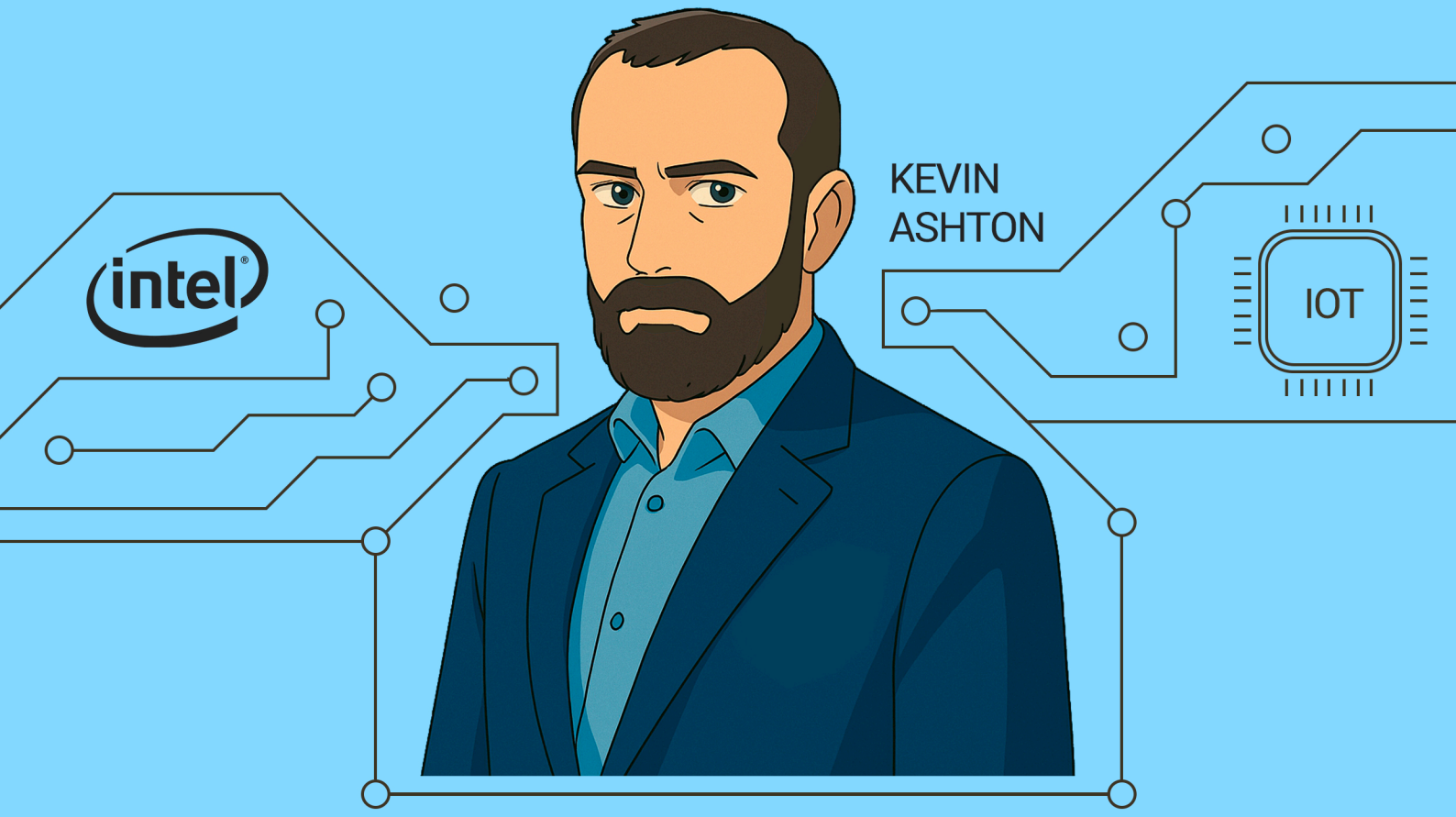
ماتریس ۴

صاحب امتیاز : انجمن علمی مهندسی کامپیوتر
دانشگاه شاهد | فروردین ماه ۱۴۰۴



در این شماره میخوانیم :


بسم الله الرحمن الرحيم

شناسنامه

نشریه ماتریس

صاحب امتیاز: انجمن علمی مهندسی کامپیوتر دانشگاه شاهد

مدیر مسئول: علی بقائی راوری

سر دبیر: فربد فیضی

تیم تحریریه این شماره: فاطمه غلامی | سارا امیرحسینی | امیرحسین ملکی | سارا کاظم زاده عطار | محمدمهدی بابابیک | محدثه جوان | فربد فیضی

تیم ویراست این شماره: محدثه جوان | نیما آذری

طراح جلد: محمدرضا ناحی داریانی

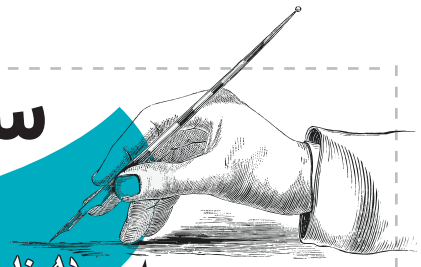
طراح مجله: علی بقائی راوری

شبکه‌های اجتماعی: @MatrisMagazine

شماره چهارم | فروردین ماه ۱۴۰۴

نشریه ماتریس نشریه‌ای است که با همت دانشجویان کامپیوتر دانشگاه شاهد در دی ماه ۱۴۰۳ با صاحب امتیازی انجمن علمی مهندسی کامپیوتر دانشگاه شاهد شروع به کار کرده است.

فهرست



سخن مدیر مسئول

به نام خالق فصل‌های عمیق و ایده‌های روشن

چهارمین شماره ماتریس، آغاز سال ۱۴۰۴ را با هیاهوی نوروز پشت سر گذاشته و با تمرکز بر عمق و کیفیت، قدمی تازه در مسیر خود برمی‌دارد.

ماتریس دیگر فقط یک مجله نیست؛ صدایی‌ست برای نسلی که می‌خواهد فکر کند، بسازد، و آینده را طراحی کند. فضایی برای آن‌هایی که به‌جای پاسخ‌های سریع، دل‌خوش پرسیدن سؤال‌های عجیب و دیدن دنیا از زوایای غیرمنتظره‌اند.

در این شماره، سعی کردیم فراتر از مرزهای کلاس و سرفصل، به دغدغه‌هایی بپردازیم که در دل گفت‌وگوهای دانشجویی شکل گرفته‌اند. ما ماتریس را نه به عنوان یک پروژه‌ی موقت، بلکه به عنوان بخشی از یک شبکه‌ی فکری می‌بینیم؛ نقطه‌ای کوچک ولی زنده در جریان رشد جمعی.

بازخوردهای گسترده و پرمهر شما در شماره‌های قبلی، این باور را تقویت کرد که آنچه آغاز کرده‌ایم، ارزش ادامه دارد. با همین انگیزه، این شماره را پیشکش شما می‌کنیم.

علی بقائی راوری

مدیر مسئول

کدهایی که میفهمن!

فاطمه غلامی

وقتی برای اولین‌بار یک مدل زبانی تونست جمله‌ی ناتمام رو تموم کنه، جهان از خودش پرسید: «آیا ماشین واقعاً می‌تونه زبان ما رو بفهمه؟»

از همون لحظه بود که موجی عظیم به راه افتاد، موجی که امروز اسمش رو همه شنیدن: Transformer

حالا توی این شماره، می‌خواهیم یک سفر هیجان انگیز به دل یکی از مهم‌ترین انقلاب‌های هوش مصنوعی داشته باشیم؛ سفری از دل معماری‌های عمیق مغزیِ ماشین‌ها تا سکویی به نام Hugging Face که مسیر یادگیری ماشین رو برای همه باز کرده.

اگر هیچ‌چیز در مورد این مفاهیم نمی‌دونی، نگران نباش؛ ما اینجاییم تا از صفر شروع کنیم و دنیایی رو بهت نشون بدیم که آینده قراره توش رقم بخوره.

بیا شروع کنیم ...

Transformer یعنی چی؟ چرا این‌قدر مهمه؟

در دنیای هوش مصنوعی، همیشه یه سؤال کلیدی وجود داشته: «چطور می‌تونیم به ماشین یاد بدیم که زبان انسان رو بفهمه؟». قبل از اینکه مدل‌های Transformer وارد صحنه بشن، ابزار اصلی ما برای این کار شبکه‌های عصبی بازگشتی مثل RNN و LSTM بودن. مدل‌های RNN و LSTM، زبان رو مرحله به مرحله می‌خوندن. مثلاً برای فهمیدن جمله‌ی: «من دیروز به پارک رفتم»، مدل باید اول «من» رو پردازش می‌کرد، بعد «دیروز» و همین‌طور تا آخر. این یعنی مدل فقط اطلاعات گذشته رو داشت، نه آینده‌ی جمله. ایراد بزرگ چی بود؟« اگر جمله خیلی طولانی می‌شد، مدل کلمات اول رو فراموش می‌کرد! » نتیجه چی میشد؟ «مدل‌ها توی ترجمه یا درک دقیق جمله‌ها، مخصوصاً وقتی طولانی بودن، خیلی ضعیف عمل می‌کردن.»

اما در سال ۲۰۱۷، تیم تحقیقاتی گوگل در مقاله‌ای به‌نام«Attention Is All You Need» مدلی معرفی کرد به اسم Transformer که ساختارش متفاوت با

هر چیزی بود که تا اون زمان دیده بودیم.

به‌جای خواندن مرحله‌به‌مرحله، Transformer می‌تونه همه‌ی کلمات جمله رو به صورت هم‌زمان ببینه و مهم‌تر از اون، بفهمه هر کلمه به کدوم کلمه‌ی دیگه مربوطه.

مغز ما چطوری زبان رو می‌فهمه؟ وقتی می‌شنویم: «کتابی که دوستم بهم داده بود...» ما ناخودآگاه دنبال ادامه‌اش می‌گردیم و می‌فهمیم «کتاب» به «داده بود» مربوطه، حتی اگه فاصله‌ی زیادی بینشون باشه.

Transformer هم همین کارو می‌کنه، با تکنیکی به‌نام: «Self-Attention» اینجا هر کلمه، می‌تونه به بقیه‌ی کلمات جمله نگاه کنه و تصمیم بگیره که به کدوم‌ها بیشتر توجه کنه. مثلاً توی جمله‌ی: «من گربه‌ای که دیشب دیدم رو خیلی دوست دارم!» مدل متوجه می‌شه که «گربه» به «دیدم» مربوطه، نه «دوست»

پس ترنسفورمرها مهم هستندچون این ساختار باعث شد: مدل‌ها بتونن هم‌زمان همه‌ی ورودی رو پردازش کنن برخلاف RNN ، آموزش موازی بشه و سرعت چند برابر! ، روابط پیچیده بین کلمات دور از هم درک بشه و قابلیت گسترش به مدل‌های عظیم مثل GPT-4 یا BERT فراهم بشه.

از وقتی Transformer معرفی شد، تقریباً تمام پیشرفت‌های چشم‌گیر در حوزه NLP (و حتی پردازش تصویر!) بر پایه‌ی اون بنا شده.

مدل‌هایی مثل: BERT (ساخته‌ی گوگل)، T5، GPT، LLaMA، BLOOM، و حتی مدل‌های تصویری مثل ViT (Vision Transformer) همه از همین معماری الهام گرفتن.

پس اگه یه انقلاب توی فهم زبان توسط ماشین‌ها بوده باشه، بدون شک اون انقلاب اسمش Transformer بوده. و اگه امروز می‌تونیم با یک ربات چت کنیم، مقاله خلاصه کنیم، یا از یه مدل بخوایم شعر بگه، باید بدونیم همه‌ش از همین جا شروع شد.

Hugging Face چیه؟ چرا همه ازش حرف می زنن ؟

اگه شنیدی کسی اسم Hugging Face رو می‌بره و از مدل‌ها و کتابخانه‌های متنوعش صحبت می‌کنه، بدون که داره درباره یکی از مهم‌ترین منابع دنیای هوش مصنوعی حرف می‌زنه.

Hugging Face در ابتدا به عنوان یک استارتاپ در سال ۲۰۱۶ در نیویورک شروع به کار کرد. هدف اصلی‌ش این بود که مدل‌های یادگیری ماشین رو برای توسعه‌دهندگان و محققین در دسترس‌تر کنه. اما خیلی زود این شرکت تبدیل به مرکز اصلی برای مدل‌های زبانی پیشرفته مثل GPT، BERT، و T5 شد. Hugging Face بیشتر از اینکه یک شرکت معمولی باشه، به نوعی یک پلتفرم تمام‌عیار برای کار

با مدل‌های هوش مصنوعی به شمار میاد. این پلتفرم شامل چندین بخش مهمه:

۱. کتابخانه‌های پُرکاربرد:
این کتابخانه‌ها به برنامه‌نویس‌ها و محققین این امکان رو میدن که مدل‌های زبانی پیشرفته رو با کمترین دردسر بسازن و استفاده کنن. معروف‌ترین کتابخانه‌های Hugging Face عبارتند از:

- Transformers : برای استفاده از مدل‌های پیشرفته NLP مثل BERT، GPT، T5 و غیره.
- Datasets : یک کتابخانه برای کار با داده‌های آماده‌شده برای آموزش مدل‌ها.
- Gradio : برای ساخت رابط‌های کاربری جذاب و تعاملی که مدل‌ها رو به نمایش می‌ذاره.
- Spaces : برای اشتراک‌گذاری و تست پروژه‌ها به صورت آنلاین.

۲. Model Hub :
یکی از بزرگترین نقاط قوت Hugging Face، ویژگی Model Hub شه. جایی که شما می‌تونید انواع مختلف مدل‌ها رو پیدا کنید، از مدل‌های ساده‌ای مثل text classification گرفته تا مدل‌های پیچیده‌تری مثل question answering یا حتی image classification . این مدل‌ها از طرف پژوهشگران، توسعه‌دهندگان و شرکت‌های بزرگ در اختیار عموم قرار می‌گیرند.

چطور به نفع‌مون باشه؟ به‌جای اینکه شما بخواهید مدل خودتون رو از صفر بسازید، می‌تونید از مدل‌های موجود روی Hugging Face استفاده کنید و حتی اون ها رو fine-tune کنید تا برای نیازهای خاص‌تون بهینه بشن.

چرا Hugging Face برای توسعه‌دهندگان و پژوهشگران محبوب شده؟ یکی از مزایای Hugging Face اینه که حتی برای کسانی که تجربه زیادی در زمینه‌ی AI ندارن، استفاده از مدل‌های پیچیده رو خیلی ساده کرده. با چند خط کد، می‌تونید مدل‌های پیشرفته رو اجرا کنید. علاوه بر اون یک جامعه جهانی از توسعه‌دهندگان، محققان و علاقه‌مندان به AI رو جمع کرده. شما می‌تونید مدل‌های خودتون رو به اشتراک بذارید، به مدل‌های دیگه کمک کنید یا از پروژه‌های دیگه استفاده کنید. چون مدل‌های Hugging Face به‌طور مداوم به‌روزرسانی می‌شن، شما همیشه می‌تونید از آخرین و بهترین مدل‌ها استفاده کنید. مثلاً GPT-3، RoBERTa، BERT و حتی مدل‌های جدیدتر که مخصوص کارهای پیچیده‌تر طراحی شدن. برای کاربرانی که از تجربه برنامه‌نویسی کمتری برخوردارن، Hugging Face ابزارهایی مثل Gradio رو معرفی کرده که به سادگی می‌تونید یه رابط کاربری تعاملی برای مدل‌های خودتون بسازید. حتی می‌تونید با دوستان یا همکارانتون مدل رو به اشتراک بذارید و بازخورد بگیرید.

چرا Hugging Face اینقدر محبوبه؟

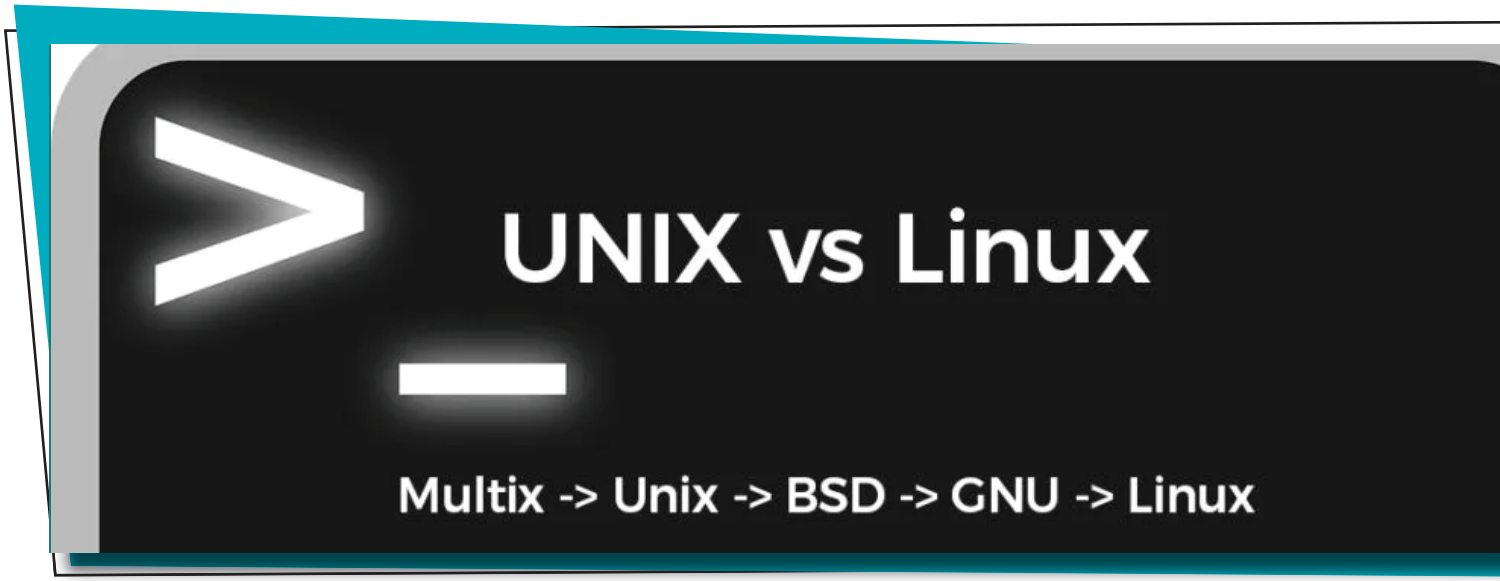
یکی از دلایل مهم موفقیت Hugging Face اینه که این پلتفرم دنیای تحقیقاتی و صنعت رو بهم پیوند داده. در حالی که خیلی از پلتفرم‌های مشابه فقط به تحقیقات علمی محدود می‌شن، Hugging Face به‌شدت عملی و کاربردی هست. محققین می‌تونن مدل‌های پیچیده رو بسازن و آزمایش کنن، در حالی که توسعه‌دهندگان می‌تونن همون مدل‌ها رو توی پروژه‌های تجاری خودشون پیاده‌سازی کنن.

در واقع، Hugging Face به نوعی GitHub برای مدل‌های یادگیری ماشین شده. شما می‌تونید مدل‌های از پیش آموزش‌داده‌شده رو دانلود کنید، اونا رو تست کنید، و حتی اگه نیاز به تغییراتی داشتند، خودتون مدل رو تغییر بدید و به اشتراک بذارید.

خبر داغ

مدل‌های کوچک، قدرت‌های بزرگ!

تو اسفند ۱۴۰۳، Hugging Face از یک مدل سبک و جدید رونمایی کرد:



SmolVLM-256M یـه مـدل تصویری-زبانـی (Vision-Language) کـه فقـط بـا ۲۵۶ میلیـون پارامتـر روی دسـتگاه‌های ضعیـف (مثـل گوشـی یـا لپ‌تـاپ) هـم اجـرا می‌شـه!

چـرا ایـن مهمـه؟ چـون هـوش مصنوعـی داره بـه سـمت «لبـه» (Edge AI) مـی‌ره

یعنـی به‌جـای اینکـه همه‌چـی تـوی سـرورهای ابـری سـنگین اجـرا بشـه، قـراره مدل‌هـا بیـان روی گوشـی‌مون، سـاعت هوشـمند، دوربیـن و حتـی یخچـال!

بـرای کشـورهایی مثـل مـا کـه منابـع محاسـباتی محـدوده، ایـن یـه فرصـت طلاییـه.

تاریخچه یونیکس و لینوکس

سفری هیجان‌انگیز از آزمایشگاه‌های بل تا انقلاب منبع باز

سارا امیرحسینی
امیرحسین ملکی

در دنیـای پرهیاهـوی فنـاوری اطلاعـات، دو نـام ماننـد سـتون‌های اسـتوار و افسـانه‌ای ایسـتاده‌اند: یونیکـس و لینوکـس. ایـن دو سیسـتم‌عامل نه‌تنهـا پایه‌هـای بسـیاری از فناوری‌هـای امـروزی را شـکل داده‌انـد، بلکـه داسـتان‌هایی پـر از نـوآوری، نبـوغ، چالـش و پیـروزی را در دل خـود جـای داده‌انـد کـه هـر علاقه‌منـدی بـه دنیـای کامپیوتـر را بـه وجـد مـی‌آورد. از روزهایـی کـه کامپیوترهـا غول‌هایـی آهنـی و دسـت‌نیافتنی بودنـد تـا عصـر پروژه‌هـای جهانـی منبـع بـاز، یونیکـس و لینوکـس مسـیری را طـی کرده‌انـد کـه تاریـخ فنـاوری را بـرای همیشـه دگرگـون کـرده اسـت. در ایـن مقالـه، مـا بـه سـفری مهیـج در دل تاریخچـه ایـن دو سیسـتم‌عامل می‌رویـم؛ از علـل پیدایـش یونیکـس در دهـه ۱۹۶۰ تـا ظهـور لینوکـس و تأثیـر آن‌هـا بـر دنیـای امـروز.

مقدمه: تولد ایده‌هایی که دنیا را تغییر داد

دهـه ۱۹۶۰ را تصـور کنیـد: زمانـی کـه کامپیوترهـا موجوداتی عظیم‌الجثـه، گران‌قیمـت و پیچیـده بودنـد. در آن روزگار، سیسـتم‌عامل‌ها ماننـد پادشـاهانی مسـتبد عمـل می‌کردنـد؛ سـنگین، ناکارآمـد و دور از دسـترس کاربـران عـادی. امـا در ایـن میـان، گروهـی از ذهن‌هـای خـلاق تصمیـم گرفتنـد ایـن وضعیـت را تغییـر دهنـد. داسـتان یونیکـس از دل یـک شکسـت بـزرگ آغـاز شـد و بـه یـک افسـانه تبدیـل شـد. سـپس، سـال‌ها بعـد، لینوکـس از خاکسـتر همـان ایده‌هـا برخاسـت و انقلابـی را در دنیـای فنـاوری بـه راه انداخـت. ایـن مقالـه نه‌تنهـا تاریخچـه و علـل پیدایـش یونیکـس را کاوش می‌کنـد، بلکـه ارتبـاط آن بـا لینوکـس و دنیـای مـدرن را نشـان می‌دهـد. آمـاده

باشید تا با داستان‌هایی از پشت صحنه، نوآوری‌های شگفت‌انگیز و چالش‌های نفس‌گیر همراه شوید!

یونیکس: از خاکستر یک رویا تا پایه‌گذاری یک امپراتوری

ریشه‌ها در پروژه Multics

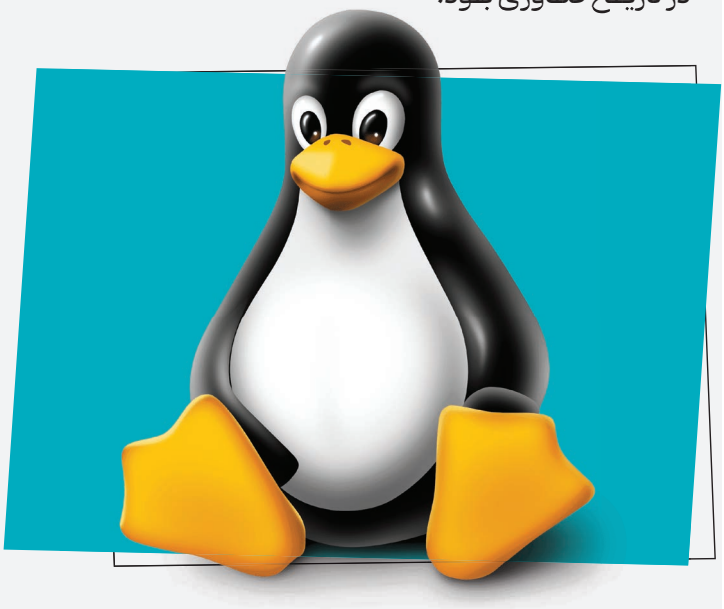
برای فهمیدن اینکه چرا یونیکس به وجود آمد، باید به سال ۱۹۶۴ بازگردیم. در آن زمان، شرکت AT&T، MIT و جنرال الکتریک دست به دست هم دادند تا پروژه‌ای جاه‌طلبانه به نام Multics (سرویس اطلاعات و محاسبات چندگانه) را راه‌اندازی کنند. هدف؟ ساخت یک سیستم‌عامل چندکاربری و چندوظیفه‌ای که بتواند چندین نفر را همزمان به یک کامپیوتر متصل کند. ایده‌ای انقلابی بود، اما اجرا؟ یک کابوس تمام‌عیار! Multics بیش از حد پیچیده شد، هزینه‌ها سر به فلک کشید و پروژه در سال ۱۹۶۹ توسط AT&T متوقف شد. اما این پایان ماجرا نبود، بلکه آغاز یک افسانه بود.

در آزمایشگاه‌های بل، چند نفر از توسعه‌دهندگان Multics، از جمله کن تامپسون و دنیس ریچی، از این شکست ناامید نشدند. آن‌ها به جای تسلیم، تصمیم گرفتند رویاهایشان را ساده‌تر کنند. تامپسون که از پیچیدگی‌های Multics خسته شده بود، می‌خواست سیستمی بسازد که کارآمد، سبک و قابل فهم باشد. اینجاست که جرقه یونیکس زده شد.

تولد یونیکس: یک شروع ساده اما درخشان

در سال ۱۹۶۹، کن تامپسون روی یک کامپیوتر قدیمی PDP-7 کار را آغاز کرد. او با استفاده از زبان اسمبلر، یک سیستم‌عامل کوچک نوشت که شامل یک سیستم فایل ساده، چند ابزار اولیه و یک پوسته (shell) بود. این نسخه ابتدایی، که بعدها به یونیکس معروف شد، یک طعنه آشکار به Multics بود. نام «Unix» (به معنای «تک‌کاره») با بازیگوشی انتخاب شد تا نشان دهد این سیستم برخلاف Multics، روی سادگی و کارایی تمرکز دارد.

تا سال ۱۹۷۱، یونیکس به کامپیوتر PDP-11 منتقل شد و اولین ویرایش رسمی آن منتشر شد. این نسخه شامل ابزارهایی مثل اسمبلر، ویرایشگر متن و دستورات ابتدایی بود. اما چیزی که یونیکس را واقعاً متمایز کرد، در سال ۱۹۷۳ رخ داد. دنیس ریچی تصمیم گرفت کل سیستم را با زبان برنامه‌نویسی جدیدی به نام C بازنویسی کند. این زبان که خودش آن را اختراع کرده بود، یونیکس را به اولین سیستم‌عامل قابل حمل تبدیل کرد؛ یعنی سیستمی که می‌توانست روی سخت‌افزارهای مختلف اجرا شود، بدون اینکه نیاز به بازنویسی کامل داشته باشد. این لحظه، نقطه عطفی در تاریخ فناوری بود.



گسترش و محبوبیت یونیکس در دانشگاه‌ها

در دهه ۱۹۷۰، یونیکس به لطف سادگی و قدرتش، در دانشگاه‌ها و مراکز تحقیقاتی محبوب شد. در سال ۱۹۷۵، ویرایش ششم یونیکس منتشر شد و به پایه‌ای برای BSD (توزیع نرم‌افزاری برکلی) تبدیل شد که توسط دانشگاه کالیفرنیا، برکلی توسعه یافت. BSD بعدها با افزودن پشته شبکه TCP/IP در نسخه BSD 4.2 (سال ۱۹۸۳)، اینترنت را متحول کرد. تصور کنید بدون یونیکس، شاید امروز اینترنت به شکل امروزی‌اش وجود نداشت!

چالش‌های تجاری‌سازی و جنگ‌های حقوقی

اما همه چیز به این سادگی پیش نرفت. در دهه

۱۹۸۰، AT&T تصمیم گرفت یونیکس را از یک پروژه تحقیقاتی به یک محصول تجاری تبدیل کند. مجوزهای گران‌قیمت و محدودیت‌های قانونی اعمال شدند و این کار بسیاری از توسعه‌دهندگان را خشمگین کرد. در همین زمان، BSD که نسخه‌ای رایگان از یونیکس بود، با AT&T وارد دعواهای حقوقی شدیدی شد. این نبردها در نهایت به ظهور نسخه‌های منبع باز مثل FreeBSD منجر شد که تا امروز هم زنده و فعال هستند.

لینوکس: انقلابی از اتاق خواب یک دانشجو

پیش‌زمینه ی Minix و جرقه یک ایده

در حالی که یونیکس درگیر تجاری‌سازی و دعواهای حقوقی بود، در سال ۱۹۹۱، یک دانشجوی جوان فنلاندی به نام لینوس تورووالدز وارد صحنه شد. او که از سیستم‌عامل Minix (یک سیستم مشابه یونیکس که برای آموزش طراحی شده بود) استفاده می‌کرد، احساس کرد محدودیت‌های آن دست و پایش را بسته است. Minix، ساخته اندرو تانه‌نبام، عالی بود اما برای کارهای واقعی، کافی نبود. لینوس تصمیم گرفت خودش دست به کار شود و یک هسته سیستم‌عامل جدید بسازد.

در ۲۵ آگوست ۱۹۹۱، او در گروه خبری comp.os.minix پیامی نوشت که دنیا را تکان داد:

«من در حال کار روی یک سیستم‌عامل رایگان (فقط برای سرگرمی) هستم... اگر کسی علاقه‌مند است، خوشحال می‌شوم نظراتش را بشنوم.»

تولد لینوکس و قدرت منبع باز

اولین نسخه هسته لینوکس در سپتامبر ۱۹۹۱ منتشر شد. در سال ۱۹۹۲، لینوس آن را تحت مجوز GPL (مجوز عمومی گنو) قرار داد که به هر کسی اجازه می‌داد کد را ببیند، تغییر دهد و پخش کند. این تصمیم، لینوکس را از یک پروژه شخصی به یک حرکت جهانی تبدیل کرد. توسعه‌دهندگان از سراسر جهان به او پیوستند و کد را بهبود دادند.

تا سال ۱۹۹۴، نسخه ۱.۰ لینوکس منتشر شد و نشان داد که این سیستم به بلوغ رسیده است. شرکت‌هایی مثل IBM و Oracle از آن حمایت کردند و توزیع‌هایی مثل Red Hat، Debian و Ubuntu ظهور کردند. لینوکس به سرعت در سرورها، ابررایانه‌ها و حتی دستگاه‌های کوچک جا باز کرد.

اینترنت اشیا و حوادث جاده ای!

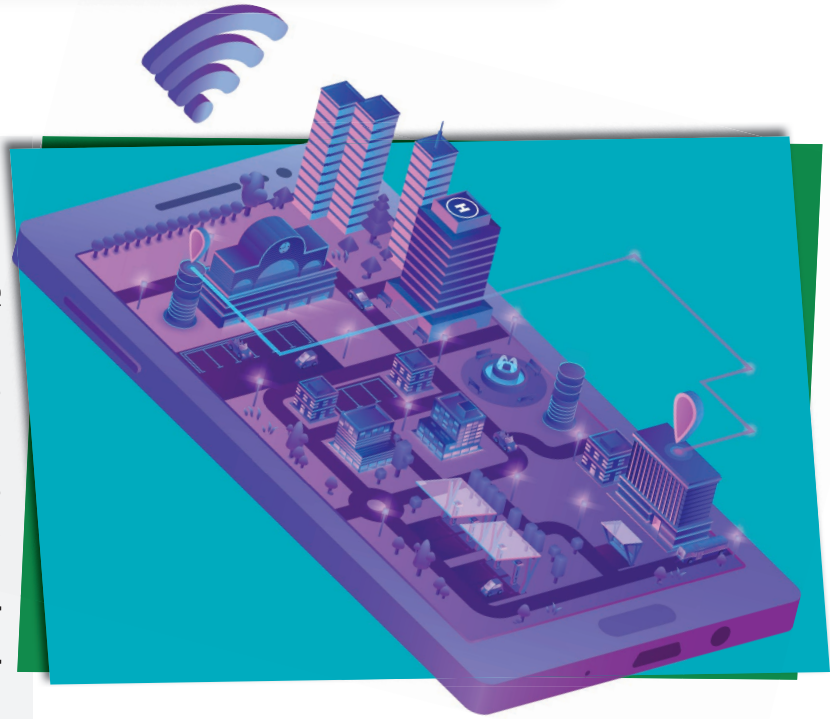
سارا کاظم زاده عطار

با فرا رسیدن سال نو، بسیاری از مردم از فرصت تعطیلات برای مسافرت استفاده می‌کنند. متأسفانه، این دوران با افزایش تردد در جاده‌ها همراه است که احتمال وقوع حوادث رانندگی را بیشتر می‌کند. در این شرایط، استفاده از فناوری‌های نوین می‌تواند نقشی کلیدی در کاهش خطرات و افزایش ایمنی سفرها ایفا کند. یکی از این فناوری‌ها، اینترنت اشیا (IoT) است که با ایجاد شبکه‌ای از دستگاه‌های هوشمند، امکان پیشگیری و مدیریت بهتر حوادث جاده‌ای را فراهم می‌کند. در این مقاله، به بررسی تأثیر IoT در بهبود ایمنی جاده‌ها و کاهش تصادفات می‌پردازیم.

اینترنت اشیا چیست؟

اینترنت اشیا (Internet of Things) به معنای اتصال دستگاه‌ها و اشیای فیزیکی مختلف به اینترنت است. این دستگاه‌ها می‌توانند شامل هر چیزی باشند، از یخچال و ماشین لباسشویی گرفته تا خودروها، ساعت‌های هوشمند، چراغ‌های خیابانی و حتی سنسورهای کشاورزی. این اشیا قادرند اطلاعات را از محیط اطراف جمع‌آوری کرده و با استفاده از نرم‌افزارها و پردازش‌های هوشمند، اقدامات مناسب را انجام دهند.

این دستگاه‌ها مجهز به سنسورها، نرم‌افزارها و فناوری‌های دیگری هستند که به آن‌ها اجازه می‌دهند داده‌ها را پس از جمع‌آوری، پردازش کرده و با سایر دستگاه‌ها و سیستم‌ها از طریق اینترنت تبادل کنند. این فرایند امکان نظارت و کنترل هوشمند را فراهم



کرده که موجب افزایش بهره‌وری، کاهش هزینه‌ها و بهبود ایمنی در حوزه‌های مختلف، ازجمله حمل‌ونقل می‌شود.

خودروهای متصل

خودروهای متصل به خودروهایی اطلاق می‌شود که با استفاده از تکنولوژی‌های اینترنت اشیا، قادر به تبادل اطلاعات با سایر خودروها، زیرساخت‌های جاده‌ای و شبکه‌های ابری هستند. این اتصال به خودروها امکان می‌دهد تا اطلاعات مربوط به موقعیت جغرافیایی، وضعیت خودرو و شرایط ترافیک را به صورت زنده دریافت و ارسال کنند و مسیر بهینه را پیشنهاد دهند.

سیستم‌های هشداردهنده هوشمند

سیستم‌های هشداردهنده هوشمند تصادف، بر اساس اطلاعات دریافتی از خودروهای دیگر عمل می‌کنند. این سیستم‌ها می‌توانند به رانندگان هشدار دهند و از وقوع تصادفات جلوگیری کنند. با استفاده از این فناوری، احتمال بروز تصادفات به میزان قابل توجهی کاهش می‌یابد.

نظارت فنی بر خودروها

سنسورهای نصب‌شده در خودروها می‌توانند عملکرد موتور، فشار لاستیک‌ها، میزان مصرف سوخت و سایر عوامل فنی را کنترل کنند. در صورت بروز نقص فنی، این سیستم‌ها به راننده هشدار داده و حتی امکان ارسال خودکار اطلاعات به تعمیرگاه را فراهم می‌کنند تا از وقوع مشکلات ناگهانی در جاده جلوگیری شود.

پاسخگویی سریع به حوادث

در صورت وقوع تصادف، سیستم‌های IoT می‌توانند موقعیت دقیق حادثه را شناسایی کرده و اطلاعات لازم را به مراکز امدادی ارسال کنند. این فناوری باعث کاهش زمان واکنش تیم‌های امدادی شده و می‌تواند جان بسیاری از افراد را نجات دهد.

چالش‌ها و محدودیت‌های اینترنت اشیا در صنعت خودروسازی

مسائل امنیتی

امنیت سایبری و حفاظت از داده‌ها باید در طراحی و پیاده‌سازی سیستم‌های هوشمند رانندگی مورد توجه قرار گیرد؛ زیرا هکرها و مهاجمین می‌توانند به اطلاعات حساس دسترسی پیدا کنند یا حتی کنترل خودروها را به دست بگیرند.

مشکلات فناوری و هزینه

پیاده‌سازی و نگهداری فناوری‌های اینترنت اشیا در صنعت خودروسازی نیازمند زیرساخت‌های پیشرفته و هزینه‌های بالاست. همچنین، هماهنگی و استانداردسازی فناوری‌های مختلف برای اطمینان از عملکرد صحیح و همگام‌سازی آن‌ها نیز یک چالش مهم است.

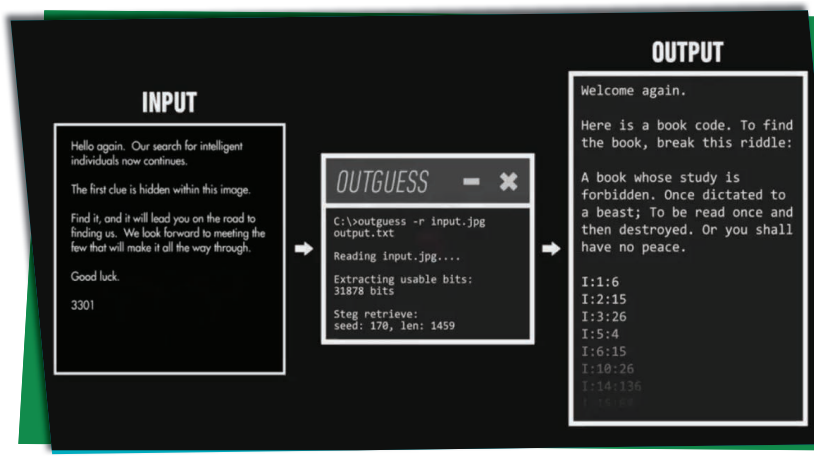
امیدواریم که با پذیرش این تکنولوژی‌ها، شاهد جاده‌هایی ایمن‌تر و سفرهایی بدون حادثه باشیم.

سیکادای رازآلود ۲!

امیرحسین ملکی

مرحله دوم ۲۰۱۳:

در سال ۲۰۱۳، دقیقاً یک سال و یک روز پس از اولین پیام، سیکادا۳۳۰۱ دوباره ظاهر شد. این بار هم پیام از طریق تصویری رمزآلود در فروم /b/ وب‌سایت 4chan منتشر شد و مانند قبل، امضای رمزنگاری‌شده‌ی PGP



اصالت آن را تأیید می‌کرد.

کاربران با استفاده از نرم‌افزار OutGuess موفق به کشف پیامی پنهان شدند که به کتاب اسرارآمیز Liber AL vel Legis اشاره داشت؛ اثری نوشته‌ی آلیستر کراولی که به موضوعات عرفانی و رازآلود می‌پردازد.

این انتخاب نشان داد که سیکادا همچنان به منابع غیرمعمول علاقه‌مند است و معماها قرار است جدی‌تر و پیچیده‌تر شوند. با استفاده از رمزنگاری بر اساس متن کتاب، شرکت‌کنندگان به لینکی در Dropbox هدایت شدند که حاوی یک فایل ISO با حجم ۱۳۰ مگابایت بود.

با باز کردن سند، سه فایل جداگانه نمایان شد که یکی از آن‌ها فایل صوتی‌ای[1] به نام ۷۶۱ بود، سرنخی تازه برای کسانی که هنوز در این بازی ذهنی باقی مانده بودند.

نکته‌ی جالب‌توجه در این فایل صوتی، متادیتای آن بود. در بخش توضیحات، عنوان The Instar Emergence درج شده بود؛ عبارتی که از نظر نمادین به چرخه‌ی رشد حشره‌ی سیکادا اشاره دارد و به‌نوعی با ماهیت فلسفی پروژه هم‌راستا بود. نام هنرمند نیز به‌طور مرموزی «۳۳۰۱» ثبت شده بود، که بازتابی مستقیم از عدد کلیدی و هویت این معمای بزرگ بود.

ویژگی جالب دیگر، مدت‌زمان این فایل بود: ۱۶۷ ثانیه، عددی که معکوس نام فایل (۷۶۱) محسوب می‌شود. جالب‌تر اینکه هر دو عدد، اول هستند، جزئیاتی که به‌سختی می‌توان آن‌ها را تصادفی دانست.

این فایل صرفاً یک قطعه‌ی موسیقی نبود؛ لایه‌هایی از اطلاعات درون آن پنهان شده بود که تنها از طریق پردازش‌های فنی قابل کشف بودند. واضح بود که سیکادا حتی از جزئی‌ترین عناصر، مثل متادیتا، زمان، و نام فایل، برای انتقال مفاهیم عمیق‌تر استفاده کرده است.

نمایان شدن این پیام از طریق باز کردن فایل با برنامه‌ی hexdump یکی از رمزآلودترین و فلسفی‌ترین بخش‌های معماهای سیکادا ۳۳۰۱ است. پیام:

```
The Instar Emergence

Parable 1,595,277,641
Like the instar, tunneling to the surface
We must shed our own circumferences;
Find the divinity within and emerge.
```

ظهور پوست‌اندازی مثل ۱,۵۹۵,۲۷۷,۶۴۱ همانند پوست‌اندازی، تونل زدن به سطح باید پیله‌ی دور خود را بشکافیم؛ ربانیت درون را پیدا کنیم و بیرون آییم.

این متن نه‌تنها به‌صورت نمادین به مفهوم «رشد» و «تبدیل شدن به نسخه‌ای برتر از خود» اشاره دارد، بلکه با به‌کارگیری تصاویری مانند پوست‌اندازی و تونل‌زنی، فرایند تحول فردی را به شکلی استعاری و عمیق به تصویر می‌کشد.

عدد ۱,۵۹۵,۲۷۷,۶۴۱ که در پیام ظاهر شده بود، به‌احتمال زیاد فراتر از یک عدد ساده است. ممکن است این عدد نقش یک کلید رمزنگاری دیگر را داشته باشد یا ارجاعی به داده‌ای خاص باشد. این دقیقاً همان چیزی است که سیکادا را منحصربه‌فرد می‌کند: پیوند ظریف بین ریاضیات، رمزنگاری، فلسفه و تفکر تحلیلی.

در همین حین که کاربران سرگرم رمزگشایی فایل صوتی بودند، فردی موفق شد حساب کاربری‌ای در توییتر پیدا کند که نشانه‌هایی از سبک خاص سیکادا داشت. این

1 https://b2n.ir/b71965

حساب کمی بعد از اولین دانلود فایل ایجاد شده بود و آدرس آن (1231507051321@) با عددی که هنگام بوت شدن فایل ظاهر می‌شد، هم‌خوانی داشت.

توییت‌های این حساب در فواصل زمانی کوتاه، رشته‌هایی از کدهای هگزادسیمال را منتشر می‌کرد. سرانجام، یکی از کاربران IRC با استفاده از همین توییت‌ها و تحلیل دقیق آن‌ها، موفق شد پیام صوتی را رمزگشایی کند. خروجی این فرآیند، تصویری با فرمت JPG بود که جدول جماتریا را نمایش می‌داد- جدولی شامل سه ستون: الفبای رون، حرف، و ارزش عددی. بررسی بیشتر نشان داد که این الفبا، رون‌های آنگلوساکسونی هستند؛ سیستمی که بعدها در حل معماهای مرحله‌ی سوم سیکادا کاربرد پیدا کرد.

Gematria Primus

an order and a value as revealed through 3301

Rune	Letter	Value	Rune	Letter	Value
ᚠ	F	2	ᛋ	S/Z	53
ᚢ	U	3	ᛏ	T	59
ᚦ	TH	5	ᛒ	B	61
ᚩ	O	7	ᛖ	E	67
ᚱ	R	11	ᛗ	M	71
ᚳ	C/K	13	ᛚ	L	73
ᚷ	G	17	ᛝ	NG/ING	79
ᚹ	W	19	ᛟ	OE	83
ᚻ	H	23	ᛞ	D	89
ᚾ	N	29	ᚪ	A	97
ᛁ	I	31	ᚫ	AE	101
ᛃ	J	37	ᚣ	Y	103
ᛇ	EO	41	ᛡ	IA/IO	107
ᛈ	P	43	ᛠ	EA	109
ᛉ	X	47			

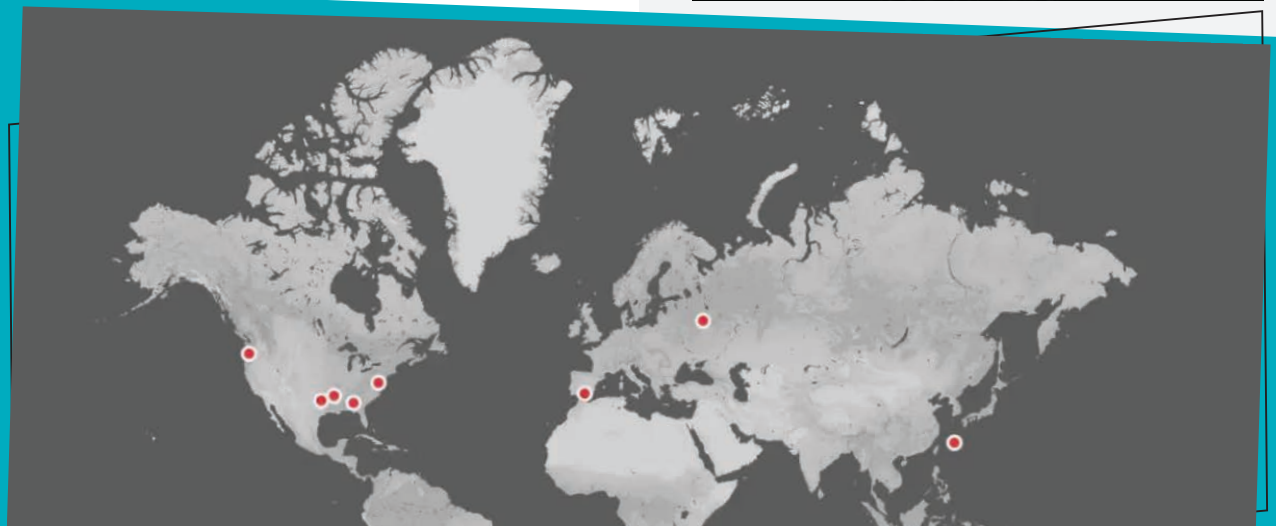
سطح بالای جزئیات و نمادپردازی در این مرحله، نشان می‌دهد که سیکادا با دقتی کم‌نظیر معماهای خود را طراحی کرده؛ نه‌فقط از نظر فنی، بلکه از نظر مفهومی، تاریخی و فرهنگی نیز چالش‌برانگیز و جذاب است.

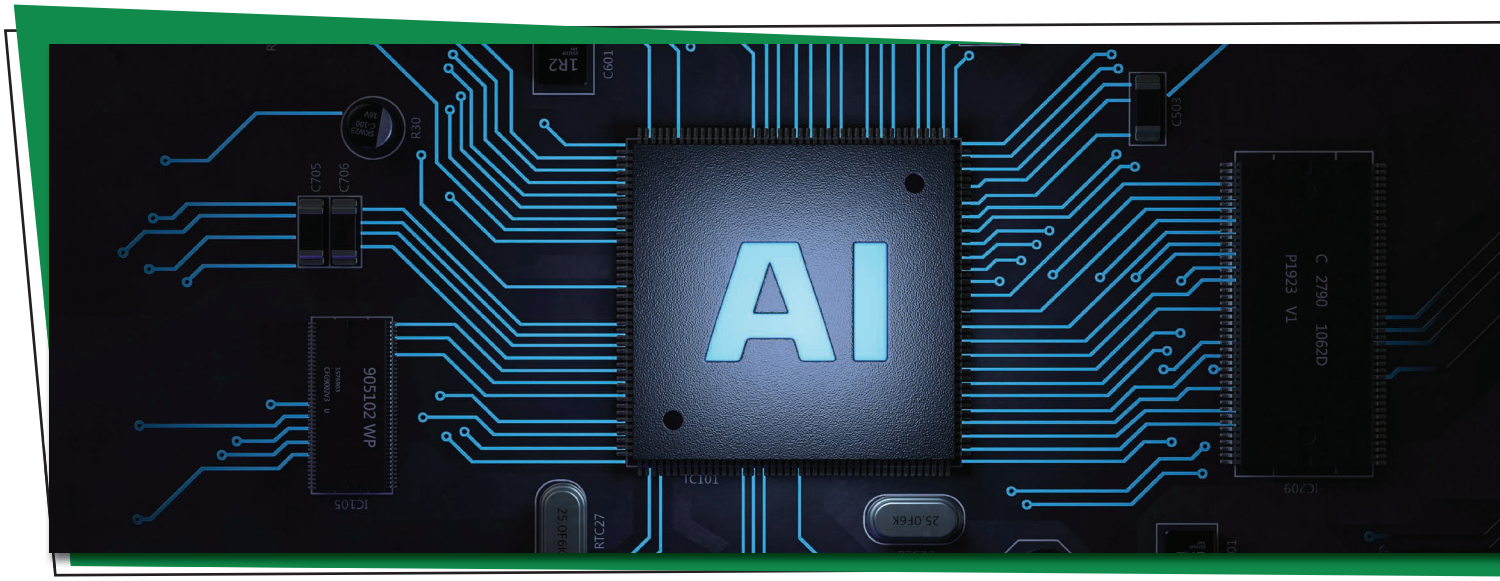
با باز کردن تصویر کشف‌شده در نرم‌افزار OutGuess و حل مجموعه‌ای دیگر از معماها، پیامی جدید به‌دست آمد که شامل آدرس وب‌سایتی در شبکه‌ی تور بود با طی مسیر این سایت و عبور از سه لینک دیگر، شرکت‌کنندگان به مختصات جغرافیایی جدیدی رسیدند؛ نقاطی در ژاپن، اسپانیا، روسیه و ایالات متحده

در این مکان‌ها، پوسترهایی فیزیکی یافت شد که هرکدام حاوی شماره تلفن‌هایی بودند که به اعداد مرموز ۳۳۰۱ یا ۱۰۳۳ ختم می‌شدند؛ عددهایی که بار دیگر تأکید می‌کردند این پروژه تا چه اندازه به نمادپردازی وفادار است.

با تماس گرفتن با این شماره‌ها و وارد کردن کدهای مخصوص، لینک‌های جدیدی نمایان شد که برای هر موقعیت جغرافیایی منحصربه‌فرد بود. این مرحله نه‌تنها مسیر حل معما را گسترش می‌داد، بلکه هیجان و حس تعلیق را نیز تقویت می‌کرد. ترکیب خلاقانه‌ای از دنیای واقعی و فضای دیجیتال که نشان‌دهنده‌ی ماهیت چندلایه، پیچیده و هوشمندانه‌ی پروژه‌ی سیکادا بود.

این داستان در نسخه بعدی ماتریس ادامه دارد...





TinyML به زودی در همه جا خواهد بود!

محمدمهدی بابابیک

TinyML به زودی در همه جا خواهد بود و نسل بعدی دستگاه های هوشمند تعبیه شده را تامین می کند. این دستگاه‌ها در خانه‌های ما و در مکان‌های بسیار دوردست خواهند بود و امکان نظارت از راه دور را برای صنعت و محیط زیست فراهم می‌کنند. امروزه در این تنظیمات مانیتورینگ از راه دور، ۹۹ درصد از داده‌های خام حسگر دور ریخته می‌شود، که انبوهی از داده‌ها برای یادگیری ماشینی است!

TinyML می تواند یک راه حل منحصر به فرد ارائه دهد: با خلاصه کردن و تجزیه و تحلیل داده ها در لبه دستگاه‌های تعبیه شده کم مصرف TinyML می‌تواند آمار خلاصه هوشمندی ارائه دهد که این الگوها، ناهنجاری ها و تجزیه و تحلیل های پیشرفته را که قبلاً گم شده است، در نظر می گیرد.

در این بخش، ما چند حوزه کاربردی نوظهور را بررسی می کنیم که پتانسیل بالایی برای TinyML دارند. این فهرست یک پیش‌نمایش کوچک از انبوه برنامه‌های موجود در افق است. بعداً در این دوره، مروری اساسی بر یادگیری ماشین انجام خواهیم داد، که برخی از شما کمتر از دیگران به آن نیاز دارید، اما باز هم مرور خوبی است. در دو دوره بعدی ما در اطراف TinyML بسیار عمیق تر پیش خواهیم رفت.

تعمیر و نگهداری پیشگویانه صنعتی:

در محیط صنعتی، TinyML در حال حاضر برای ارائه حسگر هوشمندتری استفاده می شود که نظارت پیشرفته را قادر می سازد و بهره وری و ایمنی را بهبود می‌بخشد. به عنوان مثال، نگهداری و نظارت بر توربین های بادی از راه دور می تواند بسیار چالش برانگیز و زمان بر باشد. با این حال، اگر بتوانیم به طور فعال پیش‌بینی کنیم که چه زمانی دستگاه دچار مشکل می‌شود، می‌توانیم پیش‌بینی تعمیر و نگهداری را پیش از هر گونه خرابی انجام دهیم. چنین «تعمیر و نگهداری پیش‌بینی‌شده»‌ای می‌تواند منجر به صرفه‌جویی قابل‌توجهی در هزینه‌ها، به دلیل کاهش زمان‌های خرابی و در دسترس بودن بهتر سیستم‌ها برای قابلیت اطمینان بالاتر در محصول شود، که منجر به کیفیت کلی بالاتر خدمات برای کاربران نهایی/ مشتریان می‌شود.

برنامه های بسیاری از TinyML برای نگهداری پیش بینی وجود دارد. به عنوان مثال، یک استارت آپ

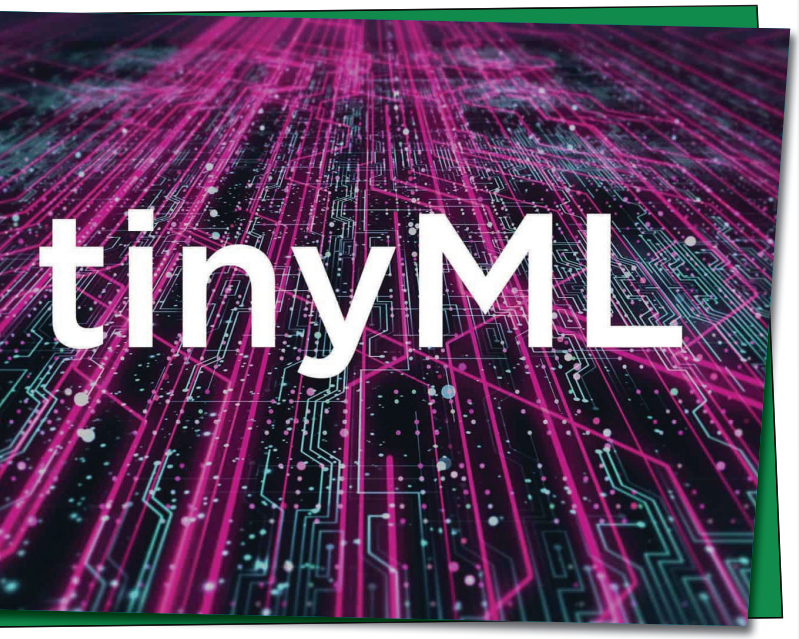
استرالیایی Ping Services، یک دستگاه جدید اینترنت اشیا را معرفی کرده است که به طور مداوم و مستقل یک توربین را در حین کار بازرسی می کند. با اتصال مغناطیسی به بیرون هر توربین و تجزیه و تحلیل داده‌های دقیق در لبه و داده‌های خلاصه در ابر، دستگاه می‌تواند به طور موثر و مؤثری از هرگونه مشکل احتمالی قبل از بروز مشکل در داخل توربین هشدار دهد.

کاربرد در کشاورزی:

هر روز، شرکت کاساوا برای بیش از ۵۰۰ میلیون آفریقایی غذا فراهم می کند. با این حال، این اصطبل حیاتی به طور مداوم تحت حمله انواع بیماری ها قرار دارد. تیم PlantVillage به سرپرستی دکتر آماندا رامچاران، اپلیکیشن Nuru را برای کمک به کشاورزان در شناسایی و درمان این بیماری ها توسعه داده اند. با اجرای یادگیری ماشینی با استفاده از TensorFlow Lite در تلفن‌های همراه، این برنامه کاهش بی‌درنگ را بدون نیاز به دسترسی به اینترنت امکان‌پذیر می‌کند - یک نیاز حیاتی برای بسیاری از کشاورزان از راه دور. نسل بعدی این سیستم فراتر خواهد رفت - با بهره گیری از tinyML و فناوری هایی مانند TensorFlow برای میکروکنترلرها، برای استقرار حسگرها در مزارع راه دور، برای امکان ردیابی و تجزیه و تحلیل بهتر-

کاربرد در بهداشت و درمان:

پروژه Solar Scare Mosquito پلتفرم‌های رباتیک هوشمند کوچک اینترنت اشیاء (IoT) را برای کمک به جلوگیری از گسترش اپیدمی‌های منتقله از طریق پشه مانند مالاریا، دنگی و ویروس زیکا به کار می‌گیرد. این سیستم با ایجاد اختلال در چرخه تولید مثل پشه با هم زدن آبی که احتمالاً حاوی لارو پشه است، کار می کند. این سیستم از سنسورهای باران و آکوستیک استفاده می‌کند تا تعیین کند که چه زمانی باید آب را به هم بزند تا باتری را حفظ کند و بتواند به طور نامحدود با انرژی خورشیدی کار کند. همچنین آمار خلاصه و هشدارهای هوشمندی را ارسال می کند تا در مورد رویدادهای احتمالی تکثیر انبوه پشه ها از طریق پروتکل های ارتباطی با سرعت پایین تر هشدار دهد. با ساختن یک سیستم خودکفا، کوچک و مقرون به صرفه، می‌توان این دستگاه ها را به طور گسترده ای مستقر کرد و از گسترش پشه ها جلوگیری کرد. تمام اجزای لازم در یک جزء کوچک‌تر از اندازه یک توپ فوتبال گنجانده شده است.



حفاظت از حیات وحش:

TinyML همچنین در حال حاضر برای نظارت بر محیط زیست استفاده می شود. به عنوان مثال، در طول ۱۰ سال گذشته، خط راه آهن Siliguri-Jalapaiguri در هند بیش از ۲۰۰ تصادف مرگبار با فیل داشته است. محققان آزمایشگاه بیوآکوستیک کاربردی در دانشگاه پلی‌تکنیک کاتالونیا یک سیستم حسگر صوتی و حرارتی هوشمند با استفاده از مدل‌های یادگیری ماشین سفارشی که با انرژی خورشیدی کار می‌کنند به عنوان یک سیستم هشدار اولیه طراحی کردند (بسته‌ی همه‌جانبه با منبع انرژی خودپایدار، نزدیکی به راه‌آهن را بدون زیرساخت‌های اضافی، به عنوان مثال خطوط برق، امکان‌پذیر می‌سازد).

سفر در دنیای تاریکی به‌نام TOR

محدثه جوان

در اعماق تاریک اینترنت، جایی که چشم، چشم را نمی‌بیند، فراتر از چشمان کنجکاو، شبکه‌ای وجود دارد که ردپای شما را پاک می‌کند. نامش Tor است؛ محافظی خاموش و ساخته‌شده برای کسانی که سایه را به نور ترجیح می‌دهند.

کسی هرگز نخواهد فهمید پیام‌ها از کجا آمده و به کجا می‌روند، چون مبدا ناشناس است، رد پاها پوشیده شده و تنها سایه‌ای از یک مسیر طی شده باقیست.

البته برخلاف تصور رایجی که Tor را تنها به فضای دارک‌وب نسبت می‌دهد، بسیاری از کاربران این شبکه شامل روزنامه‌نگاران، کنش‌گران اجتماعی، محققان، و حتی افراد عادی هستند که دغدغه‌ی حفظ حریم خصوصی را دارند.

وقتی در شبکه‌ی TOR پیامی ارسال می شود، دیگر به‌سادگی یک بسته‌ی داده نیست. این‌بار، پیام در چندین لایه‌ی رمزنگاری پیچیده می‌شود؛ لایه‌هایی شبیه پوست پیاز. هر لایه، مخصوص یکی از نگهبانان مسیر است؛ گره‌هایی که پیام را از این سوی جهان به آن سویش می‌برند، بی‌آن‌که بدانند پیام از کجا آمده یا به کجا خواهد رفت.

لشکرهای تک نفره:

ساختاری که از آن صحبت می‌کنیم، به TOR یا همان The Onion Router شناخته شده است.

این شبکه برای پنهان کردن هویت واقعی شما طراحی شده. نه با یک فریب ساده، بلکه با رمزگذاری لایه‌لایه،

کاربرد در دریا:

سامانه‌های مشابهی نیز در آبراه‌های اطراف سیاتل و ونکوور برای جلوگیری از حملات نهنگ‌ها در خطوط شلوغ کشتی‌رانی مستقر شده‌اند. این سنسورهای هوشمند مجهز به tinyML، نظارت زمان واقعی ثابت و افزایش تراکم استقرار حسگرها را امکان پذیر می‌کنند و کارایی کلی سیستم را بهبود می‌بخشند.

همانند پوست یک پیاز.

وقتی پیامی در شبکه Tor ارسال می‌کنید، آن پیام سه بار رمزگذاری می‌شود، هر بار با کلید عمومیِ یکی از سه گره‌ی تصادفی در شبکه.

سپس از سه ایستگاه ناشناس عبور می‌کند:

- گره ورود (Entry Node)
- گره میانی (Relay Node)
- گره خروج (Exit Node)

و سپس هر گره فقط لایه‌ی خود را رمزگشایی می‌کند. هیچ‌کس مسیر کامل را نمی‌بیند؛ حتی گره خروجی هم نمی‌داند پیام از کجا آمده است.

مثل لشکری که به صف ایستاده اند و هر کس، رمزی که سرباز قبلی در گوشش می‌خواند را به نفر بعدی منتقل می‌کند.

هیچکس نخواهد فهمید این رمز از کجا آمده و به دست چه کسی می‌رسد، تنها چیزی که واضح است، اهمیت حفظ پیام و انتقال آن به نفر بعدی است.

فرستنده، ابتدا مسیر را مشخص می‌کند: چند گره، چند نگهبان، چند محافظ رمز.

برای هر گره، یک کلید رمزنگاری تولید می شود؛ و سپس، لایه‌به‌لایه، پیام در این سکوتِ رمزها پنهان می‌شود.

در مرحله ی اول، رمزنگاری مقصد نهایی در دستور کار قرار می‌گیرد.

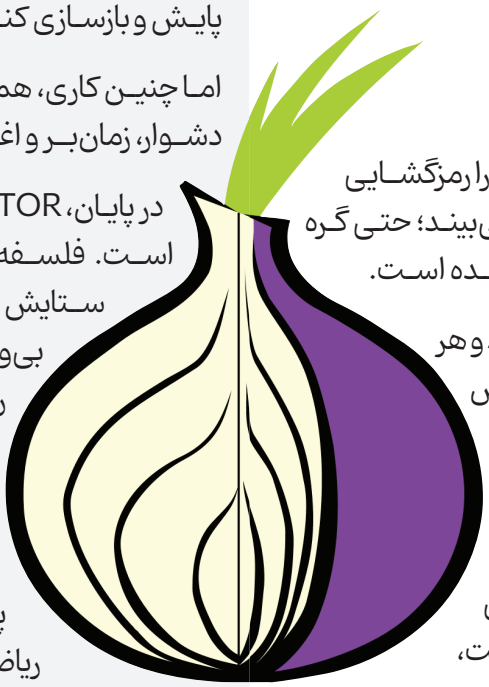
سپس، لایه ی بعدی صاحب رمز میشود، و این اتفاق آن قدر در چرخه تکرار می‌ماند که به رمزنگاری برای بیرونی ترین لایه ی آن می‌رسیم.

اکنون این پیام، چیزی است که با عبور از دروازه های پیاپی موجود در مسیرش، و رمز گشایی در هر مرحله، چهره اش را برای مقصد آشکار می‌کند.

اما آیا Tor شکست‌ناپذیر است؟

در دل این شبکه رمزآلود، همیشه خطر وجود دارد. اگر مهاجمی هر دو گره ورودی و خروجی را کنترل کند، ممکن است با تحلیل زمان‌بندی و حجم داده، ارتباط را پایش و بازسازی کند.

اما چنین کاری، همانند شکستن یک کد کهن است؛ دشوار، زمان‌بر و اغلب بی‌نتیجه.



در پایان، TOR یک تکنولوژی نیست، یک فلسفه است. فلسفه‌ای در ستایش ناشناخته‌بودن، در ستایش اختیار در جهانی که نظاره‌گرها بی‌وقفه در حال پایش اند. اگر این راه ادامه یابد، خواهی فهمید که تاریکی همیشه ترسناک نیست؛ گاه تنها راه رهایی‌ست.

شاید هم دروازه‌ای است به جهانی پنهان، جایی که اطلاعات با جادوی ریاضی و رمزنگاری محافظت می‌شود.

جهانی که در آن، ناشناس بودن قدرت است.

اما همان‌طور که نور نمی‌تواند بدون سایه باشد، امنیت نیز هرگز مطلق نیست...

گروه ScarCruft

فربد فیضی

گروه جاسوسی سایبری ScarCruft، که همچنین با نام‌های APT27 و Reaper شناخته می‌شود، یکی از گروه‌های تهدید پیشرفته و پایدار (APT) است که در سال‌های اخیر حملات سایبری بزرگی را علیه سازمان‌ها و نهادهای مختلف در سراسر جهان انجام داده است. این گروه که به کره شمالی نسبت داده می‌شود،

به‌طور مستمر به انجام حملات سایبری علیه نهادها و افراد مختلف در سرتاسر جهان پرداخته است و با استفاده از ابزارهای پیچیده و تکنیک‌های پیشرفته، توانسته است به اطلاعات حساس در بخش‌های مختلف از جمله دولت‌ها، صنایع و سیستم‌های اقتصادی دسترسی پیدا کند.

حمله به سازمان‌های دولتی و اقتصادی در کره جنوبی

یکی از مهم‌ترین حملات گروه ScarCruft در سال‌های ۲۰۱۷ و ۲۰۱۸ به سازمان‌های دولتی و اقتصادی در کره جنوبی صورت گرفت. این حملات عمدتاً به‌منظور جمع‌آوری اطلاعات محرمانه و استراتژیک انجام شد. ScarCruft از بدافزارهای پیچیده‌ای استفاده کرد که خود را به‌عنوان برنامه‌های کاربردی قانونی معرفی می‌کردند تا سیستم‌های هدف را به‌طور پنهانی آلوده کنند.

جزئیات حمله:

- هدف‌گذاری‌ها: سازمان‌های دولتی، اقتصادی و نظامی کره جنوبی هدف اصلی این حملات بودند. اطلاعاتی از قبیل داده‌های نظامی، تجاری و سیاست‌های داخلی به سرقت رفت.
- استفاده از بدافزار: گروه ScarCruft از KoSpy (که به تازگی متوجه استفاده این گروه از این ابزار شده اند و این ابزار را شناسایی کرده‌اند) و سایر ابزارهای مشابه برای نفوذ به سیستم‌های آلوده استفاده کرد. این بدافزارها از تکنیک‌های مهندسی اجتماعی و فریبندگی برای جلب توجه کاربران به خود بهره بردند.
- روش‌های حمله: مهاجمان از حملات فیشینگ و P2P (همتا به همتا) برای دسترسی به شبکه‌های حساس استفاده کردند. در این حملات، ابزارهای جاسوسی به‌طور مخفیانه در سیستم‌ها مستقر می‌شدند و اطلاعات حساس به‌طور مداوم استخراج می‌شد.

این حملات منجر به افشای اطلاعات حساس و استراتژیک شد که می‌توانستند تهدیدات امنیتی جدی برای کشور کره جنوبی ایجاد کنند. دسترسی به این داده‌ها، به ویژه در زمینه‌های نظامی و اقتصادی، می‌توانست موجب آسیب به امنیت ملی و منافع اقتصادی کره جنوبی شود.

حمله به صنعت ارزهای دیجیتال

یکی دیگر از حملات برجسته گروه ScarCruft به صنعت ارزهای دیجیتال بود. در این حملات، این گروه از بدافزارهای جاسوسی برای نفوذ به صرافی‌های ارز دیجیتال و شرکت‌های مرتبط با این بخش استفاده کرد.

گروه جاسوسی سایبری ScarCruft، به تازگی به استفاده از ابزار جاسوسی جدیدی به نام KoSpy مرتبط شده است. KoSpy یک بدافزار پیچیده است که کاربران زبان‌های انگلیسی و کره‌ای را هدف قرار می‌دهد و خود را به‌عنوان برنامه‌های کاربردی قانونی نظیر «File Manager» یا «Smart Manager» معرفی می‌کند.

عملکرد KoSpy پس از نصب

پس از نصب شدن این بدافزار بر روی دستگاه موبایل، KoSpy به جمع‌آوری اطلاعات حساس کاربران می‌پردازد. این اطلاعات شامل داده‌هایی از قبیل پیامک‌ها، گزارش تماس‌ها، موقعیت جغرافیایی و حتی ضبط صدا می‌شود. به‌واسطه این ویژگی‌ها، مهاجمان قادر خواهند بود به‌طور دقیق به جزئیات زندگی خصوصی کاربران دسترسی پیدا کنند.

سیستم فرمان و کنترل (C2) دو مرحله‌ای

این بدافزار از یک سیستم فرمان و کنترل (C2) دو مرحله‌ای استفاده می‌کند که به مهاجمان امکان می‌دهد بدون شناسایی و به‌طور مخفیانه دستورات جدیدی به دستگاه‌های آلوده ارسال کنند. تنظیمات مربوط به بدافزار از Firebase Firestore دریافت می‌شود. این روش به مهاجمان این امکان را می‌دهد که آدرس سرور فرمان و کنترل خود را تغییر دهند، بدون

اینکه شناسایی شوند یا فعالیت‌های آن‌ها توسط ابزارهای امنیتی ردیابی شود.

سیستم فرمان و کنترل (Command and Control) یا C2 به مجموعه‌ای از ابزارها و سرورها اطلاق می‌شود که مهاجمان از آن‌ها برای ارسال دستور به بدافزارها و دریافت اطلاعات از دستگاه‌های آلوده استفاده می‌کنند. سیستم‌های C2 به مهاجمان این امکان را می‌دهند که فعالیت‌های خود را بدون شناسایی یا ردیابی انجام دهند.

Firebase Firestore یک سرویس ذخیره‌سازی ابری از Google Firebase است که برای ذخیره‌سازی داده‌ها در برنامه‌های موبایل استفاده می‌شود. این سرویس به مهاجمان کمک می‌کند تا اطلاعات و تنظیمات لازم برای کنترل بدافزار را به‌طور مخفیانه ارسال کنند.

تهدیدات سایبری کره شمالی

حملات مرتبط با KoSpy تنها یک نمونه از کمپین‌های گسترده‌تر سایبری کره شمالی است که به‌طور خاص بر هدف‌گذاری دستگاه‌های موبایل و بخش‌های مرتبط با ارزهای دیجیتال تمرکز دارند. استفاده از این نوع ابزارهای جاسوسی به‌طور مستقیم به منافع اقتصادی و اطلاعاتی این کشور مربوط می‌شود.

اقدامات مثبت برای مقابله با تهدید

در پاسخ به تهدیدات ایجادشده توسط KoSpy و بدافزارهای مشابه، Google Play Store برخی از این برنامه‌ها را حذف کرده است و همچنین Google Play Protect به‌عنوان یک لایه امنیتی اضافی برای شناسایی و حذف نسخه‌های شناخته‌شده بدافزارها از دستگاه‌های کاربران فعال است. این اقدامات به‌طور قابل توجهی توانسته‌اند خطرات موجود را کاهش دهند، اما همچنان نیاز به اقدامات اضافی برای محافظت از کاربران باقی‌مانده است.

همکاری در نشریه ی ماتریس

نشریه ی ماتریس در جهت ارتقای کیفیت نشریه و مشارکت همه دانشجویان در سه تیم تحریریه، ویراستاری و طراحی گرافیک عضو همکار می پذیرد.

جهت ارتباط با روابط عمومی نشریه و همکاری در تهیه نشریه با ما در ارتباط باشید.

@MatrisMagazine