeringPolicy>
  </appender>

  <logger name="SLF4JAuditWriter" level="INFO" additivity="false">
    <appender-ref ref="SLF4JAuditWriterAppender"/>
  </logger>
```

مطابق با این تنظیمات رویدادهای ممیزی در فایل audit.log در مسیر ${Cassandra.logdir}/audit نوشته می‌شوند.

فعال‌سازی ممیزی		
۱	فعال‌سازی ممیزی	audit_logging_options: enabled: true
۲	ثبت رویدادهای ممیزی در جدول یا فایل	logger: CassandraAuditWriter logger: SLF4JAuditWriter
۳	تعیین انواع رویدادهای ممیزی (عدم تعیین مقدار برای این دو پارامتر = ممیزی از کلیه‌ی رویدادها)	included_categories: AUTH excluded_categories:
۴	تنظیم فضای کلید (عدم تعیین مقدار برای این دو پارامتر = ممیزی از کلیه‌ی فضاهای کلید)	# included_roles: # excluded_roles:

۵	تنظیم زمان نگهداری رویدادهای ثبت‌شده (صفر یعنی حفظ تمامی رویدادها به مدت نامحدود)	retention_time: 0
۶	تنظیم محل فایل ممیزی در فایل پیکربندی logback.xml	<file>${Cassandra.logdir}/audit/audit.log</file>

۵-۳ استخراج و تجزیه و تحلیل اطلاعات

در این بخش نحوه‌ی استخراج اطلاعات برای هر یک از منابع ذکرشده در بخش‌های قبل بیان می‌شود. با استفاده از اطلاعات جمع‌آوری شده، تحلیل‌گر باید تلاش برای ورود به پایگاه‌داده را بررسی نماید. نمونه‌ای از اطلاعات ثبت شده از تلاش‌های ناموفق برای ورود به پایگاه‌داده در ممیزی در شکل ۱۵ نشان داده شده است.

```
INFO  [IOThread-0] 2018-08-02 11:04:28,366  SLF4JAuditWriter.java:88 - host:localhost/127.0.0.1|source:/127.0.0.1|user:cassandra|
authenticated:cassandra|timestamp:1533191668365|category:AUTH|type:LOGIN_ERROR|operation:Failed login attempt for user - cassandr
a
INFO  [IOThread-0] 2018-08-02 11:04:37,234  SLF4JAuditWriter.java:88 - host:localhost/127.0.0.1|source:/127.0.0.1|user:cassandra|
authenticated:cassandra|timestamp:1533191677234|category:AUTH|type:LOGIN_ERROR|operation:Failed login attempt for user - cassandr
a
INFO  [IOThread-0] 2018-08-02 11:04:42,471  SLF4JAuditWriter.java:88 - host:localhost/127.0.0.1|source:/127.0.0.1|user:cassandra
authenticated:cassandra|timestamp:1533191682471|category:AUTH|type:LOGIN_ERROR|operation:Failed login attempt for user - cassandr
a
INFO  [IOThread-0] 2018-08-02 11:04:50,382  SLF4JAuditWriter.java:88 - host:localhost/127.0.0.1|source:/127.0.0.1|user:cassandra|
authenticated:cassandra|timestamp:1533191690382|category:AUTH|type:LOGIN|operation:Successful login for user - cassandra
```

شکل ۱۵: ممیزی از تلاش‌های ناموفق برای ورود به پایگاه‌داده

همان‌طور که در شکل ۱۵ دیده می‌شود، در فواصل زمانی کوتاه، نام کاربری cassandra برای ورود تلاش کرده است و با شکست روبرو شده است. این رویدادهای ثبت شده نشان‌دهنده‌ی حمله‌ی brute force برای یافتن رمز عبور مربوط به یک نام کاربری است.

استخراج و تحلیل اطلاعات ممیزی ثبت شده در جدول
SELECT * FROM dse_audit.audit_log;

۵-۴ ترمیم

با استفاده از دستور زیر می‌توان نام کاربری تعریف کرد:

```
create role <role_name> with password = <password> and superuser = false and login = true;
```

در این دستور با تعیین مقدار true برای پارامتر login، در حقیقت توانایی ورود به پایگاه‌داده به کاربر اعطا شده است. در صورتی که کاربری بتواند با یک نام‌کاربری مشخصی، توانایی ورود به پایگاه‌داده را پیدا کند و با استفاده از رویدادهای ثبت شده بتوان نشانه‌هایی از حمله‌ی brute force را تشخیص داد، برای جلوگیری از

حدس رمز عبور مربوط به نام کاربری، می‌توان با استفاده از دستور زیر اجازه‌ی ورود به پایگاه‌داده را موقتاً از حساب کاربری سلب کرد.

```
alter role <role_name> with password = <password> and login = false;
```

حال در صورت تلاش برای ورود به پایگاه‌داده با نام کاربری مورد نظر، پیامی مشابه شکل ۱۶ تولید می‌شود.

```
Connection error: ('Unable to connect to any servers', {'127.0.0.1': AuthenticationFailed('Failed to authenticate to 127.0.0.1:
 Error from server: code=0100 [Bad credentials] message="cassandrais not permitted to log in"',)})
```

شکل ۱۶: سلب اجازه‌ی ورود به پایگاه‌داده از یک نام کاربری

پارامتر یا دستوری برای غیرفعال کردن خودکار کاربر پس از تعداد مشخصی تلاش ناموفق برای ورود به پایگاه داده در Cassandra وجود ندارد.

سلب اجازه‌ی ورود به پایگاه‌داده از حساب کاربری
alter role <role_name> with password = <password> and login = false;

۵-۵ ارائه‌ی مستندات

برای این منظور می‌بایست به هنگام بررسی جرم، جدول زیر به طور دقیق تکمیل و به مدیریت سازمان برای پیگیری‌های لازم و طرح دعوی ارائه گردد.

جدول ۹: تهیه‌ی مستند از فرآیند جرم‌شناسی برای رویداد تلاش برای یافتن نام کاربری یا کلمه‌ی عبور

تلاش برای یافتن نام کاربری یا کلمه‌ی عبور			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی	ممیزی ☐		
منابع رویدادنگاری	فاقد منبع رویدادنگاری است.		
شیوه یا ابزار تحلیل	فاقد شیوه یا ابزار تحلیل است.		
امکان ترمیم	قفل کردن نام کاربری به صورت دستی ☐		
توضیحات			

۵-۶ جمع‌بندی

در این فصل، تلاش برای ورود به پایگاه‌داده در صورت نداشتن نام کاربری یا کلمه‌ی عبور مورد بحث و بررسی قرار گرفت. برای این منظور، پس از شناسایی جرم، رویکرد ممیزی برای جمع‌آوری شواهد و اطلاعات مربوط

به جرم معرفی گردید. در پایان نیز برای رویکرد معرفی شده، تنظیمات مربوط به فعال‌سازی، استخراج اطلاعات و شواهد، تحلیل شواهد، ترمیم و تهیه مستندات تشریح گردید.

۶ خلاصه مطالب

جرم‌شناسی پایگاه‌داده فرآیندی است که طی آن تلاش می‌شود تا اطلاعاتی چون زمان/ چگونگی/ چرایی و فرد مجرم برای یک رخداد غیرمجاز در سامانه مشخص شود. جرم‌شناسی پایگاه‌داده زمانی رخ می‌دهد که از مأمور ممیزی، کشف چگونگی وقوع نقض امنیتی و شخص مجرم درخواست شود. جرم‌شناسی پایگاه‌داده، چالش‌ها و مسائل زیادی به همراه دارد که آن را تبدیل به یک موضوع پیچیده و مبهم کرده‌است.

با استفاده از مدل‌های فرآیند جرم‌شناسی پایگاه‌داده می‌توان به صورت ساختاریافته عملیات مربوط به جرم‌شناسی پایگاه‌داده را پیش برد. در مراحل مختلف فرآیند جرم‌شناسی، تمرکز اصلی بر روی پایگاه‌داده و اطلاعات موجود در آن برای شناسایی جرم است. در حقیقت تنها از اطلاعات ثبت‌شده در پایگاه‌های داده به منظور شناسایی جرم استفاده می‌شود و استفاده از اطلاعات ثبت‌شده بر روی سیستم عامل، داده‌های موجود در حافظه و شبکه خارج از حوزه‌ی مورد بحث است. در یک دسته‌بندی کلی، جرم‌شناسی پایگاه‌داده شامل گام‌های زیر است:

۱. شناسایی جرم،
۲. جمع‌آوری اطلاعات و شواهد،
۳. تجزیه و تحلیل،
۴. ترمیم،
۵. ارائه‌ی مستندات.

هر سامانه پایگاه‌داده، شواهد مربوط به رویدادهای مختلف را در فایل‌های مختلف برای استفاده در تجزیه و تحلیل جرم‌شناسی ذخیره می‌کند. این به معنای آن است که برای تجزیه و تحلیل جرم‌شناسی باید نسبت به چگونگی عملکرد پایگاه‌داده و محل فایل‌ها و مصنوعات مختلف اطلاع داشت. لازم به ذکر است که جمع‌آوری مصنوعات و شواهد می‌تواند سبب تغییر در پایگاه‌داده شود، بنابراین پیش از استخراج اطلاعات از پایگاه‌داده یا خارج از پایگاه‌داده باید نسبت به این موضوع و پایدار یا ناپایدار بودن اطلاعات آگاهی پیدا کرد. همچنین دانستن این نکته خالی از لطف نیست که اگر مهاجم جدولی را حذف کند و برای پاک کردن ردپای خود اطلاعات موجود در فایل‌های رویدادنگاری و ممیزی را دستی تغییر دهد، به عنوان نمونه، کاربر ثبت شده به عنوان کاربر اجراکننده‌ی پرسمان را تغییر دهد، مطلوب است که بتوان تغییر غیرمجاز در فایل‌های رویدادنگاری و ممیزی را متوجه شد؛ هرچند چنین قابلیتی در سیستم مدیریت پایگاه‌داده‌ی Cassandra وجود ندارد.

برای هر یک از رویدادهای غیرمجاز در سامانه پایگاه‌داده به هنگام بررسی جرم، جداولی برای ارائه مستندات لازم در نظر گرفته شده است که می‌بایست در طول فرآیند جرم‌شناسی به طور دقیق تکمیل و به مدیریت سازمان برای پیگیری‌های لازم و طرح دعوی ارائه گردد. جدول ۱۰، کلیه جداول اطلاعاتی مربوطه را به تصویر کشیده است.

جدول ۱۰: تهیه‌ی مستند از فرآیند جرم‌شناسی در پایگاه‌داده‌ی Cassandra

درج غیرمجاز در جدول			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی	ممیزی ☐		
منابع رویدادنگاری	رویدادنگاری پرسمان‌های زمانبر☐		
شیوه یا ابزار تحلیل	فاقد شیوه یا ابزار تحلیل است.		
امکان ترمیم	حذف سطر(های) درج شده☐		
توضیحات			
حذف غیرمجاز محتوای جدول			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی	ممیزی ☐		
منابع رویدادنگاری	رویدادنگاری پرسمان‌های زمانبر☐		
شیوه یا ابزار تحلیل	فاقد شیوه یا ابزار تحلیل است.		
امکان ترمیم	استفاده از محتوای SSTable-ها و ابزار SSTabledump☐		
توضیحات			
مشاهده غیرمجاز محتوای جدول			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی	ممیزی ☐		
منابع رویدادنگاری	رویدادنگاری پرسمان‌های زمانبر☐		
شیوه یا ابزار تحلیل	فاقد شیوه یا ابزار تحلیل است.		
امکان ترمیم	فاقد امکان ترمیم است.		

توضیحات			
بروزرسانی غیرمجاز محتوای جدول			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی	ممیزی ☐		
منابع رویدادنگاری	رویدادنگاری پرسمان‌های زمانبر☐		
شیوه یا ابزار تحلیل	فاقد شیوه یا ابزار تحلیل است.		
امکان ترمیم	استفاده از محتوای SSTable-ها و ابزار SSTabledump☐		
توضیحات			
تغییر غیرمجاز شمای پایگاه‌داده			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی	ممیزی ☐		
منابع رویدادنگاری	رویدادنگاری پرسمان‌های زمانبر☐		
شیوه یا ابزار تحلیل	فاقد شیوه یا ابزار تحلیل است.		
امکان ترمیم	استفاده از قابلیت auto_snapshot ☐		
توضیحات			
تلاش برای یافتن نام کاربری یا کلمه‌ی عبور			
وقوع جرم	عدم وقوع ☐	وقوع خصمانه ☐	وقوع ناخواسته ☐
شیوه ممیزی	ممیزی ☐		
منابع رویدادنگاری	فاقد منبع رویدادنگاری است.		
شیوه یا ابزار تحلیل	فاقد شیوه یا ابزار تحلیل است.		
امکان ترمیم	قفل کردن نام کاربری به صورت دستی ☐		
توضیحات			

۷ منابع

[1]. DB audit and security 360, version 5.0, SoftTree Technologies, Inc.

[2]. http://www.dba-oracle.com

[3]. Al-Dhaqm, A., Razak, S.A., Othman, S.H., Nagdi, A. and Ali, A., 2016. A GENERIC DATABASE FORENSIC INVESTIGATION PROCESS MODEL. Jurnal Teknologi, 78.

[4]. R. Ramakrishnan and J. Gehrke. Database Management Systems (Third Edition). McGraw-Hill, Inc. New York, NY, USA, 2003.

[5]. G. Palmer, A Road Map for Digital Forensic Research, DFRWS Technical Report, DTR-T001-01 Final, Air Force Research Laboratory, Rome, New York, 2001.

[6]. https://docs.oracle.com

[7]. https://en.wikipedia.org/wiki/Transaction_log

[8]. J. Shital, Forensic Investigation for Database Tampering using Audit Logs, International Journal of Engineering Research & Technology (IJERT), Vol. 4 Issue 03, March 2015

[9]. K. Fowler, SQL Server database forensics, presented at the Black Hat USA Conference, 2007.

[10]. Fasan, O.M. and Olivier, M.S., 2012. On Dimensions of Reconstruction in Database Forensics. In WDFIA (pp. 97-106).

[11]. Al-Dhaqm, A., Razak, S.A., Othman, S.H., Nagdi, A. and Ali, A., 2016. A GENERIC DATABASE FORENSIC INVESTIGATION PROCESS MODEL. Jurnal Teknologi, 78.

[12]. https://solutioncenter.apexsql.com/recover-sql-server-database-using-only-a-transaction-log-file-ldf-and-old-backup-files/

[13]. H. Q. Beyers, "Database forensics: Investigating compromised database management systems", 2013.

[14]. Khanuja, H.K., Adane, D.S.: A Framework For Database Forensic Analysis. Published in Computer Science & Engineering: An International Journal (CSEIJ) 2(3) (2012)

[15]. Finnigan, P., *Oracle Incident Response and Forensics:* Preparing for and Responding to Data Breaches, 2018, Apress, Berkeley, CA.

[16]. https://dbatricksworld.com/ora-38707-media-recovery-is-not-enabled/

[17]. http://www.innovateus.net/science/what-forensics

[18]. R. Urbano, 2017, Oracle Database Administrator's Guide, 12c Release 2 (12.2)

[19]. http://www.kpipartners.com/dse-vs-Cassandra-open-source

[20]. https://docs.datastax.com/en/dse/6.0/dse-admin/

[21]. https://teddyma.gitbooks.io/learnCassandra/content/model/where_is_data_stored.html

[22]. http://abiasforaction.net/apache-Cassandra-memTable-flush/

[23]. https://docs.datastax.com/en/dse/6.0/dse-arch/datastax_enterprise/dbInternals/dbIntHowDataWritten.html

[24]. http://saumitra.me/blog/how-Cassandra-stores-data-on-filesystem/

[25]. https://docs.datastax.com/en/dse/6.0/cql/cql/cql_reference/cql_commands/cqlCreateTable.html#compactSubprop__STCS

[26]. https://blog.rdx.com/cassandra-and-relational-database-schema-comparison-query-vs-relationship-modeling/