جستجو روی داده‌های رمزگذاری‌شده: قابلیت‌ها و چالش‌ها

انیسه نجفی*[1]، مجید بیات[2] و سید حمید حاج‌سیدجوادی[3]

[1] دانشجوی دکترای جبر، دانشکده علوم پایه، دانشگاه شاهد، تهران، ایران
ensiyeh.najafi@shahed.ac.ir

[2] استادیار گروه مهندسی کامپیوتر، دانشکده فنی، دانشگاه شاهد، تهران، ایران
mbayat@shahed.ac.ir

[3] دانشیار گروه ریاضی و علوم کامپیوتر، دانشکده علوم پایه، دانشگاه شاهد، تهران، ایران
h.s.javadi@shahed.ac.ir

چکیده

رشد تولید داده، ظرفیت‌ها و نیازمندی‌هایی را در جهان به همراه دارد. از یک سو، نگهداری داده‌های تولیدشده، امکان به‌کارگیری مجدد و تحلیل روی داده‌ها را فراهم می‌کند که منجر به تولید علم داده می‌شود؛ از سوی دیگر، حجم زیاد داده‌ها، نیازمند فضای ذخیره‌سازی و امکان جستجو برای بهره برداری از آن‌ها است. محاسبات ابری، یک مدل فناورانه و عملیاتی است که محدودیت‌های ذخیره‌سازی و محاسباتی را در نگهداری و بهره برداری از داده‌ها مرتفع می‌کند. همین‌طور رمزگذاری جستجوپذیر به‌عنوان یک روش پرکاربرد مبتنی بر محاسبات ابری، علاوه‌بر حفظ امنیت داده‌ها، امکان جستجو روی آن‌ها را فراهم می‌کند. در این مقاله، روش‌های رمزگذاری جستجوپذیر، همراه با محدودیت‌ها و قابلیت‌های هر یک از آن‌ها مورد بررسی قرار می‌گیرد. در پایان، توضیحاتی درباره چگونگی به‌کارگیری رمزگذاری جستجوپذیر در داده‌های پزشکی ارائه می‌شود.

واژگان کلیدی: رمزگذاری جستجوپذیر متقارن، رمزگذاری کلید عمومی با جستجوی کلیدواژه، محاسبات ابری، ذخیره‌سازی ابری، پرونده الکترونیک سلامت

۱- مقدمه

ذخیره‌سازی ابری، بارگذاری داده‌ها روی سرورهایی است که کاربر به‌الزام به آن‌ها دسترسی فیزیکی ندارد. این داده‌ها توسط ارائه‌دهندهٔ خدمات ابر یا سرور ابر، نگهداری و مدیریت می‌شوند و مورد جستجو قرار می‌گیرند. ذخیره‌سازی در ابر امکان دسترسی سریع و آسان و همه‌جایی به داده‌های برون‌سپاری‌شده، کاهش بار محاسباتی وکاهش هزینه ذخیره‌سازی را فراهم می‌کند. از دیگر فواید به‌کارگیری ذخیره‌سازی ابری، جلوگیری از خرابی یا ازبین‌رفتن اطلاعات است. با توجه به مزایای ذخیره سازی ابری، بسیاری از افراد حقیقی، سازمان‌ها و مراکز مختلف که به نگهداری و تحلیل داده‌های خود نیاز دارند، علاقه‌مند به استفاده از این فناوری هستند؛ اما از آنجا که در بیش‌تر موارد این اطلاعات، محرمانه است و مالکان داده تمایلی به آشکارسازی آن‌ها ندارند؛ حفظ حریم خصوصی داده‌ها از ضروریات به‌کارگیری این فناوری است. برای رفع این نیازمندی، ساده‌ترین راه حل، رمزگذاری داده‌ها توسط مالک، قبل از برون‌سپاری آن‌ها در ابر است. اما این روش، جستجو در داده‌ها و بازیابی آن‌ها را بدون رمزگشایی غیر ممکن می‌کند. رمزگذاری جستجوپذیر، روشی برای رمزگذاری داده‌ها است تا علاوه‌بر حفظ حریم خصوصی آن‌ها، امکان جستجوی کارا درداده‌های رمزگذاری‌شده، بدون نیاز به رمزگشایی وجود داشته باشد. تمام پژوهش‌های انجام‌شده در این حوزه، تبادل بین کارایی جستجو، کارامدی طرح و حفظ حریم خصوصی داده‌هاست.

در سناریوی رمزگذاری جستجوپذیر، مالک داده برای برون‌سپاری داده‌های خود، آن‌ها را با یکی از روش‌های رمزگذاری مانند AES به‌طور محلی رمزگذاری می‌کند؛ اما برای حفظ قابلیت جستجو در داده‌های رمزگذاری‌شده، مالک داده برخی از مهم‌ترین کلیدواژه‌های متن اصلی را استخراج

*نویسندهٔ مسؤول

کرده و آن‌ها را به‌صورت رمزگذاری‌شده، همراه با داده‌های رمزگذاری‌شده، به سرور ابر ارسال می‌کند. وظیفه سرور، ذخیره‌سازی، جستجو و بازیابی داده‌های رمزگذاری شده است. از سوی دیگر، کاربری که قصد جستجو در ابر را دارد، کلیدواژه یا کلیدواژه‌های رمزگذاری‌شدهٔ مورد نظر خود را به‌صورت توکن یا ترپدر[1] به سرور ابر ارسال می‌کند. سرور بدون این‌که از محتوای ترپدر و محتوای کلیدواژه‌های همراه فایل‌های رمزگذاری‌شده اطلاع داشته باشد از طریق الگوریتم جستجو، فایل‌های مرتبط با کلیدواژه‌های جستجوشده را یافته و به‌عنوان نتایج جستجو به کاربر برمی‌گرداند. در این سناریو فرض می‌شود که توان محاسباتی خدمات‌دهندهٔ ابر بسیار بیش‌تر از کاربران است. هم‌چنین در ادامه، منظور از رمزگذاری و رمزگشایی داده، رمزگذاری و رمزگشایی واژگان استخراجی از فایل‌هاست و رمزگذاری و رمزگشایی خود فایل‌ها در این جا موضوع بحث نیست.

ساده‌ترین روش رمزگذاری جستجوپذیر، رمزگذاری همهٔ کلیدواژه‌های یک فایل است. بدین ترتیب در هر بار جستجو، باید تمام کلیدواژه‌های فایل با به‌کارگیری ترپدر و الگوریتم جستجو مورد بررسی قرار گیرد. به‌عبارت دیگر برای جستجو روی n فایل رمزگذاری‌شده که هر کدام w کلیدواژه داشته باشند، باید O(nw) بار الگوریتم جستجو شود [۱]. برای کاهش پیچیدگی جستجو، به‌ازای هر فایل یک فهرست یا ایندکس از کلیدواژه‌های منتخب آن فایل استخراج می‌شود [۲] که آن را فهرست مستقیم[2] می‌نامند. با به‌کارگیری فهرست مستقیم، پیچیدگی جستجو به O(n)، یعنی تعداد فایل‌های مورد جستجو کاهش می‌یابد؛ بنابراین استفاده از این نوع فهرست، به میزان قابل توجهی پیچیدگی جستجو را کاهش و عملکرد طرح را نسبت به روش قبل بهبود می‌دهد. فهرست دیگری که برای کاهش پیچیدگی جستجو پیشنهاد شد، تخصیص فهرستی از شناسه‌های فایل‌ها به هر کلیدواژه است [۳] که آن را فهرست معکوس[3]، می‌نامند. با استفاده از فهرست معکوس، پیچیدگی جستجو به O(w)، یعنی تعداد واژگان دیکشنری کاهش می‌یابد. در همین اواخر بعضی از طرح‌ها نیز، از ساختمان دادهٔ درخت برای ساماندهی واژگان استخراجی استفاده می‌کنند که پیچیدگی جستجو را به پیچیدگی لگاریتمی بر اساس تعداد فایل‌ها کاهش می‌دهد.

تمام روش‌های رمزگذاری جستجوپذیر به دو دسته قابل تقسیم است؛ دستهٔ نخست، رمزگذاری جستجوپذیر متقارن (SSE[4]) است که یک اولیه رمزنگاری است و تعریف صریح و عملی آن برای نخستین‌بار توسط سنگ[5] و همکاران در سال ۲۰۰۰ ارائه شد [۱]. در این روش، مالک داده، واژگان یا عبارات مجموعه داده‌های خود را به‌کمک کلید خصوصی و یک الگوریتم، رمزگذاری می‌کند. هنگام جستجو در داده‌های رمزگذاری‌شده در ابر، مالک داده یا هر کاربر مجاز دیگر، ترپدر کلیدواژه‌های مورد جستجوی خود را باید با همان کلید خصوصی، تولید و برای سرور ارسال کند.

دستهٔ دوم با استفاده از یک اولیه رمزگذاری دیگر با نام رمزگذاری کلید عمومی با قابلیت جستجوی کلیدواژه (PEKS[6]) انجام می‌شود که برای نخستین‌بار توسط بونه و همکاران [۴] در سال ۲۰۰۴ مطرح شد. در این روش، مالک داده، فهرست متناظر با مجموعه‌داده‌های خود را به‌کمک یک الگوریتم رمزگذاری نامتقارن و کلید عمومی کاربری که می‌خواهد به او اجازهٔ جستجو دهد، رمزگذاری کرده و به خدمات‌دهندهٔ ابر ارسال می‌کند. به‌منظور جستجو روی داده‌های بارگذاری‌شده، کاربر با کلید خصوصی خود، ترپدر متناظر با کلیدواژه مورد نظرش را تولید و برای سرور ارسال می‌کند. واضح است که فقط، کاربری می‌تواند ترپدر مناسب برای جستجو در داده‌های رمزگذاری‌شده را بسازد که فهرست متناظر با داده‌ها، با کلید عمومی او رمزگذاری شده باشد. تفاوت این روش با روش نخست در این است که مالک داده بدون این‌که به کاربر مجاز کلیدی ارسال کند به او اجازه جستجو و دسترسی به داده‌های خود را می‌دهد. با توسعهٔ رمزگذاری کلید عمومی و تولید اولیه‌های جدید، PEKS نیز گسترش یافت. از جملهٔ این اولیه‌های رمزنگاری، رمزگذاری مبتنی بر ویژگی ۷ و وکالت برای رمزگذاری مجدد ۸ است. هر دوی این اولیه‌ها در رمزگذاری جستجوپذیر با عناوین رمزگذاری مبتنی بر ویژگی با قابلیت جستجوی کلیدواژه (ABKS[7]) و وکالت برای رمزگذاری مجدد با قابلیت جستجوی کلیدواژه (PRES[8]) مطرح شده‌اند.

در مقایسه با SSE، طرح‌های مبتنی بر PEKS به‌خاطر عدم نیاز به مدیریت کلید، کارامدتر، اما از لحاظ محاسباتی کارایی طرح‌های مبتنی بر SSE بیشتر از PEKS است.

با توجه به ارتباط تنگاتنگ ابر و اینترنت اشیا، هر یک از این روش‌ها کاربردهایی در بهداشت و درمان [۵، ۶، ۷، ۸]،

[1] Trapdoor
[2] Forward index
[3] Inverted index

[4] Symmetric Searchable Encryption
[5] Song
[6] Public key Encryption with Keyword Search
[7] Attribute based encryption with keyword search
[8] Proxy re-encryption with keyword search

شهر و ساختمان‌های هوشمند [۹]، حمل و نقل هوشمند [۱۰]، صنایع هوشمند [۱۱] و ... دارند.

سه عامل کارایی، امنیت و عملکرد، نقش اساسی در رمزگذاری جستجوپذیر بازی می‌کنند. منظور از کارایی طرح، پیچیدگی ساخت فهرست، ترپدر و فرایند جستجو و هر قابلیت دیگر طرح است. همچنین میزان فضای لازم برای ذخیره‌سازی و تعداد دفعات لازم برای برقراری ارتباط بین کاربر و خدمات‌دهنده ابر در کارایی طرح مؤثر است.

حفظ حریم خصوصی داده‌های رمزگذاری‌شده، کلیدواژه‌های استخراج‌شده از هر فایل و کلیدواژه‌های جستجو[1] و همچنین نتایج جستجو[2] امنیت رمزگذاری جستجوپذیر را تضمین می‌کند. هریک از این موارد می‌تواند در دو مدل امنیتی مورد بررسی قرار گیرند. در مدل نخست، سرور ابر، راست‌گو اما کنجکاو[3] در نظر گرفته می‌شود. در چنین حالتی، سرور، الگوریتم جستجو را به‌درستی اجرا می‌کند؛ اما ممکن است از داده‌هایی که به‌طور طبیعی در اختیار دارد، اطلاعاتی را به‌دست آورد. به‌عبارت دیگر در این مدل مهاجم غیر فعال است. اما در مدل دوم، سرور ممکن است، شبه‌راست‌گو اما کنجکاو[4] و یا بدخواه[5]باشد. اگر سرور شبه‌راست‌گو و کنجکاو باشد، ممکن است، بخشی از داده‌ها یا فهرست به‌روزشده‌ای را که در ابر ذخیره شده حذف کند یا ممکن است، برای صرفه‌جویی در منابع محاسباتی یا پهنای باند، عمل جستجو را به‌طور کامل انجام ندهد؛ اما سرور بدخواه سعی می‌کند با تحلیل داده‌هایی که در اختیار دارد، حملاتی چون DoS را انجام دهد. واضح است که مدل دوم، محدودیت‌های امنیتی ضعیف‌تری دارد و طرح‌های امن در مدل دوم، دارای امنیتی بیشتری نسبت به مدل نخست هستند.

عملکرد طرح، قابلیت‌های اضافه‌شده به طرح است. حذف و اضافه‌کردن و یا تغییر در فایل‌های برون‌سپاری‌شده، صراحت بیش‌تر درکلیدواژه‌های جستجو مانند جستجوی چند کلیدواژه به‌طور عطفی[6] یا غیرعطفی[7]، جستجوی فازی، جستجوی بولی، جستجوی کلمات هم‌معنی، وارسی نتایج جستجو، رتبه‌بندی نتایج جستجو، جستجوی یک مجموعه فایل توسط چند کاربر، تعیین اجازه دسترسی به فایل‌ها جزء قابلیت‌ها و درواقع سبب عملکرد بهتر یک طرح رمزگذاری جستجوپذیر است.

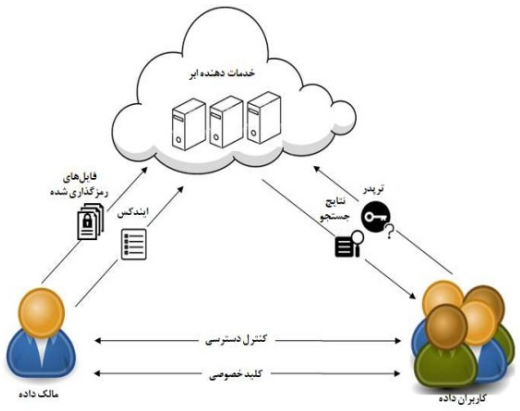
در ادامه هر یک از دو روشSSE و PEKSو سه عامل بالا را در آن‌ها مورد بررسی قرار می‌دهیم. در پایان به بررسی بعضی کاربردهای رمزگذاری جستجوپذیر می‌پردازیم.

۲-رمزگذاری جستجوپذیر متقارن

هر طرح رمزگذاری SSE شامل چهار الگوریتم زمان چندجمله‌ای (احتمالی) به‌صورت زیر است:

- الگوریتم راه‌اندازی[8] که پارامتر امنیتی را دریافت و پارامترهای عمومی طرح را به همراه کلید خصوصی مالک داده بر می‌گرداند.
- الگوریتم رمزگذاری که کلمات استخراجی با هر نوع ساختمان داده را به همراه پارامترهای عمومی و کلید خصوصی دریافت می‌کند و ساختمان دادهٔ رمزگذاری‌شده را بر می‌گرداند.
- الگوریتم تولید ترپدر که کلمه یا عبارت مورد نظر را به همراه پارامترهای عمومی و کلید خصوصی دریافت و ترپدر را تولید می‌کند.
- الگوریتم جستجو که ترپدر، پارامترهای عمومی و ساختمان دادهٔ کلمات استخراجی رمزگذاری‌شده را دریافت و مجموعه فایل‌های انتخاب‌شده را درصورت وجود بر می‌گرداند.

شکل (۱)، سناریو کلی را در یک طرح SSE نشان می‌دهد.



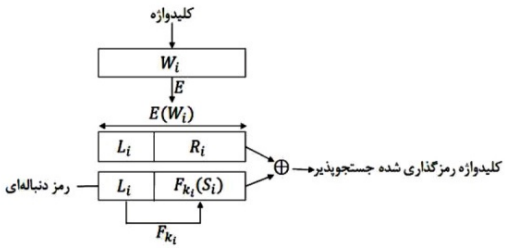
(شکل-۱): مدل سامانهٔ رمزگذاری جستجوپذیر متقارن

سنگ و همکاران [۱] در سال ۲۰۰۰ نخستین طرح رمزگذاری جستجوپذیر متقارن را معرفی کردند. در این طرح تمام واژگان یا عبارات متن به‌صورت جداگانه رمزگذاری

[1] Access pattern
[2] Search pattern
[3] Honest but curious
[4] Semi-honest but curious
[5] malicious
[6] Conjunctive
[7] Disjunctive
[8] Set up

می‌شود و برای جستجو، از بررسی متوالی واژگان رمزگذاری شده استفاده می‌شود. جزئیات طرح به‌صورت زیر است:

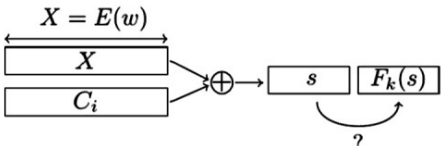
مالک داده هر کلیدواژه را با یک تابع رمزگذاری مانند E به یک رشته n بیتی تبدیل و این رشته را به دو قسمت L_i به‌طول n-m و R_i به‌طول m تقسیم می‌کند. از آن‌جایی‌که تابع رمزگذاری در این طرح، قطعی است، مالک داده برای پوشاندن یکسانی کلیدواژه‌های رمزگذاری‌شده در متن، از XOR هر کلیدواژه با یک رشتهٔ شبه‌تصادفی مختص آن کلیدواژه استفاده می‌کند. برای تولید این رشتهٔ شبه‌تصادفی n بیتی، مالک داده، رشتهٔ شبه‌تصادفی n-m بیتی S_i را با یک مولد شبه‌تصادفی تولید کرده و برای تولید m بیت تصادفی دیگر از تابع شبه‌تصادفی F_{k_i} با ورودی S_i و خروجی m بیتی استفاده می‌کند که در آن k_i، به‌صورت $k_i = f_K(L_i)$ محاسبه می‌شود و K، یک رشتهٔ تصادفی منتخب توسط مالک داده است. حال کلمه رمزگذاری‌شده، با این رشتهٔ تصادفی n بیتی XOR می‌شود. رشتهٔ حاصل، کلیدواژهٔ رمزگذاری‌شدهٔ جستجوپذیر است که توسط مالک داده در فضای ابر قرار می‌گیرد. شکل (۲) فرآیند رمزگذاری هر کلیدواژه را نشان می‌دهد.



(شکل-۲): نخستین الگوریتم رمزگذاری جستجوپذیر

کاربر برای جستجوی کلیدواژهW در فضای ابر، باید ترپدر متناظر با این کلیدواژه را بسازد. برای ساخت ترپدر، کلیدواژه با همان تابع رمزگذاری معین E رمزشده و رشتهٔ حاصل به دو رشتهٔ n-m بیتی L و m بیتی R تقسیم می‌شود. کاربر مقدار $k = f_K(L)$ را محاسبه کرده و همراه با کلیدواژه رمزگذاری‌شده به سرور ارسال می‌کند.

سرور برای جستجو، ترپدر را با هر یک از کلیدواژه‌های رمزگذاری‌شده XOR می‌کند؛ سپس n-m بیت ابتدای رشتهٔ حاصل را جدا کرده و F_kآن را محاسبه می‌کند. اگر مقدار حاصل، برابر با m بیت بعدی باشد، ترپدر با کلیدواژه رمزگذاری‌شده تطبیق یافته است. شکل (۳)، روش جستجو در این طرح را نشان می‌دهد.



(شکل-۳): مدل سامانهٔ رمزگذاری جستجوپذیر متقارن

۲-۱-امنیت در طرح‌های SSE

امنیت نخستین طرح رمزگذاری جستجوپذیر تحت فرض امن‌بودن اولیه‌های به‌کاررفته در این طرح، IND-CPA[1] است. این سطح از امنیت برای یک اولیه رمزگذاری قابل قبول است؛ به این معنا که مهاجم، تنها به عبارات رمزگذاری‌شده دسترسی دارد و هر دو عبارت متمایز رمزگذاری‌شده، از دیدگاه او قابل تمیز نیست؛ اما رمزگذاری جستجوپذیر در این مدل، یک اولیه امن به حساب نمی‌آید؛ زیرا با وجود اینکه هر یک از کلمات رمزگذاری شده قابل رمزگشایی نیست؛ اما بعد از چند بار درخواست، با در دست داشتن ترپدرهای هر درخواست و نتایج به‌دست آمده از هر بار عملیات جستجو، سرور ابر می‌تواند با تحلیل آماری روی کلمات رمزگذاری شده آن‌ها را رمزگشایی کند. به‌منظور بهبود امنیت رمزگذاری جستجوپذیر متقارن، گو[2] [۲] مفهوم جدیدی از امنیت با نام IND-CKA[3] را معرفی کرد و نشان داد، طرحش در این مدل امن است. در این مدل، مهاجم نمی‌تواند با مشاهده ایندکس و داده‌های رمزگذاری‌شده اطلاعاتی را در مورد مجموعه‌داده به جز تعداد و طول فایل‌ها به‌دست آورد. گو برای اثبات امنیت طرحش در مدل IND-CKA، بین مهاجم و چالش‌گر یک بازی تعریف می‌کند که خلاصه آن به‌صورت زیر است:

فرض کنیم مهاجم دو مجموعه کلیدواژه با طول مساوی را به چالش‌گر ارسال می‌کند؛ به‌طوری‌که تمام کلیدواژه‌ها در دو مجموعه، یکسان نباشند. چالش‌گر تصادفی فهرست متناظر با یکی از فایل‌ها را تولید و برای مهاجم ارسال می‌کند. اگر مهاجم با احتمالی که به میزان قابل توجه، از ۰/۵ فاصله دارد، نتواند فهرست مورد نظر را به‌درستی حدس بزند، فهرست، اطلاعاتی را در مورد فایل آشکار نمی‌کند. به‌عبارت دیگر، اگر تمایزدادن بین دو فهرست متعلق به هر یک از فایل‌ها، کاری دشوار باشد، حدس‌زدن دست‌کم یک کلیدواژه که در بین دو فهرست مشترک نیست، نیز کاری دشوار است. در مدل دیگر گو با نام IND2-CKA، شرط تعداد مساوی کلیدواژه‌ها در دو مجموعه حذف شده است. واضح است که طرح‌های امن در

[1] Indistinguish chosen plaintext attack
[2] Goh
[3] Indistinguish chosen keyword attack

مدل دوم، امنیت بیشتری را نسبت به مدل نخست تضمین می‌کند. به هر حال این دو مدل، تنها امنیت طرح را با در نظر گرفتن نشت احتمالی اطلاعات از فهرست، مورد بررسی قرار می‌دهند و اطلاعات به‌دست‌آمده از طرق دیگر، مانند رابطهٔ بین ترپدر و فهرست و همین‌طور نتایج جستجوی قبلی را در نظر نمی‌گیرند.

در سال ۲۰۰۵، چنگ[1] و میزنماکر[2] [۱۲]، این دو مدل را برای تأمین امنیت ناکافی دانسته و یک نوع مدل شبیه‌سازی را در نظر می‌گیرند که در آن تمام اطلاعات موجود از داده‌های مالک که در اختیار سرور قرار گرفته و یا قابل استنتاج است، در نظر گرفته می‌شود. در این جا یک بازی طراحی‌شده که در آن چالش، تشخیص طرح SSE، با پارامترهای واقعی و طرح SSE، با پارامترهای ایده‌آل (شبیه‌سازی‌شده بر اساس نشت داده‌ها) می‌باشد و طرح زمانی امن است که مزیت مهاجم برای تشخیص این دو مدل با اختلاف ناچیزی، ۰/۵باشد.

کورتمولا[3] و همکاران [۳]، شبیه‌سازی اخیر را از حالت غیر انطباقی[4] به انطباقی بهبود داده و مدل IND-CKA2 را معرفی کردند. درواقع در این مدل فرض می‌شود که مهاجم می‌تواند به تعداد چندجمله‌ای‌بار، فهرست رمزگذاری‌شده متناظر با فهرست درخواستی خود را دریافت کند و هر درخواست، تابعی از درخواست‌های قبلی او باشد.

مدل ترکیب‌پذیر سراسری[5] (UC)، ایده‌ای است که در آن ادعا می‌شود، هرگاه طرح با امنیت IND-CKA2، بخشی از یک پروتکل بزرگتر باشد، باز هم امن است [۱۳]. کوروساوا[6] و اوتاکی[7] این مدل را پیشنهاد دادند.

در بعضی از طرح‌های SSE برای ساخت ترپدر از تابع یا جای‌گشت شبه‌تصادفی استفاده شده که خروجی آن‌ها مقدار قطعی است. بنابراین ترپدرهای یکسان نشان‌دهنده‌های عبارات جستجوی یکسان است. از آن جا که ترپدرهای یکسان نتایج یکسانی را نیز به‌همراه دارد، این نوع طرح‌ها با نشت الگوی جستجو و الگوی دسترسی، سبب استنتاج اطلاعاتی از فایل‌های رمزشده توسط سرور خواهد شد. برای حل این مسئله، شن[8] و همکاران [۱۴]، یک روش رمزگذاری کلید متقارن با نام رمزگذاری گزاره‌ای[9] پیشنهاد کردند که در آن، ترپدر به‌صورت احتمالی تولید می‌شود و یک مفهوم امنیتی جدید را با نام حریم خصوصی گزاره‌ای[10] معرفی کردند. ترپدرها در طرح‌های امن بر اساس این مدل، هیچ اطلاعاتی را راجع به کلیدواژه‌های درخواستی آشکار نمی‌کنند. آن‌ها همچنین مفهوم امنیتی دیگری با نام حریم خصوصی متن اصلی[11] تعریف کردند. این مفهوم امنیتی، مربوط به حریم خصوصی فهرست است که در مقالات قبل، IND-CKA نامیده شد. آن‌ها بیان داشتند که هر طرح SSE که هر دو ویژگی اخیر را داشته باشد، دارای امنیت کامل[12] است.

یکی از حملاتی که به‌تازگی مطرح شده، حملهٔ تزریق فایل[13] است [۱۵]. در این حمله، سرور تعدادی فایل بدخواه را با روشی مانند ارسال رایانامه با فایل‌های مالک داده مخلوط و مالک آن‌ها را با کلید خصوصی خود رمزگذاری و همراه فایل‌های خود، به سرور ارسال می‌کند. به‌کمک این فایل‌های بدخواه، سرور می‌تواند در مورد محتوای تمام ترپدرهای فعلی، قبلی و بعدی حدس‌هایی بزند یا آن‌ها را به‌دست آورد. برای تعدیل نشت اطلاعات از طریق این حمله، امنیت روبه‌جلو[14] و امنیت روبه‌عقب[15]، دو نیازمندی امنیتی است که در تعداد کمی از طرح‌های SSE، مورد توجه قرار گرفته است. این دو ویژگی در طرح‌های پویا مفهوم پیدا می‌کند. در اغلب طرح‌های پویا که امکان به‌روزرسانی مجموعه‌داده وجود داشته، سه نوع نشت اطلاعات وجوددارد:

- مقدار تابع درهم‌ساز، با ورودی کلیدواژه‌های جستجو (الگوی جستجو)
- شناسه فایل‌های بازیابی‌شده در جستجو یا حذف و یا اضافه‌کردن فایل (الگوی دسترسی)
- تعداد جفت‌های فایل-کلیدواژه در مجموعه فایل (الگوی اندازه)

بعد از هربار به‌روزرسانی، ممکن است، سرور از ترپدرهای قبل برای جستجو در فایل‌هایی که به‌تازگی اضافه شده و یا از ترپدرهای جدید برای جستجو در فایل‌های حذف‌شده استفاده کند تا اطلاعاتی را از محتوای ترپدرها و فهرست‌ها به‌دست آورد. در طرح‌های با امنیت روبه‌جلو، امکان جستجو روی فایل‌های جدید با استفاده از ترپدرهای قبلی

[1] Cheng
[2] Mitzenmacher
[3] Curtmola
[4] Adoptive
[5] Universalcomposability
[6] Kurosawa
[7] Ohtaki
[8] Shen
[9] Predicate encryption
[10] Predicate privacy
[11] Plaintext privacy
[12] Full security
[13] File-injection attack
[14] Forward security
[15] Backward security

وجود ندارد. همچنین در طرح‌های با امنیت روبه‌عقب امکان جستجو روی فایل‌های حذف‌شده با استفاده از ترپدرهای جدید وجود ندارد. نخستین طرح رمزگذاری جستجوپذیر با امنیت روبه‌جلو توسط استفانوو[1] و همکاران [۱۶] در سال ۲۰۱۴ پیشنهاد شد که در آن از روش ORAM[2] استفاده کردند. قبل از طرح استفانوو، چنگ و میزنماکر [۱۲] طرحی ارائه دادند که دارای امنیت روبه‌جلو است؛ اما در این مقاله به امنیت روبه‌جلو اشاره نشده است. به‌کارگیری ORAM، سبب پنهان‌شدن الگوی دسترسی می‌شود و امنیت بالایی را تضمین می‌کند؛ اما با وجود زمان جستجوی زیرخطی[3]، سربار ارتباطی بالا در آن سبب کارایی بسیار پایین شده است. به‌طورکلی، استفاده از ORAM، HE[4]، FHE[5] و MPC[6]، امن‌ترین طرح‌ها را در SSE خلق کرده است؛ اما این طرح‌ها دارای کارایی پایین و درعمل غیر قابل استفاده هستند. در سال ۲۰۱۶، بست[7] [۱۷] نخستین تعریف رسمی از حریم خصوصی روبه‌جلو را ارائه و آن را در طرح‌های SSE با فهرست معکوس پیاده کرد. در همین اواخر طرح‌هایی با رفع محدودیت‌های طرح‌های قبل مانند ارتقای قابلیت جستجوی یک کلیدواژه به چند کلیدواژه، بهبود زمان جستجو، کاهش فضای لازم برای ذخیره‌سازی توسط مالک داده ارائه شده است [۱۸].

۲-۲-قابلیت‌ها در SSE

صراحت در عبارت جستجو، پویایی و وارسی‌پذیری از قابلیت‌های طرح‌های SSE است. روش‌های جستجوی کلیدواژه در طرح‌های ارائه‌شده، به جستجوی یک کلیدواژه، جستجوی چند کلیدواژهٔ عطفی یا غیر عطفی، جستجوی بولی، جستجوی فازی و جستجوی بسط‌یافته یا معنایی تقسیم‌بندی شده است. در جستجوی عطفی، هر فایل که به‌عنوان نتیجهٔ جستجو به‌دست می‌آید، شامل تمام کلیدواژه‌های جستجو است؛ اما در جستجوی غیرعطفی، کافی است، دست‌کم یکی از کلیدواژه‌ها در فهرست متصل به فایل وجود داشته باشد. عبارت جستجو در جستجوی بولی، مجموعه‌ای از واژگان جستجو با AND، OR و NOT منطقی بین واژگان است. در جستجوی فازی، درصورت وجود غلط املایی در واژگان جستجو یا دردست‌داشتن بخشی از واژه، امکان به‌دست‌آمدن نتایج صحیح فراهم می‌شود. جستجوی بسط‌یافته یا معنایی، این امکان را به طرح می‌دهد تا نه‌تنها خود واژهٔ جستجو، بلکه واژگانی که به هر علت با واژگان جستجو، قرابت معنایی دارند، مورد جستجو قرار بگیرند تا به این ترتیب فایل‌های مرتبط بیشتر یا فایل‌های مرتبط‌تری به‌عنوان نتایج جستجو به‌دست آیند. با این قابلیت‌ها دقت نتایج جستجو افزایش می‌یابد. در بعضی از طرح‌های SSE، کاربر تنها تعداد مشخصی از فایل‌ها را درخواست می‌کند که مرتبط‌ترین فایل‌ها به کلیدواژه‌های جستجو است. در چنین حالتی مسئلهٔ رتبه‌بندی نتایج جستجو مطرح می‌شود که از دیگر قابلیت‌های طرح‌های SSE است. در این طرح‌ها برگرداندن مرتبط‌ترین فایل‌ها به‌عنوان نتیجهٔ جستجو، علاوه برجلوگیری از سردرگمی کاربر، سبب می‌شود که در پهنای باند و هزینه‌های بارگیری نیز صرفه‌جویی شود.

جستجو با یک کلیدواژه. در نخستین طرح SSE، ساده‌ترین روش جستجو، یعنی امکان جستجوی یک کلیدواژه توسط کاربر به‌کار رفته است. در این طرح، سرور باید تک‌تک واژگان رمزگذاری‌شده را مورد بررسی قرار دهد؛ اما کلیدواژه‌ها برای رمزگذاری باید دارای طول مساوی باشند که از معایب این طرح، به‌حساب می‌آید. همچنین به‌جز پویایی، دیگر قابلیت‌های طرح‌های SSE در این طرح وجود ندارد.

نخستین طرح مبتنی بر فهرست مستقیم در سال ۲۰۰۳، توسط گو ارائه شد [۲]. در این طرح به‌ازای هر فایل، با استفاده از واژگان استخراجی از فایل و ساختمان داده بلوم فیلتر[8]، یک فهرست ساخته می‌شود. با به‌کارگیری بلوم‌فیلتر، به جای تعداد واژگان موجود در همه فایل‌ها، زمان جستجو به رابطهٔ خطی با تعداد فایل‌ها کاهش می‌یابد. در طرح گو، هر کلیدواژه بعد از اعمال دوبار تابع شبه‌تصادفی روی آن، وارد ساختمان دادهٔ بلوم‌فیلتر می‌شود. ورودی تابع شبه‌تصادفی دوم، خروجی تابع شبه‌تصادفی نخست همراه با شناسه فایل است تا به این صورت، مقدار بلوم‌فیلتر کلیدواژه‌های یکسان

[1] Stefanov
[2] Oblivious random-access memory
[3] Sublinear
[4] Homomorphic encryption
[5] Fully homomorphic encryption
[6] Multi-party computation
[7] Raphael Bost

[8] بلوم‌فیلتر (Bloom Filter)، یک آرایه m-بیتی با درایه‌های صفر همراه با l تابع درهم‌ساز مستقل است. هر تابع درهم‌ساز هر عضو یک مجموعه را به یک درایه در آرایه نگاشت می‌کند و مقدار آن درایه را یک قرار می‌دهد. بلوم فیلتر برای بررسی عضویت یک کلیدواژه در یک مجموعه به کار می‌رود به این صورت که به‌ازای هر کلیدواژه در مجموعه مورد نظر، مقدار هر l تابع هش کلیدواژه، محاسبه شده و در درایه متناظر با آن عدد یک قرار می‌گیرد. حال برای بررسی عضویت یک کلیدواژه باید مقدار l تابع هش به‌ازای آن کلیدواژه محاسبه شود. اگر تمامی درایه‌های به‌دست‌آمده دارای مقدار یک بود با احتمال بالا آن کلیدواژه در مجموعه قرار دارد.

در فایل‌های مختلف، متمایز باشند و از نشت اطلاعات جلوگیری شود. یکی از پیامدهای استفاده از بلوم‌فیلتر، احتمالی‌بودن صحت عضویت یک کلیدواژه در فهرست است که به آن نرخ خطای پاسخ مثبت[1] می‌گویند. در نرخ خطای پاسخ مثبت ممکن است، عضویت یک کلیدواژه در فهرست تأیید شود؛ درحالی‌که این چنین نیست. با انتخاب مناسب پارامترها می‌توان این خطا را کاهش داد. یکی دیگر از معایب استفاده از بلوم‌فیلتر، تعداد یک‌ها در آرایه است که به تعداد کلیدواژه‌های فهرست بستگی دارد. بنابراین ممکن است از این طریق، تعداد کلیدواژه‌ها در هر فایل مشخص شود. برای حل این مشکل، تعدادی کلیدواژه دلخواه اضافه می‌شود تا این اطمینان به‌دست آید که تعداد کلیدواژه‌های یک فایل به‌طورتقریبی به‌اندازهٔ فایل‌های دیگر است. البته این کار نرخ خطای پاسخ مثبت را افزایش می‌دهد.

این طرح بعضی از محدودیت‌های طرح سنگ مانند پیچیدگی جستجو و طول ثابت کلیدواژه را برطرف کرده است. همچنین چون زمان اجرای بلوم‌فیلتر، مستقل از تعداد کلیدواژه‌ها است، پس زمان اجرای الگوریتم جستجو با تعداد فایل‌های رابطه خطی دارد.

در سال ۲۰۰۵، چنگ و میزنماکر ایده استفاده از فهرست مستقیم را با به‌کارگیری یک دیکشنری از پیش تعریف‌شده برای مجموعه فایل، بهبود دادند [۱۲]. در این طرح یک دیکشنری برای کلیدواژه‌های قابل جستجو تعریف می‌شود. فهرست معادل با هر فایل در واقع یک آرایه m بیتی است که در ابتدا مقدار درایه‌های آن صفر است. به‌ازای هر کلیدواژه موجود در فایل، در درایه معادل با آن کلیدواژه، عدد یک قرار داده می‌شود. همچنین عمل به‌روزرسانی به‌راحتی قابل انجام است. نسبت به طرح‌های قبل، این طرح سربار ارتباطی کمتری دارد.

کورتمولا و همکاران با استفاده از فهرست معکوس نخستین طرح با پیچیدگی جستجوی زیرخطی[2] را معرفی کردند [۳]. درواقع پیچیدگی جستجو به جای تعداد فایل‌ها، متناظر با تعداد کلیدواژه‌ها است که سبب بهبود قابل توجهی نسبت به طرح‌های قبل، مبتنی بر فهرست مستقیم شده است. این فهرست شامل یک آرایه A شامل یک فهرست پیوندی L برای هر کلیدواژه و یک جدول جستجوی T برای شناسایی نخستین گره در A است. برای ساخت آرایه A، از فهرست پیوندی L_i، مرتبط به کلیدواژه w_i آغاز می‌کنیم. هر گره $N_{i,j}$، از L_i، دارای سه رشته به‌صورت $\langle a\|b\|c\rangle$ است که در آن، a شناسه فایل شامل کلیدواژه، b، کلید $k_{i,j}$ است که برای رمزگذاری هر گره استفاده می‌شود و c نشان‌گر گرهٔ بعدی و یا $\emptyset$ است. گره‌ها در آرایه A با یک ترتیب تصادفی به هم ریخته و سپس رمزگذاری می‌شود. گره $N_{i,j}$ با کلید $k_{i,j-1}$ رمزگذاری می‌شود که در گره $N_{i,j-1}$ ذخیره می‌شود؛ بنابراین T جدولی است که به‌ازای هر کلیدواژه w_i، یک گره $N_{i,0}$ وجود دارد که شامل نشان‌گر نخستین گره $N_{i,1}$ در L_i و کلید $k_{i,0}$ متناظر با آن است. گره $N_{i,0}$ در جدول T با $f_y(w_i)$که یک تابع شبه‌تصادفی وابسته به کلیدواژه w_i است، رمز می‌شود. در آخر $N_{i,0}$ رمزشده در درایه$\pi_z(w_i)$ذخیره می‌شود که در آن π یک جای‌گشت شبه‌تصادفی است. به این خاطر که کلید رمزگشایی و محلذخیره‌سازی هر گره هر دو وابسته به کلیدواژه هستند، تولید ترپدر ساده و به شکل $T_w = (\pi_z(w), f_y(w))$ است. ترپدر به سرور اجازه شناسایی و رمزگشایی گره صحیح را در T می‌دهد که درآن، T شامل محل گرهٔ نخست و کلید رمزگشایی آن است. به‌خاطر ساختار فهرست پیوندی، برای یک محل و کلید رمزگشایی صحیح آن، سرور می‌تواند همه گره‌ها و متن رمزگشایی‌شده همهٔ گره‌های مرتبط با شناسه‌های آن فایل را به‌دست آورد. بدین ترتیب زمان جستجو به تعداد فایل‌هایی که شامل آن کلیدواژه است یا بیشینهٔ فایل‌هایی که می‌تواند شامل آن کلیدواژه باشد، کاهش می‌یابد. در این طرح، زمان جستجو به مقدار بهینهٔ خود دست می‌یابد؛ اما ایراد بنیادین در این روش عدم کارایی به‌روزرسانی داده‌ها توسط مالک داده است.

شن و همکاران [۱۴] در سال ۲۰۰۹، نخستین طرح رمزگذاری متقارن گزاره‌ای را بر اساس ضرب داخلی بردار کلمه جستجو و کلمه رمزشده استخراجی از فایل، معرفی کردند که در آن هر کلمه به‌صورت بردار با طول ثابت در نظر گرفته می‌شود و ضرب داخلی بین بردار کلیدواژه جستجو و بردار کلیدواژه در فهرست انجام می‌شود. این طرح نخستین طرح با امنیت کامل است.

تاکنون طرح‌های زیادی مبتنی بر جستجوی یک کلیدواژه ارائه شده است [۱۹, ۲۰, ۲۱, ۲۲] که هر یک بهبودهایی را نسبت به طرح‌های قبلی خود به انجام رساندند.

صراحت کلیدواژه جستجو. به‌منظور نزدیک‌شدن نتایج جستجو به آنچه کاربر مدنظر دارد، به کاربر امکان داده می‌شود تا در قالب‌های مختلف بتواند اهداف خود از جستجو را بهتر انتقال دهند.

[1] False positive rate
[2] sublinear

نخستین طرح جستجوپذیر متقارن با امکان جستجوی چند کلیدواژه غیرعطفی، در سال ۲۰۱۴ توسط کاأ[1] [۲۳] و همکاران پیشنهاد شد. در این طرح که از فهرست مستقیم استفاده می‌شود، هر فهرست و ترپدر، درواقع برداری به طول دیکشنری است که با درنظرگرفتن ترتیبی برای کلیدواژه‌ها در دیکشنری، هر درایه از این بردارها متناظر با یک کلیدواژه از دیکشنری است. در هر درایه از بردار فهرست متناظر با یک فایل، میزان ارتباط آن کلیدواژه با فایل به‌صورت یک عدد قرار داده می‌شود. در هر درایه از بردار ترپدر نیز، درصورتی‌که کلیدواژه متناظر با آن درایه، مورد جستجو قرار گیرد مقدار درایه، عدد یک یا اولویت آن کلیدواژه از دیدگاه کاربر است. بردارهای فهرست و ترپدر با الگوریتم kNN[2] رمزگذاری شده و برای جستجو از ضرب داخلی هر یک از بردارهای فهرست با بردار ترپدر استفاده می‌شود. از ویژگی‌های این طرح، قابلیت رتبه‌بندی نتایج بر اساس میزان ارتباط کلیدواژه‌های جستجو با فایل‌ها و اولویت کاربر در کلیدواژه‌های جستجو است. به این ترتیب سرور می‌تواند با توجه به درخواست کاربر، k فایل نخست که بیش‌ترین ارتباط را با ترپدر دارد، به‌عنوان نتایج جستجو به کاربر برگرداند. این فناوری، سبب کاهش سربار محاسباتی و مخابراتی برای سرور و کاربر می‌شود.

سان[3] و همکاران [۲۴] در سال ۲۰۱۳ طرحی مبتنی بر ساختمان دادهٔ درخت ارائه دادند که دارای قابلیت جستجوی چند کلیدواژه و رتبه‌بندی نتایج جستجو است. این طرح از جستجوی یک کاربر پشتیبانی می‌کند و ترپدر ساخته‌شده در آن احتمالی است؛ اما امنیت آن اثبات نشده است.

دو طرح [۲۵] و [۲۶]، نیز از جستجوی چندکلیدواژه با نتایج رتبه‌بندی‌شده، اما مبتنی بر فهرست معکوس پشتیبانی می‌کنند.

گل و همکاران [۲۷]، بر اساس نگاشت دوخطی، نخستین طرح رمزگذاری جستجوپذیر با امکان جستجوی چند کلیدواژه عطفی را معرفی کردند. امنیت این طرح براساس مسئله تصمیم دیفی و هلمن اثبات شده است. از ویژگی‌های طرح گل، تقسیم درخواست جستجو به دو بخش برخط و برون‌خط برای بهبود عملکرد طرح است.

نخستین طرح SSE با امکان جستجوی بولی و بر اساس فهرست مستقیم، توسط موآتاز[4] و شیکفا[5] معرفی شد [۲۸]. موآتاز در این طرح هر کلیدواژه را به‌صورت یک بردار در نظر می‌گیرد و از فرایند گرام اشمیت برای متعامدسازی و متعامد یکه‌سازی بردارها استفاده می‌کند. سرور برای جستجو از ضرب داخلی روی بردارهای متعامد یکه استفاده می‌کند که بسیار کاراست. این طرح، در مدل IND-CKA2 امن است و ترپدر در آن به‌صورت تصادفی تولید می‌شود.

کورساوا[6] [۲۹] در سال ۲۰۱۴، طرحی با قابلیت جستجوی چند کلیدواژه عطفی و قابلیت درخواست بولی معرفی کرد که در آن از فهرست مستقیم استفاده شده است. در این طرح، برای نخستین‌بار، فرمول جستجو (AND، OR و ...)، مخفی نگه داشته می‌شود.

کش[7] و همکاران [۳۰] در سال ۲۰۱۳، ویژگی‌های طرح موآتاز را در طرحی براساس فهرست معکوس و چندکلیدواژهٔ عطفی پیاده کردند. جارکی[8] و همکاران [۳۱] در همان سال، طرح کش را از یک کاربر به چند کاربر و چندکلیدواژهٔ عطفی ارتقا داد.

در سال ۲۰۱۵، فابر و همکاران [۳۲] روش کش با امکان پشتیبانی از یک کاربر و قابلیت جستجوی بولی چند کلیدواژهٔ عطفی را به پشتیبانی از چند کاربر و جستجو با قابلیت جستجوی بازه، جستجوی زیررشته[9]، جستجوی وایدکارد[10] و جستجوی عبارت[11] ارتقا دادند. ساختار این طرح مبتنی بر فهرست معکوس و درخت است. همچنین این طرح قابلیت حذف و اضافه و یا اصلاح فایل را دارد.

دیمتزیس[12] [۳۳] در سال ۲۰۱۶ یک طرح پویا با قابلیت جستجوی بازه و چندکلیدواژه در محدودهٔ مشخص را معرفی کرد که در آن نشت اطلاعات را از طریق ساختار فهرست کاهش می‌دهد و میزان خطای پاسخ مثبت در آن کمینه می‌شود. در این طرح ساختمان دادهٔ به‌کاررفته در فرایند جستجو، درخت است.

پویایی و وارسی‌پذیری. اغلب طرح‌های مبتنی بر فهرست مستقیم، به‌خاطر استقلال فهرست‌های متناظر با هر فایل، به‌طور ذاتی قابلیت پویایی را دارند؛ اما برای بهره‌برداری از سرعت بیشتر جستجو، از فهرست معکوس استفاده می‌شود

[1] Cao
[2] k-Nearest Neighborhood
[3] Sun
[4] Moataz
[5] Shikfa
[6] Kurosawa
[7] Cash
[8] Jarecki
[9] substring
[10] wildcard
[11] phrase
[12] Demertzis

که قابلیت به‌روزرسانی در آن‌ها وجود ندارد. کامارا و همکاران [۳۴]، در سال ۲۰۱۲ نخستین روش رمزگذاری جستجوپذیر متقارن پویا (DSSE[1]) را براساس فهرست معکوس طراحی کردند. آن‌ها یک تعریف رسمی از امنیت DSSE ارائه دادند. درواقع امنیت تطابقی مقاوم در مقابل حملات کلیدواژه انتخابی را از SSE به DSSE منتقل کردند؛ اما در این طرح، مقدار درهم‌سازی‌شده کلیدواژه‌های موجود در فایل‌های اضافه‌شده یا حذف‌شده آشکار می‌شود. کامارا و پاپامانتو[2] [۳۵]، در طرح خود این نقطه‌ضعف را برطرف کردند.

درهمین‌اواخر ژنگ و همکاران [۳۶]، طرحی با قابلیت جستجوی یک کلیدواژه ارائه دادند که از فهرست معکوس و جستجوی دودیی استفاده می‌کند. به این منظور ترتیبی برای کلیدواژه‌ها در دیکشنری درنظر گرفته می‌شود. ترپدر و هر یک از فهرست‌ها به‌صورت برداری نزدیک به‌طول دیکشنری در نظر گرفته می‌شود. برای رمزکردن فهرست، از تابع درهم‌ساز برای هر کلیدواژه در فهرست، تصادفی‌سازی مقادیر تابع درهم‌ساز و سپس از نگاشت دوخطی استفاده می‌شود. ترپدر نیز به روشی مشابه با فهرست به‌دست می‌آید. در الگوریتم جستجو، یک تساوی از نگاشت‌های دوخطی، با ورودی فهرست و ترپدر، مورد بررسی قرار می‌گیرد. امنیت این طرح، امنیت کامل است.

چای[3] و گونگ[4] [۳۷] نخستین طرح رمزگذاری جستجوپذیر متقارن را با قابلیت وارسی نتایج (VSSE[5]) در حضور سرور شبه‌راست‌گو پیشنهاد کردند. کوروساوا و اوتاکی [۱۳] با استفاده از کد احراز اصالت، یک طرح VSSE در حضور خدمات‌دهندهٔ بدخواه معرفی کردند.

سان و همکاران [۳۸] یک طرح VSSE پیشنهاد دادند که امکان به‌روزرسانی مجموعه‌داده در آن وجود دارد و کاربر داده می‌تواند از جستجوی عطفی استفاده کند. در این طرح، وارسی نتایج می‌تواند توسط خود کاربر یا هر مرجع امن عمومی دیگری انجام شود. این طرح در حضور خدمات‌دهندهٔ بدخواه در مدل UC، امن است.

بوست و همکاران [۱۷] یک طرح VSSE پویا را با امنیت روبه‌جلو در حضور خدمات‌دهندهٔ بدخواه در حوزهٔ تئوری و عملی مورد مطالعه قرار دادند و در پایان ثابت کردند که طرح پیشنهادشده دارای امنیت کامل است.

[1] Dynamic Symmetric Searchable Encryption
[2] Papamanthou
[3] Chai
[4] Gong

درهمین‌اواخر یک طرح پویا با قابلیت وارسی‌پذیری نتایج، توسط ویوک و همکاران [۳۹] معرفی شد. این طرح با هدف نزدیک‌کردن ویژگی‌های مهاجم به شرایط واقعی، خدمات‌دهندهٔ ابر را شبه‌راست‌گو اما کنجکاو در نظر می‌گیرد و در چنین شرایطی نشان می‌دهد که طرح پیشنهادشده، در مدل Dynamic IND-CKA2، امن است.

جدول (۱)، مقایسه‌ای بین پژوهش‌های انجام‌شده از سال ۲۰۱۴ تا ۲۰۱۸ در هفت ویژگی طرح‌های SSE را به نمایش گذاشته است.

(جدول-۱): طبقه‌بندی مقالات با موضوع SSE بین سال‌های ۲۰۱۴ تا ۲۰۱۸

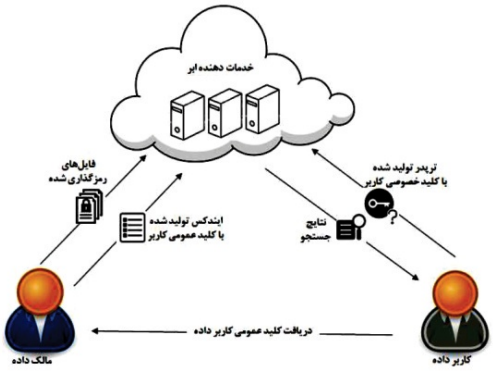
	۲۰۱۴	۲۰۱۵	۲۰۱۶	۲۰۱۷	۲۰۱۸
صراحت کلیدواژه جستجو	[۲۵، ۲۳، ۴۰، ۲۹، ۴۲، ۴۱، ۴۳، ۴۴]	[۳۲، ۴۵، ۳۸، ۴۶]	[۲۶، ۴۷، ۴۸، ۳۳، ۴۹، ۵۰، ۵۱، ۵۲]	[۵۳، ۵۴، ۵۵، ۵۶]	[۵۷، ۵۸، ۵۹، ۶۰]
رتبه‌بندی نتایج	[۲۳، ۴۰، ۲۵، ۴۲، ۴۳، ۴۴]	[۴۶]	[۲۶، ۵۰، ۵۱، ۴۸، ۵۲]	[۵۳، ۵۴، ۵۶]	[۵۷، ۵۸، ۵۹، ۶۰]
پویایی	[۶۱، ۶۲، ۶۳، ۶۴، ۶۲]	[۳۸]	[۲۶، ۱۷، ۶۵]	[۶۶، ۶۷، ۶۸، ۶۹]	[۳۹، ۱۸، ۵۷، ۷۰، ۶۰، ۷۱، ۷۲، ۷۳]
وارسی‌پذیری	[۴۲، ۷۴، ۴۴]	[۳۸، ۷۵، ۷۶، ۷۷]	[۱۷، ۵۱]	[۶۹]	[۳۹، ۶۰، ۷۳]
امنیت کامل	-	-	[۳۶]	-	-
امنیت‌روبه‌جلو	[۱۶]	-	[۱۷، ۶۵]	[۶۶، ۶۷، ۶۸]	[۱۸، ۷۰، ۵۷، ۷۱، ۷۲]
امنیت روبه‌عقب	-	-	-	[۶۶]	-

۳- رمزگذاری کلید عمومی با جستجوی کلیدواژه

رمزگذاری کلید عمومی با جستجوی کلیدواژه[6] (PEKS) برای نخستین‌بار در سرویس رایانامه به‌کار رفت تا رایانامه‌هایی که شامل کلیدواژهٔ تعیین‌شده هستند، بدون آشکارشدن محتوای رایانامه، قابل شناسایی باشند. کارایی این قابلیت در حوزه‌های دیگر سبب شد، پژوهش‌ها درخصوص این اولیه گسترش یابد.

[5] Verifiable Symmetric Searchable Encryption
[6] Public key Encryption with Keyword Search

نخستین اولیه رمزگذاری کلید عمومی با جستجوی کلیدواژه، با استفاده از رمزگذاری شناسهٔ مبنا[1]، توسط بونه و همکاران [۴] در سال ۲۰۰۴ ساخته شد. عبدالله و همکاران[2] [۷۸]، بازگشت‌پذیری[3] را در PEKS تعریف کردند و یک تبدیل از رمزگذاری شناسهٔ مبنا به PEKS ارائه دادند. شکل (۴)، سناریوی PEKS را به تصویر می‌کشد.



(شکل-۴): مدل سامانهٔ رمزگذاری جستجوپذیرنامتقارن

در این اولیه رمزنگاری، مانند رمزگذاری جستجوپذیر متقارن، شامل چهار الگوریتم زمان چندجمله‌ای (احتمالی) است:

- الگوریتم راه‌اندازی (Setup) که با دریافت پارامتر امنیتی و پارامترهای عمومی طرح، کلید عمومی و کلید خصوصی را تولید می‌کند.
- الگوریتم رمزگذاری (Enc) که به‌کمک پارامترهای عمومی و با دریافت فهرست رمزگذاری‌نشده و کلید عمومی، فهرست رمزگذاری‌شده را بر می‌گرداند.
- الگوریتم تولید ترپدر (Trap) که با دریافت کلیدواژهٔ جستجو، کلید خصوصی و پارامترهای عمومی ترپدر را بر می‌گرداند.
- الگوریتم جستجو(search) که با دریافت ترپدر، فهرست رمزگذاری‌شده و پارامترهای عمومی، فایل‌های به‌دست‌آمده را در صورت وجود بر می‌گرداند.

بونه و همکاران [۴] طرح خود را مبتنی بر مسئله دیفی[4] و هلمن[5] به‌صورت زیر مطرح کردند:

فرض کنیم G_1 و G_2، دو گروه دوری از مرتبه p و $e: G_1 \times G_1 \longrightarrow G_2$، یک نگاشت دوخطی بین این دو گروه باشد، به‌طوری‌که در سه شرط زیر صدق کند:

- **محاسبه‌پذیری:** به‌ازای هر دو عضو در G_1، الگوریتم زمان چندجمله‌ای وجود داشته باشد، که نگاشت دوخطی این دو عضو را محاسبه کند.
- **دوخطی‌بودن:** به‌ازای هر دو عدد صحیح $r, s \epsilon [1, p]$، $e(g^r, g^s) = e(g, g)^{rs}$.
- **غیر تبهگن‌بودن:** اگر g در G_1 باشد، $e(g, g)$ در G_2 قرار گیرد.

مقدار پارامتر p با توجه به پارامتر امنیتی مشخص می‌شود. برای تعریف الگوریتم‌های PEKS، به دو تابع هش $h_1: \{0,1\}^* \longrightarrow G_1$ و $h_2: G_2 \longrightarrow \{0,1\}^{\log p}$ نیاز داریم. الگوریتم‌های PEKS، به صورت زیر تعریف می‌شود:

Setup(n): با توجه به پارامترهای امنیتی، این الگوریتم گروه‌های G_1 و G_2 از مرتبه p را تولید می‌کند. همچنین الگوریتم، عدد تصادفی $\alpha \epsilon \mathbb{Z}_p^*$ و $g \epsilon G_1$ را انتخاب و خروجی $A_{pub} = [g, h = g^\alpha]$ و $A_{pir} = \alpha$ را بر می‌گرداند.

Enc(A_{pub}, W): ابتدا به‌ازای عدد تصادفی $r \epsilon \mathbb{Z}_p^*$، مقدار $t = e(h_1(W), h^r) \epsilon G_2$ را محاسبه کرده و دوتایی $[g^r, H_2(t)]$ را به‌عنوان خروجی بر می‌گرداند.

Trap(A_{priv}, W): مقدار $T = H_1(W)^\alpha$ را به‌عنوان خروجی بر می‌گرداند.

search($A_{pub}, [g^r, H_2(t)]$,T): اگر $H_2\big(e(T, h^r)\big) = H_2(t)$ خروجی الگوریتم، یک و در غیر این صورت صفر است.

به این ترتیب در رمزگذاری جستجوپذیر، مبتنی بر کلید عمومی هر کاربر می‌تواند محتوای جستجوپذیر را با کلید عمومی یک موجودیت تولید کند و تنها همان موجودیت امکان ساخت ترپدر و جستجو بین داده‌های رمزگذاری‌شده را دارد. می‌توان دید که هرگاه W را در الگوریتم ترپدر، به‌عنوان شناسه در نظر بگیریم، این اولیه در قالب یک طرح IBE، قابل اجرا است.

۳-۱- امنیت در PEKS

به‌طورکلی می‌توان گفت که با توجه به روش رمزگذاری کلیدواژه‌ها و ساخت فهرست در طرح‌های امنPEKS، هیچ اطلاعاتی از راه فهرست رمزگذاری‌شده نشت نمی‌کند. به بیان دیگر، امنیت کلیدواژه‌ها در فهرست به امنیت اولیه رمزگذاری کلید عمومی برمی‌گردد. آنچه در PEKS، یک چالش جدی به‌حساب می‌آید، نقص ذاتی آن در حفظ حریم خصوصی کلیدواژه‌های جستجو است [۷۹]؛ زیرا هر مهاجم می‌تواند

[1] Identity based encryption
[2] Abdalla
[3] Consistency
[4] Diffie
[5] Helman

کلیدواژه مورد نظر خود را با کلید عمومی کاربر مورد نظر رمزگذاری کند و با داشتن ترپدر تولیدشده توسط همان کاربر، حضور یا عدم حضور یک کلیدواژه در آن را مورد بررسی قرار دهد. از آنجا که تعداد کلیدواژه‌ها در دیکشنری محدود است و از یک تعداد چندجمله‌ای تجاوز نمی‌کند، مهاجم می‌تواند حمله دیکشنری برون‌خط[1] یا حملهٔ حدس کلیدواژه[2] را انجام دهد [۸۰]. به همین علت در PEKS، از کانال امن برای انتقال ترپدر استفاده می‌شود و این امر یک محدودیت برای طرح به‌حساب می‌آید. علاوه‌براین، استفاده از کانال امن، طرح را تنها در مقابل حملات مهاجم خارجی مقاوم می‌کند و سرور ابر که درواقع مهاجم داخلی محسوب می‌شود، می‌تواند حمله حدس کلیدواژه را روی ترپدر دریافتی انجام دهد.

برای حل این مسئله چندین راه‌کار تا به حال ارائه شده است که در ادامه به‌اختصار به آن‌ها می‌پردازیم.

در سال ۲۰۰۹، تنگ[3] و چین[4] [۸۱]، روشی را با نام رمزگذاری کلید عمومی با جستجوی کلیدواژهٔ ثبت‌نام‌شده[5] معرفی کردند که در آن هرگاه مالک داده بخواهد روی اطلاعات خود با کلید عمومی موجودیت دیگر رمزگذاری قابل جستجو انجام دهد، باید نخست کلیدواژه‌های مورد نظر خود را به دارندهٔ کلید عمومی ارسال کرده و یک پیش ایندکس را دریافت کند؛ سپس به‌کمک آن، ایندکس بسازد. این کار سبب می‌شود هر کسی قادر به تولیدکلیدواژهٔ رمزگذاری‌شده و بررسی ترپدر نباشد؛ اما با این حال، در این طرح نیز نیاز به کانال امن وجود دارد؛ زیرا ممکن است، مالک دادهٔ کنجکاو، با سرور تبانی کند و پیش‌ایندکس‌ها را به او بدهد و یا ترپدر را از سرور بگیرد. در این شرایط برای ارسال پیش‌ایندکس و ترپدر نیاز به دو کانال جداگانه هست.

در طرحی دیگر، سرور دارای کلید عمومی و خصوصی است که به‌ترتیب در تولید ایندکس و الگوریتم جستجو به‌کار می‌رود [۸۲]. برخلاف ادعای نویسنده که این طرح را بی‌نیاز از کانال امن می‌داند، یاو[6] [۸۳] نشان می‌دهد که ترپدر تولیدشده حتی برای مهاجم خارجی قابل تمیز است و این طرح در حمله حدس کلیدواژه برون‌خط، درواقع هیچ مزیتی نسبت به PEKS ندارد.

رئی[7] و همکاران [۸۴]، مفهوم جدیدی به نام تمایزناپذیری ترپدر در PEKS را مطرح کردند و نشان دادند که در مدل اوراکل تصادفی، تمایزناپذیری ترپدر معادل با امنیت در مقابل حمله حدس برون‌خط کلیدواژه است. منظور از تمایزناپذیری ترپدر، شانس ناچیز مهاجم در تمایزدادن بین ترپدرهای معادل با دو کلیدواژهٔ انتخابی توسط خود مهاجم است. در این طرح مانند طرح قبل، سرور دارای کلید عمومی و خصوصی است؛ با این تفاوت که کلید عمومی سرور، در تولید ترپدر نیز به‌کار می‌رود.

نیشیوکا[8] [۸۵] در سال ۲۰۱۲، مفهوم جدیدی را با نام حریم خصوصی کامل کلیدواژه[9] (PKP) تعریف کرد که علاوه‌بر تمایزناپذیری در ایندکس، حریم خصوصی الگوی جستجو را حفظ می‌کند و به این ترتیب، امنیت ترپدر را فرمول‌بندی کرد. او با فرض وجود یک تابع به‌طورکامل یک‌طرفه، نشان داده است که طرح او تمایزناپذیری ترپدر را دارد.

در سال ۲۰۱۳، زو[10] و همکاران [۸۶] برای حفظ حریم خصوصی کلیدواژهٔ جستجو، طرحی ارائه دادند که در آن کاربر، دو ترپدر فازی و دقیق می‌سازد و ترپدر فازی را برای جستجو به سرور ارسال می‌کند. پس از ارسال نتایج توسط سرور، کاربر روی نتایج جستجو با ترپدر دقیق، الگوریتم جستجو را اجرا می‌کند و نتایج دقیق را به‌دست می‌آورد. هدف از ارائهٔ این طرح، جلوگیری از حدس برون‌خط کلیدواژه توسط مهاجم داخلی و خارجی است. اگرچه با این روش محتوای دقیق ترپدر شناخته نمی‌شود، اما مجموعه کلیدواژه‌های احتمالی را کاهش می‌دهد. بنابراین این طرح به‌طور کامل جلوی این حمله را نمی‌گیرد و تنها اثر آن را تعدیل می‌کند.

شاو[11] و یانگ[12] [۸۷] یک طرح مقاوم در مقابل حدس کلیدواژه توسط مهاجم داخلی با استفاده از گواهی‌نامهٔ زیرساخت کلید عمومی و امضای دیجیتال معین معرفی کردند. چن[13] و همکاران [۸۸] نیز طرحی با همین ویژگی و با فرض وجود دو سرور که با یکدیگر تبانی نمی‌کنند، پیشنهاد کردند.

هوآنگ[14] و لی[15] [۸۹] یک طرح رمزگذاری کلید عمومی احراز اصالت‌شده با جستجوی کلیدواژه در مدل اوراکل تصادفی معرفی کردند که در آن ارسال‌کننده یا مالک داده،

[1] Offline dictionary attack
[2] keyword guessing attack
[3] Tang
[4] Chen
[5] Publickey encryption with registered keyword search
[6] Yau
[7] Rhee

[8] Nishioka
[9] Keywordprivacy
[10] Xu
[11] Shao
[12] Yang
[13] Chen
[14] Huang
[15] Li

کلیدواژه رمزگذاری‌شده را با استفاده از کلید خصوصی خود، احراز اصالت می‌کند. کاربر نیز در ساخت ترپدر از کلیدخصوصی خود و کلید عمومی مالک داده، استفاده می‌کند. به این ترتیب تنها مالک داده می‌تواند ایندکس بسازد و نه سرور به‌عنوان مهاجم داخلی و نه مهاجم خارجی نمی‌توانند ایندکس بسازند و ترپدر را مورد بررسی قرار دهند. این ایده برای نخستین‌بار در [۹۰] معرفی شد و یکی از محدودیت‌های آن لزوم ارسال ترپدر متفاوت توسط کاربر برای جستجو روی مجموعه‌داده‌ها با مالکان متفاوت است. برخلاف ادعای نویسندگان، طرح [۹۰]، در مقابل مهاجم داخلی امن نیست [۹۱].

وو[1] و همکاران [۹۲] نیز طرحی را بر اساس فرض دیفی–هلمن تعریف کردند که در مقابل حملهٔ حدسِ کلیدواژهٔ داخلی و حملهٔ تزریقِ فایل، مقاوم است.

در [۹۳]، یک طرح مقاوم در برابر حملهٔ تزریق[2] پیشنهاد شده است که در آن، مهاجم تعدادی فایل همراه با فهرست متناظر با هر یک از آن‌ها تولید و برای سرور ارسال می‌کند. بنابر هدف مهاجم از حمله، این فهرست‌ها می‌توانند، شامل مجموعه کلیدواژه‌های مورد نظر مهاجم یا کل کلیدواژه‌های ممکن باشند. بعد از ارسال ترپدر توسط کاربر، سرور فایل‌های مرتبط با آن را به او بر می‌گرداند. درصورتی‌که محتوای ترپدر، یکی از کلمات موردنظر مهاجم باشد، فایل‌های رمزگذاری‌شده توسط او، در نتایج جستجو قرار دارد و برای مهاجم بر خط، قابل مشاهده است؛ به‌علاوه در این حمله، مهاجم می‌تواند با استفاده از حمله مرد میانی[3]، در فهرست ارسال‌شده توسط مالک داده، تغییراتی ایجاد کند و سبب اختلال در سامانه شود. این حمله، درواقع تعمیمی از حملهٔ حدس کلیدواژه برخط[4] و نوع خاصی از حملهٔ تزریقِ فایل است.

۳-۲- قابلیت‌ها در PEKS

صراحت کلیدواژهٔ جستجو. در سال ۲۰۰۷، سه طرح به ترتیب با قابلیت جستجو کلیدواژه‌های عطفی [۹۴]، کلیدواژه‌های عطفی و بازه [۹۵] و کلیدواژه‌های عطفی، بازه و زیرمجموعه [۹۶] پیشنهاد شد. درهمین‌اواخر میائو[5] و همکاران [۹] بر اساس فرض تصمیم دیفی و هلمن، طرحی با امکان جستجوی کلیدواژه‌های عطفی پیشنهاد کردند که نیاز به کانال امن ندارد.

پویایی و وارسی‌پذیری. از دیگر قابلیت‌های یک طرح PEKS، امکان به‌روزرسانی و بررسی نتایج جستجو است. تنها طرح با قابلیت به‌روزرسانی توسط بونه و همکاران در سال ۲۰۰۶ ارائه شد [۹۷]. در این طرح با استفاده از بلوم‌فیلتر و رمزگذاری هوموروفیک، امکان جستجوی یک کلیدواژه و حذف فایل برای دارندهٔ کلید خصوصی وجود دارد. هم‌چنین الگوی جستجو در این طرح، از دید سرور ابر پنهان است.

اغلب طرح‌های موجود با قابلیت وارسی نتایج در طرح‌های SSE مطرح شده و تعداد کمی‌از طرح‌های مبتنی بر PEKS، این قابلیت را دارند. در سال ۲۰۱۶، یک طرح رمزگذاری جستجوپذیر با نام PVSAE بر مبنای فهرست معکوس [۹۸] ارائه شد. براساس این طرح، دو طرح دیگر با نام‌های l-PVSAE و 3-PVSAE معرفی شد که هر دوی آن‌ها دارای هر دو ویژگی امنیتی مهم در PEKS هستند؛ اما طرح دوم در مقایسه با طرح‌های موجود با ویژگی‌های مشابه، دارای کارایی بیشتر، و سربار ذخیره‌سازی کمتر است و نیاز کمتری به تعامل کاربر و سرور ابر دارد. در سال ۲۰۱۷، طرحی با نام VCSE [۸۶]، پیشنهاد شد که در آن کاربر می‌تواند صحت و تمامیت نتایج را مورد بررسی قرار دهد.

در بین تمام طرح‌های PEKS، طرح [۱۰۷]، به‌خاطر استفاده از درخت، دارای اندازه فهرست و زمان جستجوی لگاریتمی است و بنابراین کارایی بیشتری دارد. جدول (۲)، پژوهش‌های انجام‌شده را در PEKS، از سال ۲۰۱۴ تا سال ۲۰۱۸ نشان می‌دهد.

(جدول-۲): : دسته‌بندی مقالات با موضوع PEKS بین سال‌های ۲۰۱۴ تا ۲۰۱۸

	۲۰۱۴	۲۰۱۵	۲۰۱۶	۲۰۱۷	۲۰۱۸
صراحت در کلیدواژه‌های جستجو	-	-	[۹۸, ۹۹]	[۱۰۰, ۱۰۱]	[۱۰۲]
پویایی	-	-	-	-	-
وارسی پذیری	-	-	[۹۸]	[۱۰۳, ۱۰۱]	[۱۰۴]
امن در حضور مهاجم خارجی	[۱۰۵, ۱۰۶]	[۸۷, ۸۸, ۱۰۷, ۱۰۸]	[۹۸]	[۱۰۹, ۱۰۱]	[۱۱۰]
امن در حضور مهاجم داخلی	-	[۹۰]	[۱۱۱]	[۸۹, ۱۱۲]	[۱۱۳, ۱۱۴]
امنیت رو به جلو	-	-	-	[۸۹]	[۱۱۵]

[1] Wu
[2] Injection attack
[3] Man in the middel
[4] Online keyword geussing attack
[5] Miao

۳-۳- رمزگذاری مبتنی بر ویژگی با جستجوی کلیدواژه

رمزگذاری مبتنی بر ویژگی با جستجوی کلیدواژه[1] (ABKS) یک اولیه رمزگذاری جستجوپذیر است که به‌کارگیری آن، مالک داده را قادر می‌سازد تا بدون تعامل با کاربران داده، اجازه دسترسی دقیق[2] را برای آن‌ها تعیین کند؛ هم‌چنین در سامانه‌های مبتنی بر ویژگی چند مالک داده می‌توانند داده‌های خود را رمزگذاری کنند و چند کاربر روی آن‌ها عمل جستجو را انجام دهند. ایده اولیه انتقال رمزگذاری مبتنی بر ویژگی به رمزگذاری جستجوپذیر برای نخستین‌بار در [۱۱۶] مطرح شد. رمزگذاری مبتنی بر ویژگی با جستجوی کلیدواژه، به دو روش، قابل پیاده‌سازی است. در روش نخست که آن را KP-ABKS[3] می‌نامند، ویژگی‌ها در ایندکس و قوانین دسترسی در کلید خصوصی کاربران تعبیه شده است. در روش دوم با نام CP-ABKS[4]، ایندکس، مشخص‌کنندهٔ قوانین دسترسی و کلید خصوصی کاربران، تعیین‌کنندهٔ ویژگی‌ها هستند. در هر دو روش، کاربر داده درصورتی می‌تواند روی داده‌های رمزگذاری‌شده جستجو کند که ویژگی‌ها در قوانین دسترسی صدق کنند. درABKS با یک‌بار رمزگذاری داده‌ها، اجازهٔ دسترسی به تمام کاربرانی داده می‌شود که دارای ویژگی‌های مورد نظر هستند؛ درحالی‌که در رمزگذاری PEKS، اجازه دسترسی فقط به یک نفر یا یک موجودیت داده می‌شود. در ادامه الگوریتم‌های لازم در این اولیه رمزگذاری شرح داده می‌شود:

- الگوریتم راه‌اندازی (Setup) که با دریافت پارامتر امنیتی، پارامترهای عمومی طرح و کلید خصوصی اصلی را تولید می‌کند؛
- الگوریتم تولید کلید که با دریافت کلید خصوصی اصلی و مجموعه ویژگی‌های کاربر (در CP-ABKS) یا قوانین دسترسی (در KP-ABKS)، کلید خصوصی مختص کاربر را بر می‌گرداند؛
- الگوریتم رمزگذاری (Enc) که کلیدواژه را دریافت کرده و معادل رمزگذاری‌شدهٔ آن را تولید می‌کند؛
- الگوریتم تولید ترپدر (Trap) که با دریافت کلیدواژه جستجو، کلید خصوصی و پارامترهای عمومی، ترپدر را بر می‌گرداند؛
- الگوریتم جستجو (search) که با دریافت ترپدر، فهرست رمزگذاری‌شده و پارامترهای عمومی، اگر ویژگی‌ها در قوانین دسترسی صدق کند و محتوای ترپدر و فهرست، یکسان باشد، خروجی یک و در غیر این صورت صفر را بر می‌گرداند.

۳-۳-۱- امنیت در ABKS

همان‌طورکه بیان شد، در طرح اولیه PEKS، امنیت محتوای ترپدر ضمانت نشده است؛ زیرا مهاجم می‌تواند کلیدواژه مورد نظر خود را رمزگذاری کرده و محتوای ترپدر را با آن مورد بررسی قرار دهد. ژنگ[5] و همکاران [۱۱۶] در مقالهٔ خود مفهومی را با نام اختفای کلیدواژه[6] تعریف کردند. اختفای کلیدواژه به این معناست که احتمال این‌که مهاجم با داشتن ترپدر و متن رمزگذاری‌شده، بتواند محتوای ترپدر را بیابد به میزان ناچیزی با احتمال حدس تصادفی و صحیح کلیدواژه متفاوت باشد.

همان‌طورکه بیان شد، مجموعهٔ کلیدواژه‌ها از اندازه چندجمله‌ای است؛ به همین علت حمله حدس کلیدواژه روی ترپدر در PEKS امکان‌پذیر است. در طرح‌های ABKS، علاوه‌بر محتوای ترپدر، ساختار دسترسی برای حملهٔ برون‌خط حدس کلیدواژه لازم است. در طرح [۱۱۶]، آشکار بودن ساختار دسترسی منجر به این حمله خواهد شد. نیشید[7] و همکاران [۱۱۷] یک اولیه با نام رمزگذاری ویژگی مبنا با قوانین پاره‌ای پنهان[8] پیشنهاد دادند. در چنین سازوکاری قواعد دسترسی برای کسانی که ویژگی‌های آن‌ها در قانون دسترسی صدق نمی‌کند، پنهان است. بر این اساس چیو[9] و همکاران [۱۱۸] در سال ۲۰۱۷، یک اولیه جدید با نام HP-CPABKS[10] معرفی کردند که در آن ساختار دسترسی از دید کاربر غیر مجاز، پنهان است. به این ترتیب با این اولیه، امکان حملهٔ حدس کلیدواژه برون‌خط نیز وجود ندارد. در همین اواخر عامری و همکاران [۱۱۹]، یک اولیه رمزگذاری جدید در مدل استاندارد با نام KP-ABTKS[11] معرفی کردند که در آن توکن تولید شده توسط کاربر مجاز، تنها در یک بازه زمانی مشخص، برای جستجو در فایل های رمزگذاری‌شده معتبر است.

۳-۳-۲- قابلیت‌ها در ABKS

در نخستین طرح ABKS، ژنگ و همکاران [۱۱۶] امکان بررسی صحت و تمامیت نتایج توسط کاربر داده را فراهم

[1] Attribute based encryption with keyword search
[2] Fine-grained access control
[3] Key-policy attribute based encryption with keyword search
[4] Ciphertext-policy attribute based encryption with keyword search

[5] Zheng
[6] Keyword secrecy
[7] Nishide
[8] Attribute-basedencryption with partially hidden policy
[9] Qiu
[10] hidden policy ciphertext-policy attribute-based encryption withkeyword search
[11] Key-Policy Attribute-based Temporary Keyword Search

کردند؛ اما امکان حذف کاربر مجاز در این طرح وجود ندارد. سان و همکاران [۱۲۰] یک طرح ABKS با قابلیت وارسی نتایج پیشنهاد کردند که در آن امکان حذف کاربر مجاز وجود دارد و نتایج جستجو توسط کاربر قابل وارسی است. در این طرح، امکان جستجوی چند کلیدواژه به‌صورت عطفی فراهم شده است. هان[1] و همکاران [۶۳] به‌تازگی یک طرح ABKS با امکان حذف کاربر مجاز معرفی کردند که اندازهٔ متن رمزگذاری‌شدهٔ جستجوپذیر و اندازهٔ ترپدر در آن ثابت است. در طرح‌های قبل اندازهٔ متن رمزگذاری‌شدهٔ جستجوپذیر متناسب با تعداد ویژگی‌ها بود. در این طرح قوانین دسترسی، از درگاه AND و NOT پشتیبانی می‌کند و هزینه رمزگذاری و الگوریتم جستجو ثابت است.

جدول (۳)، پژوهش‌های انجام‌شده درخصوص اولیه ABKS را از سال ۲۰۱۴ تا ۲۰۱۸ نشان می‌دهد.

(جدول-۳): طبقه‌بندی مقالات با موضوع ABKS بین سال‌های ۲۰۱۴ تا ۲۰۱۸

	۲۰۱۴	۲۰۱۵	۲۰۱۶	۲۰۱۷	۲۰۱۸
صراحت در کلیدواژه‌های جستجو	-	-	[۱۲۱, ۱۲۲]	[۱۲۳, ۶۳, ۱۲۴]	[۱۲۵]
وارسی پذیری	[۱۱۶]	-	[۱۲۰]	[۱۲۶]	[۹, ۱۲۷]
قابلیت حذف کاربر مجاز	-	[۱۲۸]	[۱۲۰, ۱۲۹]	[۶۳, ۱۳۰, ۱۲۶]	[۱۲۷]
اختفای کلیدواژه	[۱۳۱, ۱۱۶]	-	[۱۲۲]	-	[۱۱۹]
مخفی نگهداشتن الگو دسترسی	-	-	-	[۱۱۸]	-

۳-۴- وکالت برای رمزگذاری مجدد با جستجوی کلیدواژه

وکالت برای رمزگذاری مجدد با جستجوی کلیدواژه[2] (PRES) به کاربر مجاز اجازه می‌دهد تا با استفاده از وکالت برای رمزگذاری مجدد، امکان جستجو روی داده‌های رمزگذاری‌شده را به کاربر جایگزین (نماینده) منتقل کند. در این سناریو که برای نخستین‌بار توسط شاو[3] و همکاران [۱۳۲] معرفی شد کاربری که هم اجازه دسترسی به داده‌های رمزگذاری‌شده و هم وکالت برای انتقال اجازهٔ جستجو و اجازهٔ دسترسی را دارد، می‌تواند با رمزگذاری مجدد داده‌های رمزگذاری‌شده این امکان را برای کاربران دیگر فراهم کند.

[1] Han
[2] Proxy re-encryption with keyword search
[3] Shao

سه ویژگی برای هر طرح PRES وجود دارد:

۱- جهت انتقال[4]: اگر PRES، یک طرفه[5] باشد، سرور با داشتن کلید رمزگذاری مجدد تنها می‌تواند متن رمزگذاری‌شده را در یک جهت (به‌عنوان‌مثال از آلیس به باب) مجدد رمزگذاری کند؛ اما اگر کلید رمزگذاری مجدد در جهت عکس هم بتواند مجدد رمزگذاری کند، PRES را دو طرفه[6] می‌نامند.

۲- دفعات انتقال[7]: اگر متن رمزگذاری‌شده بتواند چندین‌بار توسط سرور، رمزگذاری مجدد شود (به‌عنوان‌مثال از آلیس به باب و از باب به کارل)، طرح PRES را چندکاره[8] و در غیر این صورت تک‌کاره[9] می‌نامند.

۳- قالب متن رمزگذاری‌شدهٔ انتقال‌یافته[10]: گوییم طرح PRES، شفاف[11] است، اگر متن دوباره رمزگذاری‌شده با کلید رمزگذاری مجدد، به‌لحاظ محاسباتی از رمزگذاری‌شدهٔ همان متن، با کلید خصوصی نماینده، قابل تمایز نباشد.

در ادامه الگوریتم‌های مورد نیاز برای پیاده‌سازی این اولیه بیان می‌شود:

- الگوریتم راه‌اندازی (Setup) که با دریافت پارامتر امنیتی، پارامترهای عمومی‌طرح را تولید می‌کند؛
- الگوریتم تولید کلید (KeyGen) که با دریافت پارامترهای عمومی، کلید عمومی و خصوصی کاربر داده را بر می‌گرداند؛
- الگوریتم تولید مجدد کلید (ReKeyGen) که با دریافت پارامترهای عمومی، کلید خصوصی نخست و کلیدخصوصی دوم یک کلید رمزگذاری مجدد تولید می‌کند؛
- الگوریتم رمزگذاری (Enc) که پارامترهای عمومی، کلید عمومی کاربر و کلیدواژه را دریافت کرده و معادل رمزگذاری‌شدهٔ آن را تولید می‌کند؛
- رمزگذاری مجدد (ReEnc) که پارامترهای عمومی، کلید رمزگذاری مجدد و کلیدواژه رمزگذاری‌شده را دریافت و کلیدواژه رمزگذاری‌شده را برای امکان جستجو توسط کاربر دوم دوباره رمزگذاری می‌کند؛

[4] Transformationdirection
[5] Unidirectional
[6] Bidirectional
[7] Transformationtime
[8] Multi-use
[9] Single-use
[10] Format of the transformed ciphertext
[11] Transparent

- الگوریتم تولید ترپدر (Trap) که با دریافت پارامترهای عمومی، کلیدواژه جستجو و کلید خصوصی، ترپدر را بر می‌گرداند؛
- الگوریتم جستجو (search) که با دریافت ترپدر، کلیدواژه رمزگذاری‌شده و پارامترهای عمومی، اگر محتوای ترپدر و متن رمزگذاری‌شده یکسان باشد؛ خروجی یک و در غیر این صورت صفر را بر می‌گرداند.

۳-۴-۱- امنیت در PRES

امنیت نخستین طرح PRES یک‌طرفهٔ چندکاره، بر اساس مسئله تصمیم دیفی-هلمن در مدل اوراکل تصادفی توسط شاو و همکاران [۱۳۲] ثابت شد. در این طرح، فایل‌ها و کلیدواژه‌ها با یک الگوریتم، رمزگذاری می‌شوند. شاو مدل امنیتی خود را در دو بخش تعریف کرد. در بخش نخست، حریم خصوصی کلیدواژه‌های درخواستی مورد بررسی قرار گرفته است و فرض می‌شود که مهاجم به اوراکل رمزگشا و اوراکل تولید ترپدر دسترسی دارد و تنها ترپدر دو کلیدواژه چالش را در اختیار ندارد. در چنین وضعیتی مهاجم نمی‌تواند با احتمال قابل توجه، کلیدواژه چالش را بدرستی حدس بزند. به این ترتیب ضمانت می‌شود که تنها کسی که ترپدر را دارد، بتواند عملیات جستجو را انجام دهد. در بخش دوم، مهاجم به اوراکل رمزگشا و اوراکل تولید ترپدر دسترسی دارد و تنها متن رمزگذاری‌شدهٔ دو کلیدواژه چالش را نمی‌داند. به هر حال مهاجم نمی‌تواند با داشتن متن رمزگذاری‌شده، تشخیص دهد کدام کلیدواژه چالش رمزگذاری شده است. به این ترتیب می‌تواند اطمینان داشته باشد که تنها دارندهٔ کلید خصوصی می‌تواند متن رمزگذاری‌شده را رمزگشایی کند.

یاو و همکاران [۱۳۳]، یک طرح PRES دوطرفهٔ چندکاره با یک مدل امنیتی در مدل اوراکل تصادفی معرفی کردند که در مقابل حمله حدس کلیدواژه مقاوم است. در این طرح، الگوریتم رمزگذاری فایل‌ها و کلیدواژه‌ها متفاوت است تا با توجه به شرایط، انعطاف‌پذیری در به‌کارگیری طرح وجود داشته باشد. در این مدل، مهاجم بدون داشتن ترپدر کاربر یا نماینده نباید بتواند حدس بزند کدام کلیدواژه، متناظر با ایندکس داده شده است.

۳-۴-۲- قابلیت‌ها در PRES

ونگ و همکاران [۱۳۴] طرحی با نام CPRE-CKS[1] معرفی کردند که امکان جستجوی کلیدواژه عطفی را فراهم می‌کند. در این طرح، کاربر جایگزین می‌تواند تنها متن‌های رمزگذاری شده‌ای را دوباره رمزگذاری کند که شامل کلیدواژه‌های مورد نظر باشند. امنیت این طرح در مدل اوراکل تصادفی، مشابه با طرح شاو است.

شی[2] و همکاران [۱۳۵] با به‌کارگیری رمزگذاری ویژگی مبنا در PRES، طرحی با نام ABRKS[3] معرفی کردند که به مالک داده اجازه می‌دهد با اعمال قوانین کنترل دسترسی، امکان اجازهٔ جستجوی واژگان کلیدی را از طریق داده‌های رمزگذاری‌شدهٔ خود به کاربران مجاز دیگر منتقل کند. این طرح امکان جستجو توسط چند کاربر را برای نخستین بار با اعمال کنترل دسترسی دقیق، در یک طرح PRES فراهم می‌کند. با فرض وجود اوراکل تصادفی، ABRKS در مقابل حمله حدس کلیدواژه مقاوم است.

در سال ۲۰۱۶، یک طرح PRES، با نام Re-dtPECK وابسته به زمان و با قابلیت جستجوی چند کلیدواژه عطفی [۱۳۶] پیشنهاد شد که در آن اجازهٔ جستجو توسط کاربر جایگزین تنها در یک بازهٔ زمانی معتبر است و این بازه زمانی قابل کنترل می‌باشد. این طرح در مقابل مهاجم فعال خارجی امن است و امنیت آن در مدل استاندارد ثابت شده است.

در ادامه به جایگاه رمزگذاری جستجوپذیر در حوزه خدمات الکترونیک سلامت می‌پردازیم که از حوزه‌های پرکاربرد رمزگذاری جستجوپذیر است.

۴- کاربرد رمزگذاری جستجوپذیر در خدمات الکترونیک سلامت

برون‌سپاری داده‌های پزشکی در ابر به‌صورت رمزگذاری‌شده و امکان جستجو روی این داده‌ها، علاوه‌برکاهش هزینه‌های نگهداری پرونده‌های الکترونیک سلامت[4](EHR)، به بیمار اجازه می‌دهد پرونده پزشکی خود را در اختیار پزشکان مورد نظر خود قرار دهد، بدون این‌که اطلاعات شخصی بیمار برای پزشکان یا خدمات‌دهندهٔ ابر، قابل مشاهده باشد. این فناوری سبب می‌شود، بیمار راحت‌تر بتواند با متخصصان مورد نظر ارتباط بر قرار کرده و نیاز خود را برطرف کند.

[1] Constrainedsingle-hop unidirectional proxy re-encryption supporting conjunctive keywords search

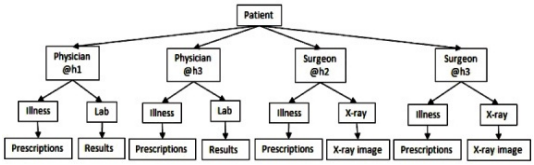
[2] Shi

[3] Attribute-based proxy re-encryption with keyword search

[4] Electronic Health Records

EHR یک مجموعه دیجیتال از اطلاعات شخصی بیمار مانند سن، جنسیت، قومیت، تاریخچهٔ سلامت، داروها، آلرژی‌ها، نتایج آزمایش‌ها، شیوه‌نامهٔ ترخیص از بیمارستان و اطلاعات صورت‌حساب است. اگر این پرونده بین چند متخصص یا خدمات‌دهندگان بهداشت و درمان به اشتراک گذاشته شود هر تغییر در پرونده، مانند آزمایش یا تشخیص جدید، تجویز دارو و غیره، توسط تمام کاربران مجاز قابل مشاهده است. به‌عنوان مثال اگر دو دارو که توسط دو متخصص تجویز شده با یکدیگر تداخل داشته باشند، توسط دیگر خدمات‌دهندگان نیز قابل مشاهده است. بنابراین تشخیص خطا و جلوگیری از آن با سرعت بیشتری انجام می‌شود و دقت درمان افزایش می‌یابد.

با توجه به نیازمندی‌ها در خدمات الکترونیک سلامت، در رمزگذاری ایندکس متناظر با EHR، یکی از دو روش رمزگذاری جستجوپذیر متقارن و رمزگذاری جستجوپذیر نامتقارن به‌کار می‌رود. هرگاه دارندهٔ EHR، تنها قصد ذخیره‌سازی پرونده و جستجو روی آن را داشته باشد، از رمزگذاری جستجوپذیر متقارن استفاده می‌کند؛ اما درصورتی‌که بخواهد به موجودیت‌های دیگر اجازه جستجو بدهد برای راحتی در مدیریت کلید و استفاده از مزایای دیگر، رمزگذاری جستجوپذیر نامتقارن را به‌کار می‌گیرد. شکل (۵) ساختار ایندکس یک EHR را به‌صورت درخت نشان می‌دهد.



(شکل-۵): ایندکس متناظر با یک پرونده الکترونیک سلامت

در این شکل، ریشهٔ درخت یا سطح نخست آن، گرهٔ بیمار است. سطح دوم، نشان‌دهندهٔ کار پزشک برای بیمار و بیمارستانی که بیمار ویزیت شده است، می‌باشد. سطح سوم، به تشخیص، نسخه‌ها و آزمایش‌های انجام‌شده اشاره می‌کند و هر یک از گره‌های سطح سوم، به داده‌های مربوطه متصل است. برای حفظ حریم خصوصی داده‌های بیمار، علاوه‌بر رمزگذاری EHR به یکی از روش‌های معمول رمزگذاری، ایندکس متناظر با آن نیز باید به یکی از روش‌های رمزگذاری جستجوپذیر، رمزگذاری شود. در ساخت ایندکس، گره‌های سطح دوم می‌تواند به‌عنوان کلیدواژه مورد استفاده قرار گیرد.

رمزگذاری EHR با استفاده از SSE، تنها به بیمار (مالک داده) یا هر کاربر مجاز دیگر (پزشک، جراح یا هر موجودیت دیگر) که بیمار کلید خصوصی را به‌طور مستقیم در اختیار او قرار داده است، اجازهٔ جستجو می‌دهد. به‌کارگیری PEKS در رمزگذاری EHR موجب می‌شود، بدون نیاز به تبادل امن کلید بین بیمار و کاربر، بیمار تنها به یک کاربر اجازهٔ جستجو روی داده‌های رمزگذاری‌شدهٔ خود را بدهد. کاربر مجاز می‌تواند با کلید خصوصی خود ترپدر بسازد و روی داده‌های رمزگذاری‌شده عملیات جستجو را انجام دهد. تفاوت به‌کارگیری SSE و PEKS، عدم نیاز به انتقال کلید رمزگشایی و کلید خصوصی برای جستجو بین مالک و کاربر در PEKS و کمتربودن پیچیدگی محاسباتی در SSE است. هرگاه بیمار بخواهد اجازهٔ جستجو روی EHR رمزگذاری‌شدهٔ خود را به بیش از یک کاربر بدهد و در عین حال این کاربران، کاربران مجاز باشند، ایندکس خود را به روش ABKS رمزگذاری می‌کند. رمزگذاری ویژگی مبنا با اعمال کنترل دسترسی بر مبنای ویژگی‌های کاربر، بیمار را قادر می‌سازد که تنها به کاربرانی که ویژگی‌های آن‌ها در قوانین دسترسی صدق می‌کند، اجازهٔ جستجو دهد. هرگاه بیمار قادر به اعمال اجازهٔ دسترسی به کاربران دیگر نبود (به‌عنوان مثال در شرایط بیهوشی یا کما)، کاربر مجاز جایگزین می‌تواند EHR او را دوباره رمزگذاری کند تا با این کار در شرایط اورژانسی، اجازهٔ دسترسی به پرونده بیمار توسط متخصصان دیگر داده شود.

۵- نتیجه‌گیری

در این مقاله رمزگذاری جستجوپذیر به‌عنوان یک روش ضروری در ذخیره‌سازی ابری مورد بررسی قرار گرفت. هر یک از اولیه‌های تعریف‌شده در این روش، محدودیت‌ها و قابلیت‌هایی را به‌همراه دارد که بنابر سناریوی کاربردی، مورد استفاده قرار می‌گیرد. اولیه‌ها در رمزگذاری جستجوپذیر، به دو دستهٔ کلی رمزگذاری جستجوپذیر متقارن و رمزگذاری جستجوپذیر نامتقارن تقسیم می‌شود. پژوهش‌های انجام‌شده در این حوزه، در زمینهٔ بهبود قابلیت‌ها، کارایی و امنیت لازم است. خدمات الکترونیک سلامت، یکی از حوزه‌های مطالعاتی در اینترنت اشیا است که اطلاعات حساسی را شامل می‌شود. در پایان این مقاله، چگونگی به‌کارگیری رمزگذاری جستجوپذیر، به‌عنوان یک الزام برای حفظ حریم خصوصی داده‌های پزشکی، مورد پژوهش قرار گرفت.

۶- مراجع

[1] Song, Dawn Xiaoding and Wagner, David and Perrig, Adrian, "Practical techniques for searches on encrypted data," in *Security and Privacy, 2000. S\&P 2000. Proceedings. 2000 IEEE Symposium on*, 2000.

[2] Goh, Eu-Jin and others, "Secure indexes," *IACR Cryptology ePrint Archive,* vol. 2003, p. 216, 2003.

[3] Curtmola, Reza and Garay, Juan and Kamara, Seny and Ostrovsky, Rafail, "Searchable symmetric encryption: improved definitions and efficient constructions," *Journal of Computer Security,* vol. 19, no. 5, pp. 895--934, 2011.

[4] Boneh, Dan and Di Crescenzo, Giovanni and Ostrovsky, Rafail and Persiano, Giuseppe, "Public key encryption with keyword search," in *International conference on the theory and applications of cryptographic techniques*, 2004.

[5] Liu, Zheli and Weng, Jian and Li, Jin and Yang, Jun and Fu, Chuan and Jia, Chunfu, "Cloud-based electronic health record system supporting fuzzy keyword search," *Soft Computing,* vol. 20, no. 8, pp. 3243--3255, 2016.

[6] Li, Hongwei and Yang, Yi and Dai, Yuanshun and Bai, Jian and Yu, Shui and Xiang, Yong, "Achieving secure and efficient dynamic searchable symmetric encryption over medical cloud data," *IEEE Transactions on Cloud Computing,* 2017.

[7] Ma, Mimi and He, Debiao and Khan, Muhammad Khurram and Chen, Jianhua, "Certificateless searchable public key encryption scheme for mobile healthcare system," *Computers \& Electrical Engineer-ing,* vol. 65, pp. 413--424, 2018.

[8] Liu, Zheli and Weng, Jian and Li, Jin and Yang, Jun and Fu, Chuan and Jia, Chunfu, "Cloud-based electronic health record system supporting fuzzy keyword search," *Soft Computing,* vol. 20, no. 8, pp. 3243--3255, 2016.

[9] Miao, Yinbin and Ma, Jianfeng and Jiang, Qi and Li, Xiong and Sangaiah, Arun Kumar, "Verifiable keyword search over encrypted cloud data in smart city," *Computers \& Electrical Engineering,* vol. 65, pp. 90--101, 2018.

[10] Ferrag, Mohamed Amine and Ahmim, Ahmed, "ESSPR: an efficient secure routing scheme based on searchable encryption with vehicle proxy re-encryption for vehicular peer-to-peer social network," *Telecommunication Systems,* vol. 66, no. 3, pp. 481--503, 2017.

[11] Wen, Mi and Lu, Rongxing and Zhang, Kuan and Lei, Jingsheng and Liang, Xiaohui and Shen, Xuemin, "PaRQ: A privacy-preserving range query scheme over encrypted metering data for smart grid," *IEEE Transactions on Emerging Topics in Computing,* vol. 1, no. 1, pp. 178--191, 2013.

[12] Chang, Yan-Cheng and Mitzenmacher, Michael, "Privacy preserving keyword searches on remote encrypted data," in *International Conference on Applied Crypto-graphy and Network Security*, 2005.

[13] Kurosawa, Kaoru and Ohtaki, Yasuhiro, "UC-secure searchable symmetric encryption," in *International Conference on Financial Cryptography and Data Security*, 2012.

[14] Shen, Emily and Shi, Elaine and Waters, Brent, "Predicate privacy in encryption systems," in *Theory of Cryptography Conference*, 2009.

[15] Zhang, Yupeng and Katz, Jonathan and Papamanthou, Charalampos, "All Your Queries Are Belong to Us: The Power of File-Injection Attacks on Searchable Encryption," in *USENIX Security Symposium*, 2016.

[16] Stefanov, Emil and Papamanthou, Charalampos and Shi, Elaine, "Practical Dynamic Searchable Encryption with Small Leakage.," in *NDSS*, 2014.

[17] Bost, Raphael and Fouque, Pierre-Alain and Pointcheval, David, "Verifiable Dynamic Symmetric Searchable Encryption: Optimality and Forward Security," *IACR Cryptology ePrint Archive,* vol. 2016, p. 62, 2016.

[18] Song, Xiangfu and Dong, Changyu and Yuan, Dandan and Xu, Qiuliang and Zhao, Minghao, "Forward Private Searchable Symmetric Encryption with Optimized I/O Efficiency," *IEEE Transactions on Dependable and Secure Computing,* 2018.

[19] Asharov, Gilad and Naor, Moni and Segev, Gil and Shahaf, Ido, "Searchable symmetric encryption: Optimal locality in linear space via two-dimensional balanced allocations," in *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, 2016.

[20] Kamara, Seny and Papamanthou, Charalam–pos and Roeder, Tom, "Dynamic searchable symmetric encryption," in *Proceedings of the 2012 ACM conference on Computer and communications security*, 2012.

[21] Chase, Melissa and Kamara, Seny, "Structured encryption and controlled dis-closure," in *International Conference on the Theory and Application of Cryptology and Information Security*, 2010.

[22] Van Liesdonk, Peter and Sedghi, Saeed and Doumen, Jeroen and Hartel, Pieter and Jonker, Willem, "Computationally efficient searchable symmetric encryption," in *Workshop on Secure Data Management*, 2010.

[23] Cao, Ning and Wang, Cong and Li, Ming and Ren, Kui and Lou, Wenjing, "Privacy-preserving multi-keyword ranked search over encrypted cloud data," *IEEE Transactions on parallel and distributed systems,* vol. 25, no. 1, pp. 222--233, 2014.

[24] Sun, Wenhai and Wang, Bing and Cao, Ning and Li, Ming and Lou, Wenjing and Hou, Y Thomas and Li, Hui, "Privacy-preserving multi-keyword text search in the cloud supporting similarity-based ranking," in *Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security*, 2013.

[25] Strizhov, Mikhail and Ray, Indrajit, "Multi-keyword similarity search over encrypted cloud data," in *IFIP international information security conference*, 2014.

[26] Xia, Zhihua and Wang, Xinhui and Sun, Xingming and Wang, Qian, "A Secure and Dynamic Multi-Keyword Ranked Search Scheme over Encrypted Cloud Data," *IEEE Trans. Parallel Distrib. Syst.,* vol. 27, no. 2, pp. 340--352, 2016.

[27] Golle, Philippe and Staddon, Jessica and Waters, Brent, "Secure conjunctive keyword search over encrypted data," in *International Conference on Applied Cryptography and Network Security*, 2004.

[28] Moataz, Tarik and Shikfa, Abdullatif, "Boolean symmetric searchable encryption," in *Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security*, 2013.

[29] Kurosawa, Kaoru, "Garbled searchable symmetric encryption," in *International Conference on Financial Cryptography and Data Security*, 2014.

[30] Cash, David and Jarecki, Stanislaw and Jutla, Charanjit and Krawczyk, Hugo and Ro{\c {s}}u, Marcel-C{\u{a}}t{\u{a}}lin and Steiner, Michael, "Highly-scalable searchable symmetric encryption with support for boolean queries," *Advances in cryptology--CRYPTO 2013,* pp. 353--373, 2013.

[31] Jarecki, Stanislaw and Jutla, Charanjit and Krawczyk, Hugo and Rosu, Marcel and Steiner, Michael, "Outsourced symmetric private information retrieval," in *Proceedings of the 2013 ACM SIGSAC conference on Computer \& communications security*, 2013.

[32] Faber, Sky and Jarecki, Stanislaw and Krawczyk, Hugo and Nguyen, Quan and Rosu, Marcel and Steiner, Michael, "Rich queries on encrypted data: Beyond exact matches," in *International Conference on Financial Cryptography and Data Security*, 2015.

[33] Demertzis, Ioannis and Papadopoulos, Stavros and Papapetrou, Odysseas and Deligiannakis, Antonios and Garofalakis, Minos, "Practical private range search revisited," in *Proceedings of the 2016 International Conference on Management of Data*.

[34] Kamara, Seny and Papamanthou, Charalampos and Roeder, Tom, "Dynamic searchable symmetric encryption," in *Proceedings of the 2012 ACM conference on Computer and communications security*, 2012.

[35] Kamara, Seny and Papamanthou, Charalampos, "Parallel and dynamic searchable symmetric encryption," in *International Conference on Financial Cryptography and Data Security*, 2013.

[36] Zhang, Rui and Xue, Rui and Yu, Ting and Liu, Ling, "Dynamic and Efficient Private Keyword Search over Inverted Index--Based Encrypted Data," *ACM Transactions on Internet Technology (TOIT),* vol. 16, no. 3, p. 21, 2016.

and Kolesnikov, Vladimir and Malkin, Tal and Bellovin, Steven M, "Malicious-client security in blind seer: a scalable private DBMS," in *Security and Privacy (SP), 2015 IEEE Symposium on*, 2015.

[46] Fu, Zhangjie and Sun, Xingming and Liu, Qi and Zhou, Lu and Shu, Jiangang, "Achieving efficient cloud search services: multi-keyword ranked search over encrypted cloud data supporting parallel computing," *IEICE Transactions on Communications,* vol. 98, no. 1, pp. 190-200, 2015.

[47] Ishai, Yuval and Kushilevitz, Eyal and Lu, Steve and Ostrovsky, Rafail, "Private large-scale databases with distributed searchable symmetric encryption," in *Cryptographers' Track at the RSA Conference*, 2016.

[48] Orencik, Cengiz and Selcuk, Ayse and Savas, Erkay and Kantarcioglu, Murat, "Multi-Keyword search over encrypted data with scoring and search pattern obfuscation," *International Journal of Information Security,* vol. 15, no. 3, pp. 251-269, 2016.

[49] Sun, Shi-Feng and Liu, Joseph K and Sakzad, Amin and Steinfeld, Ron and Yuen, Tsz Hon, "An efficient non-interactive multi-client searchable encryption with support for boolean queries," in *European symposium on research in computer security*, 2016.

[50] Fu, Zhangjie and Huang, Fengxiao and Sun, Xingming and Vasilakos, Athanasios and Yang, Ching-Nung, "Enabling semantic search based on conceptual graphs over encrypted outsourced data," *IEEE Transactions on Services Computing,* 2016.

[51] Fu, Zhangjie and Ren, Kui and Shu, Jiangang and Sun, Xingming and Huang, Fengxiao, "Enabling personalized search over encrypted outsourced data with efficiency improvement," *IEEE transactions on parallel and distributed systems,* vol. 27, no. 9, pp. 2546-2559, 2016.

[52] Fu, Zhangjie and Wu, Xinle and Guan, Chaowen and Sun, Xingming and Ren, Kui, "Toward Efficient Multi-Keyword Fuzzy Search over Encrypted Outsourced Data with Accuracy Improvement," *IEEE Transactions on Information Forensics and Security,* vol. 11, no. 12, pp. 2706-2716, 2016.

[53] Fu, Zhangjie and Wu, Xinle and Wang, Qian and Ren, Kui, "Enabling central keyword-based semantic extension search over

[37] Chai, Qi and Gong, Guang, "Verifiable symmetric searchable encryption for semi-honest-but-curious cloud servers," in *Communications (ICC), 2012 IEEE International Conference on*, 2012.

[38] Sun, Wenhai and Liu, Xuefeng and Lou, Wenjing and Hou, Y Thomas and Li, Hui, "Catch you if you lie to me: Efficient verifiable conjunctive keyword search over large dynamic encrypted cloud data," in *Computer Communications (INFOCOM), 2015 IEEE Conference on*, 2015.

[39] Vivek, S Sree and Ramasamy, Rajkumar and George, Praveen and Kshatriya, Bharat S Rawal, "Dynamic Verifiable Encrypted Keyword Search," *Journal of Signal Processing Systems,* pp. 1-15.

[40] Xia, Zhihua and Zhu, Yanling and Sun, Xingming and Chen, Lihong, "Secure semantic expansion based search over encrypted cloud data supporting similarity ranking," *Journal of Cloud Computing,* vol. 3, no. 1, p. 8, 2014.

[41] Pappas, Vasilis and Krell, Fernando and Vo, Binh and Kolesnikov, Vladimir and Malkin, Tal and Choi, Seung Geol and George, Wesley and Keromytis, Angelos and Bellovin, Steve, "Blind seer: A scalable private dbms," in *Security and Privacy (SP), 2014 IEEE Symposium on*, 2014.

[42] Sun, Wenhai and Wang, Bing and Cao, Ning and Li, Ming and Lou, Wenjing and Hou, Y Thomas and Li, Hui, "Verifiable privacy-preserving multi-keyword text search in the cloud supporting similarity-based ranking," *IEEE Transactions on Parallel and Distributed Systems,* vol. 25, no. 11, pp. 3025-3035, 2014.

[43] Wang, Bing and Yu, Shucheng and Lou, Wenjing and Hou, Y. Thomas, "Privacy-preserving multi-keyword fuzzy search over encrypted data in the cloud," in *Proceedings - IEEE INFOCOM*, 2014.

[44] Fu, Zhangjie and Shu, Jiangang and Sun, Xingming and Linge, Nigel, "Smart cloud search services: Verifiable keyword-based semantic search over encrypted cloud data," *IEEE Transactions on Consumer Electronics,* vol. 60, no. 4, pp. 762-770, 2014.

[45] Fisch, Ben A and Vo, Binh and Krell, Fernando and Kumarasubramanian, Abishek

Privacy (SP), 2014 IEEE Symposium on, 2014.

[62] Cash, David and Jaeger, Joseph and Jarecki, Stanislaw and Jutla, Charanjit S and Krawczyk, Hugo and Rosu, Marcel-Catalin and Steiner, Michael, "Dynamic searchable encryption in very-large databases: data structures and implementation," in *NDSS*, 2014.

[63] Han, Jinguang and Yang, Ye and Liu, Joseph K and Li, Jiguo and Liang, Kaitai and Shen, Jian, "Expressive attribute-based keyword search with constant-size ciphertext," *Soft Computing,* pp. 1-15, 2017.

[64] Bosch, Christoph and Peter, Andreas and Leenders, Bram and Lim, Hoon Wei and Tang, Qiang and Wang, Huaxiong and Hartel, Pieter and Jonker, Willem, "Distributed searchable symmetric encryp-tion," in *Privacy, Security and Trust (PST), 2014 Twelfth Annual International Con-ference on*, 2014.

[65] Bost, Raphael, "∑ oφoς: Forward Secure Searchable Encryption," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016.

[66] Bost, Rapha{\"e}l and Minaud, Brice and Ohrimenko, Olga, "Forward and backward private searchable encryption from constrained cryptographic primitives," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017.

[67] Wang, Xingchen and Zhao, Yunlei, "File-injection Attack and Forward Security for Order-revealing Encryption.," *IACR Cryptology ePrint Archive,* vol. 2017, p. 1086, 2017.

[68] Kim, Kee Sung and Kim, Minkyu and Lee, Dongsoo and Park, Je Hong and Kim, Woo-Hwan, "Forward secure dynamic searchable symmetric encryption with efficient updates," in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, 2017.

[69] Rajkumar, Ramasamy and Vivek, S.Sree and George, Praveen and S. Rawal Kshatriya Bharat , "Dynamic Verifiable Encrypted Keyword Search Using Bitmap Index and Homomorphic MAC," *Journal of Signal Processing Systems,* pp. 1-15, 2017.

encrypted outsourced data," *IEEE Transactions on Information Forensics and Security,* vol. 12, no. 12, pp. 2986-2997, 2017.

[54] Chen, Lanxiang and Qiu, Linbing and Li, Kuan-Ching and Shi, Wenbo and Zhang, Nan, "DMRS : an efficient dynamic multi-keyword ranked search over encrypted cloud data," *Soft Computing,* vol. 21, no. 16, pp. 4829-4841, 2017.

[55] Yang, Yang and Liu, Ximeng and Deng, Robert, "Multi-user Multi-Keyword Rank Search over Encrypted Data in Arbitrary Language," *IEEE Transactions on Dependable and Secure Computing,* no. 1, pp. 1-1, 2017.

[56] Sharma, Saumya and Bhagtani, Amrita and Agarwal, Parth and Mohite, Ankit, "N-Gram Fuzzy Keyword Search On Encrypted User Data in Cloud," *International Journal of Open Information Technologies,* vol. 5, no. 9, pp. 21-26, 2017.

[57] Du, Minxin and Wang, Qian and He, Meiqi and Weng, Jian, "Privacy-Preserving Indexing and Query Processing for Secure Dynamic Cloud Storage," *IEEE Transactions on Information Forensics and Security,* vol. 13, no. 9, pp. 2320-2332, 2018.

[58] Raghavendra, S and Girish, S and Geeta, CM and Buyya, Rajkumar and Venugopal, KR and Iyengar, SS and Patnaik, LM, "Split keyword fuzzy and synonym search over encrypted cloud data," *Multimedia Tools and Applications,* vol. 77, no. 8, pp. 10135-10156, 2018.

[59] Fu, Zhangjie and Xia, Lili and Sun, Xingming and Liu, Alex X and Xie, Guowu, "Semantic-Aware Searching Over Encrypted Data for Cloud Computing," *IEEE Transactions on Information Forensics and Security,* vol. 13, no. 9, pp. 2359-2371, 2018.

[60] Kabir, Tasnim and Adnan, Muhammad Abdullah, "A dynamic searchable encryption scheme for secure cloud server operation reserving multi-keyword ranked search," in *2017 4th International Conference on Networking, Systems and Security (NSysS)*, 2017.

[61] Naveed, Muhammad and Prabhakaran, Manoj and Gunter, Carl A, "Dynamic searchable encryption via blind storage," in *Security and*

[70] Etemad, Mohammad and K{\"u}p{\c {c}}{\"u}, Alptekin and Papa-manthou, Charalampos and Evans, David, "Efficient dynamic searchable encryp-tion with forward privacy," *Proceedings on Privacy Enhancing Technologies,* vol. 2018, no. 1, pp. 5-20, 2018.

[71] Ozmen, Muslum Ozgur, "Forward-Private Dynamic Searchable Symmetric Encryption with Efficient Search," in *2018 IEEE International Conference on Communications (ICC)*, 2018.

[72] Ocansey, Selasi Kwame and Ametepe, Wolali and Li, Xiao Wei and Wang, Changda, "Dynamic searchable encryption with privacy protection for cloud comput-ing," *International Journal of Communi-cation Systems,* vol. 31, no. 1, p. e3403, 2018.

[73] Vivek , S. Sree and Ramasamy, Rajkumar and George Praveen and S. Rawal Kshatriya, Bharat, "Dynamic Verifiable Encrypted Keyword Search Using Bitmap Index," *J Sign Process Syst,* 2017.

[74] "Verifiable Searchable Symmetric Encryption from Indistinguishability Obfuscation Categories and Subject Descriptors," in *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*, 2015.

[75] Kurosawa, Kaoru, and Yasuhiro Ohtaki, "How to Construct UC-Secure Searchable Symmetric Encryption Scheme," *IACR Cryptology ePrint Archive,* pp. 1-33, 2015.

[76] Taketani, Shunsuke, and Wakaha Ogata, "Improvement of UC secure searchable symmetric encryption scheme," in *Interna-tional Workshop on Security*, 2015.

[77] Cheng, Rong, Jingbo Yan, Chaowen Guan, Fangguo Zhang, and Kui Ren, "Verifiable searchable symmetric encryption from indistinguishability obfuscation," in *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security*, 2015.

[78] Abdalla, Michel and Bellare, Mihir and Catalano, Dario and Kiltz, Eike and Kohno, Tadayoshi and Lange, Tanja and Malone-Lee, John and Neven, Gregory and Paillier, Pascal and Shi, Haixia, "Searchable encryp-tion revisited: Consistency properties, rela-tion to anonymous IBE, and extensions," in *Annual International Cryptology Con-ference*, 2005.

[79] Nasrollah Pakniat, "Public key encryption with keyword search and keyword guessing attack: a survey," in *13th International Iranian Society of Cryptology Conference on Information Security and Cryptology (ISCISC)*, 2016.

[80] Byun, Jin Wook and Rhee, Hyun Suk and Park, Hyun-A and Lee, Dong Hoon, "Off-line keyword guessing attacks on recent keyword search schemes over encrypted data," in *Workshop on Secure Data Management*, 2006.

[81] Tang, Qiang and Chen, Liqun, "Public-key encryption with registered keyword search," in *European Public Key Infrastructure Workshop*, 2009.

[82] Baek, Joonsang and Safavi-Naini, Reihaneh and Susilo, Willy, "Public key encryption with keyword search revisited," in *Interna-tional conference on Computational Science and Its Applications*, 2008.

[83] Yau, Wei-Chuen and Heng, Swee-Huay and Goi, Bok-Min, "Off-line keyword guessing attacks on recent public key encryption with keyword search schemes," in *International Conference on Autonomic and Trusted Computing*, 2008.

[84] Rhee, Hyun Sook and Park, Jong Hwan and Susilo, Willy and Lee, Dong Hoon, "Trapdoor security in a searchable public-key encryption scheme with a designated tester," *Journal of Systems and Software,* vol. 83, no. 5, pp. 763--771, 2010.

[85] Nishioka, Mototsugu, "Perfect keyword privacy in PEKS systems," in *International Conference on Provable Security*, 2012.

[86] Xu, Peng and Jin, Hai and Wu, Qianhong and Wang, Wei, "Public-key encryption with fuzzy keyword search: A provably secure scheme under keyword guessing attack," *IEEE Transactions on computers,* vol. 62, no. 11, pp. 2266--2277, 2013.

[87] Shao, Zhi-Yi and Yang, Bo, "On security against the server in designated tester public key encryption with keyword search," *Information Processing Letters,* vol. 115, no. 12, pp. 957-961, 2015.

[88] Chen, Rongmao and Mu, Yi and Yang, Guomin and Guo, Fuchun and Wang, Xiaofen, "A new general framework for secure public key encryption with keyword search," in

searchable encryption service framework for outsourced encrypted data," in *Web Services (ICWS), 2016 IEEE International Conference on*, 2016.

[99] Liang, Kaitai and Huang, Xinyi and Guo, Fuchun and Liu, Joseph K, "Privacy-Preserving and Regular Language Search over Encrypted Cloud Data," *IEEE Transac-tions on Information Forensics and Security,* vol. 11, no. 10, pp. 2365-2376, 2016.

[100] Lu, Yang and Wang, Gang and Li, Jiguo and Shen, Jian, "Efficient designated server identity-based encryption with conjunctive keyword search," *Annals of Telecommunica-tions,* vol. 72, no. 5-6, pp. 359-370, 2017.

[101] Miao, Yinbin and Ma, Jianfeng and Liu, Ximeng and Jiang, Qi and Zhang, Junwei and Shen, Limin and Liu, Zhiquan, "VCKSM: Verifiable conjunctive keyword search over mobile e-health cloud in shared multi-owner settings," *Pervasive and Mobile Computing,* vol. Elsevier, pp. 205-219, 2017.

[102] Yang, Yang and Zheng, Xianghan and Rong, Chunming and Guo, Wenzhong, "Efficient Regular Language Search for Secure Cloud Storage," *IEEE Transactions on Cloud Computing,* 2018.

[103] Miao, Yinbin and Ma, Jianfeng and Liu, Ximeng and Zhang, Junwei and Liu, Zhiquan, "VKSE-MO: verifiable keyword search over encrypted data in multi-owner settings," *Science China Information Sciences,* vol. 60, no. 12, p. 122105, 2017.

[104] Miao, Yinbin and Ma, Jianfeng and Liu, Ximeng and Liu, Zhiquan and Shen, Limin and Wei, Fushan, "VMKDO: Verifiable multi-keyword search over encrypted cloud data for dynamic data-owner," *Peer-to-Peer Networ-king and Applications,* vol. 11, no. 2, pp. 287-297, 2018.

[105] Arriaga, Afonso and Tang, Qiang and Ryan, Peter, "Trapdoor privacy in asym- metric searchable encryption schemes," in *Inte-rnational Conference on Cryptology in Africa*, 2014.

[106] Wu, Tsu-Yang and Tsai, Tung-Tso and Tseng, Yuh-Min, "Efficient searchable id-based encryption with a designated server," *annals of telecommunications-annales des t{\'e}l{\'e} communications,* vol. 69, no. 7-8, pp. 391-402, 2014.

Australasian Conference on In-formation Security and Privacy, 2015.

[89] Huang, Qiong and Li, Hongbo, "An efficient public-key searchable encryption scheme secure against inside keyword guessing attacks," *Information Sciences,* vol. 403, pp. 1--14, 2017.

[90] Li, Chun-Ta and Lee, Chin-Wen and Shen, Jau-Ji, "An extended chaotic maps-based keyword search scheme over encrypted data resist outside and inside keyword guessing attacks in cloud storage services," *Nonlinear Dynamics,* vol. 80, no. 3, pp. 1601-1611, 2015.

[91] Noroozi, Mahnaz and Eslami, Ziba and Pakniat, Nasrollah, "Comments on a chaos-based public key encryption with keyword search scheme," *Nonlinear Dynamics,* pp. 1-6, 2018.

[92] Wu, Libing and Chen, Biwen and Zeadally, Sherali and He, Debiao, "An efficient and secure searchable public key encryption scheme with privacy protection for cloud storage," *Soft Computing,* pp. 1--12, 2018.

[۹۳] عرب‌نوری، آرین، ابراهیمی آتانی، رضا, "تحلیل امنیتی الگوریتم های رمزنگاری قابل جستجو نامتقارن در برابر حملات تزریق in ", *پانزدهمین کنفرانس بین‌المللی انجمن رمز ایران، تهران*، 1397.

[94] Hwang, Yong Ho, and Pil Joong Lee, "key encryption with conjunctive keyword search and its extension to a multi-user system," in *International conference on pairing-based cryptography*, 2007.

[95] Shi, Elaine and Bethencourt, John and Chan, TH Hubert and Song, Dawn and Perrig, Adrian, "Multi-dimensional range query over encrypted data," in *Security and Privacy, 2007. SP'07. IEEE Symposium on*, 2007.

[96] Boneh, Dan and Waters, Brent, "Conjunctive, subset, and range queries on encrypted data," in *Theory of Cryptography Conference*, 2007.

[97] Boneh, Dan and Kushilevitz, Eyal and Ostrovsky, Rafail and Skeith, William E, "Public key encryption that allows PIR queries," in *Annual International Cryptology Conference*, 2007.

[98] Zhang, Rui and Xue, Rui and Yu, Ting and Liu, Ling, "PVSAE: a public verifiable

[107] Zhang, Rui and Xue, Rui, "Efficient keyword search for public-key setting," in *Military Communications Conference, MILCOM 2015-2015 IEEE*, 2015.

[108] Emura, Keita and Miyaji, Atsuko and Rahman, Mohammad Shahriar and Omote, Kazumasa, "Generic constructions of secure-channel free searchable encryption with adaptive security," *Security and communication networks,* vol. 8, no. 8, pp. 1547-1560, 2015.

[109] Deng, Zuojie and Li, Kenli and Li, Keqin and Zhou, Jingli, "A multi-user searchable encryption scheme with keyword authoriza-tion in a cloud storage," *Future Generation Computer Systems,* vol. 72, pp. 208-218, 2017.

[110] Zhang, Xiaojun and Xu, Chunxiang, "Trapdoor Security Lattice-Based Public-Key Searchable Encryption with a Designated Cloud Server," *Wireless Personal Communications,* vol. 100, no. 3, pp. 907-921, 2018.

[111] Chen, Rongmao and Mu, Yi and Yang, Guomin and Guo, Fuchun and Wang, Xiaofen, "Dual-server public-key encryption with keyword Search for secure cloud storage," *IEEE transactions on information forensics and security,* vol. 11, no. 4, pp. 789-798, 2016.

[112] Jiang, Peng and Mu, Yi and Guo, Fuchun and Wen, Qiao-Yan, "Private Keyword-Search for Database Systems Against Insider Attacks," *Journal of Computer Science and Technology,* vol. 32, no. 3, pp. 599-617, 2017.

[113] Wu, Libing and Chen, Biwen and Zeadally, Sherali and He, Debiao, "An efficient and secure searchable public key encryption scheme with privacy protection for cloud storage," *Soft Computing,* pp. 1-12, 2018.

[114] Sun, Lixue and Xu, Chunxiang and Zhang, Mingwu and Chen, Kefei and Li, Hongwei, "Secure searchable public key encryption against insider keyword guessing attacks from indistinguishability obfuscation," *Science China Information Sciences,* vol. 61, no. 3, p. 038106, 2018.

[115] Wu, Libing and Chen, Biwen and Choo, Kim-Kwang Raymond and He, Debiao, "Efficient and secure searchable encryption protocol for cloud-based Internet of Things," *Journal of Parallel and Distributed Computing,* pp. 152-161, 2018.

[116] Zheng, Qingji and Xu, Shouhuai and Ateniese, Giuseppe, "VABKS: verifiable attribute-based keyword search over outsourced encrypted data," in *Infocom, 2014 proceedings IEEE*, 2014.

[117] Nishide, Takashi, Kazuki Yoneyama, and Kazuo Ohta, "Attribute-based encryption with partially hidden encryptor-specified access structures," in *International Con-ference on Applied Cryptography and Network Security*, 2008.

[118] Qiu, Shuo and Liu, Jiqiang and Shi, Yanfeng and Zhang, Rui, "Hidden policy ciphertext-policy attribute-based encryption with keyword search against keyword guessing attack 可抵抗关键字猜测攻击的隐藏策略属性基可检索加密方案," *Science China Information Sciences,* vol. 60, no. 5, p. 052105, 2017.

[119] Ameri, Mohammad Hassan and Delavar, Mahshid and Mohajeri, Javad and Salmasi-zadeh, Mahmoud, "A Key-Policy Attribute-Based Temporary Keyword Search scheme for Secure Cloud Storage," *IEEE Transactions on Cloud Computing,* 2018.

[120] Sun, Wenhai and Yu, Shucheng and Lou, Wenjing and Hou, Y Thomas and Li, Hui, "Protecting your right: verifiable attribute-based keyword search with fine-grained owner-enforced search authorization in the cloud," *IEEE Transactions on Parallel and Distributed Systems,* vol. 27, no. 4, pp. 1187-1198, 2016.

[121] Liu, Zheli and Weng, Jian and Li, Jin and Yang, Jun and Fu, Chuan and Jia, Chunfu, "Cloud-based electronic health record system supporting fuzzy keyword search," *Soft Computing,* vol. 20, no. 8, pp. 3243-3255, 2016.

[122] Miao, Yinbin and Ma, Jianfeng and Liu, Ximeng and Wei, Fushan and Liu, Zhiquan and Wang, Xu An, "m2-ABKS: Attribute-Based Multi-Keyword Search over Encrypted Personal Health Records in Multi-Owner Setting," *Journal of medical systems,* vol. 40, no. 11, p. 246, 2016.

[123] Zhu, Binrui and Sun, Jiameng and Qin, Jing and Ma, Jixin, "Fuzzy matching: multi-authority attribute searchable encryption without central authority," *Soft Computing,* pp. 1-10, 2017.

[124] Yang, Yang and Zheng, Xianghan and Chang, Victor and Ye, Shaozhen and Tang, Chunming, "Lattice assumption based fuzzy information retrieval scheme support multi-user for secure multimedia cloud," *Multi-media Tools and Applications,* pp. 1-15, 2017.

[125] Cui, Hui and Wan, Zhiguo and Deng, Robert H and Wang, Guilin and Li, Yingjiu, "Efficient and Expressive Keyword Search Over Encrypted Data in the Cloud," *IEEE Transactions on Dependable and Secure Computing,* vol. 18, no. 3, pp. 409-422, 2018.

[126] Ye, Jun and Wang, Jianfeng and Zhao, Jiaolian and Shen, Jian and Li, Kuan-Ching, "Fine-grained searchable encryption in multi-user setting," *Soft Computing,* vol. 21, no. 20, pp. 6201-6212, 2017.

[127] Yang, Yang and Liu, Ximeng and Zheng, Xianghan and Rong, Chunming and Guo, Wenzhong, "Efficient Traceable Authorization Search System for Secure Cloud Storage," *IEEE Transactions on Cloud Computing,* 2018.

[128] Shi, Yanfeng and Zheng, Qingji and Liu, Jiqiang and Han, Zhen, "Directly revocable key-policy attribute-based encryption with verifiable ciphertext delegation," *Information Sciences,* vol. 295, pp. 221-231, 2015.

[129] Wang, Shangping and Zhang, Xiaoxue and Zhang, Yaling, "Efficiently Multi-User Searchable Encryption Scheme with Attri-bute Revocation and Grant for Cloud Storage," *PloS one,* vol. 11, no. 11, p. e0167157, 2016.

[130] Wang, Guangbo and Wang, Jianhua, "Research on Ciphertext-Policy Attribute-Based Encryption with Attribute Level User Revocation in Cloud Storage," *Mathematical Problems in Engineering,* vol. 2017, 2017.

[131] Liu, Pengliang and Wang, Jianfneg and Ma, Hua and Nie, Haixin, "Efficient verifiable public key encryption with keyword search based on KP-ABE," in *Broadband and Wireless Computing, Communication and Applications (BWCCA), 2014 Ninth International Conference on*, 2014.

[132] Shao, Jun and Cao, Zhenfu and Liang, Xiaohui and Lin, Huang, "Proxy re-encryption with keyword search," *Infor-mation Sciences,* vol. 180, no. 13, pp. 2576-2587, 2010.

[133] Yau, Wei-Chuen and Phan, Raphael C-W and Heng, Swee-Huay and Goi, Bok-Min, "Proxy re-encryption with keyword search: new definitions and algorithms," in *Security Technology, Disaster Recovery and Business Continuity*, Springer, 2010, pp. 149-160.

[134] Wang, Xu An and Huang, Xinyi and Yang, Xiaoyuan and Liu, Longfei and Wu, Xuguang, "Further observation on proxy re-encryption with keyword search," *Journal of Systems and Software,* vol. 85, no. 3, pp. 643-654, 2012.

[135] Shi, Yanfeng and Liu, Jiqiang and Han, Zhen and Zheng, Qingji and Zhang, Rui and Qiu, Shuo, "Attribute-based proxy re-encryption with keyword search," *PloS one,* vol. 9, no. 12, p. e116325, 2014.

[136] Yang, Yang and Ma, Maode, "Conjunctive Keyword Search With Designated Tester and Timing Enabled Proxy Re-Encryption Function for E-Health Clouds," *IEEE Trans. Information Forensics and Security,* vol. 11, no. 4, pp. 746-759, 2016.

انیسه نجفی دانشجوی دکترای جبر دانشگاه شاهد در تهران است. حوزه پژوهشی مورد علاقه او، رمزنگاری است.

مجید بیات دکترای خود را از گروه علوم ریاضی و کامپیوتر دانشگاه خوارزمی دریافت کرده و در حال حاضر استادیار و عضو هیئت علمی گروه مهندسی کامپیوتر دانشگاه شاهد در تهران است. حوزهٔ پژوهشی مورد علاقهٔ او، پروتکل‌های رمزنگاری، شبکه‌های هوشمند و امنیت IOT است.

سید حمید حاج سیدجوادی مدرک کارشناسی، کارشناسی ارشد و دکترای خود را از دانشگاه امیرکبیر دریافت کرده و در حال حاضر، دانشیار و عضو هیئت علمی گروه علوم ریاضی و کامپیوتر دانشگاه شاهد در تهران است. زمینه پژوهشی مورد علاقه ایشان جبر، امنیت و شبکه‌های بی‌سیم است