دوفصل‌نامه علمی منادی امنیت فضای تولید و تبادل اطلاعات (افتا)
سال ۱۴۰۳، جلد ۲۲، شماره ۱ (۱۵-۳۷)
https://monadi.isc.org.ir

مروری بر روش‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها و بررسی چارچوب کلی آنها

سعید بنائیان‌فر* و مریم رجب‌زاده عصار

گروه مخابرات، دانشکده مکانیک، برق و کامپیوتر، دانشگاه آزاد اسلامی، واحد علوم و تحقیقات تهران، ایران

اطلاعات مقاله

تاریخچه مقاله:
تاریخ دریافت: ۱۸ دی ۱۴۰۲
تاریخ پذیرش: ۲۶ اردیبهشت ۱۴۰۳
انتشار آنلاین: ۶ خرداد ۱۴۰۳

کلمات کلیدی:
امنیت و حریم خصوصی
زنجیرهٔ قالب‌ها
پروتکل‌های ممیزی داده
صحت داده
وارسی پذیری عمومی

نوع مقاله: مروری

چکیده

برون‌سپاری داده‌ها به مراکز قابل اعتماد برای نگه‌داری، محافظت و دسترس‌پذیر بودن داده‌ها یک راه ساده و کم هزینه است و نیازی به داشتن زیرساخت‌های فیزیکی، سخت‌افزاری، نرم‌افزاری و منابع انسانی ندارد. اما اتفاقات دنیای واقعی و تحقیقات اخیر نشان داده‌اند که حتی مراکز قابل اعتماد نیز می‌توانند از اعتماد کاربران سوء استفاده کنند. به طور مثال، ۱) در داده‌هایی که در اختیار دارند تغییر ایجاد کنند، ۲) آنها را حذف کنند و یا ۳) موقتاً و یا دائماً از دسترس خارج کنند. روش‌های ممیزی داده این اطمینان را به مالکان داده می‌دهند که داده ثبت شده در پایگاه‌داده همان دادهٔ ارسال شده توسط کاربر است و تغییرات ایجاد شده در آن را آشکار می‌کند. اما فقط مشکل اول را حل می‌کنند. معرفی زنجیرهٔ قالب‌ها به عنوان یک فناوری جدید که دارای ویژگی‌های جذابی از جمله شفافیت، تغییرناپذیری و خودمختاری بود، سبب شد تا مشکلات بسیاری از سامانه‌ها که نیاز به ویژگی‌های ذکر شده را دارند حل شوند. در این مقاله، پس از مرور و بررسی چندین معماری و پروتکل ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها، یک چارچوب کلی ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها را بررسی می‌کنیم. در نهایت مقایسه‌ای بین کارهای بررسی شده ارائه می‌دهیم و برخی افق‌های آینده این حوزه را مشخص می‌کنیم.

۱ مقدمه

امروزه با گسترش شبکه‌های رایانه‌ای و افزایش ارتباطات راه دور و به تناسب آنها افزایش کاربران بدخواه نیاز به ممیزی داده بیشتر حس شده است به طوری که کاربران عادی شبکه نیز می‌توانند برای حصول اطمینان از عدم تغییر داده‌هایشان و ثبت موفقیت آمیز آنها از ساز و کارهای ممیزی داده استفاده کنند.

ذخیره‌سازی دارایی‌های دیجیتال یکی از مسائلی است که شرکت‌ها و سازمان‌ها به آن توجه ویژه‌ای دارند به گونه‌ای که داده را می‌توان یکی از باارزش‌ترین دارایی آنها در نظر گرفت. نحوه ذخیره‌سازی دارایی‌ها و استفاده از آنها می‌تواند ارتباط مستقیمی با کیفیت خدمت رسانی، پاسخگویی و رضایت مشتریان و کارمندان آن سازمان داشته باشد. امروزه بسیاری از شرکت‌ها و سازمان‌های بزرگ همچنان از روش سنتی ذخیره‌سازی استفاده می‌کنند به گونه‌ای که کلیه اطلاعات و دارایی‌ها در یک پایگاه‌داده قرار دارند و دسترسی به آنها فقط از طریق پایگاه داده مشخص شده امکان‌پذیر است. روش دیگر ذخیره‌سازی توسعه روش متمرکز است به گونه‌ای که در آن برای مقابله با حوادث و اتفاقات سایبری و یا طبیعی پایگاه داده به چند بخش مختلف تقسیم شود تا در زمان بروز مشکل در بخشی، سایر بخش‌ها بتوانند پاسخگوی مشتریان باشند و به خدمت رسانی ادامه دهند [۱].

برای ثبت اطلاعات در پایگاه‌های داده با مدیریت متمرکز یا همکاری

*نویسنده مسئول.
آدرس‌های رایانامه: saeed.banaeian.far@gmail.com (سعید بنائیان‌فر)، m.r.asaar@iau.ac.ir (مریم رجب‌زاده عصار)

با شرکت‌های بزرگ به ناچار باید به سیاست‌ها و مواضع آنها احترام گذاشت و نظرات تحمیلی آنها را پذیرفت. این امر باعث می‌شود مالکان بتوانند با وضع قوانین لحظه‌ای و با سلیقهٔ شخصی کاربران را ناراضی کنند و از وابستگی و اعتماد آنها سوء استفاده کنند. در نتیجه این رفتارها، سامانه‌های با مدیریت متمرکز یا خصوصی می‌توانند کمتر مورد توجه کاربران قرار بگیرد. در مقابل سامانه‌هایی که مبتنی بر اجماع [1] و رأی اکثریت کار می‌کنند محبوبیت بیشتری خواهند داشت [۲]. جذابیت این سامانه‌های مبتنی بر اجماع از آنجایی بیشتر است که اختیارات مدیر سامانه کمتر می‌شود و رای اعضای حاضر نیز می‌تواند در سیاست گذاری‌ها و اقدامات تاثیر گذار باشد. در این نوع سامانه‌ها امکان تقلب و تخلف مدیران و افراد داخلی به حداقل می‌رسد و در نتیجه اعتماد کاربران و اعضا بالا می‌رود.

ممیزی داده [2] مبتنی بر اینترنت می‌تواند چالش‌هایی از جنس حملات سایبری یا چالش‌های مشترک با سامانه‌های متمرکز داشته باشد. این چالش‌ها می‌توانند اعتماد کاربران به سامانه و راه‌های ارتباطی را کاهش دهند و منجر به کاهش تعداد کاربران و ارائه دهندگان این خدمات شوند. برای جلوگیری از این موضوع شرکت‌ها و سازمان‌های ارائه دهنده این نوع خدمت‌ها تلاش می‌کنند تا ساز و کاری قابل اعتماد برای کاربران را فراهم کنند.

در سال ۲۰۰۸ فناوری دفتر کل توزیع شده [3] به صورت عملیاتی تحت پروژهٔ بیتکوین توسط یک نویسنده ناشناس (با نام مستعار ساتوشی ناکاموتو) پیاده سازی شد. در این پروژه از مفهومی با عنوان دفتر کل توزیع شده استفاده شد که تمام سوابق تراکنش‌های مالی (نقل و انتقالات بیتکوین بین حساب‌های کاربری یا کلیدهای عمومی کاربران) پس از تأیید توسط اکثریت اعضای شبکه در آن ثبت و در اختیار کاربران قرار می‌گرفت. در این ساختار هر تراکنش مرتبط با تراکنش قبلی تأیید و ثبت می‌شود، در نتیجه آن ارتباط بین داده‌های ثبت شده ایجاد می‌شود به طوری که می‌توان گفت آخرین قالب شامل تعدادی تراکنش یا یک نظر [4] ثبت شده به صورت یک رشته در دفتر کل به نوعی مرتبط با اولین قالب ثبت شده در زنجیرهٔ قالب‌ها [5] است. مرتبط بودن تمام قالب‌های تولید شده و یا تراکنش‌های انجام شده مفهومی به عنوان زنجیرهٔ قالب‌ها را مطرح می‌کند [۳].

در زنجیرهٔ قالب‌ها اگر یک دادهٔ ثبت شده از انواع مختلف (به عنوان مثال، داده متنی یا حتی داده‌های چند رسانه‌ای) تغییر کند، تمام اطلاعات ثبت شده بعدی آن نیز تغییر خواهد کرد و آن تغییر بر همگان آشکار می‌شود. لذا اطلاعات ثبت شده در زنجیرهٔ قالب‌ها پس از ثبت، دیگر قابل تغییر نخواهند بود و این امر با توافق اکثریت اعضای شبکه امکان‌پذیر است (که البته احتمال انجام این توافق بسیار اندک است). در ادامه تعریف دقیق‌تر از زنجیرهٔ قالب‌ها و ساختار آن ارائه می‌شود.

مفهوم ممیزی داده و روش‌های مختلف آن سالهاست که مورد استفاده قرار می‌گیرد و قبل از پیاده‌سازی آن در فضای دیجیتال هم مورد استفاده بوده است. در گذشته بیشترین موارد استفاده آن در کاربردهای مالی بوده است، ولی در دو دهه اخیر جامعیت بیشتری پیدا کرده است به طوری که در ارسال داده‌های شخصی کاربران، دستگاه‌های هوشمند و اطلاعات پزشکی هم مورد استفاده قرار گرفته است [۴].

انگیزه مقاله در این مقاله برخی از موارد مهم در حوزه پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها بررسی می‌شوند. موارد مورد توجه در این مقاله ۱) گمنامی [6] و حفظ حریم خصوصی کاربران شبکه و حفظ محرمانگی پیام‌های تبادل شده، ۲) در نظر گرفتن مدیر شبکه به عنوان فرد بدخواه که می‌تواند برخی امور را تحت کنترل خود قرار دهد و از ثبت و تأیید پیام‌ها جلوگیری کند، ۳) همکاری افراد مخرب (حتی مدیر) در شبکه، و ۴) جلوگیری از پردازش خارج از زنجیره ثبت یا ذخیره‌سازی پیام‌های اشتباه یا هرز (گزارش، نظر، تراکنش‌های مالی، چند رسانه‌ای، یا هر موضوع دیگر) بر روی پایگاه داده یا دفتر کل توزیع شده، هستند.

هدف این تحقیق معرفی و ارائهٔ تعریفی عمومی از پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها است. سپس تحلیل و بررسی کارهای انجام شده در سال‌های اخیر که متمرکز روی موارد فوق در شبکه‌های مبتنی بر زنجیرهٔ قالب‌ها بوده‌اند. در نهایت بیان برخی راهکارها که می‌توانند در ارائه خدمات و بهبود موارد فوق (اعم از امنیت، کارایی، هزینه محاسبات، پهنای باند اشغال شده و ...) در پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها تأثیرگذار باشند است تا بتواند گامی مؤثر در این راه برداشته باشد.

سازمان مقاله ادامه این مقاله به شرح زیر سازمان‌دهی شده است. بخش ۲ تعریف‌های و پیشنیازهای این تحقیق شامل تعریف پروتکل‌های ممیزی داده، زنجیرهٔ قالب‌ها، روش‌های ذخیره‌سازی داده و تعریف پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها را ارائه می‌دهد. بخش ۳ به مرور و بررسی چندین طرح معماری و پروتکل ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها می‌پردازد. بخش ۴ چارچوب کلی این پروتکل‌ها را بررسی و تحلیل می‌کند. بخش ۵ و ۶ به ترتیب مقایسه‌ای بین کارهای انجام شده اخیر و جمع‌بندی مقاله را ارائه می‌دهند.

۲ تعریف‌ها و پیش‌نیازها

این بخش به بیان تعریف‌ها و پیش‌نیازهای مقاله می‌پردازد.

۱.۲ پروتکل‌های ممیزی داده

موضوع اصلی که در این مقاله روی آن متمرکز هستیم، پروتکل‌های ممیزی داده است. ممیزی داده یکی از راه‌های اطمینان از صحت داده‌های ثبت/ارسال شده است. در پروتکل‌های ممیزی داده، صحت داده توسط اشخاص سوم نامعتمد که کاربران ممیزی نامیده می‌شوند بررسی می‌شود. آنها صحت داده ارسالی را با استفاده از شواهدی که در دست دارند بررسی

[1] Consensus [2] Data auditing [3] Distributed ledger technology
[4] Comment [5] Blockchain
[6] Anonymity

می‌کنند و در نهایت نتیجه را تأیید یا رد می‌کنند [۵، ۶].

۲.۲ زنجیرهٔ قالب‌ها

زنجیرهٔ قالب‌ها را می‌توان یکی از بارزترین مصادیق ذخیره‌سازی توزیع شده دانست. زنجیرهٔ قالب‌ها مجموعه‌ای از قالب‌های به هم پیوسته است که هر تغییری در قالب‌های قبلی منجر به ایجاد تغییر در قالب‌های بعدی/جدید می‌شود. فناوری زنجیرهٔ قالب‌ها با تکیه بر چند مفهوم ساده ارائه شده است [۳، ۷، ۸] که در ادامه هر کدام از آنها تشریح می‌شوند. لازم به ذکر است استفاده از الگوریتم‌های اجماع، ساز و کار انتشار خطا و ارتباط بین داده‌های ثبت شده را می‌توان از مهمترین تفاوت‌های زنجیرهٔ قالب‌ها با روش ذخیره‌سازی توزیع شده دانست.

همبندی نظیر به نظیر در این نوع همبندی تمام گره‌های شبکه امکان ارتباط با یکدیگر را بدون دخالت و بدون نیاز به حضور یک نهاد مرکزی دارند و هر کاربر یا عضو شبکه می‌تواند مستقل از دیگر اعضا اطلاعاتی را برای سایرین ارسال کند. زنجیرهٔ قالب‌ها به عنوان یک پایگاه داده توزیع شده با همبندی نظیر به نظیر[1] این امکان را فراهم می‌کند تا کاربرانی که قصد ارسال داده به پایگاه داده یا دریافت اطلاعات از آن را دارند به سادگی و با کمک نزدیک‌ترین گره این کار را انجام دهند.

الگوریتم‌های اجماع ثبت هر گونه اطلاعات در زنجیرهٔ قالب‌ها پس از حصول اجماع در شبکه انجام می‌شود. با توجه به نوع زنجیرهٔ قالب‌ها و سیاست‌گذاری‌های آن از الگوریتم اجماع مناسب استفاده می‌شود [۹، ۱۰].

انتشار خطا مفهوم ارتباط قالب‌های داده‌های ثبت شده در زنجیرهٔ قالب‌ها با استفاده از ساز و کار انتشار خطا و ایجاد چکیدهٔ قالب قبلی در قالب فعلی ایجاد می‌شود. بدین ترتیب در صورتی که پس از ایجاد هر قالب از داده و ثبت آن هر یک از قالب‌های قبلی تغییر کنند، این تغییر تا قالب فعلی انتشار می‌یابد و این تغییر به سادگی قابل وارسی است.

تابع چکیده‌ساز[2]، به عنوان اصلی ترین بخش هر زنجیرهٔ قالب‌ها، به عنوان یک نگاشت غیر یک‌به‌یک تعریف می‌شود. این نگاشت یک طول نامعین و به دلیل ملاحظات عملی کوچک‌تر از یک مقدار مشخص از داده را به عنوان ورودی دریافت می‌کند و طول محدود و مشخص (به طور مثال ۲۵۶بیت) را به عنوان خروجی برمی‌گرداند [۱۱]. تابع چکیده‌ساز را می‌توان به صورت $h : \{0,1\}^* \rightarrow \{0,1\}^l$ (به طوری که $*$ مقدار نامشخص اما محدود و l یک مقدار ثابت در نظر گرفته می‌شود) تعریف کرد. تابع چکیده‌ساز را می‌توان مهم‌ترین مؤلفهٔ هر زنجیرهٔ قالب‌ها دانست به طوری که امنیت زنجیرهٔ قالب‌ها را وابسته به آن می‌دانیم.

لازم به ذکر است در هنگام استفاده از توابع چکیده‌ساز در زنجیرهٔ قالب‌ها طول ورودی تابع چکیده‌ساز، ورودی‌های آن و ترتیب ورودی‌های آن کاملاً مشخص است (به عنوان مثال و برای درک بهتر؛ تعداد مشخصی تراکنش، ریشه درخت مرکل و یک تکبار که کاملاً مشخص است به عنوان ورودی در نظر گرفته می‌شوند که طول این ورودی‌ها یک مقدار مشخص و محدود است).

مفهوم پیوسته بودن قالب‌ها با استفاده از تابع چکیده‌ساز $h(\cdot)$ و مفهوم زنجیرهٔ قالب‌ها در شکل ۱ نشان داده شده است. لازم به ذکر است نحوهٔ نمایش زنجیرهٔ قالب‌ها در این شکل غالباً در انواعی از زنجیرهٔقالب استفاده می‌شود که مبتنی بر اجماع گواه اثبات کار[3] پیاده سازی شده‌اند (زنجیرهٔ قالب‌های بیتکوین معروف‌ترین آنها است) و جامعیت ندارند، اما سایر زنجیرهٔ قالب‌ها نیز ساختاری مشابه دارند. به دلیل کاربرد زیاد این روش اجماع، در این مقاله به این ساختار اشاره شده است. با توجه به شکل ۱، هر قالب از داده‌های ثبت شده در زنجیرهٔ قالب‌ها دارای چهار بخش اصلی است که در ادامه تشریح می‌شوند.

دادهٔ اصلی، ریشهٔ داده یکی از بخش‌های اصلی هر قالب دادهٔ اصلی[4] است که اطلاعات ثبت شده در آن قالب را نشان می‌دهد. علاوه بر دادهٔ اصلی مقدار ریشهٔ آن نیز با توجه به درخت مرکل [۱۱] محاسبه می‌شود و سپس در آن قالب قرار داده می‌شود.

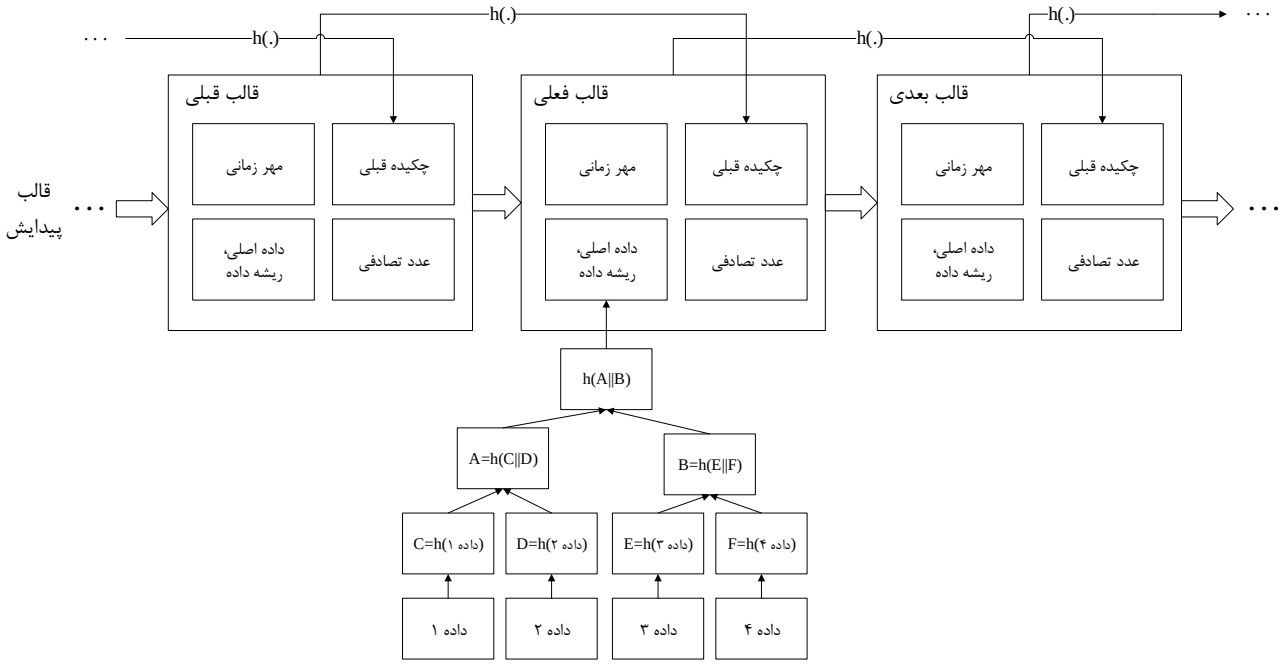
درخت مرکل به عنوان یک درخت از مجموعه نگاشت‌های دو-به-یک از توابع چکیده‌ساز تعریف می‌شود، به طوری که ورودی هر تابع چکیده‌ساز حاصل خروجی دو تابع چکیده‌ساز لایه پایین‌تر است. درخت مرکل[5] به عنوان یک نگاشت به صورت $h : \{0,1\}^{2l} \rightarrow \{0,1\}^l$ نمایش داده می‌شود. به ورودی‌های درخت مرکل برگ‌های درخت و در نهایت به بالاترین لایهٔ درخت، ریشه درخت مرکل گفته می‌شود. در واقع برگ‌های درخت مرکل همان تراکنش‌های تأیید شده و موجود در قالب مورد بحث می‌باشند.

دلیل استفاده از درخت مرکل در زنجیرهٔقالب‌ها ایجاد یک چکیدهٔ ساختار یافته از تراکنش‌های تأیید شده توسط استخراج‌گرها است. بدین ترتیب و با توجه به ساختار درخت مرکل می‌توان گواهی داد که یک تراکنش خاص در قالب استخراج شده وجود دارد. همچنین استفاده از ریشهٔ درخت مرکل در چکیدهٔ هر قالب منجر به تسهیل و سرعت بخشیدن به بررسی صحت قالب تولید شده توسط هر استخراج‌گر می‌شود. لازم به ذکر است، اگر چه در ظاهر استفاده از درخت مرکل موجب پیچیده به نظر رسیدن پیاده سازی زنجیرهٔ قالب‌ها می‌شود، اما مزایای ذکر شده انگیزهٔ طراحان را برای بکارگیری آن افزایش داده است.

مهر زمانی مهر زمانی[6] به زمان تولید آن قالب اشاره دارد.

چکیدهٔ قبلی این بخش از قالب اشاره به قالب قبلی دارد که به صورت چکیدهٔ قالب قبلی (با استفاده از تابع چکیده‌ساز $h(\cdot)$) در قالب فعلی ظاهر می‌شود. عامل اصلی اتصال قالب‌ها به یکدیگر به صورت زنجیره‌ای این بخش هست، که این روند تا اولین قالب (قالب پیدایش) ادامه دارد. در واقع می‌توان گفت شروع این روند از قالب پیدایش بوده است.

[1] Peer-to-Peer [2] Hash function
[3] Proof of Work [4] Main data [5] Merkle tree [6] Timestamp



شکل ۱. ساختار زنجیرهٔ قالب‌ها [۳]

تکبار (nonce) این عدد مرتبط با قالب است و در فرآیند استخراج یا تولید قالب ایجاد می‌شود و اعتبار سنجی قالب ایجاد شده توسط استخراج‌گر با این عدد بررسی می‌شود.

لازم به ذکر است اطلاعات دیگری (به عنوان سایر داده‌ها) نظیر نسخهٔ زنجیرهٔ قالب‌ها، شماره قالب، زبان کد نوشته شده، توابع استفاده شده، سربرگ قالب و ... نیز در هر قالب ثبت می‌شود.

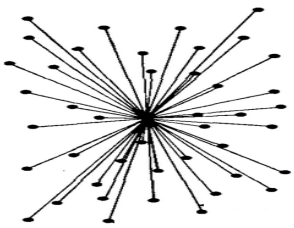
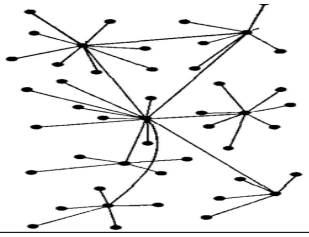
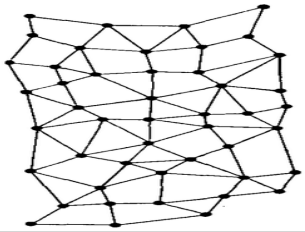
۳.۲ روش‌های ذخیره‌سازی اطلاعات

پس از روش سنتی ذخیره‌سازی داده در ابرها و به طور متمرکز، در دو دهه اخیر روش ذخیره‌سازی داده به طور نامتمرکز رواج پیدا کرده است. پس از محبوبیت فناوری زنجیرهٔ قالب‌ها روش دیگری از ذخیره‌سازی داده با عنوان ذخیره‌سازی داده به طور توزیع شده رایج شده است. این بخش این سه روش را تعریف می‌کند و در جدول ۱ مقایسه‌ای از آنها را ارائه می‌دهد.

روش ذخیره‌سازی متمرکز: یکی از مسائل اساسی در شبکه‌های رایانه‌ای حفظ حریم خصوصی کاربران و حفاظت از داده‌های حساس آنها است. در روش‌های سنتی داده‌ها در یک پایگاه داده مرکزی ذخیره می‌شوند که به آن ابر می‌گویند. در ابر مدیر سامانه و دیگر افراد مجاز اجازه دسترسی به داده‌های ذخیره شده را دارند. در این روش، داده‌هایی که در آن ابر ذخیره شده‌اند باید توسط آن مرکز تحت مراقبت‌های ویژه اعم از حفاظت سایبری و حفاظت فیزیکی قرار داشته باشد و این امر هزینه‌های زیادی را به ابر تحمیل می‌کند. علاوه بر این موضوع، در صورت بروز هرگونه مشکل در بخش مرکزی سازمان اعم از بخش مدیریت یا مرکز ذخیره‌سازی اطلاعات (مشکلات می‌توانند از نوع حملات سایبری و یا حتی حوادث طبیعی پیش‌بینی نشده مثل سیل یا زلزله باشند)، تا پایان زمان رفع مشکل سازمان مربوطه قادر به خدمت رسانی و پاسخگویی به کاربران و یا مشتریان خود نخواهد بود.

روش ذخیره‌سازی نامتمرکز: برای رفع مشکل فوق شرکت‌ها و سازمان‌های بزرگ تصمیم بر تمرکز زدایی ذخیره‌سازی اطلاعات و دارایی‌های خود (دیجیتال یا فیزیکی) گرفتند. در این روش آنها اطلاعات و دارایی‌های خود را در چند نقطه مختلف ذخیره‌سازی می‌کنند تا در صورت بروز مشکل در هر بخش، بخش‌های دیگر بتوانند امور را مرتفع کنند و پاسخگوی کاربران و مشتریان باشند. لازم به ذکر است به این نوع از ذخیره‌سازی داده نیز ابری می‌گویند. به عنوان مثال شرکت‌های بزرگ حوزه فناوری مانند گوگل و یا سایر شرکت‌های بزرگ پایگاه‌های ذخیره‌سازی خود را در چندین کشور مختلف قرار داده‌اند و یا فروشگاه اینترنتی بزرگ آمازون انبار کالاهای خود را در چندین کشور دنیا قرار داده است. این امر موجب افزایش سرعت خدمت رسانی به مشتریان یا کاربران در لحظه می‌شود. همچنین در صورت بروز مشکل برای هر یک از مراکز ذخیره‌سازی (ذخیره‌سازی اطلاعات یا انبار ذخیره‌سازی کالا) سازمان می‌تواند به کار خود ادامه دهد و در زمان مناسب در رفع مشکل تلاش کند. به طور کلی می‌توان گفت امروزه شرکت‌ها و سازمان‌های بزرگ مدیریت نامتمرکز دارایی‌های خود را در دستور کارشان قرار داده‌اند. مسئله مورد توجه در این نوع ذخیره‌سازی و دسترسی این است که در این روش بخش‌های نامتمرکز به صورت مرکزی مدیریت می‌شوند و هیچ کدام از اعضا یا کاربران اجازه تصمیم‌گیری جزئی یا کلی برای سازمان، کارکنان و مشتریان را ندارند.

جدول ۱. بررسی و مقایسهٔ روش‌های مختلف ذخیره‌سازی داده [۱]

نام روش	متمرکز (Centralized)	نامتمرکز (Decentralized)	توزیع شده (Distributed)
همبندی			
دخیره‌سازی	متمرکز	نامتمرکز	توزیع شده
مدیریت	متمرکز	متمرکز	توزیع شده
مزایا	اعمال تغییرات به صورت مرکزی، توانایی کنترل کاربران از سوی نهاد مرکزی، مدیریت و پیاده سازی ساده برای سیاست‌های شبکه.	مقاومت بیشتر در مقابل تهدیدها، امکان ادامه کار و اصلاح مشکل در زمان بروز حادثه، اعتماد بیشتر به عملکرد سامانه.	دسترسی آسان و سریع به اطلاعات برای تمام کاربران، شفافیت کامل اطلاعات و نقل و انتقالات، مقاومت در مقابل تخریب، دستکاری و حذف اطلاعات، عدم نیاز به حفاظت از دارایی‌ها، تغییر ناپذیر بودن اطلاعات ثبت شده، دقت بالا به دلیل حذف منابع انسانی و خودمختاری، انتقال مالکیت به کاربران
معایب	آسیب پذیری در مقابل حملات سایبری و حوادث طبیعی، از کار افتادن کل سامانه در صورت بروز مشکل در پایگاه داده، عدم وجود شفافیت و نظارت کاربران، احتمال تخلف توسط نهاد مرکزی. مشکلات مدیریت متمرکز، عدم کنترل داده توسط مالک	پیاده سازی سیاست‌ها به صورت مرکزی و بدون نظارت و خواست اکثریت کاربران (مدیریت متمرکز همچنان یک مشکل است)، عدم شفافیت، عدم کنترل داده توسط مالک	عدم همزمانی گره‌های شبکه، تأخیر، نیاز به داشتن فضای ذخیره‌سازی کافی در گره‌های اصلی. در صورت استفاده به جای ابر، بسیار گران است و مقرون بصرفه نیست.

روش ذخیره‌سازی توزیع شده: این روش دسترسی بسیار فراتر از روش نامتمرکز است به صورتی که تعداد پایگاه‌ها و یا به زبان ساده‌تر تعداد مراکز ذخیره‌سازی اطلاعات می‌توانند حتی تا تعداد کاربران آن سازمان افزایش یابند (این ایده‌آل‌ترین حالت است که به صورت عملیاتی ممکن نیست). در واقع می‌توان گفت در این روش برای هر کاربر این امکان فراهم است که در صورت داشتن فضای ذخیره‌سازی یک نسخه کامل از اطلاعات را در اختیار داشته باشد (از نظر سیاست گذاری سازمانی منعی وجود ندارد) و بعد از هر بروزرسانی اطلاعات، نسخهٔ در دسترس کاربر هم به روز رسانی شود. در این صورت تمام کاربران می‌توانند به صورت برخط جریان تغییرات اطلاعات جامعه‌ای که با آن در ارتباط اند را دنبال کنند. نکته دیگر این است که در این روش هیچ مرکز متمرکز یا مراکز با تعداد محدود وجود ندارد. در نتیجه علاوه بر امکان دسترسی به اطلاعات برای تمام کاربران، اطلاعات ثبت شده از بین نمی‌رود. همچنین جامعه سازمانی و سیاست‌های آن با همکاری کلیه کاربران و با اجماع آنها به روز می‌شود. یکی از مفاهیم و فناوری‌های مورد استفاده در این مقاله زنجیرهٔ قالب‌ها است که همپوشانی زیادی با روش ذخیره‌سازی و دسترسی توزیع شده دارد، اما در حال حاضر یک فناوری گران در نظر گرفته می‌شود که چالش‌های زیادی نیز دارد و در حال حاضر برای ذخیره‌سازی داده‌های حجیم مورد استفاده قرار نمی‌گیرد.

۴.۲ پروتکل‌های امنیتی مبتنی بر زنجیرهٔ قالب‌ها

با توجه به هدف این مقاله که متمرکز بر پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها است، بهتر است در ابتدا تعریفی مختصر از پروتکل‌های امنیتی مبتنی بر زنجیرهٔ قالب‌ها داشته باشیم. سپس، در زیربخش بعد (۵.۲) به طور خاص تعریفی از پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها ارائه می‌دهیم.

پروتکل‌های امنیتی به پروتکل‌هایی گفته می‌شود که با هدف ارائه یک ویژگی امنیتی (به طور مثال احراز هویت، توافق کلید، حفظ حریم خصوصی و ...) ایجاد می‌شوند. به طور مشابه، پروتکل‌های امنیتی مبتنی بر زنجیرهٔ قالب‌ها به آن دسته از پروتکل‌ها گفته می‌شود که از زنجیرهٔ قالب‌ها به عنوان زیرساخت و پایگاه داده استفاده می‌کنند [۱۲، ۱۳]. اگر بخواهیم ارائه کلی از پروتکل‌های امنیتی مبتنی بر زنجیرهٔ قالب‌ها ارائه کنیم، می‌توان در یک دید کلی آنها را در سه لایه که در شکل ۲ نشان داده شده است، تعریف کنیم. این سه لایه در ادامه تعریف می‌شوند.

- **لایه کاربرد:** در این لایه کاربران و استفاده کنندگان پروتکل‌ها قرار دارند و با استفاده از نرم‌افزاری که در دستگاه هوشمند یا رایانه شخصی‌شان قرار دارد می‌توانند پروتکل را اجرا کنند.
- **لایه انتقال:** همانطور که مشخص شد، زنجیرهٔ قالب‌ها یک دفتر کل توزیع شده مبتنی بر اینترنت است. در نتیجه لایه انتقال دقیقا همان

شکل ۲. نمای کلی از پروتکل‌های امنیتی مبتنی بر زنجیرهٔ قالب‌ها [۱۴، ۱۵]

لایه اینترنت است که مبتنی بر پروتکل TCP/IP کار می‌کند.

- **لایه زنجیرهٔ قالب‌ها:** این لایه مهم‌ترین و وجه تمایز اصلی پروتکل‌های امنیتی مبتنی بر زنجیرهٔ قالب‌ها است. در پروتکل‌های امنیتی، زنجیرهٔقالب‌ها، به عنوان شبکه‌ای از گره‌هایی که به طور نظیر به نظیر با یکدیگر ارتباط دارند و دفتر کل توزیع شدهٔ تراکنش‌ها در نظر گرفته می‌شود. در این لایه قرار دارد و الگوریتم‌های اجماع توسط گره‌ها در لایه اجرا می‌شوند.

۵.۲ پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها (تعریف عمومی)

فرآیند ممیزی داده، فرآیندی است که سالهاست در حال انجام است و بیشترین و شناخته شده‌ترین کاربرد آن در حوزهٔ مالی است (که در ایران بیشتر در حوزه ارسال، بررسی و تأیید مالیات شرکت‌ها شناخته می‌شود)، اما محدود به این حوزه نیست. به نوعی می‌توان هدف اصلی این نوع پروتکل‌ها را تأیید صحت [1] پیام توسط اشخاص سوم (کاربران ممیزی کننده داده) دانست. روند اصلی در فرآیند ممیزی داده/اطلاعات به این صورت است که کاربران ممیزی کننده داده یک برچسب [2] از کل داده‌ها را دریافت می‌کنند و پس از تطبیق آن با دادهٔ اصلی، که معمولا از مسیر دیگر یا روش دیگری ارسال می‌شود، صحت داده را تأیید یا رد می‌کنند.

به طور کلی مدل سامانهٔ پروتکل‌های ممیزی داده و پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها به شرح زیر می‌باشند [۵، ۱۴، ۱۵] (لازم به ذکر است با توجه به کاربرد و هدف طراحی، تعداد مرحله‌ها می‌توانند متغیر باشند و مراحل ذکر شده جهت جمع‌بندی در این مقاله است و الزاما جامعیت ندارند).

در مدل سامانهٔ هر پروتکل ممیزی داده چندین نهاد و مولفه موجود است که در ادامه آنها را تعریف می‌کنیم و وظایف هر یک را شرح می‌دهیم.

- **ارائه دهنده خدمات** [3]: این نهاد مالک سامانه و ارائه دهنده خدمات ممیزی داده است. مسئولیت راه‌اندازی سامانه با مالک سامانه است. مالک سامانه یک نهاد قابل اعتماد برای اعضا در نظر گرفته می‌شود که معمولا پارامترهای خصوصی سامانه، کلید کاربران و کلید کاربران ممیزی کننده داده را تولید و توانایی کنترل کاربران را دارد.
- **ابر:** محل ذخیره‌سازی داده‌ها است که معمولا تحت کنترل مالک سامانه و در دسترس کاربران قرار دارد.
- **زنجیرهٔ قالب‌ها:** به عنوان پایگاه‌داده شفاف تغییر ناپذیر در نظر گرفته می‌شود. با توجه به پر هزینه بودن ذخیره‌سازی در زنجیرهٔ قالب‌ها، در پروتکل‌های ممیزی داده از این پایگاه داده کمتر به عنوان محل ذخیره‌سازی داده‌ها پس از ممیزی استفاده می‌شود و معمولا به عنوان محل ذخیره‌سازی شواهد ممیزی استفاده می‌شود.
- **مالک داده** [4]: مالک و ارسال کننده داده‌ای است که باید پس از ممیزی در پایگاه داده (ابر یا زنجیرهٔ قالب‌ها) ذخیره شود.
- **کاربر (کاربران) ممیزی کننده داده** [5]: کاربرانی هستند که داده را بررسی می‌کنند و در صورت تأیید موارد لازم ادامه فرآیند ممیزی را انجام می‌دهند، در غیر اینصورت ممیزی داده را رد می‌کنند.
- **قرارداد هوشمند** [6]: در بعضی طرح‌ها قراردادهای هوشمند جایگزین کاربران ممیزی کننده داده می‌شوند و روند ممیزی داده را به طور خودکار و مبتنی بر زنجیرهٔ قالب‌ها تأیید یا رد می‌کنند.

پروتکل‌های ممیزی داده معمولا از پنج الگوریتم زمان–چند جمله‌ای به شرح زیر تشکیل می‌شوند [۱۴، ۱۶، ۱۷] (اما الزامی بر طراحی و ارائه در قالب این پنج الگوریتم وجود ندارد):

(۱) **راه‌اندازی** ($Params \leftarrow Setup(\lambda)$): در این الگوریتم با توجه به پارامتر امنیتی λ، پارامترهای سامانه اعم از پارامترهای عمومی و راز مشخص می‌شوند.

این الگوریتم توسط مدیر سامانه اجرا می‌شود و مقادیر عمومی اعلام عمومی می‌شوند و پارامترهای راز به صورت محرمانه نزد مدیر سامانه نگهداری می‌شوند.

در پروتکل‌هایی که مبتنی بر زنجیرهٔ قالب‌ها طراحی می‌شوند، مجموعه پارامترهای عمومی در زنجیرهٔ قالب‌ها ارسال می‌شود و از آن زمان فرض می‌شود که پروتکل برقرار شده است. کلیدهای خصوصی و عمومی کاربران

[1] Integrity [2] Tag

[3] Service provider [4] Data owner [5] Auditor [6] Smart contract

نیز پس از تولید از طریق کانال امن توسط مدیر برای آنها ارسال می‌شود (در برخی طرح‌ها مرحلهٔ تولید کلید در یک بخش مجزا ارائه می‌شود).

(۲) تولید برچسب ($Tag \leftarrow TagGen(Data)$): در این الگوریتم یک برچسب مطابق با داده‌ای که باید ممیزی شود توسط کاربر تولید می‌شود. این الگوریتم توسط کاربر اجرا می‌شود.

(۳) بارگذاری و ارسال ($Upload(Data, Tag)$): در این مرحله برچسب بارگذاری می‌شود و داده اصلی برای ممیزی ارسال می‌شود (با توجه به اینکه تعریف به صورت عمومی است، ارسال می‌تواند به هر پایگاه‌داده یا هر شخصی از اعضای سامانه ممیزی داده صورت پذیرد). این الگوریتم توسط کاربر اجرا می‌شود.

(۴) ممیزی ($Proof \leftarrow Audit(Data, Tag)$): الگوریتم ممیزی داده برچسب و داده را با یکدیگر مطابقت می‌دهد و در صورت تأیید، فرآیند ممیزی را تأیید می‌کند. معمولا در این الگوریتم یک پروتکل ممیزی داده اجرا می‌شود که با توجه به نیاز توسط کاربران ممیزی داده یا قراردادهای هوشمند مبتنی بر زنجیرهٔ قالب‌ها اجرا می‌شود. گاها اثبات نیز در این مرحله ایجاد می‌شود. از روش‌های معمول ممیزی می‌توان به ساز و کارهای مبتنی بر چالش-پاسخ [1] یا امضا توسط کاربران ممیزی اشاره کرد.

(۵) تأیید نهایی ($\{0,1\} \leftarrow Verify(Proof)$): داده ممیزی شده توسط کاربران ممیزی کننده به عنوان داده معتبر در پایگاه داده ذخیره می‌شود و صحت آن قابل تأیید است.

محدودیت‌های کلی در طراحی: با توجه به اینکه هر کدام از الگوریتم‌های طراحی شده برای پروتکل‌های ممیزی داده توسط کدام نهاد اجرا می‌شود و همچنین با توجه به توان پردازشی هر نهاد باید الگوریتم آن به گونه‌ای طراحی شود که امکان اجرا در نهاد مد نظر وجود داشته باشد. توان ارسال اطلاعات توسط کاربر باید با توجه به ویژگی‌های آن و پهنای باند تحت پوشش در نظر گرفته شود. از دیگر مواردی که در طراحی باید در نظر گرفته شود، توانایی کاربران ممیزی در دریافت و پردازش داده‌های دریافتی و همچنین میزان پاسخگویی و مقیاس‌پذیر بودن سرور ارائه دهنده خدمات است. به طور خاص در پروتکل‌های مبتنی بر زنجیرهٔ قالب‌ها باید به کارمزد ارسال و ثبت داده روی زنجیرهٔ قالب‌ها نیز توجه شود. به طور خلاصه می‌توان محدودیت کلی در طراحی پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها را در زیر عنوان کرد:

- توان پردازشی مالکان داده
- توان ارسال مالکان داده
- حافظه ذخیره‌سازی مالکان داده
- توان پردازشی کاربران ممیزی داده
- مقیاس‌پذیر بودن کاربران ممیزی داده
- پهنای باند لازم
- پاسخگویی و مقیاس‌پذیری سرور ارائه دهنده خدمات
- کارمزد ارسال و ذخیره‌سازی اطلاعات در زنجیرهٔ قالب‌ها
- تأخیر پردازش و ثبت اطلاعات

۳ کارهای انجام شده

این بخش به تحلیل و بررسی چند طرح معماری ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها و چندین پروتکل ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها می‌پردازد.

انتخاب کارهای انجام شده و مقالات بررسی شده در این بخش غالباً با استفاده از ترکیب‌های دوتایی و سه‌تایی کلمات کلیدی شامل

auditing protocols, auditing architecture, distributed, blockchain-based, without central authority

بوده است. همچنین در پایش زمانی نیز ابتدا به چند طرح اولیه ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها در سالهای ۲۰۱۷ و ۲۰۱۸ پرداخته شده است و سایر معماری‌ها و پروتکل‌ها از کارهای انجام شده در سالهای پس از ۲۰۲۰ انتخاب شده‌اند. در ارزیابی آخر برای انتخاب مقالات بررسی شده تمرکز بر اعتبار نشریات و کنفرانس‌ها بوده است که به جز چند مورد محدود استفاده از مقالات کنفرانسی (IEEE S&P و ESORICS که از موارد معتبر محسوب می‌شوند) سایر مقالات استفاده شده از مجلات نمایه شده در پایگاه Scopus و غالباً از مواردی با رتبه‌بندی Q1 انتخاب شده‌اند.

۱.۳ بررسی چند طرح معماری

همانطور که گفته شد، این بخش مروری کوتاه بر چند طرح معماری ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها دارد و به طور خلاصه انگیزه/مشکل مطرح شده در آنها و روش حل آن را مطرح می‌کند. در این بخش، استفاده از عبارت طرح معماری [2] با توجه به عنوان طرح‌های بررسی شده و روش ارائهٔ آنها انتخاب شده است که در آن‌ها به جزئیات طرح‌ها و محاسبات مرتبط پرداخته نشده است و صرفا دورنمایی از لایه‌های طرح، نحوهٔ ارتباطات و اختیارات نهادهای موجود ارائه شده است.

در سال ۲۰۱۷ کانیش و لورنت یک طرح معماری برای ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها ارائه دادند که در آن زنجیرهٔ قالب‌ها به عنوان پایگاه‌داده توزیع شده مورد استفاده قرار گرفته بود [۱۸]. آنها ادعا کردند استفاده از زنجیرهٔ قالب‌ها، به عنوان یک پایگاه‌داده شفاف، موجب می‌شود تا کاربران زیادی به اطلاعات خصوصی دیگران دسترسی داشته باشند و این امر با افزایش تعداد کاربران شبکه مدیریت اطلاعات خصوصی کاربران را با مشکل مواجه می‌کرد. برای مقابله و حل این مشکل سه راه‌کار پیشنهاد شد: اول، تکیه بر مدیریت نامتمرکز زنجیرهٔ قالب‌ها که تغییر ناپذیری داده‌های ثبت شده را ارائه می‌دهد. دوم، استفاده از رمزنگاری مبتنی بر شناسه به طوری که هر کسی بتواند تحت نظر مدیر سامانه

[1] Challenge-response

[2] Architecture

اطلاعات خود را مدیریت کند. سوم، استفاده از قرارداد هوشمند به جای کاربران ارائه دهنده خدمات.

در سال ۲۰۱۸، نارولا و همکارانش طرح معماری با نام zkLedger ارائه دادند که علاوه بر حفظ حریم خصوصی اعضای شبکه از روش ممیزی قابل اثبات و سریع پشتیبانی می‌کرد [۱۹]. در طرح معماری ارائه شده از طرح دانایی صفر ناتراوایی اشنور استفاده شده است. zkLedger از دفتر کل ستونی استفاده می‌کند و کاربران می‌توانند تمام تراکنش‌ها را رصد کنند به گونه‌ای که هیچ تراکنشی از دید کاربران و حسابرس‌ها پنهان نمی‌ماند. در همان سال، هوانگ و همکارانش طرح معماری را با نام InfiniteChain برای زنجیرهٔ قالب‌ها عمومی ارائه دادند [۲۰]. آنها برای ایجاد برچسب اطلاعات، خارج از زنجیرهٔ اصلی، از درخت مرکل استفاده کردند به طوری که اطلاعات اصلی خارج از زنجیرهٔ قالب‌ها پردازش می‌شد. برای جلوگیری از تقلب‌های ممکن در خارج از زنجیرهٔ اصلی، زنجیرهٔ جانبی (یک زنجیرهٔ قالب‌های دیگر) استفاده می‌شود. در نتیجه پهنای باند تأیید تراکنش‌ها به میزان قابل توجهی زیاد می‌شود و امکان تلفیق سناریوهایی که خارج از زنجیرهٔ قالب‌ها پیاده شده‌اند با زنجیرهٔ قالب‌ها وجود دارد. CertChain طرح دیگری است که در همان سال توسط چِن و همکارانش ارائه شد [۲۱]. هدف اصلی طرح ارائه شده جلوگیری از تقلب علیه گواهی‌نامه‌های صادر شده برای دامنهٔ سایت‌های اینترنتی بود به طوری که گاهی گواهی‌نامه‌های صادر شده توسط مراجع مرکزی به خطر می‌افتادند یا مشتریان سایت‌های اینترنتی بدون آگاهی قبلی به گواهی‌نامه‌های باطل شده اعتماد می‌کردند. معماری ارائه شده درجهٔ قابل اطمینانی از اعتماد به گواهی‌نامه‌ها و همچنین قابلیت ردیابی را با استفاده از زنجیرهٔ قالب‌ها ارائه می‌داد.

در سال ۲۰۱۹، ژِنگ و همکارانش یک طرح معماری برای مدل‌سازی و ممیزی کلان‌داده‌های تولید شده در ساختمان‌های هوشمند با عنوان bcBIM ارائه دادند [۲۲]. طرح bcBIM یک زنجیرهٔ قالب‌های خصوصی را به عنوان پایگاه‌داده برای اشتراک گذاری کلان‌داده‌های دریافتی از تلفن‌های همراه هوشمند در نظر می‌گرفت به گونه‌ای که امنیت اطلاعات یکی از معیارهای اصلی در نظر گرفته می‌شد. در همان سال، احمد و همکارانش یک طرح معماری را با هدف حل مشکل مقیاس‌پذیری ممیزی‌ها در زنجیرهٔ قالب‌ها با عنوان BlockTrail ارائه دادند [۲۳]. در BlockTrail یک معماری سلسله مراتبی ارائه شده تا با توزیع بار به افزایش سرعت و پهنای باند کمک کند و همچنین باعث ارائه خدمات و پشتیبانی خدمت ممیزی در مقیاس بزرگتر شود. در اواخر سال ۲۰۱۹، هوانگ و همکارانش با رویکردی قوی‌تر نسبت به حفاظت از حریم خصوصی کاربران یک طرح اشتراک‌گذاری داده را مبتنی بر یک زنجیرهٔ قالب‌ها کنسرسیومی مطرح کردند [۲۴]. در معماری ارائه شده آنها از اشتراک‌گذاری داده توسط گروه‌های مختلف و غیر وابسته (مثلا کاربرانی از سازمان‌ها مختلف) از کاربران در ضمن حفظ نام مستعار و غیر قابل‌ردیابی بودن کاربران گروه‌های مختلف پشتیبانی می‌شد. علاوه بر این، هر شخص سومی، بدون تعامل با مالک یا ارسال کننده داده، می‌تواند صحت داده ممیزی شده را به درستی بررسی کند.

۲.۳ بررسی چند پروتکل

در ابتدای این بخش چهار پروتکل ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها که در سال‌های ۲۰۱۷ تا ۲۰۱۹ ارائه شدند به طور کلی تحلیل و بررسی می‌شوند. سپس چندین پروتکل بدیع که در سالهای ۲۰۲۰ و ۲۰۲۳ ارائه شده‌اند با جزئیات بیشتر بررسی می‌شوند. مشابه زیر بخش قبل، در این بخش یادآوری می‌شویم که استفاده از عبارت پروتکل به این منظور است که در کارهای بررسی شده تمام محاسبات سمت کاربر، سمت سرور و کاربران ممیزی کننده داده با جزئیات ارائه و تحلیل شده است. در همین راستا، بررسی جزئیات شامل هزینه محاسبات برای هر یک از نهادهای موجود و سربار مخابراتی در انتهای این مقاله ارائه شده است.

پس از بررسی هر پروتکل، تحلیلی مختصر در رابطه با ارائه یا عدم ارائه ویژگی‌های امنیتی مد نظر (صحت داده در تمام زمان‌ها، حفاظت از حریم خصوصی فرستنده و گیرنده داده، محرمانگی داده تحت ممیزی و مقاومت پروتکل در مقابل حضور کاربران ممیزی مخرب) ارائه می‌شوند. در سال ۲۰۱۷، هوانگ و همکارانش پروتکل SeShare را با هدف جلوگیری از تغییر اطلاعات ثبت شده توسط کاربران ناشناس ارائه دادند [۲۵]. در SeShare محل ذخیره‌سازی داده‌ها ابر است ولی شواهد، با توجه به ویژگی‌های زنجیرهٔ قالب‌ها، در زنجیرهٔ قالب‌ها ذخیره می‌شوند. در پروتکل ارائه شده آنها این فرض در نظر گرفته شده بود که کاربران اجازهٔ دسترسی و ایجاد تغییر در داده‌های ثبت شده را دارند ولی در صورتی که تعدادی از کاربران یک داده را تغییر دهند، نمی‌توان متوجه شد که کدام تغییر را کدام کاربر ایجاد کرده است. برای حل این مشکل پروتکل SeShare با تکیه بر منحصر بفرد بودن امضاهای مختلف طراحی شد که در بخش تحلیل مقاله منحصر بفرد بودن امضا اثبات شد. ایدهٔ اصلی این طراحی اضافه کردن مهر زمانی مبتنی بر زنجیرهٔ قالب‌ها به انتهای داده‌های تغییر کرده بود به طوری که ترتیب تغییرات و کاربران تغییر دهنده را مشخص می‌کرد. همچنین برای کاربرانی که تغییراتی خارج از مقررات وضع شده را مشاهده می‌کردند این امکان فراهم بود تا ممیزی انجام شده را رد کنند و تخلف انجام شده را اطلاع دهند. این پروتکل از موارد امنیتی چون ممیزی عمومی و حفظ حریم خصوصی ارسال کننده پیام پشتیبانی می‌کند.

در سال ۲۰۱۸، یو و همکارانش مدعی شدند که ممیزی کلان‌داده‌ها از اهمیت بالایی برخوردار است، بویژه زمانی که کلان‌داده‌ها مرتبط با یک شهر هوشمند باشند. در نتیجه، در صورت ممیزی نشدن داده‌ها اطلاعات اشتباه زیادی در پایگاه داده ذخیره می‌شود. این مشکل را می‌–توان با ممیزی داده‌ها حل کرد ولی ادعای دوم مطرح شده این بود که این امکان وجود دارد که کاربران ممیزی از پیش تعیین شده بدخواه باشند و با سایر کاربران بدخواه همکاری کنند [۲۶] (این ادعا را می‌توان به عنوان تهدیدات داخلی در نظر گرفت به گونه‌ای که یک نهاد ممتاز از داخل به سامانه ضربه می‌زند یا با مهاجمان خارجی همکاری می‌کند، در بخش بعد این مقاله این موضوع به طور کامل بررسی می‌شود). ایدهٔ اصلی یو و

همکارانش انتخاب و استفاده از کاربران ممیزی از بین کاربران عمومی شبکه بود (کاربران عمومی شبکه نیز می‌توانند به عنوان کاربران بدخواه در نظر گرفته شوند) که برای هر داده پس از ممیزی شدن یک توافق بیزانسی (PBFT) با هدف اجماع کاربران عمومی روی دادهٔ ممیزی شده اجرا می‌شود تا در صورت بدخواه بودن نیز پروتکل ارائه شده خطای تحمیل شده به سامانه را تحمل کند و همچنان سامانه قابل اعتماد باشد.

در سال ۲۰۱۹، ژَنگ و همکارانش اولین پروتکل راستی آزمایی عمومی در برابر کاربران ممیزی معطل (CPVPA) را ارائه دادند که رمزنگاری بدون گواهی در آن استفاده شده بود [۲۷]. در پروتکل CPVPA به دلیل استفاده از زنجیرهٔ قالب‌ها شفافیت و تغییر ناپذیری تراکنش‌های ثبت شده پشتیبانی می‌شد و همچنین زمان ثبت هر تراکنش در کنار آن و بدون تغییر باقی خواهد ماند. ایدهٔ اصلی ارائه پروتکل CPVPA ثبت هر تأیید ممیزی داده به عنوان یک تراکنش در زنجیرهٔ قالب‌ها است. در واقع می‌توان گفت اثبات ممیزی هر داده، به عنوان یک تراکنش در زنجیرهٔ قالب‌ها ثبت می‌شود. دلیل استفاده از رمزنگاری بدون گواهی در پروتکل CPVPA حل مشکل سختی مدیریت گواهی‌نامه‌های صادر شده توسط مرجع مرکزی است.

در تحلیل پروتکل CPVPA با توجه به اینکه در این پروتکل از امضای بدون گواهی استفاده می‌شود دو نوع مهاجم به عنوان مهاجمان داخلی و خارجی تعریف می‌شود و نشان داده می‌شود که امضای به کار برده شده در این پروتکل در مقابل هر دو نوع مهاجم جعل ناپذیر است.

در همان سال، ژو و همکارانش یک پروتکل ممیزی از راه دور برای ذخیره‌سازی داده‌ها در ابر با کمک زنجیرهٔ قالب‌ها ارائه دادند [۲۸]. آنها ادعا کردند که پایگاه‌های دادهٔ خارجی متمرکز برای کاربران قابل اعتماد نیستند و این امر باعث به خطر افتادن صحت داده‌ها می‌شود. ایدهٔ اصلی ژو و همکارانش برای مقابله با مشکلات یاد شده استفاده از زنجیرهٔقالب‌ها به عنوان مرجع ذخیره‌سازی شواهد (برچسب‌ها) مرتبط با داده‌هایی است که باید در ابر ذخیره شوند، همچنین در طرح ارائه شده آنها قرارداد هوشمند مبتنی بر زنجیرهٔ قالب‌ها جایگزین کاربران ممیزی از پیش تعیین شده می‌شود. همچنین در طرح ارائه شدهٔ آنها یک ساز و کار مبتنی بر قرارداد هوشمند برای صحت سنجی (داوری) دادهٔ ممیزی شده ارائه شد که ممیزی انجام شده را به صورت غیر متمرکز داوری می‌کند. با توجه به طراحی پروتکل، امنیت ویژگی‌های ذکر شده به سختی مسائل سخت CDH و QR کاهش یافته بود و به دلیل عدم توانایی مهاجم در حل کردن مسائل فوق امنیت پروتکل برقرار خواهد بود.

ژو (همان نویسندهٔ [۲۸]) و همکارانش یک طرح ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها را با استفاده از روش همانند سازی داده [1] ارائه دادند [۲۹]. آنها مدعی شدند که ساز و کارهایی که قبلا با استفاده از روش همانند سازی داده ارائه شده بودند همچنان با مشکل اتکا به اشخاص سوم معتمد [2] مواجه هستند. روش پیشنهادی اصلی استفاده شده از زوج-نگاشت دو خطی [3] برای همانند سازی و با هدف کاهش بار محاسبات در سمت کاربر است، و در ادامه از همان روش برای بررسی صحت داده استفاده می‌شود. از زنجیرهٔ قالب‌ها در این طرح به عنوان پایگاه برون‌سپاری داده استفاده می‌شود تا علاوه بر اطمینان از صحت نتایج ممیزی بتوان به نظر کاربران ممیزی غیر قابل اعتماد نیز اعتماد کرد.

طرح دیگر ارائه شده در سال ۲۰۲۰ توسط لی و همکارانش با هدف ممیزی کلان‌داده‌های ذخیره شده در ابر با کمک زنجیرهٔ قالب‌ها ارائه شد [۳۰]. مشکل و انگیزهٔ مطرح شده در طرح آنها مشابه با مشکل ارائه شده در طرح ژو [۲۹] است، اما در [۳۰] راه حل دیگری برای رفع این مشکل ارائه شد. آنها علاوه بر عدم اتکا به شخص سوم معتمد، که در دنیای واقعی به سختی پیدا می‌شود، مشکل مدیریت گواهی‌نامه‌های صادر شده را مطرح کردند و برای حل این مشکل از روش ممیزی داده بدون گواهی استفاده کردند. با توجه به اینکه مراحل ایجاد رمزنگاری بدون گواهی نیز باید در پروتکل ممیزی داده آنها اعمال شود.

لی و همکارانش یک طرح ممیزی داده با کمک زنجیرهٔ قالب‌ها برای ذخیره‌سازی کلان‌داده‌ها در ابر ارائه دادند [۵] (مشابه [۳۰]). انگیزه و مشکل مطرح شده در این طرح نیز مشابه دو طرح قبلی است ([۲۹] و [۳۰])، به طوری که لی و همکارانش نیز ادعا کردند اعتماد به اشخاص سوم سخت و پیدا کردن آنها در دنیای واقعی کاری دشوار است (لازم به ذکر است غالب طرح‌های بررسی شده، علی الخصوص طرح‌های سال ۲۰۲۰ و ۲۰۲۱، با ایدهٔ حذف اشخاص سوم و جایگزین کردن کاربران ممیزی عمومی یا قرارداد هوشمند ارائه شده‌اند). یکی از ایده‌های هوشمندانهٔ این طرح برای ایجاد برچسب داده، به طوری که سریع تولید و اعتبارش تأیید شود، استفاده از درخت مرکل برای تولید برچسب است. همچنین استفاده از درخت مرکل علاوه بر به ارمغان آوردن کارایی قابل قبول برای این طرح، باعث می‌شود تا این اطمینان حاصل شود که تمام داده به طور کامل ممیزی شده و در هیچ بخش آن تقلبی صورت نگرفته است.

ژائو و همکارانش با کمک ساز و کارهای ممیزی داده، بدون اتکا به شخص سوم، و با استفاده از ویژگی‌های جذاب زنجیرهٔ قالب‌ها پروتکلی را با هدف بررسی از راه دور صحت داده‌های اینترنت اشیا ارائه دادند [۶]. نخستینه‌های رمزنگاشتی [4] استفاده شده در این پروتکل طرح رمزنگاری الجمال، زوج‌نگاشت دوخطی و یک امضای دیجیتال با قابلیت تأیید دسته‌ای با هدف ارتقا حفاظت از حریم خصوصی هستند. مفهوم تأیید دسته‌ای در این پروتکل به این معناست که در صورتی که داده به بخش‌های متعددی تقسیم شده باشد و یا تعدادی داده متفاوت متعلق به یک کاربر باشد، می‌توان صحت تمام بخش‌های یک داده یا تمام داده‌های ثبت شده را با یک بار اجرای الگوریتم بررسی کرد.

وَنگ و همکارانش طرحی را با هدفی متفاوت ارائه دادند به طوری که از یک روش ناتراوایی با قابلیت اثبات عمومی (NI-PPDP) پشتیبانی می‌کرد. [۳۱]. تفاوت اصلی پروتکل ارائه شده آنها با سایرین در پشتیبانی

[1] Dublication [2] TTP: Trusted Third Party

[3] Bilinear pairing [4] Cryptographic primitives

از ویژگی ناتراوایی بودن و ماهیت داده‌های تحت ممیزی بود که در طرح آنها داده‌هایی که پس از ممیزی در ابر ذخیره می‌شوند تراکنش مالی بودند. در پروتکل NI-PPDP ایده و انگیزهٔ اصلی استفاده از ساز و کار ممیزی عمومی داده در ارسال تراکنش‌های مالی حفظ صحت داده‌ها بدون اتکا به شخص سوم است.

فَن و همکارانش پروتکل Dredas را با هدف ممیزی غیر متمرکز، کارا و توزیع شدهٔ داده‌های ارسالی از اینترنت اشیا صنعتی ارائه دادند [۱۴]. در این پروتکل نیز بر عدم اتکا به کاربر ممیزی سوم قابل اعتماد تاکید می‌شود به طوری که یک قرارداد هوشمند مبتنی بر شبکهٔ اتریوم جایگزین اشخاص سوم شده است و بدون هیچ نگرانی می‌توان از اشخاص سوم قابل اعتماد (منظور همان قرارداد هوشمند مبتنی بر شبکهٔ اتریوم است) استفاده کرد. نویسندگان پروتکل Dredas از دو ایده برای منحصر بفرد کردن کار خود استفاده کردند: اول، در مرحله چالش هر یک از طرفین پروتکل، به طور مستقل از هم، یک تکبار را به عنوان چالش انتخاب می‌کنند. دوم، برای جلوگیری از تقلب از قرارداد هوشمند مبتنی بر شبکهٔ اتریوم استفاده شده است به طوری که قالب‌های شبکه به عنوان مهر زمانی تغییر ناپذیر مورد استفاده قرار می‌گیرند.

ژَنگ و همکارانش یک طرح ذخیره‌سازی در ابر با کمک زنجیرهٔ قالب‌ها را ارائه دادند که علاوه بر حفظ محرمانگی داده‌های ذخیره شده در ابر، مقاومت در مقابل حملهٔ جست و جوی کامل را نیز فراهم می‌کند [۳۲]. استفاده از قرارداد هوشمند به جای کاربران ممیزی باعث می‌شود تا از اصلاح، دستکاری و تغییر غیرقانونی داده جلوگیری شود و استفاده از زنجیرهٔ قالب‌ها باعث می‌شود کاربر از صحت داده اطمینان حاصل کند.

در سال ۲۰۲۱ پروتکل DA-DS ارائه شد که در آن گروهی از کاربران ممیزی از پیش تعیین شده وظیفهٔ ممیزی داده را بر عهده دارند [۳۳]. انگیزه و هدف پروتکل DA-DS حل مسائل و مشکلاتی چون: ۱) جلوگیری از تغییر و دستکاری داده‌های ثبت شده، ۲) وجود نقطه اتکا در شبکه‌های متمرکز، ۳) جلوگیری از دسترسی به اطلاعات خصوصی کاربران توسط افراد داخلی ممتاز، ۴) جلوگیری از ثبت داده‌های نامعتبر، ۵) جلوگیری از افشای محتوای پیام‌ها/تراکنش‌ها، و ۶) تحمیل کارمزد زیاد به کاربران مطرح شده است. برای حل مشکلات ذکر شده راه حل‌های ساده‌ای با هم ترکیب شدند که منجر به طراحی این پروتکل DA-DS شد. به طور کلی می‌توان گفت زنجیرهٔ قالب‌ها به عنوان پایگاه داده تغییرناپذیر برای جلوگیری از دستکاری داده‌های استفاده شده، سامانه رمزنگاری الجمال برای جلوگیری از انتشار و به دست آوردن کلید خصوصی کاربران توسط داخلی‌های ممتاز استفاده شد، ساز و کار ممیزی داده برای جلوگیری از ثبت پیام‌ها/تراکنش‌های نامعتبر (با وجود حفظ محرمانگی داده‌ها) استفاده شد، به گونه‌ای که پیام‌ها/تراکنش‌های محرمانه قبل از ثبت در زنجیرهٔ قالب‌ها ممیزی می‌شوند.

در سال ۲۰۲۳ چندین پروتکل ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها ارائه شد که در این بخش دو مورد از آنها [۳۴، ۳۵] که برخی ویژگی‌های امنیتی با هدف برآورده کردن و حفظ حریم خصوصی نهادهای حاضر در پروتکل را ارائه می‌دهند بررسی می‌کنیم. پروتکل BB-DA [۳۴] با هدف حفظ گمنامی ثبت کنندگان تراکنش مالی از دید عموم ارائه شده بود. در این طرح نویسندگان با ارائه پیگیری‌پذیری تراکنش‌های محرمانه و غیر قابل ردیابی در قالب ایجاد یک پوستهٔ اضافی به عنوان بخش ممیزی این ویژگی را برآورده کردند. اگرچه این طرح از دید مالک سامانه و افراد ممتاز حاضر در پروتکل قابل پیگیری است، اما مهاجمان خارجی نمی‌توانستند تراکنش‌های مالی را ردیابی کنند و همچنین نمی‌توانند هویت کاربران حاضر در پروتکل را افشا کنند. بر خلاف پروتکل BB-DA، پروتکل ZK-DAP [۳۵] برای ذخیره‌سازی داده‌هایی با حجم کم پس از ممیزی طراحی شده بود. در این پروتکل، نویسندگان با استفاده از یک امضای دیجیتال مبتنی بر دانش صفر، علاوه بر احرازهویت کاربر ارسال کننده داده ساز و کاری را طراحی کردند که مالکیت داده نیز اثبات می‌شود. ویژگی‌های اصلی ارائه شده در پروتکل ZK-DAP شامل ارائه احرازهویت گمنام مالکان داده، اثبات مالکیت داده و محرمانگی داده تحت ممیزی است که با استفاده از امضای دیجیتال مبتنی بر دانش صفر برآورده می‌شد.

۴ بررسی یک چارچوب کلی پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها

یکی از انگیزه‌های اصلی ارائه این چارچوب محدودیت‌های زنجیرهٔ قالب‌ها برای ذخیره‌سازی (شامل هزینه و تأخیر) است. در سالهای اخیر، روش‌های زیادی برای ذخیره‌سازی داده در ابر با کمک زنجیرهٔ قالب‌ها ارائه شده است [۳۶]. به طور کلی در این روش‌ها، برچسب داده‌ها پس از ممیزی و وارسی صحت داده در زنجیرهٔ قالب‌ها ذخیره می‌شود و داده اصلی در ابر ذخیره می‌شود. همچنین برای تأیید صحت داده می‌توان به زنجیرهٔ قالب‌ها مراجعه کرد.

همانطور که در بخش قبل بررسی شد، تنوع پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها در سالهای اخیر بالا است. همچنین طرح‌های زیادی با اهداف مشابه و ویژگی‌های مختلف ارائه شده است. اما مسئله اصلی در ارائهٔ پروتکل‌های ذکر شده فقدان یک چارچوب جامع است که پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها از آن پیروی کنند.

۱.۴ تعاریف

در این بخش کلیات چارچوب پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها، شامل معرفی نهادهای حاضر، تعریف کلی و اهداف آن ارائه می‌شوند.

- **نهادهای حاضر:** در این قسمت نهادهای حاضر در چارچوب مد نظر تعریف می‌شوند و وظایف هر یک به تفکیک بیان می‌شوند [۳۳، ۳۷، ۳۸].
- **مالک سامانه** [1] : او کسی است که زنجیرهٔ قالب‌ها را راه‌اندازی (ایجاد) می‌کند و قرارداد هوشمند مرتبط با هر طرح یا پروتکل ممیزی داده

[1] System owner

را روی آن توسعه می‌دهد. همچنین او مالک ابر است و به کاربران مجاز اجازه دسترسی به ابر را می‌دهد.

- **زنجیرهٔ قالب‌ها:** پایگاه داده توزیع شده‌ای است که در این بخش به عنوان محل ذخیره‌سازی برچسب داده‌ها در نظر گرفته می‌شود. برچسب مرتبط با هر داده پس از تأیید، با هر پروتکل اجماع، توسط استخراج‌گرها در زنجیرهٔ قالب‌ها ذخیره می‌شود و در صورت هر گونه درخواست یا پرس و جو به کاربر بازگردانده می‌شود.
- **ابر:** محل ذخیره‌سازی داده‌ای است که برچسب مرتبط با آن روی زنجیرهٔ قالب‌ها ذخیره شده است. ابر توسط مالک سامانه کنترل می‌شود و او می‌تواند اطلاعات را، پس از ممیزی، در ابر ذخیره کند. در صورت هر گونه تغییر در داده‌ها پس از ممیزی یا حذف آنها، کاربر با استناد به برچسب ذخیره شده در ابر می‌تواند مالک سامانه را محکوم کند.
- **کاربر (مالک داده):** او مالک داده است و می‌خواهد اطلاعاتش پس از ممیزی در ابر ذخیره شود و اطمینان حاصل کند که صحت آنها حفظ می‌شود.
- **کاربران ممیزی کننده داده:** آنها را می‌توان کاربران عادی شبکه در نظر گرفت که در فرآیند تأیید و ممیزی داده، قبل از ثبت داده، مشارکت دارند.
- **سایر کاربران:** گروهی از کاربران عادی شبکه هستند که می‌توانند جزء کاربران ممیزی کننده داده نیز باشند، و امکان دسترسی به داده‌های ثبت شده در ابر و به برچسب‌های ذخیره شده در زنجیرهٔ قالب‌ها را دارند و می‌توانند صحت داده‌ها را وارسی کنند.

۲.۴ کلیات

این طرح در شش الگوریتم زمان-چندجمله‌ای زیر تعریف می‌شود (تعریف و وظایف نهادهای طرح مشابه طرح قبلی است). همچنین مدل عمومی پروتکل‌های ممیزی داده مبتنی زنجیرهٔ قالب‌ها که در این بخش بررسی می‌شود در شکل ۳ نشان داده شده است.

راه اندازی ($Setup(\lambda) \leftarrow (Params)$): مالک سامانه با استفاده از پارامتر امنیتی λ الگوریتم $Setup$ را اجرا می‌کند و مجموعه پارامترهای عمومی سامانه $Params$ را در زنجیرهٔ قالب‌ها ارسال می‌کند.

تولید برچسب ($TagGen(Data) \leftarrow (Tag)$): مالک داده الگوریتم $TagGen$ را روی دادهٔ $Data$ اجرا می‌کند و برچسب Tag را به عنوان خروجی الگوریتم دریافت می‌کند.

بارگذاری ($Upload(Data, Tag)$): پس از تولید برچسب Tag، کاربر برچسب را در زنجیرهٔ قالب‌ها ارسال می‌کند و داده اصلی $Data$ را برای مالک سامانه ارسال می‌کند.

ممیزی ($Audit(Tag) \leftarrow (Proof)$): مالک سامانه داده اصلی را برای کاربر ممیزی کننده داده ارسال می‌کند. کاربر ممیزی کننده داده با توجه به برچسب ارسالی روی زنجیرهٔ قالب‌ها و دادهٔ دریافتی، اسناد ممیزی (شامل مواردی که صحت تولید برچسب را اثبات می‌کنند) را تولید و روی زنجیرهٔ قالب‌ها ارسال می‌کند. اسناد ممیزی با توجه به شرایط و نیازمندی‌های پروتکل مختلف ایجاد می‌شوند.

تأیید ($Verify(Proof) = (\{۰,۱\})$): پس از ارسال اثبات تولید شده روی زنجیرهٔ قالب‌ها و اجرای موفق پروتکل اجماع مورد استفاده در طرح و در صورت صحیح بودن اثبات تولید شده، فرآیند ممیزی نهایی می‌شود.

ذخیره‌سازی ($Storage(Data)$): در صورتی که اجماع با موفقیت انجام شود، مالک سامانه داده را در ابر ذخیره می‌کند و در دسترس عموم قرار می‌دهد.

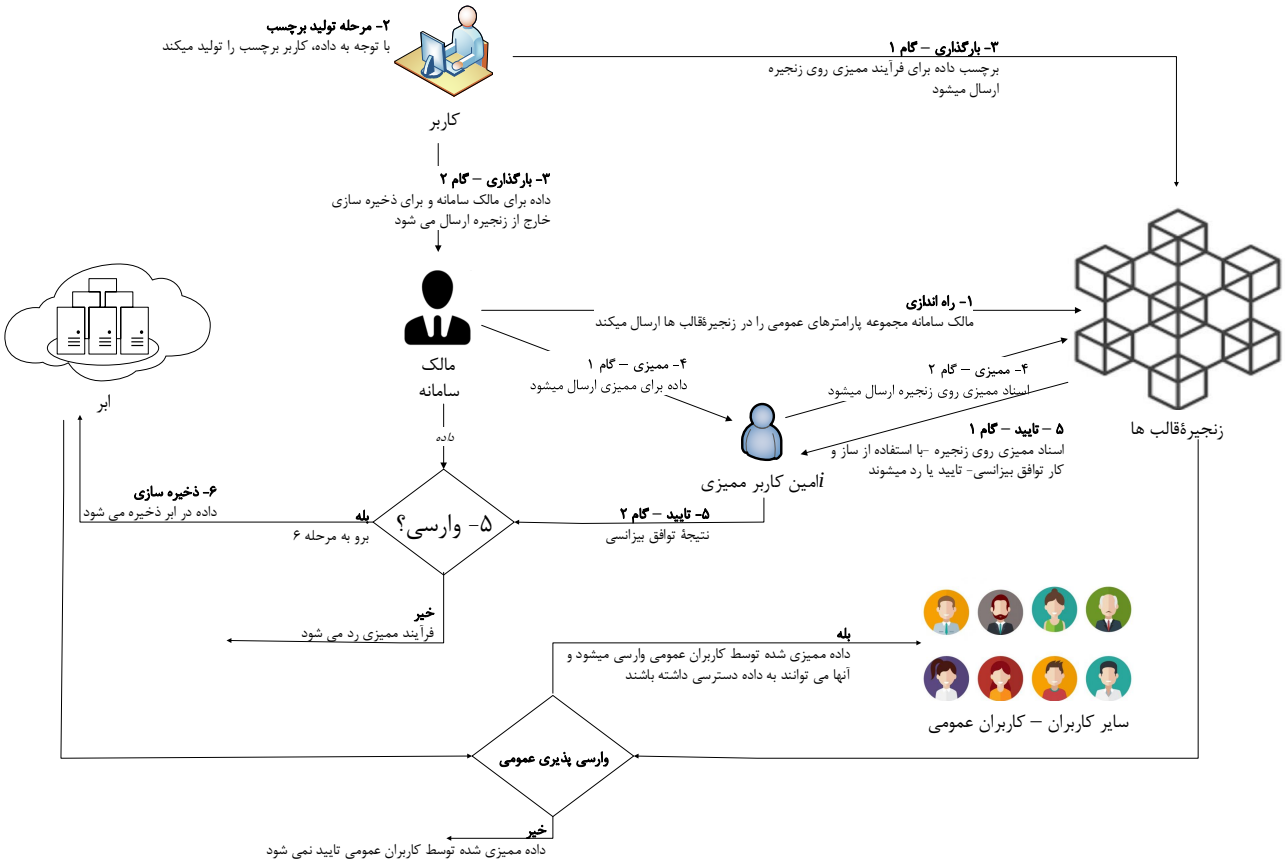
پس از ذخیره شدن داده در ابر، علاوه بر داده، برچسب و اسناد ممیزی مرتبط با آن در زنجیرهٔ قالب‌ها قابل جست و جو و وارسی هستند.

۳.۴ جزئیات

در این بخش ما چارچوب ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها را در شش مرحله اصلی با جزئیات تشریح می‌کنیم، این مراحل در شکل ۳ نمایش داده شده‌اند. همچین یک مرحله نیز به عنوان وارسی عمومی در نظر گرفته می‌شود که کاربران عادی شبکه را قادر می‌سازد تا صحت داده ممیزی شده را وارسی کنند.

راه اندازی: در این مرحله، مالک ابر چارچوب ممیزی را راه‌اندازی می‌کند و با استفاده از پارامتر امنیتی λ الگوریتم $Setup$ را فراخوانی می‌کند. سپس، پارامترهای عمومی چارچوب پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها به صورت $Param = \{N, th, corr, h(\cdot), GPK\}$ مشخص می‌شود که در آن N تعداد کاربران ممیزی کننده داده، th تعداد آستانه لازم برای انجام فرآیند ممیزی تأیید شده، $corr$ تعداد کاربران ممیزی که از همکاری امتناع می‌کنند، و $h(\cdot)$ تابع چکیده‌ساز یکطرفه است که به صورت $h : \{۰,۱\}^* \rightarrow \{۰,۱\}^l$ تعریف می‌شود که معمولاً $l = ۲۵۶$ در نظر گرفته می‌شود و کلید عمومی گروه $GPK(GroupPublicKey)$ تابعی از λ است. او سپس کلیدهای عمومی و خصوصی کاربران ممیزی PA_j را به صورت e_{PA_j} و d_{PA_j} ایجاد می‌کند و کلیدهای خصوصی آنها را از طریق کانال امن برایشان ارسال می‌کند. در انتهای این مرحله، مالک ابر پارامترهای عمومی چارچوب ممیزی $Param$ را در زنجیرهٔ قالب‌های عمومی ثبت می‌کند.

تولید برچسب: برای تولید برچسب، کاربر مالک داده U_i با شناسهٔ ID_i که قصد دارد دادهٔ $Data$ را ذخیره کند باید ابتدا برچسب Tag مرتبط با آن را مطابق با الگوریتم ایجاد ریشهٔ درخت مرکل محاسبه کند. برای این منظور ابتدا داده $Data$ را به n قسمت تقسیم می‌کند ریشهٔ درخت مرکل را به عنوان برچسب به صورت $h : \{ID_i, data_۱, data_۲, \ldots, data_n\}^{۲l} \rightarrow \{Tag\}^l$ محاسبه می‌کند و مجموعه $\{ID_i, data_۱, data_۲, \ldots, data_n\}$ را در نظر می‌گیرد. جهت داشتن گمنامی، کاربر U_i مقدار $UID_i =$



شکل ۳. چارچوب کلی پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها [۳۶]

$h(ID_i\|r\|t)$ را به جای ID_i جایگزین می‌کند که در آن r یک تکبار و t زمان تولید برچسب Tag هستند. در این بخش استفاده از درخت مرکل، به عنوان روشی برای ایجاد برچسب، این ویژگی را به ارمغان می‌آورد که کاربر می‌تواند یک برچسب کوتاه از داده که در بر دارنده شناسه او نیز هست را با روشی سریع محاسبه کند. از این مورد در بخش اثبات مالکیت داده استفاده می‌شود.

بارگذاری[1]: این مرحله در دو گام زیر انجام می‌شود.

- گام اول: کاربر U_i برچسب تولید شدهٔ Tag را، به عنوان شواهد داده $Data$، در زنجیرهٔ قالب‌ها ثبت می‌کند (لازم به ذکر است که کارمزد ثبت برچسب بسیار ناچیز است، چون برچسب تولید شده ریشهٔ درخت مرکل غالباً ۲۵۶ بیت طول دارد). سپس، داده برای فرآیند ممیزی برای کاربران ممیزی ارسال می‌شود.
- گام دوم: او سپس شناسهٔ ID_i یا UID_i و دادهٔ $Data$ (یا داده‌های $Data_1$ و $Data_2$ و غیره) برای مالک ابر برای ذخیره‌سازی نهایی در ابر ارسال می‌کند. سپس کاربر منتظر تأیید و ذخیره داده می‌ماند.

ممیزی: این مرحله نیز در دو گام به شرح زیر انجام می‌شود.

- گام اول: تعداد N کاربر ممیزی PA_j به طوری که $i < N$، به طور تصادفی توسط مالک ابر انتخاب می‌شوند و کلید عمومی گروهی GPK مرتبط با آنها ایجاد می‌شود و n, ID_i or $UID_i, Data$ به صورت عمومی پخش می‌شود. پس از دریافت n, ID_i or $UID_i, Data_1, Data_2$ از طرف کاربر، مالک ابر مجموعه کاربران ممیزی را انتخاب می‌کند.
- گام دوم: به محض دریافت n, ID_i or $UID_i, Data$ هر کاربر ممیزی کننده داده PA_j که قصد دارد در فرآیند ممیزی همکاری کند، داده $Data$ را به n قسمت به صورت $data_1, data_2, \ldots, data_n$ تقسیم می‌کند و برچسب Tag^* را به صورت

$$h : \{ID_i \text{ or } UID_i, data_1, data_2, \ldots, data_n\}^{2l} \rightarrow \{Tag^*\}^l$$

محاسبه می‌کند.

سپس، کاربر ممیزی کننده دادهٔ PA_j درخواست جست و جوی برچسب محاسبه شدهٔ Tag را به زنجیرهٔ قالب‌ها ارسال می‌کند (همان برچسبی که توسط کاربر در زنجیرهٔ قالب‌ها ثبت شده است). پس از بازگردانده شدن برچسب از زنجیرهٔ قالب‌ها، در صورتی که $Tag^* =$

[1] Upload

Tag، کاربر ممیزی کننده داده الگوریتم

$$(\sigma_j) \leftarrow ISG(Tag^*, d_(PA_j))$$

را اجرا می‌کند. در این بخش الگوریتم ISG به الگوریتم ایجاد امضای دیجیتال فردی [۱] اشاره دارد که هر کاربر ممیزی کننده داده با استفاده از کلید خصوصی خود، صحت برچسب محاسبه شده را تأیید می‌کند. در نهایت برچسب امضا شدهٔ Tag^* را در زنجیرهٔ قالب‌ها ارسال می‌کند.

وارسی [۲]: در این مرحله، برچسب ممیزی شدهٔ Tag^* در زنجیرهٔ قالب‌ها ثبت شده است. هر برچسب شامل یک امضای انفرادی است. برای تأیید برچسب و تمامیت داده ارسالی برای ذخیره‌سازی دو گام زیر طی می‌شود.

- گام اول: پس از ثبت برچسب Tag^* در زنجیرهٔ قالب‌ها توسط کاربر ممیزی کننده دادهٔ PA_j، استخراج‌گرها می‌توانند برچسب Tag را در زنجیرهٔ قالب‌ها پیدا کنند و در صورت صحت، الگوریتم $(\{۰,۱\}) \leftarrow ISV(\sigma_j, GPK)$ را اجرا کنند. الگوریتم ISV [۳] همان الگوریتم بررسی امضای دیجیتال فردی است که ورودی‌های آن امضای دیجیتال هر کاربر ممیزی کننده داده و کلید عمومی گروهی کاربران ممیزی GPK است. در صورت خروجی ۱ از الگوریتم فوق، گام بعدی اجرا می‌شود.
- گام دوم: همانطور که مشخص است، در این گام مالک سامانه/ابر باید در صورت تأیید گام قبل، صحت امضای آستانه‌ای با اجرای الگوریتم TSV [۴] بررسی می‌کند. در صورت خروجی الگوریتم به صورت $TSV(\sigma_{th}, GPK) \leftarrow ۱$ باشد، مالک ابر الگورتیم $(\sigma_{th}) \leftarrow TSG(\sigma_j)$ را اجرا می‌کند و آن را به عنوان تأیید نهایی امضای آستانه‌ای انجام شده [۵] اجرا می‌کند. در صورت عدم تأیید موارد فوق، فرآیند ممیزی رد می‌شود و ادامهٔ روند پروتکل اجرا نمی‌شود.

ذخیره‌سازی [۶]: در نهایت و پس از تکمیل تمام مراحل قبل، مالک ابر مجموعه

$$\{ID_i \text{ or } UID_i, Data, n\}$$

یا

$$\{ID_i \text{ or } UID_i, Data_۱, Data_۲, \ldots\}$$

را در ابر ذخیره می‌کند. در این زمان، داده $Data$ در ابر برای همه در دسترس است و هر کسی قادر است آن را ببیند و تمامیت آن را وارسی کند.

وارسی عمومی [۷]: این مرحله لازم نیست اجرا شود و فقط در صورت نیاز اجرا می‌شود. با امکان دسترسی به دادهٔ ذخیره شده در ابر، هر کاربر عمومی شبکه می‌تواند تمامیت آن را وارسی کند. این امکان در پنج گام زیر انجام می‌شود.

[1] Individual Signature Generation [2] Verification [3] Individual Signature Verification [4] Threshold Signature Verification [5] Threshold Signature Generation [6] Storage [7] Public verification

- کاربر داده ذخیره شده در ابر را به صورت

$$\{ID_i \text{ or } UID_i, Data, n\}$$

پیدا می‌کند.
- کاربر یک درخواست برای دسترسی به برچسب Tag به زنجیرهٔ قالب‌ها ارسال می‌کند.
- کاربر داده دریافتی $Data$ را به n بخش تقسیم می‌کند و برچسب مرتبط Tag را به صورت

$$h : \{ID_i \text{ or } UID_i, data_۱, data_۲, \ldots, data_n\}^{۲l} \rightarrow \{Tag\}^l$$

محاسبه می‌کند.
- کاربر صحت تمام امضاهای فردی روی برچسب را وارسی می‌کند.
- در نهایت، اگر تعداد آستانه‌ای از امضاهای فردی صحیح باشند کاربر صحت داده $Data$ را تأیید کند.

۴.۴ تحلیل چارچوب بررسی شده

در این بخش چارچوب بررسی شده را تحلیل می‌کنیم و نشان می‌دهیم که اهداف امنیتی مد نظر پروتکل‌های ممیزی داده را تامین می‌کند.

۱.۴.۴ امنیت

عبارت امنیت محدود به یک ویژگی خاص نیست و شامل موارد زیادی است که در این بخش نشان می‌دهیم چارچوب بررسی شده، ویژگی‌های امنیتی مشترک بین اکثر پروتکل ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها را ارائه می‌دهد.

گمنامی: برای گمنام بودن لازم است کاربر یا مالک داده علاوه بر داشتن نام مستعار امن، غیر قابل ردیابی نیز باشد.

۱) **نام مستعار امن** [۸]: برای بازیابی شناسهٔ i امین کاربر ID_i با داشتن نام مستعار او UID_i، مهاجم $\mathcal{A}$ باید بتواند معکوس تابع چکیده‌ساز را به صورت $h^{-۱}(UID_i) = ID_i \| r \| t$ محاسبه کند. برای این کار، مهاجم $\mathcal{A}$ تعداد زیادی پرس و جوی UID_i به پیشگوی تصادفی ارسال می‌کند و پاسخ‌ها را به صورت

$$Response = \{ID_i^*, r^*, t^*\} \leftarrow \mathcal{O}_h(UID_i)$$

دریافت می‌کند [۳۹-۴۱]. سپس، مهاجم $\mathcal{A}$ جدولی از پاسخ‌های دریافتی به صورت $\{UID_i, ID_i^*, r^*, t^*\}$ ایجاد می‌کند.

برای حدس زدن ID_i، مهاجم $\mathcal{A}$ باید مقادیر معتبری برای ID_i^*, r^*, t^* حدس بزند به طوری که در رابطه $UID_i = h(ID_i^* \| r^* \| t^*)$ صدق کنند.

مزیت مهاجم $\mathcal{A}$ در بازیابی مقدار معتبر برای شناسهٔ ID_i با داشتن UID_i به صورت

$$ADV_{\mathcal{A}}^{hash} = \Pr[\mathcal{A}(UID_i) = ID_i \mathcal{O}_h] < \epsilon$$

[8] Secure pseudonym

محاسبه می‌شود. در نتیجه، می‌توان گفت مهاجم $\mathcal{A}$ می‌تواند شناسه کاربر را با احتمال ناچیز بازیابی کند و چارچوب بررسی شده از نام مستعار امن پیشتیبانی می‌کند چون تابع چکیده‌ساز استفاده شده امن است.

۲) **غیر قابل ردیابی بودن**[1]: با دسترسی به داده $Data$، مهاجم $\mathcal{A}$ می‌تواند تمام مجموعه‌های $\{UID_i, Data_i\}$ را داشته باشد. می‌توان گفت چارچوب بررسی شده غیر قابل ردیابی است اگر مهاجم $\mathcal{A}$ نتواند ارتباطی بین $\{UID_i, Data_i\}$ و $\{UID_{i'}, Data_{i'}\}$ پیدا کند به طوری که $i \neq i'$. با توجه به امن بودن تابع چکیده‌ساز استفاده شده که در بند قبل نیز مزیت مهاجم $\mathcal{A}$ به صورت $ADV_{\mathcal{A}}^{hash} < \epsilon$ محاسبه شد، می‌توان نتیجه گرفت مهاجم $\mathcal{A}$ نمی‌تواند هیچ دانشی از شناسهٔ i امین کاربر ID_i یاد بگیرد.

۳) **صحت داده:** بعد از ذخیره شدن داده $Data$ در ابر، مهاجم $\mathcal{A}$ یا هر فرد ممتاز داخلی که به داده دسترسی دارد می‌تواند با استفاده از هر یک از ترفندهای نفوذ آن را تغییر دهد. اما موفقیت مهاجم زمانی در نظر گرفته می‌شود که کسی نتواند تغییرات صورت گرفته را متوجه شود یا آنها را ردیابی کند. با توجه به تعریف چارچوب ارائه شده (مرحله بارگذاری – گام اول) برچسب Tag مرتبط با داده $Data$ که امضا شده است در زنجیرهٔ قالب‌ها ذخیره شده است (σ_j). در نتیجه در صورت تغییر داده ثبت شده در ابر، با بررسی صحت امضا روی برچسب ثبت شده در زنجیرهٔ قالب‌ها هر کاربر می‌تواند متوجه تغییر شود. در نتیجه، می‌توان گفت چارچوب بررسی شده صحت داده را فراهم می‌کند و هیچ مهاجم خارجی یا افراد داخلی ممتاز نمی‌توانند داده ذخیره شده در ابر را تغییر دهند چون اطلاعات ثبت شده در زنجیرهٔ قالب‌ها تغییر ناپذیر است.

امنیت در مقابل حمله مردی در میانه (بین مالک سامانه و کاربران ممیزی کننده داده): در این حمله مهاجم $\mathcal{A}$ بین دو نهاد ذکر شده قرار می‌گیرد و تلاش می‌کند با قطع پیام و جایگزینی پیام دیگر و یا هر ترفند دیگری در روند ممیزی اختلال ایجاد کند. لازم به ذکر است، بررسی این حمله در چارچوب بررسی شده با هدف بررسی امکان ایجاد اختلال در احرازهویت کاربر یا جعل هویت کاربر نیست و مهاجم $\mathcal{A}$ قصد دارد تا مالکیت داده را تغییر دهد. در واقع، در صورت موفقیت‌آمیز بودن این حمله، مالک داده نمی‌تواند مالکیت خود را بر داده‌های ثبت شده اثبات کند. در این حمله مهاجم $\mathcal{A}$ پیروز در نظر گرفته می‌شود اگر کسی از حضور او اطلاعی پیدا نکند و او بتواند تغییرات مد نظرش را انجام دهد و به هدف خود برسد. برای پیاده سازی این حمله در چارچوب بررسی شده، مهاجم $\mathcal{A}$ مجموعه $\{n, ID_i \text{ or } UID_i, Data\}$ که شامل شناسه اصلی کاربر یا نام مستعار او است را در بین راه قطع می‌کند و مقادیر $\{n, ID_i' \text{ or } UID_i', Data'\}$ را جایگزین آن می‌کند. با دریافت $\{n, ID_i' \text{ or } UID_i', Data'\}$ در سمت کاربران ممیزی کننده داده PA_j، هر کدام از آنها برچسب Tag^* را محاسبه می‌کنند و تلاش می‌کنند تا آن را در زنجیرهٔ قالب‌ها پیدا کنند. با توجه به اینکه برچسب محاسبه شده در زنجیرهٔ قالب‌ها وجود ندارد یا $Tag \neq Tag^*$، می‌توان نتیجه گرفت که کاربران ممیزی کننده داده متوجه تغییر می‌شوند و داده نامعتبر را ممیزی نمی‌کنند. پس می‌توان نتیجه گرفت چارچوب بررسی شده در مقابل حمله مردی در میانه امن است چون اطلاعات ثبت شده در زنجیرهٔ قالب‌ها تغییر ناپذیر است.

نقض مالکیت داده (در سمت کاربر): در این حمله، مهاجم $\mathcal{A}$ تلاش می‌کند تا شناسهٔ خود را به جای کاربر U_i در برچسب داده جایگزین کند و به این طریق مالکیت داده را داشته باشد. برای انجام این کار، مهاجم $\mathcal{A}$ دو رویکرد زیر را در مقابل خود دارد:

۱) مهاجم $\mathcal{A}$ ابتدا داده مد نظر $Data$ را از کانال عمومی شنود می‌کند یا آن را در ابر پیدا می‌کند. سپس او برچسب جدید Tag' را به صورت $h : \{ID_{\mathcal{A}}, data_1, data_2, \ldots, data_n\}^{2l} \rightarrow \{Tag'\}^l$ محاسبه می‌کند و آن را در زنجیرهٔ قالب‌ها ثبت می‌کند. مهاجم $\mathcal{A}$، درنهایت بخش بارگذاری از چارچوب بررسی شده را اجرا می‌کند. درست است که او یک برچسب صحیح مالکیت داده مد نظر را در زنجیرهٔ قالب‌ها ثبت کرده است، اما کاملاً مشخص است این یک ارسال مجدد از داده ثبت شده قبلی به عنوان یک داده جدید یا تکراری است. در نتیجه، نمی‌توان مهاجم را مالک داده قبلی دانست.

۲) مهاجم $\mathcal{A}$، به عنوان یک داخلی ممتاز، این امکان را دارد تا مجموعهٔ $\{n, ID_i, Data\}$ که در ابر ذخیره شده را به $\{n, ID_i', Data'\}$، و برچسب Tag و تمام امضاهای σ_j را به برچسب Tag' و امضاهای σ_j' تغییر دهد. با وجود اینکه مهاجم $\mathcal{A}$ را یک داخلی ممتاز در نظر گرفتیم، دو مشکل سر راه او وجود دارد. اول، او باید تعداد $th+1$ تا امضای σ_j را جعل کند و همچنین او باید تمام امضاهای جعل شده را در زنجیرهٔ قالب‌ها ثبت کند چون هم امضای استفاده شده جعل ناپذیر است و هم اطلاعات ثبت شده در زنجیرهٔ قالب‌ها غیر قابل تغییر است. در نتیجه می‌توان گفت مهاجم $\mathcal{A}$ هیچ شانسی برای پیروزی در این رویکرد نیز ندارد.

با توجه به دو رویکرد بالا، چارچوب بررسی شده مالکیت داده را ارائه می‌دهد.

امنیت در مقابل داخلی‌های ممتاز (در سمت کاربران ممیزی کننده داده): این رویکرد شبیه به رویکرد دوم در بند قبل است. به عنوان توضیح تکمیلی برای این موضوع، یک فرد داخلی ممتاز که به عنوان مهاجم $\mathcal{A}$ در نظر گرفته می‌شود باید بتواند کنترل $th+1$ تا از کاربران ممیزی کننده داده PA_j یا بیش از $\frac{2}{3}$ از اعضای زنجیرهٔ قالب‌ها را در اختیار داشته باشد که این امر برای مهاجم ممکن نیست.

۲.۴.۴ اثبات مالکیت داده

همانطور که در بخش قبل ادعا شد، چارچوب بررسی شده اثبات مالکیت داده را فراهم می‌کند و هر کاربر می‌تواند مالکیت خود بر داده‌های ثبت

[1] Untraceability

شده در ابر را اثبات کند.

تئوری: چارچوب بررسی شده اثبات مالکیت داده را ارائه می‌دهد و مالک داده می‌تواند ثابت کند که مالک داده مد نظر است به طوری که هیچ شخص دیگری نتواند مدعی مالکیت باشد اگر اطلاعات ثبت شده در زنجیرهٔ قالب‌ها تغییر ناپذیر باشد.

اثبات: مجموعهٔ $\{ID_i, Data, n\}$ در ابر برای همه در دسترس است و هرکسی می‌تواند با دسترسی به داده $Data$ برچسب آن را به صورت $h : \{ID_i, data_1, data_2, \ldots, data_n\}^{2l} \rightarrow \{Tag\}^l$ محاسبه کند و اعتبار مالک داده را وارسی کند. همچنین با توجه به ویژگی‌های زنجیرهٔ قالب‌ها هیچ کس نمی‌تواند برچسب Tag' را مرتبط با همان داده و مجموعهٔ $\{ID', Data, n\}$ به صورت $h : \{ID', data_1, data_2, data_n\}^{2l}\{Tag'\}^l$ محاسبه کند و برچسب Tag' را جایگزین Tag کند. در نتیجه، می‌توان گفت پس از ممیزی و ثبت داده مالک داده به سادگی می‌تواند مالکیت داده را اثبات کند چون اطلاعات ثبت شده در زنجیرهٔ قالب‌ها تغییر ناپذیر است.

۳.۴.۴ صحت داده:

با توجه به ویژگی‌های پروتکل‌های ممیزی داده، چارچوب بررسی شده صحت داده را فراهم می‌کند و هرگونه تغییر در داده اصلی به سادگی قابل بررسی است.

تئوری: چارچوب بررسی شده صحت داده را ارائه می‌دهد و هر تغییر در داده ثبت شده به راحتی آشکار می‌شود.

اثبات: از آنجایی که مجموعه $\{ID_i, Data, n\}$ برای همه در ابر قابل دسترسی است، هر کاربر عمومی شبکه می‌تواند صحت داده را بررسی کند. برای این کار ابتدا برچسب مرتبط با داده $Data$ را به صورت

$$h : \{ID_i, data_1, data_2, \ldots, data_n\}^{2l} \rightarrow \{Tag\}^l$$

محاسبه می‌کند و با برچسب ذخیره شده مرتبط با داده در زنجیرهٔ قالب‌ها مقایسه می‌کند. علاوه بر این هیچ کس نمی‌تواند داده را به گونه‌ای تغییر دهد

$$(h : \{ID_i, data_1', data_2', \ldots, data_n'\}^{2l} \rightarrow \{Tag'\}^l)$$

که همان برچسب قبلی برای داده جدید صادق باشد.

۴.۴.۴ وارسی پذیری عمومی[1]:

چارچوب بررسی شده برای کاربران عمومی شبکه این امکان را فراهم می‌کند تا بتوانند صحت انجام فرآیند ممیزی داده را وارسی کنند. در واقع تمام کاربرن امکان بررسی صحت انجام فرآیند ممیزی داده را دارند.

تئوری: چارچوب بررسی شده وارسی پذیری عمومی را ارائه می‌دهد.

اثبات: با توجه به تشریح چارچوب بررسی شده (علی الخصوص بخش وارسی عمومی) و ویژگی‌های زنجیرهٔ قالب‌ها، بعد از انجام فرآیند ممیزی هر کاربر به داده $Data$ دسترسی دارد و می‌تواند صحت آن را وارسی کند. علاوه بر این، در طول فرآیند ممیزی هر کاربر عمومی می‌تواند به فرآیند ممیزی بپیوندد و صحت داده را وارسی کند.

۵.۴.۴ وارسی پذیری دسته‌ای[2]:

در صورتی که یک داده به چند بخش تقسیم شده باشد یا چند داده توسط یک نفر ارسال شده باشد و پس از انجام موفقیت آمیز فرآیند ممیزی در ابر ذخیره شود، می‌توان صحت تمام داده‌ها را در یک مرحله و به سرعت انجام داد.

تئوری: با توجه به ساختار درخت مرکل، چارچوب بررسی شده از وارسی پذیری دسته‌ای داده حمایت می‌کند و این امکان برای کاربران ممیزی کننده داده و همچنین کاربران عادی شبکه فراهم است تا داده‌هایی با حجم زیاد و یا چندین دادهٔ ثبت شده را به سادگی و به سرعت وارسی کنند.

اثبات: از آنجایی که در ارائه چارچوب بررسی شده بحث شد، صحت یک یا چند داده از طریق روشی سریع با استفاده از محاسبهٔ ریشهٔ درخت مرکل بررسی می‌شوند. در این روش هر کاربر می‌تواند مقادیر $ID_i, Data_1, Data_2, etc.$ را در ابر پیدا کند و با محاسبهٔ رابطهٔ زیر تأیید کند که صحت آنها حفظ شده است یا خیر.

$$h : \{ID_i, data_{1,1}, data_{1,2}, \ldots, data_{1,n}, data_{2,1}, data_{2,2}, \ldots, data_{2,n}, \ldots\}^{2l} \rightarrow \{Tag\}^l$$

همچنین با یافتن برچسب محاسبه شده در زنجیرهٔ قالب‌ها اعتبار و مالکیت داده را نیز وارسی کرد.

۵ مقایسه کارهای بررسی شده

از آنجایی که تمرکز اصلی این مقاله بر بررسی روش‌های ارائهٔ امنیت و حفظ حریم خصوصی در پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها است، این بخش جمع بندی و بحث مختصری روی پروتکل‌های بررسی شده ارائه می‌دهد. موارد ذکر شده در ادامهٔ این بخش ارائه شده‌اند. مقایسهٔ کلیات پروتکل‌های بررسی شده در جدول ۲ آورده شده است.

۱.۵ روش ذخیره‌سازی

پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها بررسی شده نشان دادند که تکیه بر زنجیرهٔ قالب‌ها و استفاده از آن الزاماً به معنای ذخیره‌سازی اطلاعات و داده‌ها در زنجیرهٔ قالب‌ها نیست، لذا ثبت اطلاعات با حجم بالا در زنجیرهٔ قالب‌های عمومی زمان‌بر و مستلزم پرداخت هزینه زیادی

[1] Public verifyability

[2] Batch verification

جدول ۲. مقایسهٔ کلیات پروتکل‌های بررسی شده

پروتکل	نوع داده	نوع کاربران	نوع کاربران ممیزی یا نحوه انتخاب	روش ممیزی	محل ذخیره‌سازی
SeShare [۲۵]	داده شخصی	شخص	از پیش تعریف شده	امضا	ابر
DAB [۲۶]	کلان داده	اینترنت اشیاء (شهر هوشمند)	تصادفی	اجماع	ابر
CPVPA [۲۷]	داده شخصی	شخص	از پیش تعریف شده	چالش-پاسخ	ابر
ژو [۲۸]	داده شخصی	شخص	قرارداد هوشمند	چالش-پاسخ	ابر
ژو [۲۹]	داده شخصی	شخص	قرارداد هوشمند	چالش-پاسخ	ابر
لی [۳۰]	داده شخصی	شخص	قرارداد هوشمند	چالش-پاسخ	ابر
لی [۵]	کلان داده شخص	قرارداد هوشمند	چالش-پاسخ	ابر	
ژائو [۶]	دادهٔ محیطی	اینترنت اشیاء	از پیش تعریف شده	امضا	ابر
Dredas [۱۴]	دادهٔ محیطی	اینترنت اشیاء صنعتی	قرارداد هوشمند	چالش-پاسخ	ابر
وَنگ [۱۷]	داده شخصی	شخص	از پیش تعریف شده	چالش-پاسخ	ابر
NI-PPDP [۳۱]	داده مالی	شخص	قرارداد هوشمند	امضا	ابر
لی [۱۵]	داده شخصی	شخص	از پیش تعریف شده	چالش-پاسخ	ابر
زَنگ [۳۲]	داده شخصی	شخص	قرار داد هوشمند	چالش-پاسخ	ابر
DA-DS [۳۳]	داده مالی	شخص	از پیش تعریف شده	امضا	زنجیرهٔ قالب‌ها
BB-DA [۳۴]	داده مالی	شخص	از پیش تعریف شده	امضا	زنجیرهٔ قالب‌ها
ZK-DAP [۳۵]	داده شخصی	شخص	قرارداد هوشمند	امضا	زنجیرهٔ قالب‌ها

است که برای کاربران عادی مقرون به صرفه نیست. همچنین یکی از دلایل برون‌سپاری داده عدم پشتیبانی محلی از حافظه کافی است. لذا ذخیره و نگهداری حجم زیادی از داده در زنجیرهٔ قالب‌ها کار مقرون به صرفه‌ای نیست و اکثر پروتکل‌هایی که با هدف ذخیره‌سازی (علاوه بر ممیزی) مبتنی بر زنجیرهٔ قالب‌ها ارائه می‌شوند شواهدی از اطلاعات را در زنجیرهٔ قالب‌ها ارسال/ذخیره می‌کنند و داده اصلی را در یک سرور ابری خارجی ذخیره می‌کنند. در نهایت برای دسترسی به اطلاعات درخواست را به سرور ابری ارسال می‌کنند و برای بررسی صحت داده ذخیره شده، اثبات ارسال و وجود داده در ابر و اطمینان از عدم تغییر آن از زنجیرهٔ قالب‌ها کمک می‌گیرند (ساز و کار ذکر شده جامعیت ندارد، ولی غالب پروتکل‌های ذخیره‌سازی از این ساز و کار استفاده می‌کنند).

۲.۵ روش ممیزی

وجه مشترک اکثر پروتکل‌های ارائه شده در سال‌های اخیر عدم اتکا به کاربران ممیزی از پیش تعیین شده است و مهم‌ترین دلیل ذکر شده برای این امر عدم وجود شخص معتمد در دنیای واقعی است. پروتکل‌های ممیزی داده که از کاربران ممیزی از پیش تعیین نشده (استفاده از کاربران عادی شبکه، یا استفاده از قرارداد هوشمند به جای کاربر ممیزی) استفاده می‌کنند پروتکل‌های ممیزی دادهٔ عمومی نامیده می‌شوند.

۳.۵ حذف کاربران ممیزی کننده داده از پیش تعین شده

این موضوع می‌تواند جذابیت زیادی بهمراه داشته باشد، چرا که تمام فرآیند ممیزی به صورت خودکار و با استفاده از اجماع مبتنی بر قراردادهای هوشمند انجام می‌شود. ولی به عنوان یکی از ضعف‌های این روش می‌توان به عدم پشتیبانی قراردادهای هوشمند از محاسبات سنگین و عدم امکان استفاده از مقادیر محرمانه در قراردادهای هوشمند اشاره کرد. چالش دیگر استفاده از قرارداد هوشمند به جای کاربران را می‌توان نیاز به ظرافت و دقت بیشتر در طراحی دانست، چرا که، در ضمن طراحی کارا باید قرارداد به گونه‌ای تنظیم شود که لازم به ذخیره‌سازی هیچ مقدار رازی در آن نباشد (به طور مثال کلید خصوصی برای امضا). پس می‌توان اینطور نتیجه گرفت که نمی‌توان در پروتکل‌هایی که از توابع و فرآیندهایی با بار محاسباتی زیاد استفاده شده است قرارداد هوشمند را جایگزین کاربران کرد. لذا نمی‌توان حکم قطعی بر حذف منابع انسانی و یا ارائه خودمختاری کامل در پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها داد.

۴.۵ ملاحظات امنیتی

یکی از مواردی که تقریبا در پروتکل‌های بررسی شده کمتر مورد توجه قرار گرفته است ملاحظات امنیتی (شامل محرمانگی داده در هنگام ممیزی، پس از ممیزی، حفظ حریم خصوصی مالکان داده، حفظ حریم خصوصی استفاده کنندگان از داده، حضور کاربران ممیزی بدخواه و...) است. به

جدول ۳. مقایسهٔ ویژگی‌های امنیتی پروتکل‌های بررسی شده

پروتکل	صحت در تمام زمان‌ها	حفظ حریم خصوصی فرستنده و گیرنده	محرمانگی داده تحت ممیزی	مقاومت در مقابل کاربران ممیزی داده مخرب	امکان پردازش روی داده محرمانه
ژو [۲۹]	✓	✗	✗	✗	✗
لی [۳۰]	✓	✗	✗	✗	✗
ژائو [۶]	✓	✗	✓	✗	✗
Dredas [۱۴]	✓	✗	✗	✓	✗
وَنگ [۱۷]	✓	✗	✗	✓	✗
لی [۱۵]	✓	✗	✗	✗	✗
ژَنگ [۳۲]	✓	✗	✗	✓	✗
DA-DS [۳۳]	✓	✗	✓	✗	✗
BB-DA [۳۴]	✓	✓	✓	✓	✓
ZK-DAP [۳۵]	✓	✓	✓	✓	✓

نظر می‌رسد به دلیل عدم نیاز و یا تحمیل سربار مخابراتی یا محاسباتی به نهادهای حاضر در پروتکل‌ها ملاحظات امنیتی در نظر گرفته نشده است.

همانطور که در جدول ۳ نشان داده شده است، با وجود ویژگی‌های جذاب زنجیرهٔ قالب‌ها مثل شفافیت، تغییرناپذیری و خودمختاری معمولا زنجیرهٔ قالب‌ها محل خوبی برای ذخیره‌سازی داده‌ها با حجم زیاد نیست و مهم‌ترین دلیل این موضوع هزینهٔ قابل توجه ثبت داده با حجم زیاد در زنجیرهٔ قالب‌ها است (این استدلال مبتنی بر پروتکل‌های بررسی شده در حوزه ممیزی داده است). لذا با توجه این موضوع می‌توان نتیجه‌گیری کرد ذخیره‌سازی داده‌ها همچنان باید در بستری ارزان و مقرون به صرفه مثل ابر باشد و در کنار آن از زنجیرهٔ قالب‌ها به عنوان یک پایگاه داده قابل دسترس و تغییرناپذیر برای ذخیره‌سازی شواهد و اثبات‌های مرتبط با داده‌های ذخیره شده در ابر مورد استفاده قرار گیرد. جدول ۳ ویژگی‌های امنیتی مد نظر ما در این مقاله را در چند پروتکل اخیر ارائه شده مرور و مقایسه می‌کند. در این جدول ۱) فقط در یکی از پروتکل‌های بررسی شده از محرمانگی داده پشتیبانی می‌شود (ژائو [۴]) به گونه‌ای که برچسب منطبق بر داده رمزنگاری شده ایجاد می‌شود، ۲) پروتکل‌هایی در مقابل کاربران ممیزی کننده داده بدخواه مقاوم هستند که از قرارداد هوشمند به جای کاربران ممیزی استفاده می‌کنند و فرض بدخواه بودن کاربر ممیزی در هیچ یک از پروتکل‌های بررسی شده در نظر گرفته نشده است و ۳) نکته قابل توجه عدم پشتیبانی تمام پروتکل‌های بررسی شده از امکان «پردازش روی داده محرمانه» و عدم پشتیبانی از «حفظ حریم خصوصی کاربران» است.

۵.۵ هزینه محاسبات، تأخیر و سربار مخابراتی

در این بخش مقایسه‌ای جزئی‌تر از پروتکل‌های بررسی شده در بخش ۲.۳ ارائه می‌کنیم. این مقایسه‌ها با توجه به جزئیات پروتکل‌های بررسی شده ارائه می‌شود و شامل مقایسهٔ هزینه محاسبات، زمان اجرای پروتکل (تأخیر) و سربار مخابراتی هر پروتکل در سمت کاربر، سمت کاربران ممیزی و سمت مدیر سامانه است. برای ارائهٔ مقایسه‌ای عادلانه فرض می‌کنیم هر گروه از نهادهای مختلف حاضر در پروتکل مشابه و طبق [۴۲] در نظر گرفته شده‌اند. در همین راستا، شرایط زیر برای هر گروه از نهادها در نظر گرفته شده است.

- کاربران عادی شبکه: تلفن‌های هوشمند Hisilicon Kirin 925، با پردازنده 2.45 GHz و حافظه 3 GB به طوری که زمان اجرای الگوریتم‌های رمزنگاری برای این دسته از کاربران به صورت $T_{Pair} = ۳۶۱ms$، $T_{Pow} = ۲۰۰ms$، $T_{Inv} = ۲۰۰ms$، $T_{Mul} = ۰٫۷ms$ و $T_H = ۱۱٫۲ms$ است.
- کاربران ممیزی کننده داده: رایانه‌هایی با پردازنده‌های Intel I7-4460S, 3.1 GHz و 16 GB حافظه به طوری که زمان اجرای الگوریتم‌های رمزنگاری برای این دسته از کاربران به صورت $T_{Pair} = ۵٫۵ms$، $T_{Pow} = ۲ms$، $T_{Inv} = ۲ms$، $T_{Mul} = ۰٫۱ms$ و $T_H = ۰٫۷ms$ است.
- سرور ارائه خدمات یا CSP: رایانه‌ای با پردازندهٔ Intel I7-6700k, 4.0 GHz و 32 GB حافظه به طوری که زمان اجرای الگوریتم‌های رمزنگاری برای CSP به صورت $T_{Pair} = ۴٫۱ms$، $T_{Pow} = ۱٫۷ms$، $T_{Inv} = ۱٫۷ms$، $T_{Mul} = ۰٫۱ms$ و $T_H = ۰٫۵ms$ است.

هزینه محاسبات: در این بخش میزان بار محاسباتی هر یک از نهادهای حاضر در پروتکل را به تفکیک بیان می‌کنیم و در شرایط مشابه با سایر پروتکل‌ها مقایسه می‌کنیم. این مقایسه در جدول ۴ آورده شده است.

جدول ۴. مقایسهٔ هزینهٔ محاسبات در پروتکل‌های بررسی شده

پروتکل	کاربر (مالک داده)		کاربران ممیزی کننده داده		مالک سامانه (CSP)	
	تولید برچسب	بارگذاری	وارسی	ممیزی	وارسی	ثبت
DAB [۲۶]	$۲Cost_{Pow}+۲Cost_{Mul}+Cost_H$	$Cost_{Pow}+Cost_{Inv}+۲Cost_{Mul}$	$۴Cost_{Pow}+۲Cost_{Mul}$	$۸Cost_{Pow}+۲Cost_{Inv}+۷Cost_{Mul}+۲Cost_H$	$۳Cost_{Pow}+Cost_{Mul}$	$۳Cost_{Pair}+۴Cost_{Pow}+۲Cost_{Mul}$
CPVPA [۲۷]	$Cost_{Pow}+۲Cost_H$	$۵Cost_{Pow}+۴Cost_{Mul}+۲Cost_H$	$۴Cost_{Pair}+۴Cost_{Pow}+۲Cost_H$	$Cost_{Pow}+Cost_{Mul}+۴Cost_H$	$۴Cost_{Pair}+۵Cost_{Pow}+۲Cost_{Mul}+۷Cost_H$	
ژو [۲۸]	$۲Cost_{Pow}+Cost_{Mul}+Cost_H$	$۲Cost_{Pair}+۳Cost_{Pow}+۲Cost_{Mul}+۳Cost_H$	$۲Cost_H$	$۲Cost_{Pair}+Cost_{Pow}+Cost_{Mul}$	$۲Cost_{Pair}+Cost_{Pow}+Cost_{Mul}+۳Cost_H$	
لی [۳۰]	$۲Cost_H$	$۶Cost_{Mul}+۲Cost_H$	$Cost_{Inv}+۴Cost_{Mul}+Cost_H$		$۴Cost_{Pair}+۸Cost_{Mul}+۶Cost_H$	
لی [۵]	$۲Cost_{Pow}+Cost_{Mul}+Cost_H$	$Cost_{Pow}+Cost_{Inv}+۲Cost_{Mul}$	$۳Cost_{Pow}+Cost_{Mul}$	$۵Cost_{Pow}+۲Cost_{Inv}+۵Cost_{Mul}+۲Cost_H$	$۲Cost_{Pair}$	
ژائو [۶]	$۲Cost_{Pair}+۶Cost_{Mul}+۳Cost_H$		$۲Cost_{Pair}+۳Cost_{Mul}+۲Cost_H$		$۲Cost_{Pair}+۳Cost_{Mul}+۲Cost_H$	
NI-PPDP [۳۱]	$۴Cost_{Pow}+۲Cost_{Mul}+۲Cost_H$		$Cost_{Pair}+۲Cost_{Pow}+۲Cost_{Mul}+۳Cost_H$		$۲Cost_{Pair}+۴Cost_{Pow}+۲Cost_{Mul}+۴Cost_H$	
DA-DS [۳۳]	$۲Cost_{Pow}+Cost_{Mul}+Cost_H$		$Cost_{Pow}+Cost_{Inv}+۳Cost_{Mul}+Cost_H$		$۳Cost_{Pow}+۲Cost_{Mul}$	
ژی [۴]	$۲Cost_{Pow}+Cost_{Inv}+Cost_{Mul}$		$۲Cost_{Pair}+۴Cost_{Pow}+۵Cost_{Mul}$		$۲Cost_{Pair}+۵Cost_{Pow}+۳Cost_{Mul}$	
BCSAI [۱۶]	$۶Cost_{Pow}+Cost_{Inv}+۳Cost_{Mul}+Cost_H$		$۲Cost_{Pair}+۶Cost_{Pow}+۶Cost_{Mul}+۳Cost_H$		$۲Cost_{Pow}+Cost_{Mul}+Cost_H$	
BB-DA [۳۴]	$Cost_H$	$۲Cost_{Pow}+Cost_{Mul}$	$Cost_H$	$Cost_{Pow}+Cost_{Inv}+۳Cost_{Mul}+Cost_H$	$۳Cost_{Pow}+(۳+th)Cost_{Mul}+Cost_H$	$thCost_{Mul}$
ZK-DAP [۳۵]	$Cost_{Pair}+۸Cost_{Mul}$		$۲Cost_{Pow}$		–	

تأخیر: با توجه جدول ۴، مقایسه تأخیر برای بخش‌های مختلف پروتکل‌های ممیزی مبتنی بر زنجیرهٔ قالب‌های بررسی شده در جدول ۵ ارائه می‌شوند.

سربار مخابراتی: طول بیت ارسالی برای هر قالب از داده که برای ممیزی ارسال می‌شود در جدول ۶ محاسبه شده است. همانطورکه مشخص است، با توجه به کمتر بودن طول بیت برای هر قالب از داده ارسالی نتیجه می‌شود که کارمزد پرداختی توسط کاربران (سمت کاربر) پروتکل‌ها کمتر است.

۶ جمع‌بندی و کارهای آینده

در این مقاله، ابتدا به تعریف کلی زنجیرهٔ قالب‌ها، به عنوان یک فناوری نوظهور و جذاب دههٔ اخیر، پرداختیم و پس از مروری کوتاه بر پروتکل‌های ممیزی داده، یک چارچوب کلی برای آنها را بررسی کردیم به گونه‌ای که می‌توان پروتکل‌های بررسی شده را مثال‌هایی از آن دانست.

کارهای آینده: به جرات می‌توان گفت انعطاف پذیری ریاضیات و تعدد نخستینه‌های رمزنگاشتی پیاده سازی هر ایده‌ای در طراحی پروتکل‌های امنیتی را ممکن می‌کند. حتی اگر هیچ نخستینهٔ رمزنگاشتی برای استفاده در یک پروتکل وجود نداشته باشد، با اتکا به انعطاف پذیری ریاضیات می‌توان آن را طراحی کرد. حذف نهاد مرکزی و واگذاری وظایف او به افراد یک جامعه همچنان یک چالش بزرگ در نظر گرفته می‌شود. در این میان، حفظ حریم خصوصی مشروط در جایی که نهاد مرکز وجود ندارد، سخت است. همچنین بسیاری از روش‌های جذاب و کاربردی وجود دارند که به سختی با فناوری زنجیرهٔ قالب‌ها سازگار می‌شوند. به طور خاص‌تر می‌توان به پروتکل‌های ممیزی داده متمرکزی اشاره کرد که به شدت وابسته به نحوه توزیع کلید متمرکز هستند (مخصوصا برای کاربران ممیزی کننده داده) و نمی‌توان به سادگی نسخه مبتنی بر زنجیرهٔ قالب‌های آنها را ارائه داد. در مقابل، در صورت حل مشکلات فوق، توزیع شدگی بالا در تقابل با قانونگذاری قرار می‌گیرد و برقراری تعادل بین توزیع‌شدگی و قانون گذاری نیازمند مطالعه و هماهنگی‌های زیادی بین نهادهای علمی و قانونی است.

به طور خاص در رابطه با پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها می‌توان چندین کاربرد عملیاتی در نظر گرفت. در مواردی که تأخیر از دید کاربر قابل چشم‌پوشی است و هزینهٔ ذخیره‌سازی اطلاعات مقرون بصرفه است، می‌توان پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها را به عنوان یک راه‌کار قابل قبول برای استفاده به عنوان یک پوستهٔ کنترل کننده در تراکنش‌های مالی و یا اطلاعات حساس در نظر گرفت. در مقابل، پیشنهاد می‌شود این نوع پروتکل‌ها در بخش‌هایی که سرعت و توان عملیاتی مورد توجه است و امکان پرداخت هزینه ثبت اطلاعات در زنجیرهٔ قالب‌ها بسهولت مقدور نیست (به طور مثال: شبکه‌های حسگر اقتضایی، یا فناوری‌های شناسایی فرکانس رادیویی) استفاده از این پروتکل توجیهی ندارد و در صورت استفاده نیز باید از چندین زیرساخت و یا فناوری‌های کمکی استفاده کرد.

به عنوان سخن آخر، به پژوهشگرانی که قصد دارند متمرکز بر پروتکل‌های امنیتی و به ویژه پروتکل‌های ممیزی داده مبتنی بر زنجیرهٔ قالب‌ها باشند توصیه می‌شود در ضمن حل مشکلات امنیتی و ارائه راهکارهای خلاقانه، نیازمندی‌های قانونی (به طور خاص در حوزه پروتکل‌های امنیتی، حفظ حریم خصوصی مشروط) نیز توجه ویژه داشته باشند.

مراجع

[1] Pourbabak, Hajir, Chen, Tao, and Su, Wencong. Centralized, decentralized, and distributed control for energy internet. in *The Energy Internet*, pp. 3–19. Elsevier, 2019.

[2] Akhtaruzzaman, Md, Hasan, Mohammad Kamrul, Kabir, S Rayhan, Abdullah, Siti Norul Huda Sheikh, Sadeq, Muhammad Jafar, and Hossain, Eklas. Hsic bottleneck based distributed deep learning model for load forecasting in smart grid with a comprehensive survey. *IEEE Access*, 8:222977–223008, 2020.

[3] Nakamoto, Satoshi. Bitcoin: A peer-to-peer electronic cash system. 2008.

[4] Zheng, Xiaokun, Zhao, Yanqi, Li, Huilin, Chen, Ruonan, and Zheng, Dong. Blockchain-based verifiable privacy-preserving data classification protocol for medical data. *Computer Standards & Interfaces*, 82:103605, 2022.

[5] Li, Jiaxing, Wu, Jigang, Jiang, Guiyuan, and Srikanthan, Thambipillai. Blockchain-based public auditing for big data in cloud storage. *Information Processing & Management*, 57(6):102382, 2020.

[6] Zhao, Quanyu, Chen, Siyi, Liu, Zheli, Baker, Thar, and Zhang, Yuan. Blockchain-based privacy-preserving remote data integrity checking scheme for iot information systems. *Information Processing & Management*, 57(6):102355, 2020.

[7] Chowdhury, Mohammad Jabed Morshed, Colman, Alan, Kabir, Muhammad Ashad, Han, Jun, and Sarda, Paul. Blockchain versus database: A critical analysis. in *2018 17th IEEE International conference on trust, security and privacy in computing and communications/12th IEEE international conference on big data science and engineering (TrustCom/BigDataSE)*, pp. 1348–1353. IEEE, 2018.

[8] Zheng, Zibin, Xie, Shaoan, Dai, Hong-Ning, Chen, Xiangping, and Wang, Huaimin. Blockchain challenges and opportunities: A survey. *International journal of web*

جدول ۵. مقایسهٔ میزان تأخیر (زمان اجرا) در پروتکل های بررسی شده (میلی ثانیه)

پروتکل	کاربر / مالک داده (تولید برچسب + بارگذاری)	کاربر ممیزی کننده داده (وارسی + ممیزی)	مالک سامانه /CSP (وارسی + ثبت)
DAB [۲۶]	$3T_{Pow}+T_{Inv}+4T_{Mul}+T_H=$ $3\times200+200+4\times0/7+11/2=814$	$12T_{Pow}+2T_{Inv}+9T_{Mul}+2T_H=$ $12\times2+2\times2+9\times0/1+2\times1/4=30/3$	$3Pair+7T_{Pow}+3T_{Mul}=$ $3\times4/1+7\times1/7+3\times0/1=24/5$
CPVPA [۲۷]	$6T_{Pow}+4T_{Mul}+4T_H=$ $6\times200+4\times0/7+4\times11/2=1247/6$	$4T_{Pair}+5T_{Pow}+T_{Mul}+6T_H=$ $4\times5/5+5\times2+0/1+6\times0/7=36/3$	$4T_{Pair}+5T_{Pow}+2T_{Mul}+7T_H=$ $4\times4/1+5\times1/7+2\times0/1+7\times0/5=28/6$
ژو [۲۸]	$2T_{Pair}+2T_{Pow}+3T_{Mul}+4T_H=$ $2\times361+2\times200+3\times0/7+4\times11/2=1368/9$	$2T_{Pair}+T_{Pow}+T_{Mul}+2T_H=$ $2\times5/5+1/7+0/1+2\times0/7=14/5$	$2T_{Pair}+T_{Pow}+2T_{Mul}+7T_H=$ $2\times4/1+1/7+2\times0/1+7\times0/5=13/5$
لی [۳۰]	$6T_{Mul}+4T_H=$ $6\times0/7+4\times11/2=49$	$T_{Inv}+4T_{Mul}+T_H=$ $2+4\times0/1+0/7=3/1$	$4T_{Pair}+8T_{Mul}+6T_H=$ $4\times4/1+8\times0/1+6\times0/5=20/2$
لی [۵]	$3T_{Pow}+T_{Inv}+3T_{Mul}+T_H=$ $3\times200+200+3\times0/7+11/2=832/2$	$8T_{Pow}+2T_{Inv}+6T_{Mul}+2T_H=$ $8\times2+2\times2+6\times0/1+2\times0/7=22$	$2T_{Pair}=2\times4/1=8/2$
ژائو [۶]	$2T_{Pair}+6T_{Mul}+3T_H=$ $2\times361+6\times0/7+3\times11/2=759/8$	$2T_{Pair}+3T_{Mul}+2T_H=$ $2\times5/5+3\times0/1+2\times0/7=12/7$	$2T_{Pair}+3T_{Mul}+2T_H=$ $2\times4/1+3\times0/1+2\times0/5=9/5$
NI-PPDP [۳۱]	$4T_{Pow}+2T_{Mul}+2T_H=$ $4\times200^2\times0/7+2\times11/2=8238$	$T_{Pair}+2T_{Pow}+2T_{Mul}+3T_H=$ $5/5+2\times2+2\times0/1+3\times0/7=11/8$	$2T_{Pair}+4T_{Pow}+2T_{Mul}+4T_H=$ $2\times4/1+4\times1/7+2\times0/1+4\times0/5=17/2$
DA-DS [۳۳]	$2T_{Pow}+T_{Mul}+T_H=$ $2\times200+0/7+11/2=411/9$	$T_{Pow}+T_{Inv}+3T_{Mul}+T_H=$ $2+2+3\times0/1+0/7=5$	$3T_{Pow}+2T_{Mul}=$ $3\times1/7+2\times0/1=5/3$
ژی [۴]	$2T_{Pow}+T_{Inv}+T_{Mul}=$ $2\times200+200+0/7=600/7$	$2T_{Pair}+4T_{Pow}+5T_{Mul}=$ $2\times5/5+4\times2+5\times0/1=19/5$	$2T_{Pair}+5T_{Pow}+3T_{Mul}=$ $2\times4/1+5\times1/7+3\times0/1=17$
BCSAI [۱۶]	$6T_{Pow}+T_{Inv}+3T_{Mul}+T_H=$ $6\times200+200+3\times0/7+11/2=1413/3$	$2T_{Pair}+6T_{Pow}+6T_{Mul}+3T_H=$ $2\times5/5+6\times2+6\times0/1+3\times0/7=25/7$	$2T_{Pow}+T_{Mul}+T_H=$ $2\times1/7+0/1+0/5=4$
BB-DA [۳۴]	$2T_{Pow}+T_{Mul}+T_H=$ $2\times200+0/7+11/2=411/9$	$T_{Pow}+T_{Inv}+3T_{Mul}+2T_H=$ $2+2+3\times0/1+2\times0/7=5/7$	$3T_{Pow}+(3+2th)T_{Mul}+T_H=$ $3\times1/7+3\times0/1+2th\times0/1+0/5=$ $5/9+0/2th=7/1(\ ifth=6)$
ZK-DAP [۳۵]	$2T_{Pow}+T_{Mul}+T_H=$ $2\times200+0/7+11/2=411/9$	$T_{Pow}+T_{Inv}+3T_{Mul}+2T_H=$ $2+2+3\times0/1+2\times0/7=5/7$	$3T_{Pow}+(3+2th)T_{Mul}+T_H=$ $3\times1/7+3\times0/1+2th\times0/1+0/5=$ $5/9+0/2th=7/1(\ ifth=6)$

[12] Banaeian Far, Saeed and Rajabzadeh Asaar, Maryam. A blockchain-based quantum-secure reporting protocol. *Peer-to-Peer Networking and Applications*, 14(5):2992–3011, 2021.

[13] Banaeian Far, Saeed, Imani Rad, Azadeh, and Rajabzade Asaar, Maryam. Bb-csp: an efficient blockchain-based collective salary payment framework using weighted functional encryption. *SN Computer Science*, 3(5):408, 2022.

[14] Fan, Kuan, Bao, Zijian, Liu, Mingxi, Vasilakos, Athanasios V, and Shi, Wenbo. Dredas: Decentralized, reliable and efficient remote outsourced data auditing scheme with blockchain smart contract for industrial iot. *Future and grid services*, 14(4):352–375, 2018.

[9] Bamakan, Seyed Mojtaba Hosseini, Motavali, Amirhossein, and Bondarti, Alireza Babaei. A survey of blockchain consensus algorithms performance evaluation criteria. *Expert Systems with Applications*, 154:113385, 2020.

[10] Nguyen, Giang-Truong and Kim, Kyungbaek. A survey about consensus algorithms used in blockchain. *Journal of Information processing systems*, 14(1), 2018.

[11] Merkle, Ralph C. A digital signature based on a conventional encryption function. in *Conference on the theory and application of cryptographic techniques*, pp. 369–378. Springer, 1987.

جدول ۶. مقایسهٔ سربار مخابراتی در پروتکل‌های بررسی شده (بیت بازای هر قالب)

پروتکل	کاربر / مالک داده	کاربر ممیزی کننده داده	مالک سامانه / CSP
DAB [۲۶]	$۶\|G_۱\|= ۶ \times ۱۰۲۴ = ۶۱۴۴$	$۹\|G_۱\|= ۹ \times ۱۰۲۴ = ۹۲۱۶$	$۴\|G_۱\|+\|G_۲\|= ۴\times۱۰۲۴+۲۵۶ = ۴۳۵۲$
CPVPA [۲۷]	$۴\|G_۱\|= ۴ \times ۱۰۲۴ = ۴۰۹۶$	$۴\|G_۱\|= ۴ \times ۱۰۲۴ = ۴۰۹۶$	$۴\|G_۱\|= ۴ \times ۱۰۲۴ = ۴۰۹۶$
ژو [۲۸]	$۲\|G_۱\|+\|G_۲\|= ۲\times۱۰۲۴+۲۵۶ = ۲۳۰۴$	$۵\|G_۱\|= ۵ \times ۱۰۲۴ = ۵۱۲۰$	$۵\|G_۱\|= ۵ \times ۱۰۲۴ = ۵۱۲۰$
لی [۳۰]	$۴\|G_۱\|= ۴ \times ۱۰۲۴ = ۴۰۹۶$	$۴\|G_۱\|= ۴ \times ۱۰۲۴ = ۴۰۹۶$	$۸\|G_۱\|= ۸ \times ۱۰۲۴ = ۷۱۹۲$
لی [۵]	$۵\|G_۱\|+\|G_۲\|= ۵\times۱۰۲۴+۲۵۶ = ۵۳۷۶$	$۲\|G_۱\|+\|G_۲\|= ۲\times۱۰۲۴+۲۵۶ = ۲۳۰۴$	$۵\|G_۱\|+\|G_۲\|= ۵\times۱۰۲۴+۲۵۶ = ۵۳۷۶$
ژائو [۶]	$۴\|G_۱\|+\|G_۲\|= ۴\times۱۰۲۴+۲۵۶ = ۴۳۵۲$	$۴\|G_۱\|= ۴ \times ۱۰۲۴ = ۴۰۹۶$	$۲\|G_۱\|+۲\|G_۲\|=$ $۲ \times ۱۰۲۴ + ۲ \times ۲۵۶ = ۲۵۶۰$
NI-PPDP [۳۱]	$۴\|G_۱\|= ۴ \times ۱۰۲۴ = ۴۰۹۶$	$۵\|G_۱\|= ۵ \times ۱۰۲۴ = ۵۱۲۰$	–
لی [۱۵]	$۲\|G_۱\|+۲\|G_۲\|=$ $۲ \times ۱۰۲۴ + ۲ \times ۲۵۶ = ۲۵۶۰$	$۲(۲\|G_۱\|+۲\|G_۲\|) =$ $۲(۲ \times ۱۰۲۴ + ۲ \times ۲۵۶) =$ $۲ \times ۲۵۶۰ = ۵۱۲۰$	$۴\|G_۱\|= ۴ \times ۱۰۲۴ = ۴۰۹۶$
DA-DS [۳۳]	$۲\|G_۱\|+\|G_۲\|= ۲\times۱۰۲۴+۲۵۶ = ۲۳۰۴$	$۴\|G_۱\|= ۴ \times ۱۰۲۴ = ۴۰۹۶$	$۴\|G_۱\|= ۴ \times ۱۰۲۴ = ۴۰۹۶$
BB-DA [۳۴]	$۲\|G_۱\|+\|G_۲\|= ۲\times۱۰۲۴+۲۵۶ = ۲۳۰۴$	$۳\|G_۱\|= ۳ \times ۱۰۲۴ = ۳۰۷۲$	$۳\|G_۱\|= ۳ \times ۱۰۲۴ = ۳۰۷۲$
ZK-DAP [۳۵]	$۲\|G_۱\|+\|G_۲\|= ۲\times۱۰۲۴+۲۵۶ = ۲۳۰۴$	$۳\|G_۱\|= ۳ \times ۱۰۲۴ = ۳۰۷۲$	$۳\|G_۱\|= ۳ \times ۱۰۲۴ = ۳۰۷۲$

finitechain: A multi-chain architecture with distributed auditing of sidechains for public blockchains. in *Blockchain–ICBC 2018: First International Conference, Held as Part of the Services Conference Federation, SCF 2018, Seattle, WA, USA, June 25-30, 2018, Proceedings 1*, pp. 47–60. Springer, 2018.

[21] Chen, Jing, Yao, Shixiong, Yuan, Quan, He, Kun, Ji, Shouling, and Du, Ruiying. Certchain: Public and efficient certificate audit based on blockchain for tls connections. in *IEEE INFOCOM 2018-IEEE conference on computer communications*, pp. 2060–2068. IEEE, 2018.

[22] Zheng, Rongyue, Jiang, Jianlin, Hao, Xiaohan, Ren, Wei, Xiong, Feng, and Ren, Yi. bcbim: A blockchain-based big data model for bim modification audit and provenance in mobile cloud. *Mathematical problems in engineering*, 2019(1):5349538, 2019.

[23] Ahmad, Ashar, Saad, Muhammad, Njilla, Laurent, Kamhoua, Charles, Bassiouni, Mostafa, and Mohaisen, Aziz. Blocktrail: A scalable multichain solution for blockchain-based audit trails. in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, pp. 1–6. IEEE, 2019.

[24] Huang, Hui, Chen, Xiaofeng, and Wang, Jianfeng. Blockchain-based multiple groups data sharing with anonymity and traceability. *Science China Information Generation Computer Systems*, 110:665–674, 2020.

[15] Li, Hongtao, Guo, Feng, Wang, Lili, Wang, Jie, Wang, Bo, and Wu, Chuankun. A blockchain-based public auditing protocol with self-certified public keys for cloud data. *Security and Communication Networks*, 2021(1):6623639, 2021.

[16] Li, Angtai, Tian, Guohua, Miao, Meixia, and Gong, Jianpeng. Blockchain-based cross-user data shared auditing. *Connection science*, 34(1):83–103, 2022.

[17] Wang, Han, Wang, Xu An, Xiao, Shuai, and Liu, JiaSen. Decentralized data outsourcing auditing protocol based on blockchain. *Journal of Ambient Intelligence and Humanized Computing*, 12(2):2703–2714, 2021.

[18] Kaaniche, Nesrine and Laurent, Maryline. A blockchain-based data usage auditing architecture with enhanced privacy and availability. in *2017 IEEE 16th International Symposium on Network Computing and Applications (NCA)*, pp. 1–5. IEEE, 2017.

[19] Narula, Neha, Vasquez, Willy, and Virza, Madars. {zkLedger}:{Privacy-Preserving} auditing for distributed ledgers. in *15th USENIX symposium on networked systems design and implementation (NSDI 18)*, pp. 65–80, 2018.

[20] Hwang, Gwan-Hwan, Chen, Po-Han, Lu, Chun-Hao, Chiu, Chun, Lin, Hsuan-Cheng, and Jheng, An-Jie. In-

traceable transactions. *Journal of Information Security and Applications*, 73:103429, 2023.

[35] Far, Saeed Banaeian, Asaar, Maryam Rajabzadeh, and Haghbin, Afrooz. Zero-knowledge-based distributed auditing protocol. *Security and Privacy*, 6(3):e289, 2023.

[36] Banaeian Far, Saeed, Rajabzadeh Asaar, Maryam, and Haghbin, Afrooz. A generic framework for blockchain-assisted on-chain auditing for off-chain storage. *International Journal of Information Security*, pp. 1–29, 2024.

[37] Far, Saeed Banaeian, Asaar, Maryam Rajabzadeh, and Haghbin, Afrooz. Distributed auditing protocol for untraceable transactions. *Journal of Information Security and Applications*, 73:103429, 2023.

[38] Far, Saeed Banaeian, Asaar, Maryam Rajabzadeh, and Haghbin, Afrooz. Zero-knowledge-based distributed auditing protocol. *Security and Privacy*, 6(3):e289, 2023.

[39] Camenisch, Jan, Drijvers, Manu, and Dubovitskaya, Maria. Practical uc-secure delegatable credentials with attributes and their application to blockchain. in *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 683–699, 2017.

[40] Camenisch, Jan, Drijvers, Manu, Gagliardoni, Tommaso, Lehmann, Anja, and Neven, Gregory. The wonderful world of global random oracles. in *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 280–312. Springer, 2018.

[41] Ernst, Johannes and Mitrokotsa, Aikaterini. A framework for uc secure privacy preserving biometric authentication using efficient functional encryption. in *International Conference on Applied Cryptography and Network Security*, pp. 167–196. Springer, 2023.

[42] Kumar, Vinod, Ahmad, Musheer, Kumari, Adesh, Kumari, Saru, and Khan, Muhammad Khurram. Sebap: a secure and efficient biometric-assisted authentication protocol using ecc for vehicular cloud computing. *International Journal of Communication Systems*, 34(2):e4103, 2021.

Sciences, 63:1–13, 2020.

[25] Huang, Longxia, Zhang, Gongxuan, Yu, Shui, Fu, Anmin, and Yearwood, John. Seshare: Secure cloud data sharing based on blockchain and public auditing. *Concurrency and Computation: Practice and Experience*, 31(22):e4359, 2019.

[26] Yu, Haiyang, Yang, Zhen, and Sinnott, Richard O. Decentralized big data auditing for smart city environments leveraging blockchain technology. *IEEE Access*, 7:6288–6296, 2018.

[27] Zhang, Yuan, Xu, Chunxiang, Lin, Xiaodong, and Shen, Xuemin. Blockchain-based public integrity verification for cloud storage against procrastinating auditors. *IEEE Transactions on Cloud Computing*, 9(3):923–937, 2019.

[28] Xu, Yang, Ren, Ju, Zhang, Yan, Zhang, Cheng, Shen, Bo, and Zhang, Yaoxue. Blockchain empowered arbitrable data auditing scheme for network storage as a service. *IEEE Transactions on Services Computing*, 13(2):289–300, 2019.

[29] Xu, Yang, Zhang, Cheng, Wang, Guojun, Qin, Zheng, and Zeng, Quanrun. A blockchain-enabled deduplicatable data auditing mechanism for network storage services. *IEEE Transactions on Emerging Topics in Computing*, 9(3):1421–1432, 2020.

[30] Li, Song, Liu, Jian, Yang, Guannan, and Han, Jinguang. A blockchain-based public auditing scheme for cloud storage environment without trusted auditors. *Wireless Communications and Mobile Computing*, 2020(1):8841711, 2020.

[31] Wang, Hao, Qin, Hong, Zhao, Minghao, Wei, Xiaochao, Shen, Hua, and Susilo, Willy. Blockchain-based fair payment smart contract for public cloud storage auditing. *Information Sciences*, 519:348–362, 2020.

[32] Zhang, Guipeng, Yang, Zhenguo, Xie, Haoran, and Liu, Wenyin. A secure authorized deduplication scheme for cloud data based on blockchain. *Information Processing & Management*, 58(3):102510, 2021.

[33] Banaeian Far, Saeed and Imani Rad, Azadeh. Distributed auditing protocol for blockchain-based transactions using a distributed signature. *Security and Privacy*, 4(3):e156, 2021.

[34] Far, Saeed Banaeian, Asaar, Maryam Rajabzadeh, and Haghbin, Afrooz. Distributed auditing protocol for un-

MonadiAfta-Bianual Scientific Journal for Cyberspace Security
2024, Volume 22, Number 1 (pp. 15–37)
https://monadi.isc.org.ir

A review of blockchain-based data auditing protocols and analyzing their general framework

Saeed Banaeian Far * and Maryam Rajabzadeh Asaar

Department of Electrical and Computer Engineering, Science and Research Branch Islamic Azad University, Tehran, Iran

ARTICLE INFO.

Article history:
Received: January 8, 2024
Accepted: May 15, 2024
Published Online: May 26, 2024

Keywords:
Security and privacy
Blockchain
Data auditing protocols
Data integrity
Public verifiability

Type: Review paper

ABSTRACT

Transferring data to reliable centers for data maintenance, protection and accessibility is a simple and low-cost way, and there is no need to have a physical infrastructure, hardware, software, and human resources does not have. However, real-world events and recent research have shown that even reliable centers can abuse users' trust. For example, 1) make changes in the data they have, 2) delete them, or 3) make them temporarily or permanently unavailable. Data audit methods give assurance to the data owners that the data recorded in the database is the same as the data sent by the user and reveals the changes made in it. But they only solve the first problem. The introduction of blockchain technology as a new technology that had attractive features such as transparency, immutability and autonomy caused the problems of many systems that need the mentioned features to be solved. In this article, after reviewing and reviewing several blockchain-based data audit architectures and protocols, we review a generic framework for blockchain-based data auditing protocol. Finally, we provide a comparison between the reviewed works and specify some future horizons of this field.

* Corresponding author
Email addresses: saeed.banaeian.far@gmail.com (Saeed Banaeian Far), m.r.asaar@iau.ac.ir (Maryam Rajabzadeh Asaar)