

D23 – Metodológia pre tvorbu časovej osi prípadu



Projekt Automatizácia digitálnej forenznej analýzy a odpovede na incident (ADFIR) financovaný Európskou úniou – Next GenerationEU prostredníctvom Plánu obnovy a odolnosti Slovenskej republiky pod číslom projektu č. 09I05-03-V02-00079.

Obsah

Popis projektu	2
1 Úvod	3
2 Proces spracovania dát pre časovú os	4
3 Nástroj na tvorbu a zobrazovanie časovej osi – ETL Timeline	6
3.1 Závislosti a prerekvizity	6
3.2 Vstupné dáta	6
3.3 Spustenie nástroja	7
3.4 Spracovanie dát na pozadí nástroja	7
3.4.1 Nahrať dáta cez UploadModal	8
3.4.2 Parsovanie a validácia	8
3.4.3 Prepojenie udalostí s anotáciami	8
3.4.4 Transformácia na TimelineCase	9
3.4.5 Kompresia časovej osi	9
3.4.6 Filtrovanie	9
3.4.7 Budovanie vizuálnych vrstiev (len pre viditeľné udalosti)	9
3.4.8 Histogram hustoty (density)	10
3.4.9 Drilldown	10
3.4.10 Panely detailov	11
3.5 Vloženie dát	11
3.6 Grafické rozhranie a jeho funkcionality	12
3.6.1 Zobrazenie grafu časovej osi	12
3.6.2 Obogatenie ETL Timeline o výstupy ďalších analýz	13
3.6.3 Práca s textovými dátami	15
3.6.4 Filtrovanie udalostí	18
4 Forenzný význam	20
5 Záver	21
6 Bibliografia	22

Popis projektu

Projekt Automatizácia digitálnej forenznej analýzy a odpovede na incident (ďalej len „ADFIR“) je financovaný Európskou úniou – Next GenerationEU prostredníctvom Plánu obnovy a odolnosti Slovenskej republiky pod číslom projektu č. 09I05-03-V02-00079. Tento projekt sa zaoberá jednou z kľúčových výziev v oblasti kybernetickej bezpečnosti a informačnej bezpečnosti – ako spracovať obrovské množstvo digitálnych dôkazov, ktoré vznikajú počas incidentov kybernetickej bezpečnosti alebo forezných vyšetrovaní. V súčasnosti je tento proces veľmi náročný z hľadiska ľudských zdrojov a času. Automatizácia pomocou metód strojového učenia môže preto výrazne zlepšiť kvalitu digitálnej forenznej analýzy a skrátiť čas potrebný na jej vykonanie. Celkovo to umožňuje bezpečnostným tímom efektívnejšie reagovať na kybernetické hrozby. Hlavné prínosy tohto projektu sú:

- 3 **Rýchlejšie riešenie incidentov v oblasti kybernetickej bezpečnosti.** Projekt ADFIR zavádza automatizované prístupy k zberu, spracovaniu a analýze digitálnych stôp. Vďaka tomu môžu bezpečnostné tímy rýchlejšie identifikovať príčiny incidentov a prijať účinné opatrenia na ich riešenie.
- 4 **Zníženie pracovnej záťaže forezných analytikov.** Rutinné a časovo náročné úlohy spojené so spracovaním digitálnych stôp budú nahradené automatizovanými metódami. To umožní analytikom sústrediť sa na zložitejšie prípady a strategické rozhodovanie.
- 5 **Vyššia kvalita a konzistentnosť výstupov.** Použitie jednotných metodík a nástrojov zaručuje, že spracované digitálne stopy budú presnejšie, konzistentnejšie a ľahšie overiteľné. To výrazne znižuje riziko chýb spôsobených ľudskými faktormi.
- 6 **Možné využitie v trestnom konaní.** Výstupy projektu budú vyvinuté v súlade s právnymi požiadavkami a normami, čo umožní, aby digitálne stopy boli akceptované ako relevantné dôkazy pre vyšetrovanie a súdne konania.

1 Úvod

Časová os predstavuje jeden zo základných nástrojov digitálnej forenznej analýzy pre vizualizáciu a interpretáciu udalostí zachytených v operačnom systéme. Windows generuje rozsiahle množstvo logov naprieč viacerými zdrojmi — Event Log, prefetch súbory, registre, plánované úlohy a ďalšie artefakty — pričom každý z týchto zdrojov prispieva vlastnou vrstvou informácií o aktivite systému a jeho používateľov. Samostatne sú tieto záznamy ťažko interpretovateľné; až ich zlúčenie do spoločnej časovej osi odhaľuje súvislosti a vzory v správaní systému.

Pri práci s takýmto objemom dát je lineárne prechádzanie záznamov prakticky neuskutočniteľné. Vizualná reprezentácia v čase umožňuje analytikovi okamžite identifikovať zhľady udalostí, nezvyčajné výpadky aktivity alebo náhle zmeny v intenzite zápisov — javy, ktoré by v surových dátach zostali nepovšimnuté.

Nástroj ETL Timeline na zobrazovanie a prácu s časovou osou dokáže navyše zvýrazniť rôzne anomálie identifikované inými analytickými nástrojmi, ktorá vznikli v rámci tohto projektu. Anomálie identifikované týmito nástrojmi sú priamo vizualizované v kontexte časovej osi, čím analytik získava integrovaný pohľad na prípad bez potreby prepínania medzi viacerými prostrediami. Takáto anomália môže indikovať neautorizované prihlásenie, spustenie neočakávaného procesu, zmenu kritických systémových súborov alebo inú udalosť vyžadujúcu bližšie preskúmanie.

Táto metodológia popisuje spôsob zostavenia, interpretácie a dokumentácie takýchto časových osí v rámci platformy, vrátane definície relevantných zdrojov dát, kritérií pre identifikáciu anomálií a požiadaviek na zaznamenanie analytických záverov.

Forenzne vyšetrovania často vyžadujú rekonštrukciu časovej osi udalostí a činností súvisiacich s digitálnym zariadením alebo používateľmi. Takéto časové osi môžu poskytnúť cenné informácie o rôznych trestných činnostiach, vrátane škodlivého softvéru, útokov hrubou silou alebo činností útočníka po získaní kontroly nad systémom. Proces analýzy časovej osi je zložitý a časovo náročný, najmä pri práci s veľkým množstvom digitálnych údajov z viacerých zdrojov. Tradičné metódy analýzy časovej osi sú založené na manuálnej analýze, ktorá môže byť subjektívna a náchylná na chyby a môže viesť k prehliadnutiu kľúčových informácií [1].

Rekonštrukcia udalostí spočíva v obnovení minulých udalostí prostredníctvom analýzy digitálnych artefaktov, čo umožňuje vyšetrovateľom určiť činnosť systému a vyvodiť podložené závery o tom, čo sa stalo. Kým tradičná forezná veda čerpá z jasne definovaného rámca, ktorý tento odbor sumarizuje, rekonštrukcia udalostí v digitálnej forenznej analýze sa často rozoberá po častiach, pričom sa pozornosť zameriava na úlohy, ako je vytváranie superčasovej osi, odhaľovanie manipulácií alebo špecifiká prostredia. V dôsledku toho sa výskum zameriaval na tieto úzke aspekty, pričom širšie výzvy zostali nedostatočne preskúmané alebo boli prehliadané. Chýbajúci jednotný pohľad viedol aj k vzniku množstva nejednotných pojmov, čo sťažuje komplexnú diskusiu o rekonštrukcii udalostí alebo vyhľadávanie relevantných výskumov; napríklad niektoré štúdie používajú pojem artefakt, iné

sa odvolávajú na pozorovateľné aspekty. Pojmy ako udalosti, akcie používateľov, interakcie alebo kliknutia sa v odbornej literatúre používajú nejednotne [2].

2 Proces spracovania dát pre časovú os

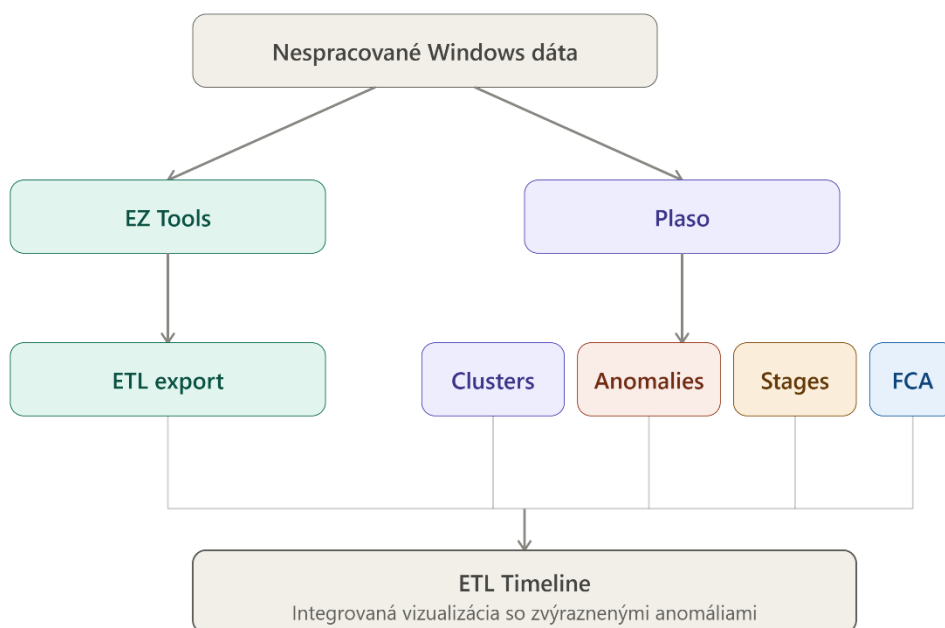
Zaistené dáta z operačného systému Windows (bitové kópie disku) prechádzajú dvoma paralelnými analytickými tokmi, ktorých výstupy sú následne zlúčené v nástroji ETL Timeline. Celá schéma spracovania je na **Chyba! Nenašiel sa žiaden zdroj odkazov.**

Prvý tok využíva Eric Zimmerman (EZ) tools [3]— sadu špecializovaných nástrojov pre parsovanie artefaktov operačného systému Windows. Výstupom sú viaceré CSV tabuľky s odlišnými štruktúrami. Ich zjednotenie prebieha prostredníctvom nástroja ETL do štandardizovaného formátu pripraveného pre vizualizáciu.

ETL podľa prednastavených situácií vytvorí tzv. metazáznamy – štandardizované záznamy o udalostiach ako vytvorenie procesu, vytvorenie plánovanej úlohy, stiahnutie súboru a pod. Tieto metazáznamy následne agreguje. Podrobnejšie fungovanie nástroja ETL od predspracovania až po vizualizáciu je popísané vo výstupe ADFIR-D17-KPB3-03-Model for aggregation and connection of digital evidence.pdf v kapitole 3 - Agregácia forenzných artefaktov.

Druhý tok pracuje s nástrojom Plaso, ktorý agreguje udalosti z rôznych zdrojov do jednotného super-timeline. Nad týmito dátami sú následne identifikované štyri typy analytických výstupov — Clusters, Anomalies, Stages a FCA, ktoré sú podrobnejšie popísané vo výstupoch KPB4 (Clusters a Anomalies vo výstupe D19-KPB4-01 Metóda na identifikáciu digitálnych stôp, Stages vo výstupe D21-KPB4-03 Metóda na určenie časového rámca a fáz útoku, FCA – vo výstupe D20-KPB4-02.01 Metóda na identifikáciu vzťahov).

Všetky exporty z ETL a výstupy analýzy nad Plasom sú nakoniec zobrazené v ETL Timeline, ktorý poskytuje integrovaný pohľad na prípad s vizuálnym zvýraznením identifikovaných anomálií a udalostí.



Obrázok 1 - Schéma spracovania dát

3 Nástroj na tvorbu a zobrazovanie časovej osi – ETL Timeline

Nástroj ETL Timeline slúži na zobrazovanie spracovaných dát z operačného systému Windows v podobe časovej osi agregovaných záznamov spolu s doplnkovými informáciami o anomáliách, klastroch podobných dát a pod. Analytikovi poskytuje prehľadný spôsob zobrazenia veľkého množstva dát tak, aby nebol množstvom dát zahltený. Zároveň mu doplnkové informácie zvýrazňujú potenciálne dôležité alebo zaujímavé časové úseky.

3.1 Závislosti a prerekvizity

Frontendová časť aplikácie je postavená na technológii **React [4]**, ktorá slúži na vývoj moderného používateľského rozhrania formou komponentov.

Na správu balíkov a závislostí sa používa **npm** (Node Package Manager) [5]. Prostredníctvom npm sú inštalované knižnice a nástroje potrebné na vývoj, zostavenie a spustenie frontendovej aplikácie.

Pred použitím ETL Timeline je potrebné najprv spustiť nástroj ETL, vďaka ktorému sa získajú požadované vstupné dáta. Podrobný postup spracovania dát do vhodnej podoby je popísaný vo výstupe D17-KPB3-03 Model na agregáciu a prepojenie digitálnych stôp.

3.2 Vstupné dáta

Na vstupe je potrebné použiť exportované dáta z nástroja ETL. Ide o exportovaný súbor vo formáte CSV, ktorý predstavuje základnú časovú os. Obsahuje jednu časovú pečiatku, popis vzniknutej udalosti a informácie k viacúrovňovej agregácii.

Ako ďalšie vstupy, ktoré umožňujú zobrazovať dodatočné informácie, predovšetkým podobnosť dát, je možné do ETL Timeline dodať nasledovné dáta vo formáte JSON, ktoré sú výstupmi z analytických nástrojov:

- *Clusters* – ide o výstup z metódy hierarchického klastrovania. Metóda vytvára skupiny (klastre) časových okien, ktoré sú si podobné. Táto podobnosť môže predstavovať prirodzené podobné aktivity ale aj podozrivé. Viac informácií je možné získať v D19-KPB4-01 Metóda na identifikáciu digitálnych stôp.
- *Attack Stages* – výstupy z časových radov. Viac informácií je možné získať v D21-KPB4-03 Metóda na určenie časového rámca a fáz útoku.
- *Anomalies* – anomálie sú časové okná, ktoré sú vyhodnotené ako najviac odlišné od ostatných časových okien. Viac informácií je možné získať v D19-KPB4-01 Metóda na identifikáciu digitálnych stôp.
- *FCA* – formálna konceptová analýza (FCA) je metóda na identifikáciu skupín podobnosti na základe spoločných atribútov na čo najväčšej množine riadkov. Viac informácií je možné získať v D20-KPB4-02.01 Metóda na identifikáciu vzťahov.

Štandardizovaný tvar voliteľných dát:

```
{
  "<object_name>": {
    "id":<id>
    "metadata": {
      "<metadata1>":<attribute1>,
      ...
    }
    "ts_objects": [
      {
        "ts_start": <time_start>,
        "ts_end": <time_end>
      },
      ...
    ]
  }
}
```

3.3 Spustenie nástroja

Nástroj ETL Timeline sa spúšťa v dvoch nasledujúcich krokoch:

1. Spustenie aplikácie:

```
npm i
```

```
npm run dev # alternatívne: npm run build && npm run preview
```

2. Otvorenie aplikácie v prehliadači:

V konzole sa zobrazí príslušný odkaz v tvare `http://localhost:<port>`, napríklad `http://localhost:5173`.

3.4 Spracovanie dát na pozadí nástroja

ETL Timeline využíva TypeScript. Spracovanie dát prebieha v nasledujúcich krokoch:

- 1) nahratie dát,
- 2) parsovanie a validácia dát,
- 3) prepojenie udalostí s anotáciami,
- 4) transformácia na TimelineCase,
- 5) kompresia časovej osi,
- 6) filtrovanie,
- 7) budovanie vizuálnych vrstiev,
- 8) tvorba histogramu hustoty,
- 9) drilldown,
- 10) budovanie panelov detailov.

3.4.1 Nahratie dát cez UploadModal

Používateľ má na nahratie dát k dispozícii 5 typov dát uvedených v **Tabuľka 1**. Pre každý typ sa používa príslušný parser na ich spracovanie. Výsledkom je spracovaný objekt.

Dáta	Formát	Povinné	Parser → spracovaný objekt
Timeline (časová os)	CSV	áno	utils/csv.ts → utils/events.ts
Klastre	JSON	nie	utils/temporalAnnotations.ts → utils/clusters.ts
Attack Stages	JSON	nie	utils/temporalAnnotations.ts → utils/stages.ts
Anomálie	JSON	nie	utils/temporalAnnotations.ts → utils/anomalies.ts
FCA faktory	JSON	nie	utils/temporalAnnotations.ts → utils/fca.ts

Tabuľka 1 - Typy dát vhodné pre nástroj ETL Timeline

3.4.2 Parsovanie a validácia

Podľa formátu vstupných dát (CSV alebo JSON) prebieha ich spracovanie (parsovanie) a validácia.

- CSV parser* (utils/csv.ts:3) – manuálny parser podporujúci polia v úvodzovkách, hlavičky mapuje na kľúče v malých písmenách.
- JSON parser* (utils/temporalAnnotations.ts:12-38) – prijíma JSON pole objektov alebo objekt s `ts_objects` poľom. Každý objekt obsahuje časové okná (`ts_start/ts_end`), `metadata`, a ID. Za účelom efektívnejšieho zobrazovania dát v grafe časovej osi sú blízke okná spájané (`clusterWindowMergeGapMs`).

3.4.3 Prepojenie udalostí s anotáciami

V prípade zadania klastrov ako jedného zo vstupov sa už počas prípravy časovej osi priradzujú jednotlivé klastre k udalostiam. Na to slúži nasledujúca funkcia:

`applyClusterAssignments` (utils/clusters.ts:14) – ku každej udalosti priradí klaster podľa toho, či časová pečiatka udalosti padne do niektorého okna daného klastra. Toto je jediné krížové prepojenie medzi datasetmi – ostatné anotácie (anomálie, FCA, stages) sa vyhodnocujú až pri renderovaní, nie pri importe.

3.4.4 Transformácia na TimelineCase

Ďalším krokom spracovania je transformácia načítaných udalostí do internej dátovej štruktúry objektu `TimelineCase`, ktorá predstavuje základnú jednotku používanú pri ďalšom spracovaní a vizualizácii časovej osi.

`buildCasesFromRows` (`utils/events.ts:35`) – z CSV riadkov vytvorí pole `SourceEvent` objektov, zoradí ich chronologicky a zabalí ich do objektu `TimelineCase`. V ňom sa teda nachádzajú všetky zdrojové udalosti.

3.4.5 Kompresia časovej osi

Proces kompresie časovej osi pozostáva z rozdelenia časového rozsahu na segmenty a vytvorenia škálovacej funkcie, ktorá prevádza časové pečiatky na ich zodpovedajúcu pozíciu na osi X. Zabezpečujú to nasledujúce dve funkcie:

`buildCompressedTimelineSegments` (`utils/compressedTimeline.ts:4`) – z `caseTimeRange` vytvorí segmenty.

`buildCompressedTimelineScale` (`utils/compressedTimeline.ts:89`) – vytvorí škálovaciu funkciu, ktorá mapuje časovú pečiatku na konkrétny pixel na osi X. Podporuje zoom cez `zoomRange` (offset range).

3.4.6 Filtrovanie

Po načítaní udalostí sa na ne aplikujú používateľom zvolené filtre, ktoré určujú, ktoré udalosti budú zahrnuté do ďalšieho spracovania a zobrazenia. Slúži na to funkcia

`filteredEvents`.

`filteredEvents` (`App.tsx:157-168`) – aplikuje 5 vrstiev filtrov (`source`, `eventType`, `situation`, `group`, `signal`). Každý filter je `Record<string, boolean>`.

3.4.7 Budovanie vizuálnych vrstiev (len pre viditeľné udalosti)

Každý typ zobrazenia prekrývajúceho časovú os (klastre, anomálie, FCA) sa buduje v dvoch úrovniach:

Úroveň 1 – buduje sa pre celý case (z `visibleCaseEvents`, čo sú udalosti v `timelineSegments`):

- `selectedCaseGroups` = `buildEventGroups(visibleCaseEvents, clusterImport.clusters, timelineSegments)` – používa sa pre filter list a detail panel
- `selectedAnomalyGroups` = `buildAnomalyGroups(visibleCaseEvents, anomalyImport.anomalies, timelineSegments)` – používa sa pre filter list a detail panel
- `selectedFcaFactorGroups` = `buildFcaFactorGroups(visibleCaseEvents, fcaImport.factors, timelineSegments)` – používa sa pre filter list a detail panel

Úroveň 2 – buduje sa pre aktuálne viditeľné priblíženie (zoom) (z `graphEvents`, čo sú udalosti v `timelineScale.viewSegments`):

- `graphClusterGroups`, `graphAnomalyGroups`, `graphFcaFactorGroups` – používa sa na vykreslenie grafu
- `graphGroups` = `packGraphGroups([...graphClusterGroups, ...graphAnomalyGroups, ...graphFcaFactorGroups])`
- `packGraphGroups` (`utils/events.ts:298`) priradí každej viditeľnej skupine číslo riadku v rozložení. Tieto skupiny sa potom vykreslia ako farebné pásy (lane-y) v hornej časti grafu.

Špeciálnym mechanizmom sú tzv. *Stage markery*: vznikajú priamo v `App.tsx:256-280` ako `TimelineStageMarker[]`, neprechádzajú cez `build*` funkcie ani `packGraphGroups`. Sú to diskkrétne vertikálne indikátory na časovej osi, nie súvislé pásy.

3.4.8 Histogram hustoty (density)

Histogram hustoty poskytuje prehľad o rozložení udalostí na časovej osi. Je vytvorený rozdelením časového rozsahu na intervaly, v ktorých sa spočíta počet udalostí, pričom veľkosť intervalov sa dynamicky prispôsobuje aktuálnemu rozsahu zobrazenia.

`buildDensityBins` (`utils/chart.ts:117`) – rozdelí viditeľný časový rozsah do binov ("kontajnerov"), spočíta udalosti v každom bine. Šírka binu je dynamická (natural density bin width – zaokrúhľuje na "peknú" hodnotu 100ms, 250ms, 500ms, 1s, 5s, ...). Výsledok je vykreslený ako oblastný graf (area chart).

3.4.9 Drilldown

Na zobrazenie detailného pohľadu na udalosti sa vytvára hierarchická stromová štruktúra, ktorá umožňuje postupné rozbaľovanie agregovaných údajov až na úroveň jednotlivých zdrojových udalostí.

`buildDrilldownRows` (`utils/events.ts:302`) – z `matchingEvents` vytvorí hierarchický strom: top agregácia → grouped agregácia (voliteľná) → zdrojové udalosti. Virtualizácia (len viditeľné riadky = `virtualDrilldown`).

3.4.10 Panely detailov

Po výbere udalosti, skupiny alebo markera sa zobrazí panel s podrobnými informáciami o vybranom prvku. Typ zobrazeného panelu závisí od charakteru výberu a poskytuje buď detailné informácie o jednej udalosti, alebo prehľad zodpovedajúcich udalostí v hierarchickej štruktúre. K dispozícii sú 2 typy panelov:

- `DetailsPane` – obsahuje metadáta, FCA faktory, surové JSON dáta.
- `MatchingDetailsPane` – drilldown strom, časové okná, možnosť zoom-to-selection

3.5 Vloženie dát

Vloženie dát je možné vykonať kliknutím na tlačidlo *Upload data*. Následným kliknutím na jednotlivé plochy je možné vložiť požadované vstupné dáta (viď. **Obrázok 2**).

Obrázok 2 - Ukážka obrazovky na vkladanie vstupných dát

3.6 Grafické rozhranie a jeho funkcionality

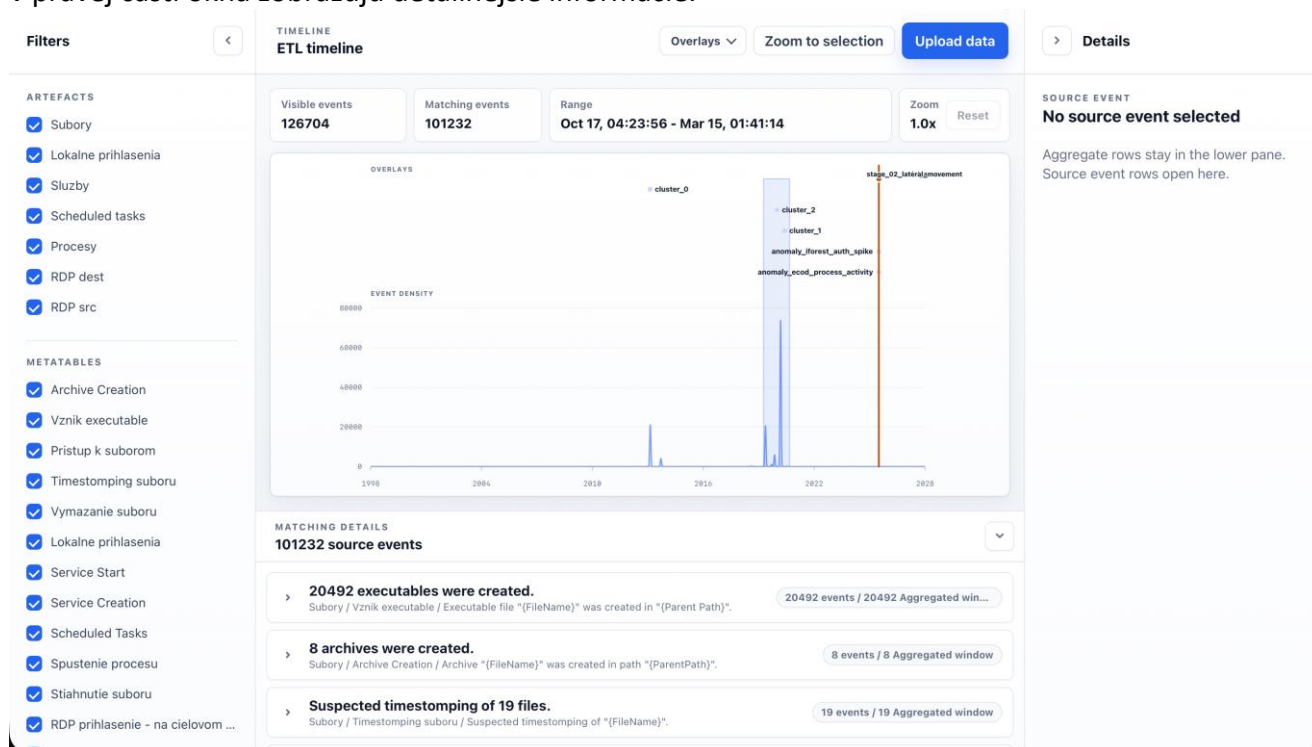
Nástroj na demonštráciu funkcionality časovej osi (viď. **Obrázok 3**) pozostáva z nasledujúcich častí:

- graf časovej osi,
- filtre,
- textové dáta,
- detaily o udalosti.

Nosnou časťou je samotný graf časovej osi, ktorý graficky zobrazuje množstvo dát v čase spolu so základnými informáciami ako počet zobrazovaných udalostí.

Filtre v ľavej časti okna umožňujú výber požadovaných typov udalostí v hlavných aj voliteľných dátach.

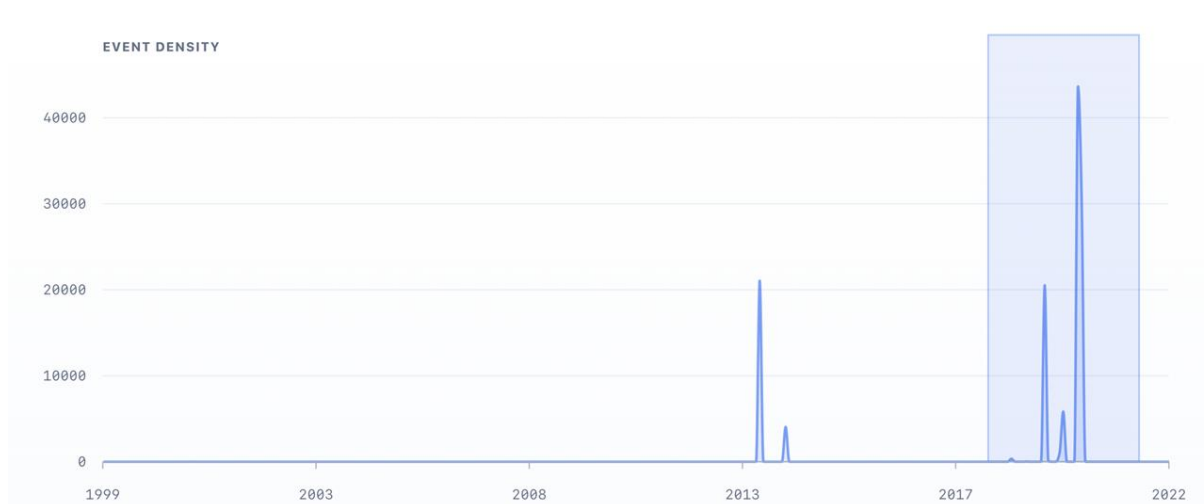
V dolnej časti okna sa zobrazujú textové informácie o udalostiach a po kliknutí na udalosť sa v pravej časti okna zobrazujú detailnejšie informácie.



Obrázok 3 - Grafické rozhranie ETL Timeline

3.6.1 Zobrazenie grafu časovej osi

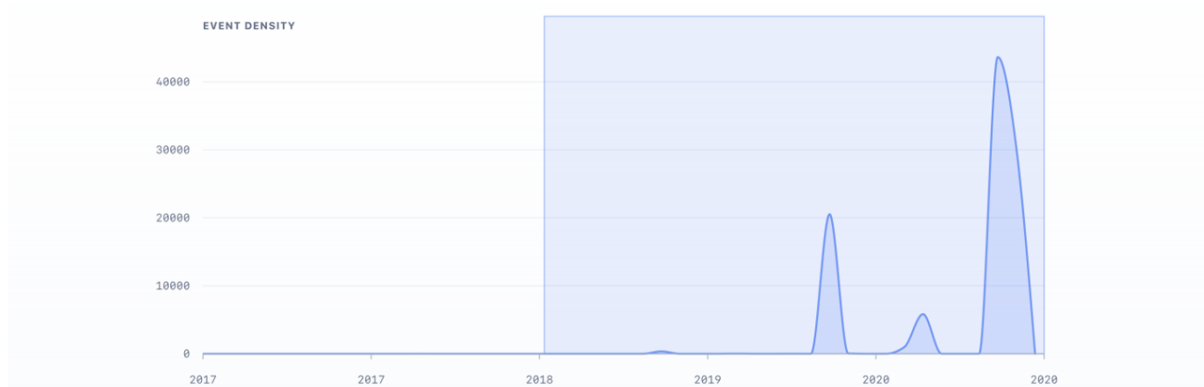
Graf časovej osi sa automaticky zobrazí po vložení povinných dát. Znázorňuje počet udalostí v danom čase. Na časovej osi je možné označiť konkrétny úsek dát, o ktorom sa majú zobraziť aj konkrétne údaje (na **Obrázok 4** je modrou farbou zvýraznený označený úsek).



Obrázok 4 – Ukážka časovej osi s výberom časového úseku

Implementovaná je aj funkcia priblíženia (zoom) – umožňuje časovú os priblížiť tak, aby sa detailnejšie zobrazil požadovaný časový úsek, čo zabezpečuje možnosť detailnejšieho preskúmania. Na

Obrázok 5 je ukážka priblíženej časovej osi na roky 2017-2020. Vhodné je to predovšetkým pri časovej osi zachytávajúcej dlhý časový úsek.



Obrázok 5 - Ukážka časovej osi s priblížením na úsek 2017 – 2020

Pre zrušenie označenia výberu stačí opätovne kliknúť na graf, zrušenie priblíženia grafu umožňuje tlačidlo **Reset**.

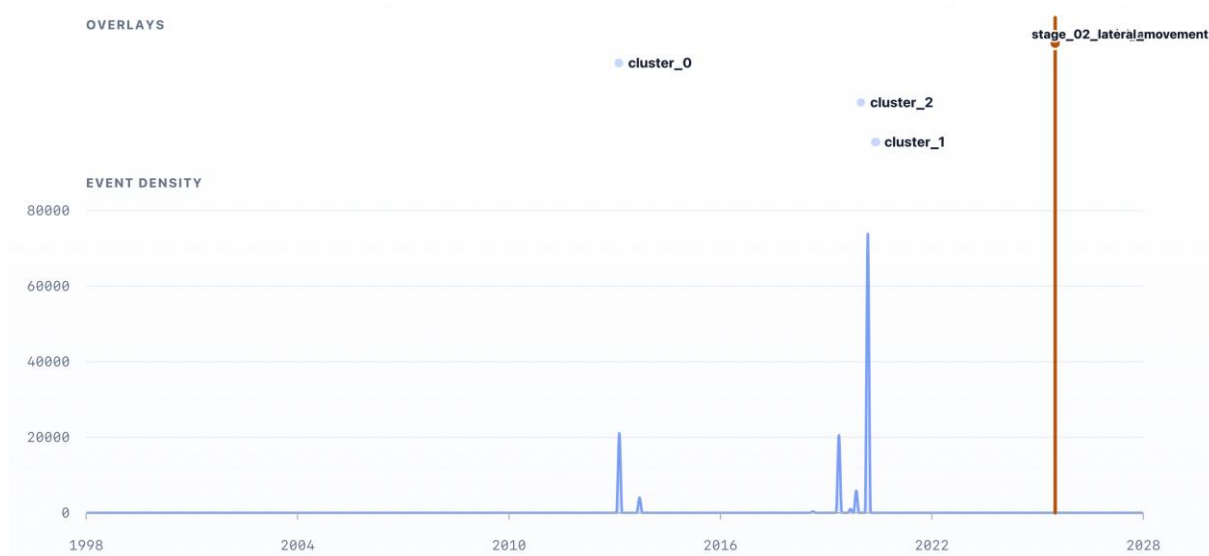
3.6.2 Obohatenie ETL Timeline o výstupy ďalších analýz

Funkcionalita obohatenia umožňuje zobrazit výstupy ostatných analytických nástrojov priamo nad časovou osou. Vďaka tomu je možné jednoduchšie porovnávať výsledky jednotlivých analýz a rýchlejšie odhadnúť časový úsek, v ktorom sa diala podozrivá aktivita.

Základným predpokladom pre zobrazenie doplnkových údajov je nahranie príslušných dát prostredníctvom rozhrania **Upload data**. Následne sú jednotlivé artefakty automaticky spracované a pripravené na zobrazenie.

Všetky podporované výstupy analýz (s výnimkou **FCA**) sa automaticky zobrazia ako prekrývajúce sa vrstvy (z angl. overlays, ďalej iba vrstvy) nad grafom časovej osi. Používateľ môže ich viditeľnosť kedykoľvek meniť prostredníctvom rozhrania **Overlays**, kde je možné jednotlivé vrstvy zapínať alebo vypínať podľa potreby. Tým je možné zobrazíť iba relevantné informácie bez zbytočného zahltenia grafu.

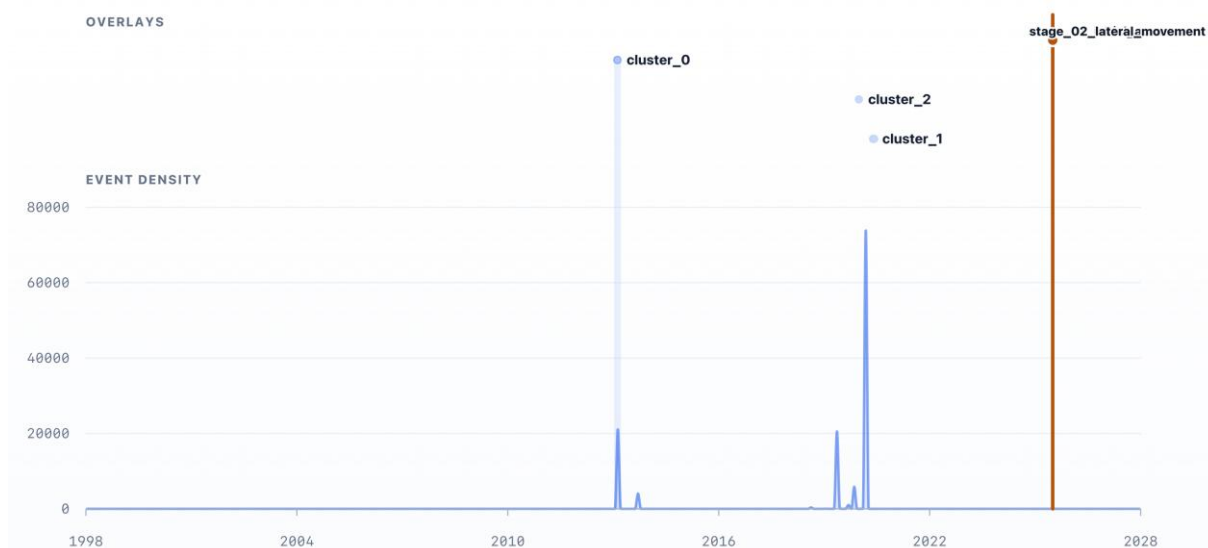
Klastre, anomálie, aj faktory z FCA sa zobrazujú ako kruhy. Klastre vo farbách modrej, anomálie v červených farbách a faktory vo farbách zelenej. Attack stages sú znázornené ako oranžové vertikálne čiary. Ukážka zobrazenia klastrov a stages je na **Obrázok 6**.



Obrázok 6 - Časová os obohatená o klastre a stages

Všetky zobrazené vrstvy sú interaktívne. Po kliknutí na konkrétny prvok sa zobrazia podrobnejšie informácie o danom výstupe, napríklad jeho identifikátor, časové údaje alebo ďalšie atribúty závislé od typu artefaktu.

Pri výstupoch z **Cluster**, **Anomalies** a **FCA** má interakcia navyše rozširujúcu funkcionality. Po výbere konkrétneho prvku sa automaticky zvýrazní aj časová oblasť grafu, ktorú daný výstup pokrýva, vid'. **Obrázok 7**. Používateľ tak okamžite vidí interval, ku ktorému sa výsledok vzťahuje, čo uľahčuje analýzu a porovnávanie s ostatnými udalosťami zobrazenými na časovej osi.



Obrázok 7 - Časová os so zvýraznením úseku pre klaster „cluster_0“

3.6.3 Práca s textovými dátami

Spodný panel s názvom **Matching Details** umožňuje pracovať s udalosťami časovej osi. Zobrazujú sa v ňom udalosti iba z vybraného časového úseku (ak nie je vybraný žiadny úsek, panel aj vzhľadom na množstvo záznamov ostáva prázdny). Panel napodobňuje prácu s udalosťami v nástroji ETL.

V pravom hornom rohu sa používateľovi zobrazí informácia o počte udalostí a agregovaných okien. Na **Obrázok 8** máme 35 udalostí a 35 agregovaných okien – každá udalosť je teda agregovaná. Po kliknutí na agregovaný záznam sa pod ním rozbalia samostatné udalosti. Zobrazovať sa však môžu priamo aj samostatné udalosti, ktoré agregované nie sú.

MATCHING DETAILS		
126704 source events		
35 executables were created in .\Windows\SysWOW64 folder. ParentPath: .\Windows\SysWOW64		35 events / 35 Aggregated window
23:35:16	Executable file mfc40.dll was created in .\Windows\SysWOW64. Subory / Vznik executable	Aggregated window
23:35:16	Executable file mfc40u.dll was created in .\Windows\SysWOW64. Subory / Vznik executable	Aggregated window
23:35:33	Executable file ctl3d32.dll was created in .\Windows\SysWOW64. Subory / Vznik executable	Aggregated window
23:35:33	Executable file msvcrt20.dll was created in .\Windows\SysWOW64. Subory / Vznik executable	Aggregated window
23:35:33	Executable file crt.dll was created in .\Windows\SysWOW64. Subory / Vznik executable	Aggregated window
23:36:39	Executable file aspnet_counters.dll was created in .\Windows\SysWOW64. Subory / Vznik executable	Aggregated window
23:36:40	Executable file msvcp120_clr0400.dll was created in .\Windows\SysWOW64. Subory / Vznik executable	Aggregated window

Obrázok 8 - Zobrazenie dát z vyznačeného časového úseku s podrobnosťami o agregovanom zázname

Kliknutím na neagregovaný záznam sa v paneli na pravej strane zobrazia jeho podrobnosti, vid'. **Obrázok 9**. V podrobnostiach je možná okrem časovej pečiatky nájsť informácie o type dát, používateľovi, či vzore použitom na vytvorenie udalosti. Ak udalosť patrí do klastra (ak sú do nástroja pridané), zobrazí sa aj prislúchajúci klaster.

V prípade, že sú dostupné vstupné údaje z FCA, zobrazí sa v detaile záznamu tlačidlo **Find all similar**. Po jeho aktivácii systém vyhľadá a zvýrazní všetky faktorové okná, v ktorých sa daný záznam nachádza. Táto funkcionlita umožňuje rýchlo identifikovať všetky výskyty záznamu v rámci výsledkov FCA analýzy a sledovať ich vzájomné súvislosti.

SOURCE EVENT	
Find all similar	1 factor
Executable file mfc40.dll was created in .\Windows\SysWOW64.	
ID	files_exe_created_total_000001:files_exe_created_by_parentpath_000212:files_21504:000007
TIME	Aug 21, 23:35:16
METADATA	Vznik executable
SITUATION	Executable file "{FileName}" was created in "{Parent Path}".
TYPE	Aggregated window
CLUSTER	Aug 21 overnight activity
HOST / USER	Unknown host / Unknown user
ARTIFACT	Subory
Executable file mfc40.dll was created in .\Windows\SysWOW64.	

Obrázok 9 - Podrobnosti o zdrojovej udalosti

V spodnej časti detailu sa zároveň zobrazujú podrobné informácie o faktore (viď. **Obrázok 10**), ku ktorému vybraný záznam patrí. K dispozícii sú informácie o počte atribútov, ktoré tvoria daný faktor, či o počte ďalších záznamov, ktoré spadajú do daného faktora.



Obrázok 10 – Ukážka detailov faktoru

3.6.4 Filtrovanie udalostí

Podľa potreby umožňuje ETL Timeline aktiváciu/deaktiváciu filtrov na rôznych úrovniach. Dajú sa vyberať záznamy podľa typu – týkajúce sa súborov, týkajúce sa procesov, týkajúce sa RDP prihlásení a pod. Ďalšia úroveň umožňuje výber konkrétnych metazáznamov, napr. vytvorenie archívu, či spustenie služby, vid'. **Obrázok 11.**

ARTEFACTS

- ☒ Subory
- ☒ Lokálne prihlásenia
- ☒ Služby
- ☒ Scheduled tasks
- ☒ Procesy
- ☒ RDP dest
- ☒ RDP src

METATABLES

- ☒ Archive Creation
- ☒ Vznik executable
- ☒ Prístup k suborom
- ☒ Timestamping suboru
- ☒ Vymazanie suboru
- ☒ Lokálne prihlásenia
- ☒ Service Start
- ☒ Service Creation
- ☒ Scheduled Tasks
- ☒ Spustenie procesu

Obrázok 11 - Ukážka filtrov

4 Forenzný význam

Hlavným zámerom nástroja ETL Timeline je spojiť rôznorodé zdroje forenzných dát do jedného prehľadného celku. Windows generuje záznamy o aktivite naprieč mnohými nezávislými zdrojmi – Event Log, prefetch súbory, register, plánované úlohy a ďalšie artefakty – pričom každý z nich má vlastnú štruktúru, formát aj mieru granularity. Samostatne poskytuje každý z týchto zdrojov iba čiastkový pohľad na dianie v systéme a ich manuálne prechádzanie a vzájomné porovnávanie je časovo náročné a náchylné na chyby. ETL Timeline tieto heterogénne zdroje zlučuje do jednej spoločnej časovej osi, vďaka čomu analytik pracuje na jednom mieste namiesto prepínania medzi viacerými nástrojmi, tabuľkami a formátmi.

Kľúčovým prínosom je nielen samotné zlúčenie dát, ale aj ich spojenie do metazáznamov (napr. vytvorenie procesu, vytvorenie plánovanej úlohy, stiahnutie súboru a pod.) a agregácia do časových okien. Namiesto desiatok, či stoviek jednotlivých nespracovaných záznamov tak analytik pracuje s výrazne menším počtom zmysluplných udalostí, čo priamo znižuje objem dát, ktoré je potrebné prehliadať. Táto redukcia je z forenzného hľadiska podstatná, keďže pri veľkom objeme dát rastie riziko, že relevantná udalosť ostane prehliadnutá. Agregácia toto riziko výrazne obmedzuje bez toho, aby sa strácala možnosť dostať sa v prípade potreby až k pôvodnému zdrojovému záznamu.

Ďalším významným prínosom je grafické zobrazenie časovej osi, ktoré umožňuje analytikovi na prvý pohľad identifikovať zhľady aktivity, výpadky alebo náhle zmeny intenzity zápisov – javy, ktoré by v surových dátach zostali skryté. Nástroj tento pohľad ešte posilňuje tým, že priamo nad časovú os vizualizuje aj výstupy ďalších analytických metód – klastre, anomálie, stages a faktory z formálnej konceptovej analýzy (FCA). Vďaka tomu nemusí analytik hľadať podozrivé úseky „naslepo“ v celom datasete, ale je na nich aktívne nasmerovaný: anomálie upozorňujú na časové okná najviac odlišné od bežnej aktivity a môžu indikovať napríklad neautorizované prihlásenie alebo spustenie neočakávaného procesu, klastre odhaľujú skupiny podobných udalostí (prirodzených aj potenciálne podozrivých) a FCA faktory umožňujú prostredníctvom funkcie „Find all similar“ rýchlo dohľadať všetky záznamy so spoločnými atribútmi. Analytik sa tak môže sústrediť práve na tie časti dát, ktoré si reálne zaslúžia bližšie preskúmanie, namiesto rovnomerného prehliadania celého objemu záznamov.

Spojením týchto vlastností – konsolidácie rôznych zdrojov na jedno miesto, agregácie znižujúcej objem prehliadaných dát a zvýrazňovania anomálií, klastrov a ďalších výstupov priamo v kontexte časovej osi – poskytuje ETL Timeline analytikovi integrovaný a efektívny pohľad na prípad, ktorý zrýchľuje analýzu a zároveň znižuje riziko prehliadnutia relevantných udalostí.

5 Záver

Nástroj ETL Timeline prináša integráciu výstupov dvoch nezávislých analytických tokov do jedného prostredia pre ich spoločnú interpretáciu. ETL Timeline prepája štandardizované metazáznamy vytvorené z výstupov EZ Tools s analytickými výsledkami získanými nad dátami spracovanými nástrojom Plaso. Vďaka tomu sú udalosti získané z forenzných artefaktov zobrazené v spoločnom časovom kontexte spolu s výsledkami analýz, ako sú identifikované klastre, anomálie, fázy útoku či vzťahy medzi udalosťami. Takýto integrovaný pohľad eliminuje potrebu manuálneho prepájania výsledkov z viacerých nástrojov a umožňuje analytikovi efektívnejšie identifikovať časové a logické súvislosti medzi jednotlivými udalosťami, čo významne podporuje proces digitálnej forenznej analýzy.

Možnosti ďalšieho rozvoja spočívajú predovšetkým v rozšírení podporovaných zdrojov dát, typov metazáznamov a integrácii ďalších analytických metód, ktorých výstupy by bolo možné zobrazovať v spoločnej časovej osi. Rozšírenie nástroja o priebežné spracovanie dát a pokročilejšie možnosti filtrovania by zároveň zvýšilo jeho využiteľnosť pri riešení komplexných bezpečnostných incidentov aj v prostredí reálnych vyšetrovaní.

6 Bibliografia

- [1] Hudan Studiawan, F. B. (2025). Towards a standardized methodology and dataset for evaluating LLM-based digital forensic timeline analysis. *Forensic Science International: Digital Investigation*, Volume 54, Supplement. doi:<https://doi.org/10.1016/j.fsidi.2025.30198>
- [2] Frank Breiting, H. S. (2025). SoK: Timeline based event reconstruction for digital forensics: Terminology, methodology, and current challenges. *Forensic Science International: Digital Investigation*, Volume 53, Supplement. doi:<https://doi.org/10.1016/j.fsidi.2025.301932>
- [3] ZIMMERMAN, Eric. *Eric Zimmerman Tools* [online]. [cit. 2026-08-04]. Dostupné na: <https://ericzimmerman.github.io/>
- [4] React [online]. [cit. 2026-08-04]. Dostupné na: <https://react.dev/>
- [5] npm, Inc., "About npm" [online]. [cit. 2026-08-04]. Dostupné na: <https://docs.npmjs.com/about-npm>