مقایسه و بررسی روش‌های انگشت‌نگاری مرورگر کاربر در جهان و ایران

دکتر شهریار محمدی[۱]، علی مجیدی کلیبر[۲]

عضو هیات علمی و معاون پژوهشی دانشکده مهندسی صنایع دانشگاه خواجه نصیرالدین طوسی (mohammadi@kntu.ac.ir)
کارشناسی ارشد دانشگاه خواجه نصیرالدین طوسی (a.majidikalibar@email.kntu.ac.ir)

چکیده

محیط وب بستر زیبایی است که امکان ورود به این بستر از طریق مرورگرها فراهم شده است. با معرفی HTML5 و CSS3، محیط وب غنی‌تر و پویاتر از قبل شده است و همین باعث شده است این محیط به طور باورنکردنی از دستگاه‌های متنوع و مختلف نظیر لپ‌تاپ، تلفن‌های هوشمند و تبلت پشتیبانی کند. تنوع بخشی از وب مدرن است و همین موضوع، امکانی برای انگشت نگاری[۱] تجهیزات فراهم کرده است. منظور از انگشت نگاری، تکنیک شناسایی بر اساس لیست گسترده ای از ویژگی های دستگاه کاربر در لایه های مختلف است. از آنجا که این قابلیت از اصول بنیادین وب است؛ بنابراین انگشت نگاری مرورگر از طریق وصله[۲] قابل اصلاح نیست. به طور معمول سرویس‌گیرنده[۳] و سرویس‌دهنده[۴] اطلاعات مربوط به دستگاه را برای بهبود عملکرد بین یکدیگر به اشتراک می‌گذارند. مفهوم اصلی انگشت نگاری مرورگر، جمع آوری اطلاعات خاص دستگاه با هدف شناسایی یا افزایش امنیت است. با این حال، هنگامی که این مفهوم پیاده‌سازی می‌شود، اصول آن دائماً در حال تغییر است، زیرا سازوکارهای آن توسط فناوری‌های مرورگر تعریف می‌شود. به‌طورمعمول، رهگیری کاربران برای چند هدف انجام می‌شود، برای نمونه برای پیشگیری از کلاه‌برداری‌های الکترونیکی و یا تجزیه‌وتحلیل برای نمایش آگهی تبلیغات برخط به کاربران. در این مقاله سعی شده است روش های مختلف انگشت نگاری مرورگر کاربر مورد بررسی قرار گیرد.

واژگان کلیدی: انگشت نگاری، شناسایی کاربر، امنیت وب، حریم شخصی،

[1] Fingerprinting
[2] Patch
[3] Client
[4] Server

۱- مقدمه

امروزه وب بخشی مهمی از تعاملات کاربران فضای مجازی محسوب می‌شود و بیشتر اطلاعات از این راه منتقل می‌شود. میلیاردها کاربر به‌صورت روزمره صفحات وب را مرور می‌کنند، حتی سایت‌های وبی وجود دارند که بیش از یک میلیارد حساب کاربری در آن ثبت شده‌اند. در این شرایط، انگشت نگاری[1] کاربران و شناسایی عادات برخط آن‌ها در فضای مجازی وب، یکی از موضوعات مورد توجه شرکت‌ها و سازمان‌های رهگیری‌کننده[2] و تبلیغ‌کننده[3] است. درحالی‌که از دید کاربران وب، ردگیری آن‌ها نقضی برای حریم شخصی[4] آن‌ها است. انگشت نگاری مرورگر کاربران از یک‌سو می‌تواند برای اهداف قانونی مورد استفاده قرار گیرد و از سوی دیگر برای اهداف غیرقانونی و نقض‌کننده حریم شخصی کاربران. به‌عنوان نمونه‌ای از کاربردهای قانونی برای انگشت نگاری مرورگر کاربر، می‌توان برای تشخیص رفتار کاربران در شبکه‌های اجتماعی و امور تبلیغاتی یادکرد، علاوه بر این از انگشت نگاری مرورگر کاربر در وب می‌توان برای شناسایی کلاه‌برداری[5] استفاده کرد. نمونه غیرقانونی آن، انگشت نگاری مرورگر کاربر و جمع‌آوری اطلاعات در خصوص سامانه مورد استفاده کاربر توسط مهاجم است، به‌طوری‌که مهاجم می‌تواند کدهای سوء استفاده‌گر[6] را متناسب با مشخصات مرورگر، افزونه‌های نصب‌شده[7] و سیستم‌عامل کاربر انتخاب نموده و پس از ارسال و نصب در سامانه کاربر، آن را اجرا کند و احتمال موفقیت‌آمیز بودن حمله را افزایش دهد. علاوه بر این، انگشت نگاری مرورگر کاربر می‌تواند برای اهداف تبلیغاتی انجام شود که سود زیادی برای شرکت‌های تبلیغاتی دارد.

۲- مقدمه ای بر انگشت نگاری کاربران وب

منظور از انگشت نگاری مرورگر، جمع آوری مجموعه اطلاعات مرتبط با دستگاه کاربر از سخت افزار گرفته تا سیستم عامل مرورگر و پیکربندی آن است. در حقیقت انگشت نگاری مرورگر، به فرآیند جمع آوری اطلاعات از طریق مرورگر وب برای انگشت نگاری یک دستگاه است. از طریق اجرای یک اسکریپت ساده درون مرورگر، یک سرویس دهنده می تواند طیف گسترده ای از اطلاعات را از رابط برنامه نویسی برنامه کاربردی[8] (API) و سرآیندهای HTTP بدست آورد [1]. یک API، رابطی است که ورودی برای اشیاء و توابع را فراهم می‌کند. از آنجا که برخی از APIها برای دسترسی به میکروفن و یا دوربین مستلزم مجوز می باشند، بنابراین اغلب آنها از طریق کدهای جاوااسکریپتی قابل دسترس هستند؛ بر خلاف سایر تکنیک های شناسایی مانند کوکی که به یک شناسه منحصر به فرد (ID) وابسته است و مستقیماً درون مرورگر ذخیره می شود، انگشت نگاری مرورگر یک تکنیک کاملاً بدون حالت[9] است و هیچ ردی از خود باقی نمی گذارد زیرا نیازی به ذخیره سازی اطلاعات در داخل مرورگر ندارد [1].

وجود پیچیدگی در نگاشت ویژگی‌های متنوع مرورگرها به انگشت‌نگاری کاربران، همچنین عدم وجود مدلی که بتواند مجموعه‌ای از ویژگی‌های بهینه از نظر قابل‌دسترس بودن در تمامی بسترها، یکتا بودن اثرانگشت از مقادیر این ویژگی‌ها و قابل‌قبول بودن زمان محاسبه این ویژگی‌ها فراهم سازد، ازجمله چالش‌هایی برای رهگیری کاربران در وب می‌باشند. در این پژوهش مدلی برای غلبه بر چالش‌های فوق‌الذکر ایجاد خواهد شد و بر اساس خروجی این مدل، یک روش انگشت نگاری ایجاد خواهد شد. این پژوهش می‌تواند مبنایی برای کاربردهای انگشت نگاری مرورگر کاربران در شبکه‌های تبلیغاتی، شناسایی کلاه‌برداری، بی‌اثر ساختن روش‌های گمراه‌سازی و پاک کردن ردپای کاربران باشد.

---

[1] Fingerprinting
[2] Tracker
[3] Advertiser
[4] Privacy
[5] Fraud
[6] Exploit Code
[7] Plug-in
[8] Application Programming Interface
[9] Stateless

در سال ۱۹۹۴ میلادی، آقای مُنتولی[1] هنگامی‌که در شرکت نت‌اسکیپ[2] مشغول به کار بود، ایده کوکی را معرفی کرد. کوکی این امکان را به کارپذیر[3] وب می‌دهد تا یک داده با حجم ناچیز روی دستگاه کاربر ذخیره کند و سپس آن را درخواست‌های بعدی کاربر به کارپذیر وب ارسال کند. از این راه یک سایت وب می‌تواند حالت را روی پروتکل بدون حالت HTTP ایجاد و پشتیبانی کند. کوکی‌ها به‌سرعت توسط شرکت‌های سازنده مرورگرها و نیز توسعه‌دهندگان وب مورد استقبال قرار گرفت. پس از مدت کوتاهی از معرفی کوکی‌ها، ماهیت حالتمند آن‌ها مورد سوءاستفاده قرار گرفت. به‌طورمعمول صفحات وب ترکیبی از منابع مختلف مانند HTML، تصاویر، جاوا اسکریپت و CSS است که می‌تواند هم روی کارپذیر وب که میزبان صفحه اصلی است قرار گیرند و یا اینکه روی کارپذیرهای وب شخص ثالث[4] باشند. با هر درخواستی که به سمت سایت وب شخص ثالث ارسال می‌شود، آن سایت وب می‌تواند کوکی روی مرورگر کاربر تنظیم کند و کوکی‌های قبلی تنظیم شده روی آن را بخواند. به‌عنوان‌مثال، فرض کنید که یک کاربر سایت travel.com را مرور می‌کند، صفحه خانگی[5] این سایت یک تصویر از سایت tracking.com استفاده کرده است. بنابراین، بخشی از فرآیند نمایش صفحه خانگی travel.com این است که مرورگر کاربر درخواستی برای نمایش آن تصویر به tracking.com ارسال می‌کند. کارپذیر وب tracking.com تصویر را به همراه یک سرآیند HTTP Set-Cookie برای تنظیم کردن یک کوکی روی سامانه کاربر از راه دامنه tracking.com ارسال می‌کند. سپس کاربر سایت وب دیگر وابسته به tracking.com را مرور می‌کند به‌عنوان مثال buy.com. در این صورت سایت وب tracking.com کوکی‌های تنظیم شده قبلی را دریافت می‌کند و کاربر را شناسایی می‌کند و نمایه‌ای[6] از عادات کاربر ایجاد می‌کند [12]. کوکی‌های شخص ثالث، با توجه به اثرات نامطلوب آن روی نقض حریم شخصی کاربر و ارتباط مستقیم آن‌ها با آگهی‌های برخط تبلیغاتی، توسط جامعه پژوهشی و رسانه‌های عمومی مورد توجه قرار گرفتند و باعث نارضایتی کاربران شدند.

محیط وب بستر زیبایی است که امکان ورود به این بستر از طریق مرورگرها فراهم شده است. با معرفی HTML5 و CSS3، محیط وب غنی‌تر و پویاتر از قبل شده است و همین باعث شده است این محیط به طور باورنکردنی از دستگاه‌های متنوع و مختلف نظیر لپ‌تاپ، تلفن‌های هوشمند و تبلت پشتیبانی کند. تنوع بخشی از وب مدرن است و همین موضوع، امکانی برای انگشت نگاری تجهیزات فراهم کرده است. منظور از انگشت نگاری تجهیز، تکنیک شناسایی بر اساس لیست گسترده ای از ویژگی های دستگاه در لایه های مختلف است. از آنجا که این قابلیت از اصول بنیادین وب است؛ بنابراین انگشت نگاری مرورگر از طریق وصله قابل اصلاح نیست. به طور معمول سرویس گیرنده ها و سرویس گیرنده ها اطلاعات مربوط به دستگاه را برای بهبود عملکرد بین یکدیگر به اشتراک می گذارند. مفهوم اصلی انگشت نگاری مرورگر، جمع آوری اطلاعات خاص دستگاه با هدف شناسایی یا افزایش امنیت است. با این حال، هنگامی که این مفهوم پیاده‌سازی می‌شود، اصول آن دائماً در حال تغییر است، زیرا سازوکارهای آن توسط فناوری‌های مرورگر تعریف می‌شود [1].

برای این منظور، قانون ملاحظات حفظ حریم شخصی در ایالات‌متحده و اروپا شرکت‌های تبلیغاتی را در ذخیره‌سازی شناسه منحصربه‌فرد روی سامانه کاربر با محدودیت مواجه کرد. البته این محدودیت‌ها تقاضا برای رهگیری کاربران را کاهش نداد. برای شرکت‌های تبلیغاتی توانایی رهگیری کاربران مزیتی است تا از این راه آگهی‌های موردنظر کاربران را به آن‌ها نمایش دهند. در این صورت ارزش‌افزوده خدمات آن‌ها چندین برابر قبل خواهد بود. در این شرایط، عدم دسترسی به کوکی‌ها شرکت‌های تبلیغی و رهگیری‌کننده[7] را ترغیب کرد تا روش جدیدی برای رهگیری کاربران و تاریخچه‌ای از بازدیدهایی که از تارنماهای وب داشته‌اند، پیدا کنند [12].

---

[1] Lou Montulli
[2] Netscape
[3] Server
[4] Third-Party
[5] Homepage
[6] Profile
[7] Tracker

انگشت‌نگاری دستگاه کاربر تهدیدی جدی برای حریم شخصی کاربران است. ماهیت بدون حالت انگشت‌نگاری، شناسایی آن را مشکل کرده است. علاوه بر این، انگشت‌نگاری در حالت خصوصی[1] مرورگرهای جدید نیز عمل می‌کند. در پژوهش آقای نیکیفاراکیس و همکارانش [12] فناوری‌های بکار گرفته شده در سه شرکت انگشت‌نگاری مورد تجزیه‌وتحلیل قرار گرفته است.
امروزه یکی از مشکلات واقعی مشکل امنیت سایبری است. هر کاربر دارای اطلاعات محرمانه ای است که نیاز به محافظت بیشتر در برابر سرقت توسط افراد غیرمجاز است؛ بنابراین، اطلاعات دزدیده شده روی کارت اعتباری صاحب کارت و تراکنش های انجام شده با آن نه تنها می تواند در اختیار افراد غیرمجاز قرار گیرد، بلکه می تواند منجر به خسارات مالی قابل توجهی برای دارنده کارت شود. اگر سرقت داده ها نه از یک شخص، بلکه در سازمان های با اهمیتی که زیرساخت های حیاتی یک دولت را تشکیل می دهند، انجام شود، مشکل می تواند گسترده تر باشد. وقتی حجم داده ها زیاد می شود، اطلاعات در برابر نشت، آسیب‌پذیرتر می‌شوند زیرا کنترل آن، و در نتیجه امنیت، دشوارتر می‌شود. اغلب این قبیل آسیب پذیری ها در سرویس دهنده های وب معمول می باشند، زیرا داده های کاربران روی آنها ذخیره می شوند. برای حل این مشکل، صاحبان سرویس دهنده های وب می توانند روش های شناسایی خاصی را پیاده سازی کنند تا قابلیت تمایز بین کاربران را با استفاده از شناسه های منحصر به فرد فراهم نماید [22]، [23].

۳- پیشینه تحقیق

در سال ۲۰۰۹ مایر[2] بررسی کرد که تفاوت هایی که منشاء آن اینترنت است آیا می تواند منجر به بی نام سازی[3] سرویس گیرنده وب شود [1]. به طور خاص، او به دنبال تفاوت هایی در محیط های مرورگر بود تا توسط سرویس دهنده برای شناسایی کاربر قابل بهره برداری باشد. او نشان داد که یک مرورگر می تواند ویژگی هایی از سیستم عامل، سخت افزار و پیکربندی مرورگر ارائه دهد. او آزمایشی را اجرا کرد و محتوایی از Navigator، صفحه نمایش، افزونه های Navigator[4] و اشیاء MimeType مربوط به Navigator[5] را جمع آوری کرد. در این آزمایش، از بین ۱۳۲۸ کاربر، ۱۲۷۸ یعنی ۹۶.۲۳٪ آنها به طور منحصر به فرد با این روش قابل شناسایی بودند. با این حال، او در این مقاله اذعان کرده است به دلیل اینکه او در این تحقیق مقیاس کوچکی برای مطالعه داشته است، همین موضوع او را از نتیجه گیری کلی منع کرده است.
یک سال بعد، پیتر اکرسلی[6]، آزمایش Panopticlick را انجام داد [40]، او با برقراری ارتباط در رسانه های اجتماعی و درگاه های وب محبوب، در مدت دو هفته، ۴۷۰.۱۶۱ اثر انگشت جمع آوری کرد. بر خلاف تحقیق مایر، با توجه به میزان اثر انگشت جمع آوری شده، در این تحقیق تصویر بسیار دقیق تری از وضعیت تنوع دستگاه های مختلف جمع آوری شد. در این تحقیق، با داده های جمع آوری شده از سرآیندهای HTTP، جاوا اسکریپت و افزونه ها[7] مانند Flash Player و Java، ۸۳.۶٪ از اثر انگشت های جمع آوری شده به صورت منحصر به فرد شناسایی شدند. او نشان داد که اگر کاربران Flash و Java را فعال کرده باشند، این تعداد به ۹۴.۲٪ افزایش می یابد، زیرا این افزونه ها اطلاعات اضافی از دستگاه کاربر ارائه می نمایند. این تحقیق، ابداعی برای اصطلاح "انگشت نگاری مرورگر" بود. پیامدهای ناشی از حفظ حریم خصوصی این روش، بسیار قوی هستند به طوریکه یک تجهیز با پیکربندی نه چندان رایج به سادگی قابل شناسایی خواهد بود.

[1] Private Mode
[2] Mayer
[3] Deanonymization
[4] Navigator.plugins
[5] Navigator.mimeTypes objects
[6] Peter Eckersley
[7] Pluging

در سال ۲۰۱۲، موری[1] و شاچم[2] برای اولین بار زمینه دو بعدی Canvas[3] را در آزمایش خود تحت عنوان "Pixel Perfect" برای تهیه اثر انگشت مورد تحقیق قرار دادند [3]، از آنجا که پشته های مدیریت قلم روی تجهیزات مختلف، متفاوت هستند، آنها نشان دادند که سیستم عامل، نسخه مرورگر، کارت گرافیک و قلم های نصب شده همگی در تولید اثرانگشت نهایی قابل مشاهده برای کاربر نقش دارند.

در تحقیق سالوماتین و همکاران [2]، دو راهکار برای اطمینان از دستیابی به این هدف پیشنهاد شده است. از یک طرف، مجموعه محدودی از ویژگی‌هایی که انگشت نگاری مرورگر کاربر را تشکیل می‌دهند در نظر گرفته می‌شوند [35]، [34] از سوی دیگر، شناسایی بر اساس ردپای دیجیتالی کاربر نه تنها با استفاده از ابزار ریاضی احتمالی-آماری، بلکه با استفاده از روش های یادگیری ماشینی [36]،[39] ،[37] انجام شده است. علاوه بر این، کارایی و اثربخشی روش پیشنهاد شده با استفاده از یک آزمایش محاسباتی روی داده‌های واقعی انجام شده است.

در تحقیق انگلهارت و همکاران [6]، نشان داده شده که هنگام خزیدن در وب با هدف یافتن ردیاب ها[4]، انگشت نگاری بر پایه AudioContext یکی از جدیدترین روش ها است، آنها اسکریپت هایی را پیدا کردند که یک سیگنال صوتی تولید شده با OscillatorNode را برای انگشت نگاری تجهیزات پردازش می کند.

در مطالعه جاستن[5] و همکاران [7] به منابع قابل دسترس وب برای شناسایی افزونه ها پرداخته شد، آنها نشان دادند که در هنگام دسترسی به یک URL خاص، می توان فهمید که یک افزونه نصب شده است یا خیر.

موزانی و همکاران [11]، روشی را برای شناسایی قابل اعتماد مرورگر بر پایه جاوااسکریپت پیشنهاد کرد، آنها با جمع آوری مجموعه داده ترکیبی در بیش از ۱۵۰ مرورگر و سیستم عامل، حداقل مجموعه آزمایش برای شناسایی منحصر به فرد هر ترکیبی از مرورگر و سیستم عامل محاسبه کردند. تحقیق آنها بر این اساس بود که مرورگرها در موتور جاوااسکریپت خود حتی در دو نسخه متوالی، تفاوت هایی دارند.

نیکیفوراکیس و همکاران [12] در تحقیق خود تغییرپذیری در navigator و اشیاء صفحه نمایش را مورد تجزیه و تحلیل قرار دادند، آنها نشان دادند که نه تنها می توان بین نسخه های اصلی مرورگر تفاوت قائل شد بلکه می توان بین نسخه ای فرعی مرورگر نیز تمایز قائل شد. در تحقیق نیکیفوراکیس و همکاران [13] که در سال ۲۰۱۳ تحت عنوان Cookieless Monster انجام شد، تا ۲۰ صفحه برای ۱۰۰۰۰ سایت برتر مشخص شده در الکسا با هدف جستجو برای اسکریپت های انگشت نگاری از سه شرکت BlueCava، Iovation و ThreatMetrix انجام شد. در این تحقیق نشان داده شده که ۴۰ سایت (۰.۴٪) از کد انگشت نگاری این شرکت ها استفاده می کردند.

شوارتز و همکاران [13] در تحقیق خود فراتر از مرورگر رفتند و اطلاعات سیستم را استخراج کردند. آنها ویژگی هایی را استخراج کردند که می توانست اختلاف در سطح سیستم عامل و معماری را آشکار کند.

در تحقیق آکار و همکاران [18] یک خزش در مقایس بزرگ برای بازدید از صفحه اصلی یک میلیون سایت معرفی شده در الکسا توسط فریم ورک FPDetective انجام شد. در این تحقیق، تغییراتی در موتور مرورگر ایجاد شد تا از دسترسی به ویژگی های مرورگر و دستگاه با هدف انگشت نگاری ممانعت شود و لاگ آن نیز ثبت شد. ایشان فایل های Flashای را که در حین خزش با آنها مواجه می شدند را دِکامپایل[6] کردند تا وجود فراخوانی های تابع انگشت نگاری را تایید کنند. تحقیق FPDetective اولین تحقیقی بود که اسکریپت های انگشت نگاری را بدون توجه به لیست شناخته شده اسکریپت های رهگیری اندازه گیری کرد،

[1] Mowery
[2] Shacham
[3] Canvas 2D context
[4] Tracker
[5] Sjösten
[6] Decompile

زیرا در این تحقیق رفتارهای مرتبط با فعالیت انگشت نگاری بررسی شد. در این تحقیق نشان داده شده که ۴۰۴ سایت از ۱ میلیون سایت، کاوش قلم مبتنی بر جاوا اسکریپت و ۱۴۵ سایت از ۱۰۰۰۰ سایت کاوش قلم مبتنی بر Flash انجام می دادند. در تحقیق آکار و همکاران [4] در سال ۲۰۱۴ تحت عنوان "The Web Never Forgets"، انگشت نگاری بر اساس Canvas در صفحات اصلی ۱۰۰۰۰ سایت وب برتر الکسا اندازه گیری شد. آنها مرورگر را برای ممانعت از فراخوانی ها و بازگشت نسبت به متدهای مربوط به Canvas مجهز کردند و سعی کردند تا موارد مثبت کاذب[1] را با بکارگیری مجموعه ای از قوانین حذف کنند. آنها ۵۵۴۲ سایت از ۱۰۰۰۰۰ را پیدا کردند که انگشت نگاری انجام می دادند.

در تحقیق انگلهارت و نارایانان [19] تحت عنوان پلتفرم OpenWPM که در سال ۲۰۱۶ انجام شد، یک فریم ورک اندازه گیری حریم شخصی وب برای جمع آوری داده برای مطالعات حریم شخصی در مقیاس هزاران تا میلیون ها درگاه وب انجام شد. آنها برای نشان دادن قابلیت های ابزار خود، یک میلیون سایت برتر الکسا را مورد تجزیه و تحلیل قرار دادند تا رفتارهای رهگیری آنلاین را شناسایی و اندازه گیری کنند. یافته های آنها نتایج دقیق تری نسبت به گذاشته ارائه داد زیرا آنها تعداد زیادی از اشیاء جاوا اسکریپت را برای ایجاد یک معیار شناسایی برای هر روش انگشت نگاری شناخته شده مورد استفاده قرار دادند. در این تحقیق نشان داده شد که از یک میلیون درگاه وب، ۱۴۳۷۱ سایت از انگشت نگاری بر پایه Canvas استفاده می کنند و ۳۲۵۰ سایت انگشت نگاری قلم بر پایه Canvas را انجام می دادند و ۷۱۵ سایت انگشت نگاری بر پایه WebRTC را انجام می دادند و تنها 67 سایت انگشت نگاری بر پایه AudioContext انجام شد.

در تحقیق الفناح و همکاران [20] در سال ۲۰۱۸، ۱۰۰۰۰ سایت وب برتر مشخص شده توسط Majestic مورد خزش قرار گرفت و آنچه که خارج از مرورگر ارسال شده است، ثبت شد. تعریف آنها از انگشت نگاری بسیار گسترده تر و فراگیر تر از دیگر مطالعات بود. در این تحقیق، اگر حداقل یک ویژگی از لیست ۱۷ تایی در payloadهای ثبت شده وجود داشته باشد، وب سایت در حال انگشت نگاری تلقی می شد. آنها ۶۸۷۶ معادل ۶۸.۶٪ از سایت های وب را شناسایی کردند که انگشت نگاری را انجام می دهند، که بسیار بیشتر از آن چیزی بود که در گذشته گزارش شده بود و در نهایت آنها ۲۸۴ ویژگی را شناسایی کردند که می تواند برای انگشت نگاری مورد استفاده قرار گیرد.

سالوماتین و همکاران [2] روشی برای شناسایی کاربران در شبکه بر اساس انگشت نگاری مرورگر با استفاده از روش‌های یادگیری ماشینی ارائه کردند. روش ارائه شده در این تحقیق اصلاحی برای شناسایی کاربر بر اساس ردپای دیجیتال[2] است که به دو دلیل می تواند کارآمدتر باشد؛ اول اینکه، انتخاب ویژگی‌ها برای یک ردپای دیجیتال از مجموعه محدودی از ویژگی‌ها برای تشکیل اثر انگشت مرورگر کاربر ایجاد شده است و ثانیاً، دقت شناسایی را می توان از طریق استفاده ترکیبی از روش های طبقه بندی[3] و رویکرد احتمالی-آماری[4] افزایش داد. به منظور بررسی عملکرد موفقیت آمیز روش ارائه شده، یک آزمایش محاسباتی بر روی داده های واقعی انجام شده است که شامل حل مشکل طبقه بندی کاربر برپایه انگشت نگاری مرورگر کاربر با استفاده از روش K نزدیکترین همسایگان است.

سالوماتین و همکاران [2] در تحقیق خود نشان دادند توسعه یک روش مؤثرتر برای شناسایی کاربران در شبکه بر اساس ردپای دیجیتال ضروری است [28]، [29]، [30]، [32]، [33]، [31]. در این تحقیق، دو راهکار برای اطمینان از دستیابی به این هدف پیشنهاد شده است. از یک طرف، مجموعه محدودی از ویژگی‌هایی که انگشت نگاری مرورگر کاربر را تشکیل می‌دهند در نظر گرفته می‌شوند [35]، [34] از سوی دیگر، شناسایی بر اساس ردپای دیجیتالی کاربر نه تنها با استفاده از یک اسباب ریاضی احتمالی-آماری، بلکه با استفاده از روش های یادگیری ماشینی انجام شد [36]،[39]،[37]. علاوه بر این، کارایی و اثربخشی روش پیشنهاد شده با استفاده از یک آزمایش محاسباتی روی داده‌های واقعی انجام شده است.

---

[1] false positives
[2] digital footprint,
[3] classification
[4] probabilistic-statistical

البانا و همکاران [22] در تحقیق خود نشان دادند که منابع مختلفی از ویژگی‌های لازم برای تشکیل ردپای دیجیتالی کاربر و تعیین منحصربه‌فرد بودن او وجود دارد. منابع را می توان سیستماتیک کرد و به مجموعه های زیر تقسیم کرد: (۱) مجموعه ای از انگشت نگاری مرورگر، (۲) مجموعه ای از انگشت نگاری شبکه، (۳) مجموعه ای از لاگ های ممیزی، (۴) مجموعه ای از مدل های رفتاری. با این حال، این دسته بندی مشروط است، زیرا صفات - استثناهایی وجود دارند که می توانند به چندین گروه تعلق داشته باشند.

انگشت نگاری مرورگر حاوی اطلاعاتی در مورد پارامترهای ثابت است که می تواند از طریق مرورگر کاربر در مرحله اتصال وی به سرور محاسبه شود [38]. اطلاعات دریافتی حاوی داده هایی در مورد نام و نسخه مرورگر کاربر، دستگاهی که کاربر استفاده می کند، موقعیت مکانی کاربر و غیره است. از آنجایی که ویژگی های تجزیه و تحلیل شده ثابت هستند، به ردپای دیجیتالی که توسط انگشت نگاری مرورگر نشان داده می شود، مزایای زیادی می دهد. اولاً، محاسبه ویژگی‌ها برای انگشت نگاری مرورگر به زمان زیادی نیاز ندارد، که می‌توان آن را در آزمایش‌های بعدی تأیید کرد. ثانیاً، ذخیره داده‌های مربوط به انگشت نگاری مرورگر به مقدار زیادی حافظه نیاز ندارد، اگرچه به تعداد ویژگی‌های انتخابی بستگی دارد. محاسبه ویژگی های مرورگر به روش های مختلفی امکان پذیر است [38].

تحقیقات جدید در این زمینه نشان می‌دهد که ردپای دیجیتالی کاربر، که یک ارزیابی چند معیاره از ویژگی‌های استاتیک و پویا است، می‌تواند به عنوان یک شناسه کاربر به طور مؤثر عمل کند [24]، [25]، [30]. با این حال، استفاده از ردپای دیجیتالی بهینه نیست، زیرا ارزیابی بر اساس ویژگی هایی که کاربران را مشخص می کند، مانند انتخاب آنها، ذهنی است یا ماهیت احتمالی دارد که در آن شناسایی کاربر را می توان با یک احتمال مشخص به درستی تعیین کرد [27]، [26]؛ بنابراین، توسعه یک روش مؤثرتر برای شناسایی کاربران در شبکه بر اساس ردپای دیجیتال ضروری است [28]، [29]، [30]، [32]، [33]، [31].

در جدول ۱، جمع بندی تحقیقات کلیدی انجام شده در حوزه انگشت نگاری مرورگر کاربر آورده شده است، این تحقیقات شالوده اصلی تحقیقات انگشت نگاری مرورگر کاربر هستند.

جدول ۱: جمع بندی تحقیقات انجام شده در حوزه انگشت نگاری مرورگر کاربر

ردیف	نویسنده	سال تحقیق	توضیح مختصر تحقیق
۱.	پیتر اکرسلی [40]	۲۰۱۰	اجرای آزمایش Panopticlick، جمع آوری انگشت نگاری با برقراری ارتباط در رسانه های اجتماعی و درگاه های وب محبوب.
۲.	مووری و شاچم [3]	۲۰۱۲	انگشت نگاری در زمینه دو بعدی Canvas[1] تحت آزمایش "Pixel Perfect".
۳.	انگلهارت و همکاران [6]	۲۰۱۶	انگشت نگاری بر پایه AudioContext
٤.	جاستن و همکاران [7]	۲۰۱۷	انگشت نگاری بر پایه افزونه ها
٥.	موزانی و همکاران [11]	۲۰۱۳	انگشت نگاری بر پایه جاوا اسکریپت
٦.	نیکیفوراکیس و همکاران [12]	۲۰۱۳	انگشت نگاری بر پایه navigator و اشیاء صفحه نمایش
۷.	شوارتز و همکاران [13]	۲۰۱۹	انگشت نگاری سیستم کاربر
۸.	آکار و همکاران [18]	۲۰۱۳	اجرای فریم ورک FPDetective با هدف ممانعت از انگشت نگاری مرورگر و دستگاه کاربر

[1] Canvas 2D context

ردیف	نویسنده	سال تحقیق	توضیح مختصر تحقیق
۹.	آکار و همکاران [4]	۲۰۱۴	انگشت نگاری بر پایه Canvas
۱۰.	الفناح و همکاران [20]	۲۰۱۸	اجرای خزشگر Majestic با هدف انگشت نگاری
۱۱.	سالوماتین و همکاران [2]	۲۰۲۱	شناسایی کاربران در شبکه بر اساس انگشت نگاری مرورگر با استفاده از روش‌های یادگیری ماشینی
۱۲.	البانا و همکاران [22]	۲۰۱۸	شناسایی منابع مختلفی از ویژگی‌ها برای تشکیل ردپای دیجیتالی کاربر و تعیین منحصربه‌فرد بودن او

٤- بررسی روش های انگشت نگاری کاربران

منظور از رهگیری کاربران از راه انگشت‌نگاری سامانه کاربر، استخراج مجموعه‌ای از ویژگی‌های سامانه کاربر است که ترکیب آن‌ها با احتمال زیاد برای هر کاربر منحصربه‌فرد است و می‌توان از آن برای شناسایی کاربر استفاده کرد. این ویژگی‌ها می‌تواند شامل مواردی از قبیل اندازه صفحه‌نمایش، نرم‌افزارهای نصب‌شده و فهرست قلم‌های نصب‌شده باشد. برای رهگیری کاربران ویژگی‌هایی که مقادیر متنوع‌تری دارند مانند فهرست قلم‌ها نسبت به مقادیری که تنوع کمتری دارند مانند نسخه سامانه عامل، کاربرد بیشتری دارند. به‌طور مشابه، ویژگی‌ها با مقادیری که در طول زمان ماندگارتر هستند و تغییرات در آن‌ها به‌ندرت انجام می‌شود، شناسایی را در مقایسه با ویژگی‌هایی که اغلب تغییر می‌کنند و قابل پیش‌بینی نیستند، ساده‌تر می‌کنند [18].

به‌عبارت‌دیگر منظور از انگشت‌نگاری سامانه کاربر، همان فرآیند جمع‌آوری اطلاعات کافی از راه مرورگر کاربر است تا کاربر بدون استفاده از سازوکارهای حالتمند در وب شناسایی شود. انگشت‌نگاری به این شیوه، به‌عنوان شناسه‌ای برای رهگیری سامانه کاربر در وب مورد استفاده قرار می‌گیرد [18].

دلایلی زیادی وجود دارد که برنامه‌های کاربردی بر پایه وب مستلزم اطلاعات سامانه کاربر باشند به‌عنوان نمونه برای تحویل درست محتوا و یا برای به خدمت گرفتن رسانه سازگار با سامانه کاربر. بنابراین واسط‌های برنامه‌نویسی برنامه کاربردی[1] وجود دارند که برنامه‌های کاربردی را قادر می‌سازد تا این ویژگی‌ها را پرس‌وجو کنند. این واسط‌های برنامه‌نویسی برنامه کاربردی می‌توانند برای استخراج مقادیر این ویژگی‌ها باهدف انگشت‌نگاری دستگاه استفاده شوند [18].

هنگامی‌که مرورگر کاربر از صفحه وبی که شامل برنامه انگشت‌نگاری است بازدید می‌کند، اطلاعات دستگاه انگشت‌نگاری شده جمع‌آوری می‌شود و با پایگاه داده‌ای از دستگاه‌ها ذخیره شده مقایسه می‌شود. در این صورت، دستگاه‌های شناخته شده با پایگاه داده مطابقت داده می‌شوند و دستگاه‌های ناشناخته به پایگاه داده اضافه می‌شوند. در این صورت، هر ورودی به پایگاه داده برای هر دستگاه، با اطلاعات رفتاری کاربر بازدیدکننده تکمیل می‌شود. در ادامه روش‌های رایج انگشت‌نگاری معرفی شده‌اند.

- انگشت نگاری بر پایه User agent

یکی از ایده‌های کلیدی توسعه اولیه وب این است که دسترسی هر کاربر وب روی هر دستگاهی با هر نوع معماری فراهم باشد. پروتکل HTTP و بستر HTML از نیاز به داشتن یک راه جهانی برای برقراری ارتباط بین ماشین ها توسعه یافتند، با این حال، از آنجایی که وب شروع به تکامل کرد تا آنچه را که به صورت آنلاین امکان پذیر است ادامه دهد، نه هر مرورگر و نه هر پلتفرمی از آخرین موارد اضافه شده پشتیبانی نمی‌کند. برخی

[1] Application Programming Interface

از مرورگرها تنها با زیرمجموعه‌ای از مشخصات مطابقت دارند و ویژگی‌های خود را توسعه دادند. برای جلوگیری از مشکلات ناسازگاری، پروتکل HTTP شامل "User-Agent requestheader" است. بنابراین مرورگرها شروع به گنجاندن نام، نسخه و حتی گاهی اوقات پلتفرمی که روی آن اجرا می‌شدند، برای جلوگیری از محدودیت‌های عامل کاربر[1] شدند [1]. به عنوان مثال، عامل کاربر نسخه ۶۸ مرورگر کروم که بر روی لینوکس اجرا می شود به شرح شکل ۱ است [1]:

```
Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko)
    Chrome/68.0.3440.75 Safari/537.36
```

شکل ۱: عامل کاربر برای نسخه ۶۸ مرورگر کروم که بر روی لینوکس اجرا می شود [1]

- انگشت نگاری بر پایه Canvas

رابط برنامه نویسی برنامه کاربردی Canvas، امکانی در HTML 5 است که اشیاء، روش ها و ویژگی هایی برای ترسیم و ویرایش اشکال گرافیکی فراهم می کند، کاربران می توانند اشکال مختلف رسم کنند و آنها را متحرک کنند علاوه بر این، می توان محتوای متنی را به طور مستقیم در مرورگر با استفاده از قابلیت های گرافیکی دستگاه ارائه نمود. در سال ۲۰۱۲، موری[2] و شاچم[3] برای اولین بار زمینه دو بعدی Canvas[4] را در آزمایش خود تحت عنوان " Pixel Perfect" برای تهیه اثر انگشت مورد تحقیق قرار دادند [3]، از آنجا که پشته های مدیریت قلم روی تجهیزات مختلف، متفاوت هستند، آنها نشان دادند که سیستم عامل، نسخه مرورگر، کارت گرافیک و قلم های نصب شده همگی در تولید اثرانگشت نهایی قابل مشاهده برای کاربر نقش دارند.

در تحقیق آکار و همکاران، یک آزمایش در مقیاس بزرگ از انگشت نگاری Canvas تحت عنوان " The Web Never Forgets" انجام شد [4]، آنها نشان دادند که اسکریپت ها از تکنیک های مشخص شده در تحقیق موری و شاچم [3] استفاده می کنند و به ویژه آنها از مزایای سازوکار قلم بازگشتی که در مرورگرهای جدید وجود دارد، برای ایجاد تفاوت بیشتر بین دستگاه ها استفاده کردند. علاوه بر این، آنها نشان دادند که اغلب اسکریپت ها یک پایگاه کد بسیار شبیه هم و مشترک دارند و آنها این شباهت را از طریق دردسترس بودن یک کتابخانه انگشت نگاری منبع باز با عنوان "fingerprintjs" در GitHub توضیح دادند[5].

- انگشت نگاری بر پایه WebGL

قابلیت WebGL یک API گرافیکی است و می تواند اشیاء سه بعدی تعاملی در مرورگر فراهم کند و آنها را بدون نیاز به افزونه از طریق جاوااسکریپت دستکاری کرد. در تحقیق موری و شاچم [3]، WebGL برای انگشت نگاری مورد بررسی قرار گرفت. تا سال ۲۰۱۷ پیشرفتی در زمینه قابلیت های WebGL برای انگشت نگاری حاصل نشد. کانو و همکاران[5] یک تکنیک انگشت نگاری طراحی کردند که برای شناسایی دستگاه ها وابسته به WebGL بود، آنها به واسطه یک سری از ۳۱ کار پرداخت صحنه[6]، به دقت پارامترهای گرافیکی کامپیوتری را آزمایش کردند و ویژگی های دستگاه را استخراج کردند و توانستند به طور منحصر به فرد بیش از ۹۹٪ از ۱۹۰۳ دستگاه های مورد آزمایش را به طور منحصر به فرد شناسایی کنند.

---

[1] User-agent
[2] Mowery
[3] Shacham
[4] Canvas 2D context
[5] 2018. Anonymous browser fingerprint - fingerprints. https://github.com/Valve/fingerprintjs.
[6] Render

- انگشت نگاری بر پایه AudioContext

رابط برنامه نویسی برنامه کاربردی AudioContext، یک واسط برای ایجاد امکان پردازش صوت فراهم می کند، با ارتباط دادن ماژول های صوتی با یکدیگر، هر کسی می تواند سیگنال صوتی تولید کند. در تحقیق انگلهارت و همکاران [6]، نشان داده شده که هنگام خزیدن در وب با هدف یافتن ردیاب ها[1]، انگشت نگاری بر پایه AudioContext یکی از جدیدترین روش ها است، آنها اسکریپت هایی را پیدا کردند که یک سیگنال صوتی تولید شده با OscillatorNode را برای انگشت نگاری تجهیزات پردازش می کند. آنها نشان دادند که فرآیند انگشت نگاری با این روش شبیه به کاری است که با انگشت نگاری بر پایه Canvas انجام می شود، زیرا سیگنال های پردازش شده به دلیل پشته نرم افزاری و سخت افزاری دستگاه، تفاوت هایی را نشان می دهند.

- انگشت نگاری بر پایه افزونه های مرورگر

مرورگرهای جدید امکان سفارشی سازی مرورگر را از طریق افزونه های مرورگر[2]، فراهم می کنند به طوریکه کاربر با بکارگیری آنها می تواند عملکرد مرورگر خود را توسعه دهد. شناسایی افزونه مرورگر یک از مباحث چالش برانگیز است زیرا هیچ APIای برای پرس و جو کردن و استخراج فهرستی از افزونه های نصب شده روی مرورگر وجود ندارد. اگرچه به دلیل اینکه افزونه ها روی مرورگر ادغام می شوند، امکان شناسایی برخی از آنها وجود دارد. در مطالعه جاستن[3] و همکاران [7] به منابع قابل دسترس وب برای شناسایی افزونه ها پرداخته شد، آنها نشان دادند که در هنگام دسترسی به یک URL خاص، می توان فهمید که یک افزونه نصب شده است یا خیر. برای مثال، برای نمایش علامت تجاری یک افزونه، مرورگر می داند که آن کجا ذخیره شده است، مرورگر برای واکشی اطلاعات افزونه به آدرس "<extension://<extensionID>/<pathToFile>" مراجعه می کند؛ اگرچه، از آنجا که می توان به این منابع در زمینه هر صفحه وب دسترسی داشت، این سازوکار می تواند توسط یک اسکریپت برای تشخیص وجود یا عدم وجود یک پسوند خاص مورد بهره برداری قرار گیرد. هر افزونه ای به این منابع دسترسی ندارد، بنابراین هر افزونه ای با این روش قابل شناسایی نیست. جاستن و همکاران [8] نشان دادند که قادر به شناسایی ۱۲۱۵۴ افزونه از ۴۳۴۲۹ افزونه روی مرورگر کروم و ۱۰۰۳ افزونه از ۱۴۸۹۶ افزونه روی مرورگر فایرفاکس را شناسایی کردند.

در مطالعه استارو[4] و نیکیفوراکیس[5] [8] نشانه های جانبی ناشی از افزونه ها شناسایی شد، برای مثال اگر یک افزونه یک دکمه در YouTube اضافه کند تا کنترل جدید روی یک ویدئو ایجاد شود و دکمه اضافه شده با تجزیه و تحلیل DOM[6] صفحه وب قابل شناسایی است (DOM ساختار صفحه را نشان می دهد). در این تحقیق، ۱۰۰۰۰ عدد از محبوب ترین افزونه ها در مرورگر کروم مورد بررسی قرار گرفت و نشان داده شده که ۹٪ از آنها تغییرات DOM را ایجاد می کنند که می توانند در هر دامنه ای شناسایی شوند و ۱۶.۶٪ آنها تغییرات قابل شناسایی در دامنه های محبوب ایجاد می کنند. در این تحقیق با انجام آزمایش روی ۸۵۴ کاربر و شناسایی ۱۶۵۶ افزونه، نشان داده شده که ۱۴.۱۰٪ کاربران منحصر به فرد هستند.

در تحقیق سانچز رولا و همکاران [8]، از یک حمله کانال جانبی زمانبندی[7] برای شناسایی افزونه های مرورگر استفاده شد، آنها منابع مربوط به توسعه های جعلی و موجود را مورد پرس و جو قرار دادند و تفاوت زمانی بین فراخوانی ها را اندازه گیری کردند، با استفاده از این روش، آنها نشان دادند که می توانند هر توسعه ای روی مرورگر را شناسایی کنند.

---

[1] Tracker
[2] Browser extensions
[3] Sjösten
[4] Starov
[5] Nikiforakis
[6] Document Object Model
[7] a timing side channel attack

در این مقاله، مطالعه موردی روی ۲۰۴ کاربر با هدف شناسایی ۲۰۰۰ توسعه روی مرورگرها انجام شد و نشان داده شد که از این طریق ۵۶.۸۶٪ کاربران به صورت منحصر به فرد شناسایی شده اند.
در تحقیق گولیاس و همکاران [10] بزرگترین مطالعه با ۱۶۳۹۳ کاربر با هدف ارزیابی منحصر به فرد بودن کاربران بر اساس توسعه مرورگر و ورود تحت وب[1] انجام شد. در این مقاله نشان داده شد که از ۷۶۴۳ کاربر کروم، ۳۹.۲۹٪ آنها بر پایه شناسایی ۱۶۷۴۳ افزونه کروم به صورت منحصر به فرد هستند. علاوه بر این، در این تحقیق نشان داده شد که برای دستیابی به همان سطح از منحصر به فرد بودن، کافی است فقط ۴۸۵ از توسعه های انتخاب شده مورد آزمایش قرار گیرند.

- انگشت نگاری بر پایه انطباق با استانداردهای جاوااسکریپت

موزانی و همکاران [11]، روشی را برای شناسایی قابل اعتماد مرورگر بر پایه جاوااسکریپت پیشنهاد کرد، آنها با جمع آوری مجموعه داده ترکیبی در بیش از ۱۵۰ مرورگر و سیستم عامل، حداقل مجموعه آزمایش برای شناسایی منحصر به فرد هر ترکیبی از مرورگر و سیستم عامل محاسبه کردند. تحقیق آنها بر این اساس بود که مرورگرها در موتور جاوااسکریپت خود حتی در دو نسخه متوالی، تفاوت هایی دارند.

نیکیفوراکیس و همکاران [12] در تحقیق خود تغییر پذیری در navigator و اشیاء صفحه نمایش را مورد تجزیه و تحلیل قرار دادند، آنها نشان دادند که نه تنها می توان بین نسخه های اصلی مرورگر تفاوت قائل شد بلکه می توان بین نسخه ای فرعی مرورگر نیز تمایز قائل شد.

شوارتز و همکاران [13] در تحقیق خود فراتر از مرورگر رفتند و اطلاعات سیستم را استخراج کردند. آنها ویژگی هایی را استخراج کردند که می توانست اختلاف در سطح سیستم عامل و معماری را آشکار کند.

- انگشت نگاری بر پایه معیار محک[2]

روش دیگر برای کشف اطلاعات در مورد یک دستگاه محک زدن ویژگی های CUP و GPU دستگاه است. از طریق جاوا اسکریپت، می توان کارهایی را آماده کرد و زمان لازم برای تکمیل آنها را اندازه گیری کرد. اگرچه، بزرگترین مشکل در هنگام استفاده از محک ها، تفسیر صحیح تفاوت ها و نوسانات است. دو مقدار زمانی می توانند متفاوت باشند، زیرا از دو دستگاه مختلف جمع آوری شده اند، با این حال در شرایطی که یک فرآیند جدید در پس زمینه، منجر به اختلال در اندازه گیری واقعی شود، ممکن است به یک دستگاه واحد نیز تعلق داشته باشد.

سانچز رولا و همکاران [14] در تحقیق خود اختلاف ساعت یک دستگاه برای انجام انگشت نگاری را اندازه گیری کردند. در این تحقیق نشان داده شد که از طریق کد بومی[3]، دستگاه های دارای سخت افزار و نرم افزار مشابه را با اندازه گیری زمان لازم برای اجرای عملکردهای خاص مانند "string::compare"، "std::regex" و "std::hash" می توان متمایز کرد. پیاده سازی محیط وب، وابسته به تابع "Crypto.getRandomValues()" است، در حالیکه نمی توان همه دستگاه ها را متمایز کرد، این روش نتایج بهتری نسبت به انگشت نگاری بر پایه Canvas یا WebGL فراهم کرده است.

- سیر تکاملی انگشت نگاری در طول زمان

یکی دیگر از جنبه های اصلی انگشت نگاری مرورگر به سیر تکاملی آنها در طول زمان مربوط می شود. با توجه به اینکه، انگشت نگاری بازتاب مستقیم دستگاه کاربر و محیط آن است، انگشت نگاری بسیار با توجه به تغییر در اجزای

[1] Web Logins
[2] Benchmarking
[3] Native code

سیستم، پیکربندی و به روزرسانی دستگاه مستعد تغییر است. بنابراین برای فعال کردن رهگیری در بلند مدت، لازم است به درک این تغییرات و پیش بینی اینکه انگشت نگاری چگونه می تواند تغییر کند توجه داشت. اکرسلی [40] اولین کسی بود که در تحقیق Panopticlick این سوال را مطرح کرد. او الگوریتمی برای تخمین اکتشافی پیاده سازی کرد تا بر پایه آن بتوان فهمید آیا یک اثر انگشت ممکن است نسخه تکامل یافته یک اثر انگشت باشد. در این تحقیق بر پایه انگشت نگاری های جمع آوری شده، نشان داده شده که در ۶۵٪ موارد می توان این موضوع را به درستی حدس زد، اما تحقیق او در این موضوع بسیار مقدماتی بود.

واستل و همکاران [15] در تحقیق خود نشان دادند که به لطف افزونه های مرورگرهای فایرفاکس و کروم، آنها از کاربران اثر انگشت جمع آوری کردند و توانستند تغییرات مختلفی را که اثر انگشت مرورگر ممکن است داشته باشد، مشاهده کردند. در این تحقیق سه نوع مختلف تکامل شناسایی شد، تحولات خودکار ناشی از به روزرسانی های ذاتی نرم افزار، تحولات وابسته به متن[1] که با تغییرات در محیط کاربر منعکس می شوند و تحولات ایجاد شده توسط کاربر ناشی از تغییر در تنظیمات مرورگر. در این تحقیق نشان داده شد که تکامل اثر انگشت به شدت به نوع دستگاه و نحوه استفاده از آن وابستگی دارد. حداقل یک تغییر برای ۴۵.۵۲٪ از اثر انگشت های جمع آوری شده به طور مشابه مشاهده شد، در حالیکه ممکن است برای دستگاه های دیگر چندین هفته طول بکشد تا یک تغییر را مشاهده کرد. در این تحقیق اثر انگشت متعلق به یک دستگاه در گذر زمان بررسی شد. با جمع آوری اثر انگشت هر سه روز یکبار، الگوریتم FP-Stalker قادر به رهگیری یک دستگاه به طور میانگین برای ۵۱.۸ روز است. آنها همچنین توانستند ۲۶٪ از دستگاه ها را برای بیش از ۱۰۰ روز رهگیری کنند و ثابت کردند که اثر انگشت مرورگر می تواند به طور موثر برای تکمیل سایر شیوه های شناسایی مورد استفاده قرار گیرد.

در جدول ۲ یک مثال کامل از ویژگی های اصلی جمع آوری شده برای انگشت نگاری مروگر به همراه منبع آنها آورده شده است [۱].

جدول ۲: مثال هایی از انگشت نگاری مرورگر [1].

ویژگی	منبع	مثال
عامل کاربر (User agent)	HTTP header	Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.119 Safari/537.36
پذیرش (Accept)	HTTP header	text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
کد گذاری محتوا (Content encoding)	HTTP	gzip, deflate, br
زبان محتوا (Content language)	header	en-US,en;q=0.9
فهرست افزونه ها (List of plugins)	JavaScript	Plugin 1: Chrome PDF Plugin. Plugin 2: Chrome PDF Viewer. Plugin 3: Native Client. Plugin 4: Shockwave Flash...
کوکی فعال شده (Cookies enabled)	JavaScript	yes
استفاده از محلی/جلسه ذخیره سازی (Use of local/session storage)	JavaScript	yes
منطقه زمانی (Timezone)	JavaScript	-60 (UTC+1)

[1] contextdependent

وضوح صفحه و عمق رنگ (Screen resolution and color depth)	JavaScript	1920x1200x24
فهرست قلم ها (List of fonts)	Flash or JS	Abyssinica SIL,Aharoni CLM,AR PL UMing CN,AR PL UMing HK,AR PL UMing TW...
فهرستی از سرآیندهای HTTP	HTTP headers	Referer X-Forwarded-For Connection Accept Cookie Accept-Language Accept-Encoding User-Agent Host
پلتفرم (Platform)	JavaScript	Linux x86_64
عدم رهگیری (Do Not Track)	JavaScript	yes
کنوس (Canvas)	JavaScript	Cwm fjordbank glyphs vext quiz, 😃 Cwm fjordbank glyphs vext quiz, 😃
فروشنده WebGL (WebGL Vendor)	JavaScript	NVIDIA Corporation
ارائه دهنده WebGL (WebGL Renderer)	JavaScript	GeForce GTX 650 Ti/PCIe/SSE2
استفاده از مسدودکننده تبلیغات (Use of an ad blocker)	JavaScript	yes

۵- مطالعات انجام شده در مقیاس بزرگ به لحاظ اثربخشی انگشت نگاری کاربر

اکرسلی [40] آزمایش Panopticlick را در سال ۲۰۱۰ انجام داد، از ۴۷۰۱۶۱ اثر انگشت، او به این نتیجه رسید که از اثر انگشت مرورگر می توان برای رهگیری کاربران استفاده کرد، زیرا ۸۳.۶٪ از اثرانگشت های جمع آوری شده منحصر به فرد بودند. این عدد درصورتیکه کاربران برنامه های Flash و Java را روی مرورگر فعال کنند به ۹۴.۲٪ افزایش می یابد. متمایزترین ویژگی ها در آن زمان، فهرست افزونه ها، قلم ها و user-agent بود.

لاپردریکس و همکاران [17] در آزمایش خود تحت عنوان AmIUnique در سال ۲۰۱۶، ۱۱۸۹۳۴ اثر انگشت را تجزیه و تحلیل کردند و نشان دادند که نتایج اکرسلی [40] را تایید می کنند زیرا ۸۹.۴٪ از اثر انگشت های جمع آوری شده به صورت منحصر به فرد بودند. اگرچه در ۶ سالی که این دو تحقیق از یکدیگر فاصله داشتند، شاهد تحول در ویژگی های متفاوتی بودیم که اثر انگشت را تشکیل می دهند. اگر چه در ابتدا فهرست افزونه ها و قلم ها عامل اصلی انگشت نگاری بودند، اما اخیرا با توجه به تهدیدات امنیتی که بکارگیری افزونه ها روی مرورگرها فراهم می کنند، بکارگیری آنها منسوخ شده است. انگشت نگاری بر اساس Canvas نتایج خوبی فراهم می کنند زیرا نشان داده شد که آنها یک آنتروپی مهمی را در مقادیر جمع آوری شده ایجاد می کنند. لاپردریکس و همکاران [17] در تحقیق خود نشان دادند، که انگشت نگاری موبایل نیز ممکن است. در مجموعه داده این تحقیق، ۸۱٪ از اثرانگشت ها مربوط به تجهیزات موبایلی بود که به صورت منحصر به فرد شناسایی شدند. در این تحقیق نشان داده شد سرآیندهای HTTP و HTML5 Canvas نقش مهمی در شناسایی مرورگرها بازی می کنند. علاوه بر این، نشان داده شد تغییرات ساده نظیر سرآیندهای HTTP و یا حذف افزونه ها باعث کاهش ۳۶٪ انگشت نگاری به صورت منحصر به فرد می شود.

در تحقیق گومز-بویکس و همکاران [16] در سال ۲۰۱۸ تحت آزمایش Hiding in the Crowd، ۲۰۶۷۹۴۲ اثر انگشت از یکی از ۱۵ سایت برتر فرانسه جمع آوری و سپس مورد تجزیه و تحلیل قرار گرفت. یافته های آنها نشان داد که در آن دامنه مورد بررسی ۳۳.۶٪ از اثرانگشت ها به صورت منحصر به فرد هستند. در مقایسه با دو مطالعه قبل، این عدد دو تا سه برابر کمتر است. در صورت در نظر گرفتن دستگاه های تلفن همراه، این تفاوت حتی بیشتر هم می شود زیر ۱۸.۵٪ از اثر انگشت تلفن همراه منحصر به فرد هستند. در این مطالعه نشان داده شد که فرآیند جمع آوری داده اهمیت دارد زیرا در گذشته اثر انگشت در

شرایطی انجام می شد که بازدیدکنندگان آن سایت به شرایط حریم شخصی آنلاین آگاه بودند و یا حتی نسبت به کاربران عادی وب محتاط تر بودند. اما در این مطالعه، داده های جمع آوری شده مربوط به یک وب سایت تجاری بود که مخاطبان جهانی داشت. در این تحقیق، ویژگی مجموعه داده جمع آوری شده که شامل تعداد بسیار زیادی اثر انگشت بود، عامل اصلی درک تفاوت منحصر به فرد بودن اثر انگشت بود. در این تحقیق نشان داده شد اغلب اثر انگشت ها در مورد دستگاه های رومیزی[1] به دلیل ترکیبی از ویژگی ها منحصر به فرد هستند، در حالیکه در دستگاه های تلفن همراه ویژگی ها هستند که منجر به منحصر به فرد بودن می شوند.

در جدول ۳ خلاصه‌ای از ویژگی‌ها در سه تحقیق Panopticlick(2010) [40]، AmIUnique(2016) [17] و Hiding in the Crowd(2018) [16] ارائه شده است.

جدول ۳: مروری بر مطالعات انجام شده در مقیاس بزرگ در خصوص انگشت نگاری کاربر

	Panopticlick (2010)	AmIUnique (2016)		Hiding in the Crowd (2018)	
	Desktop	Desktop	Mobile	Desktop	Mobile
Number of fingerprints	470,161	105,829	13,105	1,816,776	251,166
Unique fingerprints	94.2%	89.4%	81%	35.7%	18.5%

در تحقیق نیکیفوراکیس و همکاران [12] که در سال ۲۰۱۳ تحت عنوان Cookieless Monster انجام شد، تا ۲۰ صفحه برای ۱۰۰۰۰ سایت برتر مشخص شده در الکسا با هدف جستجو برای اسکریپت های انگشت نگاری از سه شرکت BlueCava، Iovation و ThreatMetrix انجام شد. در این تحقیق نشان داده شده که ۴۰ سایت (۰.۴٪) از کد انگشت نگاری این شرکت ها استفاده می کردند.

در تحقیق آکار و همکاران [18] یک خزش در مقایس بزرگ برای بازدید از صفحه اصلی یک میلیون سایت معرفی شده در الکسا توسط فریم ورک FPDetective انجام شد. در این تحقیق، تغییراتی در موتور مرورگر ایجاد شد تا از دسترسی به ویژگی های مرورگر و دستگاه با هدف انگشت نگاری ممانعت شود و لاگ آن نیز ثبت شد. ایشان فایل های Flashای را که در حین خزش با آنها مواجه می شدند را دِکامپایل[2] کردند تا وجود فراخوانی های تابع انگشت نگاری را تایید کنند. تحقیق FPDetective اولین تحقیقی بود که اسکریپت های انگشت نگاری را بدون توجه به لیست شناخته شده اسکریپت های رهگیری اندازه گیری کرد، زیرا در این تحقیق رفتارهای مرتبط با فعالیت انگشت نگاری بررسی شد. در این تحقیق نشان داده شده که ۴۰۴ سایت از ۱ میلیون سایت، کاوش قلم مبتنی بر جاوا اسکریپت و ۱۴۵ سایت از ۱۰۰۰۰ سایت کاوش قلم مبتنی بر Flash انجام می دادند.

در تحقیق آکار و همکاران [4] در سال ۲۰۱۴ تحت عنوان "The Web Never Forgets"، انگشت نگاری بر اساس Canvas در صفحات اصلی ۱۰۰۰۰ سایت وب برتر الکسا اندازه گیری شد. آنها مرورگر را برای ممانعت از فراخوانی ها و بازگشت نسبت به متدهای مربوط به Canvas مجهز کردند و سعی کردند تا موارد مثبت کاذب[3] را با بکارگیری مجموعه ای از قوانین حذف کنند. آنها ۵۵۴۲ سایت از ۱۰۰۰۰۰ را پیدا کردند که انگشت نگاری انجام می دادند.

در تحقیق انگلهارت و نارایانان [19] تحت عنوان پلتفرم OpenWPM که در سال ۲۰۱۶ انجام شد، یک فریم ورک اندازه گیری حریم شخصی وب برای جمع آوری داده برای مطالعات حریم شخصی در مقیاس هزاران تا میلیون ها درگاه وب انجام شد. آنها برای نشان دادن قابلیت های ابزار خود، یک میلیون سایت برتر الکسا را مورد تجزیه و تحلیل قرار دادند تا رفتارهای رهگیری آنلاین را شناسایی و اندازه گیری کنند. یافته های آنها نتایج دقیق تری نسبت به گذاشته ارائه داد زیرا آنها تعداد زیادی از اشیاء

[1] Desktop
[2] Decompile
[3] false positives

جاوا اسکریپت را برای ایجاد یک معیار شناسایی برای هر روش انگشت نگاری شناخته شده مورد استفاده قرار دادند. در این تحقیق نشان داده شد که از یک میلیون درگاه وب، ۱۴۳۷۱ سایت از انگشت نگاری بر پایه Canvas استفاده می کنند و ۳۲۵۰ سایت انگشت نگاری قلم بر پایه Canvas را انجام می دادند و ۷۱۵ سایت انگشت نگاری بر پایه WebRTC را انجام می دادند و تنها 67 سایت انگشت نگاری بر پایه AudioContext انجام شد.

در تحقیق الفناح و همکاران [20] در سال ۲۰۱۸، ۱۰۰۰۰ سایت وب برتر مشخص شده توسط Majestic مورد خزش قرار گرفت و آنچه که خارج از مرورگر ارسال شده است، ثبت شد. تعریف آنها از انگشت نگاری بسیار گسترده تر و فراگیر تر از دیگر مطالعات بود. در این تحقیق، اگر حداقل یک ویژگی از لیست ۱۷ تایی در payloadهای ثبت شده وجود داشته باشد، وب سایت در حال انگشت نگاری تلقی می شد. آنها ۶۸۷۶ معادل ۶۸.۶٪ از سایت های وب را شناسایی کردند که انگشت نگاری را انجام می دهند، که بسیار بیشتر از آن چیزی بود که در گذشته گزارش شده بود. و در نهایت آنها ۲۸۴ ویژگی را شناسایی کردند که می تواند برای انگشت نگاری مورد استفاده قرار گیرد.

آنتروپی برای تعیین کمیت سطح اطلاعات شناسایی در انگشت نگاری کاربرد دارد. هر چه آنتروپی بالاتر باشد، انگشت نگاری منحصر به فردتر و قابل شناسایی تر خواهد بود. در جدول ۴ ویژگی های مرورگر، آنتروپی آنها و آنتروپی نرمال شده در سه تحقیق Panopticlick (2010) [40]، AmIUnique (۲۰۱۶) [17] و Hiding in the Crowd (۲۰۱۸) [16] ارائه شده است [1].

جدول ۴: ویژگی های مرورگر، آنتروپی آنها و آنتروپی نرمال شده در سه تحقیق Panopticlick (2010) [40]، AmIUnique (2016) [17] و Hiding in the Crowd (2018) [16]

ویژگی	Panopticlick (2010)		AmIUnique (2016)		Hiding (2018)	
	آنتروپی	آنتروپی نرمال شده	آنتروپی	آنتروپی نرمال شده	آنتروپی	آنتروپی نرمال شده
عامل کاربر (User agent)	10.000	0.531	9.779	0.580	7.150	0.341
پذیرش (Accept)	-	-	1.383	0.082	0.729	0.035
کد گذاری محتوا (Content encoding)	-	-	1.534	0.091	0.382	0.018
زبان محتوا (Content language)	-	-	5.918	0.351	2.716	0.129
فهرست افزونه ها (List of plugins)	15.400	0.817	11.060	0.656	9.485	0.452
کوکی فعال شده (Cookies enabled)	0.353	0.019	0.253	0.015	0.000	0.000
استفاده از محلی/جلسه ذخیره سازی (Use of local/session storage)	-	-	0.405	0.024	0.043	0.002
منطقه زمانی (Timezone)	3.040	0.161	3.338	0.198	0.164	0.008
وضوح صفحه و عمق رنگ (Screen resolution and color depth)	4.830	0.256	4.889	0.290	4.847	0.231
فهرست قلم ها (List of fonts)	13.900	0.738	8.379	0.497	6.904	0.329
فهرستی از سرآیندهای HTTP	-	-	4.198	0.249	1.783	0.085
پلتفرم (Platform)	-	-	2.310	0.137	1.200	0.057
عدم رهگیری (Do Not Track)	-	-	0.944	0.056	1.919	0.091
کنوس (Canvas)	-	-	8.278	0.491	8.546	0.407

ویژگی	Panopticlick (2010)		AmIUnique (2016)		Hiding (2018)	
	آنتروپی	آنتروپی نرمال شده	آنتروپی	آنتروپی نرمال شده	آنتروپی	آنتروپی نرمال شده
فروشنده WebGL (WebGL Vendor)	-	-	2.141	0.127	2.282	0.109
ارائه دهنده WebGL (WebGL Renderer)	-	-	3.406	0.202	5.541	0.264
استفاده از مسدودکننده تبلیغات (Use of an ad blocker)	-	-	0.995	0.059	0.045	0.002
بدترین حالت که در آن آنتروپی حداکثر و تمام مقادیر یک ویژگی منحصر به فرد است	18.843		16.860		20.980	
تعداد انگشت نگاری	470,161		118,934		2,067,942	

۶- مطالعات انجام شده در حوزه انگشت نگاری مرورگر کاربر در ایران

در بین مقالات فارسی، تحقیقات علمی در حوزه انگشت نگاری مرورگر کاربر یافت نشد، جستجو در میان منابع مختلف فارسی منجر به تحقیقاتی در حوزه انگشت نگاری وب و تارنما شد، انگشت نگاری تارنما روشی برای تحلیل ترافیک در شرایطی است که کاربر از روش های پنهان سازی محتوای ترافیک و مقصد واقعی استفاده می کند. دو تحقیق آورده شده در ادامه مرتبط با انگشت نگاری تارنما است، که حوزه تحقیقی متفاوت از انگشت نگاری مرورگر کاربر است.

در مقاله مریم طائبی و همکاران [41] با عنوان "مروری بر حمله های انگشت نگاری تارنما" مروری جامع بر روش های انگشت نگاری تارنما انجام شده است.

در مقاله سعید شیرودی و همکاران [42] با عنوان "فانوس: راهکار مقابله با حملات انگشت نگاری وب" سازوکار دفاعی برای عدم شناسایی کاربر بر پایه انگشت نگاری به روش شبکه عصبی ارائه شده است. در این تحقیق روشی تحت نام فانوس (فریب دهنده انگشت نگاری وب سایت) ارائه شده است که با سربار پایین، نرخ خطای مهاجم در دسته‌بندی را به بالاترین مقدار خود می‌رساند.

۷- دستاوردها و نتایج

توسعه اینترنت همراه با پیشرفت در فناوری تلفن همراه، تنوع گسترده ای از دستگاه ها را به ارمغان آورده است. این تنوع باعث ایجاد انگشت نگاری مرورگر شده است، تکنیکی ساده که شامل جمع آوری اطلاعات در مورد پیکربندی و ویژگی های دستگاه کاربر است. جنبه جذاب آن این است که انگشت نگاری مرورگر در تقاطع چهار عامل شرکت های تجاری، گروه های تحقیقاتی دانشگاهی، قانون گذاران و مدافعان حریم شخصی قرار دارد. همانطور که از نتایج آزمایشگاه های تحقیقاتی خارج شده است، انگشت نگاری مرورگر تأثیر مشخصی بر وب دارد زیرا اکنون در سناریوهای دنیای واقعی استفاده می شود. از دیدگاه شرکت‌های تجاری، انگشت نگاری مرورگر جایگزینی برای روش‌های فعلی ردیابی و شناسایی است درحالیکه چشم‌انداز تبلیغات با افزایش مسدودکننده‌های تبلیغاتی دستخوش تغییرات عظیمی است. از دیدگاه گروه‌های تحقیقاتی، انگشت نگاری مرورگر سؤالات غیرمنتظره‌ای در مورد وضعیت حریم خصوصی ایجاد کرده است. از دیدگاه قانونگذاران، انگشت نگاری مرورگر نشان دهنده یک سازوکار ردیابی اضافی است که باید به گونه ای تنظیم شود تا کنترل به دست کاربران بازگردد. از دیدگاه مدافعان حریم شخصی، که متکی به محرمانگی و حریم شخصی ارتباطات خود هستند، محافظت از فعالیت های خود را ضروری می دانند.

در مجموع، انگشت نگاری مرورگر هنوز یک تکنیک نسبتا جدید است. با این حال، در مدت کوتاهی از خلق آن تأثیرات زیادی داشته است. در این مقاله تلاش برای نظام‌بندی تجربیات انجام شده در این حوزه، ثابت می‌کند که هنوز چالش‌ها و مشکلات زیادی وجود دارد که باید حل شوند زیرا محققان و توسعه‌دهندگان به پیچیدگی‌های آن پی برده اند.

۸- مراجع

[1] Pierre Laperdrix, Nataliia Bielova, Benoit Baudry, Gildas Avoine, “Browser Fingerprinting: A Survey”, ACM Transactions on the Web, 2020.

[2] AA Salomatin, AY Iskhakov, AO Iskhakova, “Web user identification based on browser fingerprints using machine learning methods”, IFAC PapersOnLine 54-13 (2021) 582–587-Elsevier, 2021.

[3] Keaton Mowery and Hovav Shacham, “Pixel Perfect: Fingerprinting Canvas in HTML5”, In Proceedings of W2SP 2012, Matt Fredrikson (Ed.), IEEE Computer Society, 2012.

[4] Gunes Acar, Christian Eubank, Steven Englehardt, Marc Juarez, Arvind Narayanan, and Claudia Diaz, “The Web Never Forgets: Persistent Tracking Mechanisms in the Wild”, In Proceedings of the 2014 ACM SIGSAC, Conference on Computer and Communications Security (CCS ’14). ACM, New York, NY, USA, 674–689, 2014.

[5] Yinzhi Cao, Song Li, and Erik Wijmans, “(Cross-)Browser Fingerprinting via OS and Hardware Level Features”, In 24nd Annual Network and Distributed System Security Symposium, NDSS, 2017.

[6] Steven Englehardt and Arvind Narayanan, “Online Tracking: A 1-million-site Measurement and Analysis”, In Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS ’16). ACM, NewYork, NY, USA, 1388–1401, 2016.

[7] Alexander Sjösten, Steven Van Acker, and Andrei Sabelfeld, “Discovering Browser Extensions viaWeb Accessible Resources”, In Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy (CODASPY’17), ACM, New York, NY, USA, 329–336, 2017.

[8] Oleksii Starov and Nick Nikiforakis, “XHOUND: Quantifying the Fingerprintability of Browser Extensions”, In 38th IEEE Symposium on Security and Privacy (S&P 2017), San Jose, United States, 2017.

[9] Iskander Sánchez-Rola, Igor Santos, and Davide Balzarotti, “Extension Breakdown: Security Analysis of Browsers Extension Resources Control Policies”, In 26th USENIX Security Symposium, 679–694, 2017.

[10] Gábor György Gulyás, Dolière Francis Somé, Nataliia Bielova, and Claude Castelluccia, “To Extend or not to Extend: on the Uniqueness of Browser Extensions and Web Logins”, In 2018 Workshop on Privacy in the Electronic Society (WPES’18). ACM, 14–27, 2018.

[11] Martin Mulazzani, Philipp Reschl, Markus Huber, Manuel Leithner, Sebastian Schrittwieser, Edgar Weippl, and FH Campus Wien, “Fast and reliable browser identification with javascript engine fingerprinting”, In Web 2.0 Workshop on Security and Privacy (W2SP), Vol. 5, 2013.

[12] Nick Nikiforakis, Alexandros Kapravelos, Wouter Joosen, Christopher Kruegel, Frank Piessens, and Giovanni Vigna, “Cookieless Monster: Exploring the Ecosystem of Web-Based Device Fingerprinting”, In Proceedings of the 2013 IEEE Symposium on Security and Privacy (SP ’13), IEEE Computer Society, Washington, DC, USA, 541–555, 2013.

[13] Michael Schwarz, Florian Lackner, and Daniel Gruss, “JavaScript Template Attacks: Automatically Inferring Host Information for Targeted Exploits”, In 26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019.

[14] Iskander Sanchez-Rola, Igor Santos, and Davide Balzarotti, “Clock Around the Clock: Time-Based Device Fingerprinting”, In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (CCS’18). ACM, New York, NY, USA, 1502–1514, 2018.

[15] Antoine Vastel, Pierre Laperdrix, Walter Rudametkin, and Romain Rouvoy, “FP-STALKER: Tracking Browser Fingerprint Evolutions”, In 39th IEEE Symposium on Security and Privacy (S&P 2018). San Fransisco, United States. 2018.

[16] Alejandro Gómez-Boix, Pierre Laperdrix, and Benoit Baudry, “Hiding in the Crowd: an Analysis of the Effectiveness of Browser Fingerprinting at Large Scale”, In WWW 2018: The 2018 Web Conference, Lyon, France. 2018.

[17] Pierre Laperdrix,Walter Rudametkin, and Benoit Baudry, “Beauty and the Beast: Diverting modern web browsers to build unique browser fingerprints”, In 37th IEEE Symposium on Security and Privacy (S&P 2016). San Jose, United States. 2016.

[18] Gunes Acar, Marc Juarez, Nick Nikiforakis, Claudia Diaz, Seda Gürses, Frank Piessens, and Bart Preneel, “FPDetective: dusting the web for fingerprinters”, In Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security (CCS ’13), ACM, New York, NY, USA, 1129–1140, 2013.

[19] OpenWPM - A web privacy measurement framework, 2018, https://github.com/citp/OpenWPM.

[20] Nasser Mohammed Al-Fannah, Wanpeng Li, and Chris J. Mitchell, “Beyond Cookie Monster Amnesia: Real World Persistent Online Tracking”, In Information Security - 21st International Conference, ISC 2018, Guildford, UK, September 9-12, 2018, Proceedings. 481–501, 2018.

[21] Takamichi Saito, Koki Yasuda, Kazuhisa Tanabe, and Kazushi Takahash, “Web Browser Tampering: Inspecting CPU Features from Side-Channel Information”, In 2017 12th International Conference on Broadband and Wireless Computing, Communication and Applications (BWCCA). 392–403, 2017. https://doi.org/10.1007/978-3-319-69811-336

[22] ElBanna, A., Abdelbaki, N., “Browsers Fingerprinting Motives, Methods, and Countermeasures”, International Conference on Computer, Information and Telecommunication Systems (CITS), pp. 1-5, Colmar, 2018.

[23] Elizarov, M., Ivanyuk, V., Soloviev, V., Tsvirkun, A., “Identification of high-frequency traders using fuzzy logic methods”, 2017 Tenth International Conference Management of Large-Scale System Development (MLSD), pp. 1-4, Moscow, 2017.

[24] Efremov, I., “Computational personality prediction based on digital footprint of a social media user”, Procedia computer science, pp. 185-193, 2019.

[25] Efremov, E.A., Kovalevsky, A.E., Ershova, Y.A., “Effective ways of information collection from open sources in the Internet”, Theoretical and practical issues of integrated safety”, pp. 217-218, 2018.

[26] Akhtar, Z., Mouree, M.R., Dasgupta, D., "Utility of Deep Learning Features for Facial Attributes Manipulation Detection", 2020 IEEE International Conference on Humanized Computing and Communication with Artificial Intelligence, 2020.
[27] Iskhakova, A.O., Iskhakov, A.Y., Meshcheryakov, R., Jharko, E., "Method of Verification of Robotic Group Agents in the Conditions of Communication Facility Suppression", IFAC-PapersOnLine, vol. 52, Issue 13, pp. 1397-1402, 2019.
[28] Kytmanov, N.S., Putilova, S.E., Kamennaya, E.V., Scherbina, I.A., "Protection from repeated internet voices with using digital printing", Marine intelligent technology, no. 44, pp. 88-92, 2019.
[29] Komissarov, A.A., Tretyakov, V.C., "Digital footprint", Endrunch Tomsk, pp. 146-153. 2019
[30] Feher, K., "Digital identity and the online-self: footprint strategies - an exploratory and comparative research study", Journal of information science, pp.1-5, 2019.
[31] Iskhakova, A.O., Iskhakov, A.Y., Meshcheryakov, Bendrau, R., Melekova, O., "Using a heatmap of user behavior in the problem of identifying the subject of an information security incident", Proceedings of the SPIIRAS, vol 6 (61). pp. 147-171, 2018.
[32] Nair, K., RoseLalson, E., "The Unique Id's You Can't Delete: Browser Fingerprints", International Conference on Emerging Trends and Innovations in Engineering and Technological Research (ICETIETR). pp. 1-5, Ernakulam, 2018.
[33] Luangmaneerote, S., Zaluska, E., Carr, L., "Inhibiting Browser Fingerprinting and Tracking", IEEE 3rd International Conference on Big Data Security on Cloud, pp. 63-68., Beijing, 2017.
[34] Jiang, W., "Tracking Your Browser with High-Performance Browser Fingerprint Recognition Model", China communications, vol. 17 (3), pp. 168-175, Beijing, 2020.
[35] Jiang, W., Wang, X., and Other, "Tracking your browser with high-performance browser fingerprint recognition model", China Communications, vol. 17, no. 3, pp. 168-175, 2020.
[36] Xu, Y., "Application Research Based on Machine Learning in Network Privacy Security", 2020 International Conference on Computer Information and Big Data Applications (CIBDA), pp. 237–240, Guiyang, 2020.
[37] Pan, F., Wen, H., and Other, "Physical layer authentication based on channel information and machine learning", 2017 IEEE Conference on Communications and Network Security (CNS), pp. 364–365, Las Vegas, 2017.
[38] Agafonov, U.M., "Deanonymization of users based on digital, fingerprint", Information Space Security: Collection of Proceedings of the XVI All-russian Scientific and Practical Conference of Students, Young Scientists. Ekaterinburg: Ural Federal university named after the first President of Russia B.N. Elcin, pp. 3-5, 2018.
[39] Platonov, T.S., Ogolyuk, A.A., "Web Application Layer Firewalls in the Modern World", Software engineering and computer technology (Major's readings), pp. 106-119, 2019.
[40] Peter Eckersley, "How Unique is YourWeb Browser?", In Proceedings of the 10th International Conference on Privacy, Enhancing Technologies (PETS'10), Springer-Verlag, Berlin, Heidelberg, 1–18, 2010.
[۴۱] مریم طائبی ، علی بهلولی ، مرجان کائدی، "مروری بر حمله های انگشت نگاری تارنما"، نشریه منادی امنیت فضای تولید و تبادل اطلاعات (افتا)، سال ششم شماره ۲ ، ۱۳۹۶.
[۴۲] سعید شیرودی، امیرمهدی صادق زاده و رسول جلیلی، "فانوس: راهکار مقابله با حملات انگشت نگاری وب"، شانزدهمین کنفرانس بین المللی انجمن رمز ایران در دانشگاه فردوسی مشهد، ۱۳۹۸.