# Data Preparation and Enrichment Algorithm for Fraud Detection System

Viktor Sahaidak[1,*,†], Kamila Storchak[1,†], Viktoriia Zhebka[1,†], Vladimir Bilavka[1,†] and Andrii Bondarchuk[2,†]

[1] State University of Information and Communication Technologies, Solomyanska Street 7, 03110, Kyiv, Ukraine
[2] Borys Grinchenko Kyiv Metropolitan University, 18/2, Bulvarno-Kudriavska St., 04053, Kyiv, Ukraine

**Abstract**

Following paper provides small overview of recent research results on fraud detection solutions, test environment, conditions that showed a lack of focus and testing on standardized data formats and following information delivery time. In paper data preparation and enrichment algorithm scheme, related tables, configurations, required fields for mapping between session detail record from VoLTE platform and billing data was described. Developed algorithm approbation results were provided and compared between case with only standardized data format (NRTRDE, TAP3) and case that combines NRTRDE, TAP3 data with enriched xDR prepared by algorithm. Conclusions were made, that following algorithm improved fraudulent traffic detection time, allows to upload and enrich data directly in Fraud detection system database, proved that data delivery time is vital KPI and should be considered during Fraud management system development phase and approbation before implementation on carrier network.

**Keywords**

Information technologies, VoLTE, Big data, network security, telecommunication fraud, KPI, ETL.

## 1. Introduction

Telecommunication fraud is very common phenomenon. Fraud can be classified depending on network generation and realization methods [1,4], impacted services [2,4], scale of influence on specific territory or network [3,4]. Review of recent researches made by cooperation of Beihang University and Technical University of Munich [5], University of Yaoundei [6], University of Moratuwa [7] show, that most focus is done for developed system probation in isolated environment. Research in [5] describes machine learning solution based on Hawkes-enhanced Sequence Model, that is trained dataset provided by one of the largest telecommunication operators in China. In research [6] Docker container-based ELK system is used with predefined conditions and dataset with SMS xDRs generated by code included in package. Research [7] describes real time detection system similar in topology, subsystems and processing logic to commercial solutions. University of Moratuwa developed call pattern-analysis algorithms that were examined on training and test dataset received from service provider. In other words, proposed solutions were examined on cleaned and already formatted dataset provided by carrier, software used records generation, that exclude transfer delay on real network.

Data delivery have significant impact on timely detection and block of such sessions. Existing standardized methods for roaming data exchange between network operators have defined delays

*Information Technology and Implementation (IT&I-2025), November 20-21, 2025, Kyiv, Ukraine*
* Corresponding author.
† These authors contributed equally.
qsagvict@ukr.net (V. Sahaidak); kpstorchak@ukr.net (K. Storchak); v.zhebka@duikt.edu.ua (V. Zhebka); wolod29@gmail.com (V. Bilavka); a.bondarchuk@kubg.edu.ua (A. Bondarchuk);
0009-0000-9724-958X (V. Sahaidak); 0000-0001-9295-4685 (K. Storchak); 0000-0003-4051-1190 (V. Zhebka); 0009-0000-5053-2601 (V. Bilavka); 0000-0001-5124-5102 (A. Bondarchuk);
© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).



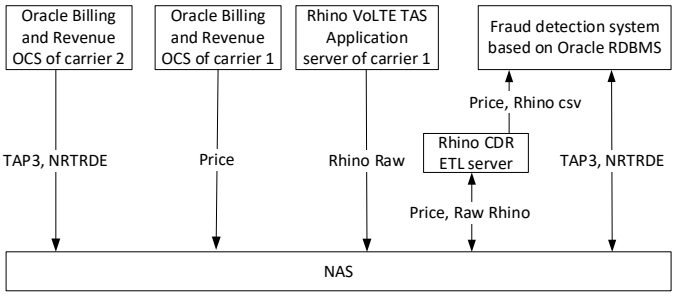
CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-02-17
CEUR-WS.org/Vol-4170/Short03.pdf

for transaction delivery time to partner network. This delivery time can vary not more than 4 hours or 8 hours (in case of outages on partner network) for NRTRDE and not more than 30 days or not more than 40 days (in case of outages on partner network) for TAP3 defined by GSMA [8,9]. In order to improve detection, data preparation and enrichment algorithm was developed and tested on environment approximate to real network.

## 2. Test environment and data flow description

In order to simulate NRTRDE and TAP3 files preparation, Oracle Billing system was used. This system has embedded feature to generate such files. For data transfer from one carrier network to another carrier DCH providers are used of some other near real time file sharing service.

Test environment (figure 1) consist of two OCS for carrier 1 and 2, Rhino VoLTE TAS Application server, Rhino CDR ETL server and Fraud detection system (FMS). All elements are connected to one and same NAS to deliver, prepare and upload files to FMS. OCS of carrier 1 provides price file to calculate cost of subscriber session in Rhino Raw. OCS of carrier 2 generates NRTRDE and TAP3 files with time delays defined in their standards [8,9], delivers to NAS and FMS collects following data from NAS. Rhino VoLTE TAS generate Rhino Raw in AVP format every 5 minutes, ETL server convert Rhino Raw from AVP to .csv format (Rhino csv) and with help of Oracle ODI collect Rhino Raw and Price file, calculate and enrich session cost, convert session time from UNIX to readable format, load result to FMS DB.




| | File | Path | Description |
|---|---|---|---|
| 1 | NRTRDE | /fraud_detection/NRTRDE/in | File in ASN1 format |
| 2 | TAP3 | /fraud_detection/TAP3/in | File in ASN1 format |
| 3 | Rhino Raw | /fraud_detection/Rhino/raw | txt file in AVP format |
| 4 | Price | /fraud_detection/Price/in | Price file in csv format |
| 5 | Rhino csv | /fraud_detection/Rhino/in | File in csv format |

**Figure 1:** Test environment of data preparation and enrichment algorithm for Fraud detection system.

## 3. ETL server file preparation and DB structure

Price files exported from OCS contain 8 fields (refer to table 1) for recognition of service name and type, TADIG, MCC, MNC, IOI, minimum price and currency. Fields from Rhino CDR OC-Service-Type, OC-Call-Type, OC-Access-Network-MCC-MNC, OC-Visited-Network-MCC-MNC, OC-IMSI-MCC-MNC, Inter-Operator-Identifier are used from Rhino csv to correlate Price file and session.

To prepare Rhino csv file, ETL server have bash code configured to run periodically through cron jobs. Rhine raw file is delivered by VoLTE in AVP (attribute value-pair) format and one AVP can contain more than one child AVP. Rhino VoLTE AVP fields are similar to ETSI standard [10] but also providing additional self-developed AVPs.

Oracle ODI logical flow (figure 2) perform next actions:

1. Files are loaded from directories «/media/sf_fraud_detection/Price/in» and «/media/sf_fraud_detection/Rhino/in» to temporary tables C$_0EXP, C$_1CDR;

2. Values from temporary tables are transformed into expressions (EXPRESSION_EXP, EXPRESSION_CDR) for join operations (JOIN_MOC, JOIN_MTC);
3. Data from tables are joined by expression for JOIN_MTC "*CDR.Service_name=EXP.Service_name and EXP.Service_type=CDR.Service_type and CDR.Destination_network=EXP.IOI"* and JOIN_MOC *"CDR.Service_name=EXP.Service_name and EXP.Service_type=CDR.Service_type and CDR.Originated_network=EXP.IOI"*;
4. After join operation was done, results are loaded in table fraud.ur_ims_records. In this table dates are transformed from UNIX OS format by expression "*(to_date('1970-01-01 00:00:00','YYYY-MM-DD HH24:MI:SS') + numtodsinterval((CDR.Start_call_time/1000),'SECOND'))"*, price for provided service is calculated by *"(EXP.Min_price*(CDR.End_call_time-CDR.Start_call_time)/60000)"*, session duration "(CDR.End_call_time-CDR.Start_call_time)/60000";
5. After record is loaded in table, system periodically checks it and loads new CDRs in application memory, where these records will be processed by fraud detection rules;
6. After ODI loaded records in table, original files that are located in «/media/sf_fraud_detection/Rhino/in» moved automatically to directory «/media/sf_fraud_detection/Rhino/done»;

**Table 1**
Price file fields definition

| Field name | Description | Value example |
|---|---|---|
| Service_name | Service name used by subscriber | SipCall |
| Service_type | Service type used by subscriber | MOC |
| Network_TADIG | Network TADIG code | USAHI |
| MCC | Mobile country code | SWE01 |
| MNC | Mobile network code | 255 |
| IOI | Inter Operator Identifier | 01 |
| Min_price | Price for provided service | bea.net |
| Currency | Price currency | 10 |

On physical level of described ODI scheme for data load from files to temporary table Loading Knowledge Module (LKM) SQL to SQL (Built-in) Global is used. It is built-in ODI module for common operations with relational databases.

After records were loaded in database, Fraud detection software starts to process it and each record goes through next lifecycle of tables located in RDBMS: fraud.ur_ims_records->fraud.pos_prealerts->total_alerted_cdrs/fraud.ims_alerted_cdrs->fraud.alerts->fraud.cases

This tables perform next functions:

- fraud.ur_ims_records – this table is used to store Rhino CDRs on FMS DB;
- fraud.pos_prealerts – this table is used by FMS to create temporary fraud alerts. It stores all types of alerts that became complete or were discarded as duplicates;
- fraud.ims_alerted_cdrs – this table stores the CDRs that were used to create a fraud alert. By default, the records in total_alerted_cdrs duplicate the information in fraud.ur_ims_records, but will also provide the time when record was added into the table after alert was created and alert ID related to record.
- fraud.total_alerted_cdrs - stores information about all CDRs that were used to create an alert, regardless of the source type (it combines CDRs from all types of data sources).
- fraud.alerts – stores information about alerts. This table is common for all data source types and provides information about the alert creation time, number of CDR records, rule

identifier based on which the alert was created, suspected entity information (subscriber number).

- fraud.cases – stores information about created fraud cases. After an alert is created, the system either adds it to an existing case or creates a new one on the alert subject if one does not exist or has been closed.

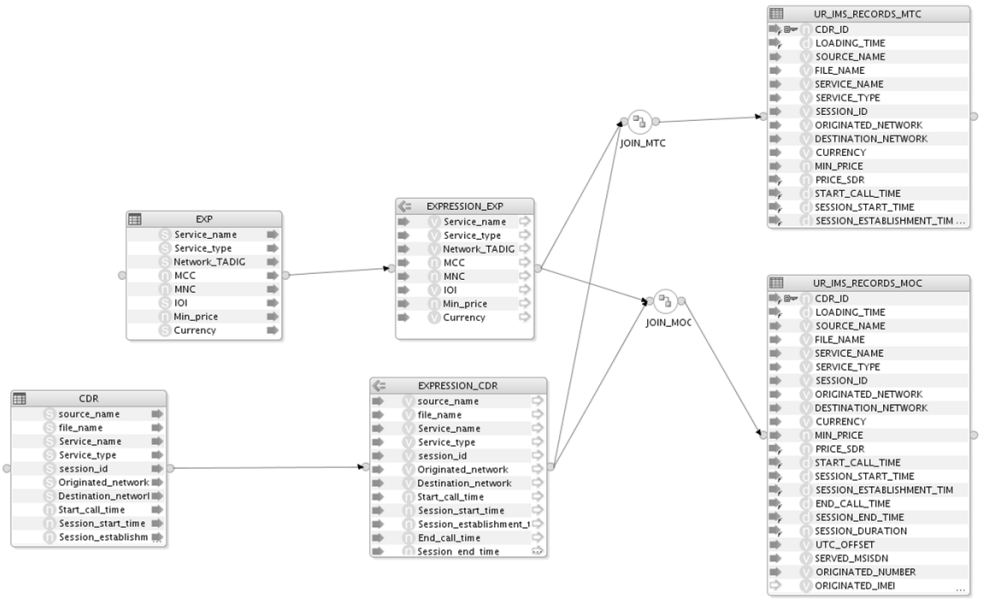



**Figure 2:** Oracle ODI logical flow with relationship description between expressions, join operations and tables in Fraud management system.

Mapping between table fraud.ur_ims_records in FMS, Rhino raw and price file, described in table 2. Depending on service type in record only one of two fields can be filled:

- Originated_imei can be filled in Rhino raw if service type is MOC
- Destination_imei can be filled in Rhino raw if service type is MTC.

Fraud detection in FMS consists of Detection, Correlation and Case management phases (figure 3). Each phase uses next lookup tables:

- fraud.sources – stores information about data sources and their unique identifiers.
- fraud.rule_sources – stores information about rules belonging to data sources defined in fraud.sources.
- fraud.rules - this table stores information about rules created by users (unique identifier of the rule, name of conditions used to detect suspicious traffic, and boundary conditions, which include the monitoring period, number of suspicious sessions required to create a fraud case, etc).
- fraud.conditions – this table contain information about conditions created by users to track a certain type of traffic. Conditions can consist of any fields of the ur_*_records tables and lists that use Boolean algebra conditions to combine or exclude a certain condition.
- fraud.hotlist - stores information about existing lists created by users. It contains information such as the list creation time, username, list name, and its unique identifier.
- fraud.hotlist_values - this table stores the values of suspicious objects or identifiers of unscrupulous carriers. Each value contains the time it was created and the time until which this value must exist in the list to which it belongs.

**Table 2**
Mapping between files and fields in table fraud.ur_ims_records

| Field name | Field type | Rhino raw field | Price file field |
|---|---|---|---|
| cdr_id | Number(16), cdr_id_seq.nextval | | - |
| Loading_time | Date, systime | | - |
| source_name | Varchar2(30 char) | hostname=mortadella | - |
| file_name | Varchar2(30 char) | Rhino file name | - |
| Service_name | Varchar2(10 char) | OC-Service-Type(Ext,Ext) | - |
| Service_type | Varchar2(10 char) | OC-Call-Type(Ext,Ext) | - |
| session_id | Varchar2(65 char) | User-Session-Id | - |
| Originated_network | Varchar2(30 char) | Originating-IOI | - |
| Destination_network | Varchar2(30 char) | Terminating-IOI | - |
| Currency | Varchar2(8 char) | | Currency |
| Min_price | Number(5) | | Min_price |
| Price_sdr | Number(20,6) | Duration*Min_price | - |
| Start_call_time | Date | OC-Start-Time | - |
| Session_start_time | Date | OC-Session-Start-Time | - |
| Session_establishment_time | Date | OC-Session-Established-Time | - |
| End_call_time | Date | OC-End-Time | - |
| Session_end_time | Date | OC-Session-End-Time | - |
| Session_duration | Number(20) | (OC-End-Time - OC-Start-Time)/1000 | - |
| UTC_OFFSET | Varchar2(5 char) | UTC+0 | - |
| Served_msisdn | Varchar2(32 char) | Subscription-Id-Data | - |
| Originated_number | Varchar2(32 char) | Calling-Party-Address | - |
| Originated_imei | Varchar2(32 char) | If OC-Call-Type(Ext,Ext)[MOC], then User-Equipment-Info-Value, Else 'Empty' | - |
| Destination_number | Varchar2(32 char) | Called-Party-Address | - |
| Destination_imei | Varchar2(32 char) | If OC-Call-Type(Ext,Ext)[MTC], then User-Equipment-Info-Value, Else 'Empty' | - |
| Id_third_party_num | Varchar2(32 char) | Called-Asserted-Identity | - |
| Id_dialed_number | Varchar2(32 char) | Requested-Party-Address | - |
| Nt_charge_id | Varchar2(20 char) | OC-Call-Id | - |
| Serving_Node_Role | Varchar2(20 char) | Role-Of-Node | - |
| Serving_Node_Function | Varchar2(20 char) | Node-Functionality | - |
| Charging_Instance | Varchar2(40 char) | OC-Charging-Instance-Name | - |
| Nt_termination_cause | Number(5) | OC-OCS-Session-Termination-Cause | - |
| Session_ICSI | Varchar2(40 char) | IMS-Communication-Service-Identifier | - |
| Session_ICI | Varchar2(40 char) | IMS-Charging-Identifier | - |

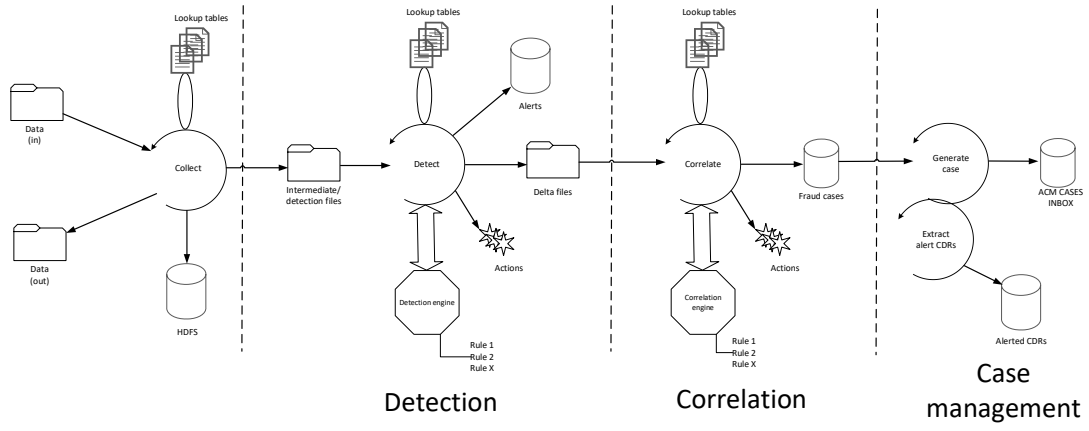



**Figure 3:** Full processing fraud management processing cycle.

NRTRDE and TAP3 files are decoded by built-in ASN1_decoder of Fraud management system to csv format and loaded in appropriate tables:

- fraud.ur_nrtrde_records – this table is used to store decoded CDRs from NRTRDE on FMS DB;
- fraud.ur_ims_records – this table is used to store decoded CDRs from TAP3 on FMS DB;

Processing flow for NRTRDE and TAP3 records by Fraud detection software is same:

- fraud.ur_nrtrde_records->fraud.pos_prealerts->total_alerted_cdrs/fraud.nrtrde_alerted_cdrs->fraud.alerts->fraud.cases;
- fraud.ur_tap3_records->fraud.pos_prealerts->total_alerted_cdrs/fraud.tap3_alerted_cdrs->fraud.alerts->fraud.cases;

## 4. Rules for testing, number of records

To detect fraudulent traffic several same detection rules were created for each type of data source (refer to table 3). Following rules are targeted to detect long and small duration sessions that are commonly recognized as fraudulent behavior.

**Table 3**
Detection rules definition

| № | Traffic definition for monitoring | Definition of suspicious behavior | Threshold for alert creation | Monitoring time frame value | Suspected entity |
|---|---|---|---|---|---|
| 1 | MOC, (terminating network - SWE01 or terminating network – bea.net), calling number is not present in whitelist | duration>=30 min and duration<=1800 min | count >=3 calls based on originated IMSI/MSISDN | 1 hour | Originated MSISDN, Originated IMSI |
| 2 | MOC, (terminating network - SWE01 or terminating network – bea.net), calling number is not present in whitelist | duration>20 sec | count >=10 calls based on originated IMSI/MSISDN | 2 hours | Originated MSISDN, Originated IMSI |
| 3 | MOC, (terminating network - SWE01 or terminating network – bea.net), calling number is not present in whitelist | duration>20 sec and duration<30 sec | sum duration >= 1 hours based on originated IMSI/MSISDN | 6 hours | Originated MSISDN, Originated IMSI |
| 4 | MOC, (terminating network - SWE01 or terminating network – bea.net), calling number is present in whitelist | duration>20 sec and duration<30 sec | sum duration >= 1 hours based on terminated IMSI/MSISDN | 3 hours | Terminated MSISDN, Terminated IMSI |
| 5 | MTC, (originating network - SWE01 or originating network – bea.net), called number is not present in whitelist | duration>=30 min and duration<=1800 min | count >=3 calls based on originated IMSI/MSISDN | 1 hour | Originated MSISDN, Originated IMSI |
| 6 | MTC, (originating network - SWE01 or originating network – bea.net), called number is not present in whitelist | duration>20 sec | count >=10 calls based on originated IMSI/MSISDN | 2 hours | Originated MSISDN, Originated IMSI |
| 7 | MTC, (originating network - SWE01 or originating network – bea.net), called number is not present in whitelist | duration>20 sec and duration<30 sec | sum duration >=1 hours based on originated IMSI/MSISDN | 6 hours | Originated MSISDN, Originated IMSI |
| 8 | MTC, (originating network - SWE01 or originating network – bea.net), called number is not present in whitelist | duration>20 sec and duration<30 sec | sum duration >= 1 hours based on terminated IMSI/MSISDN | 3 hours | Terminated MSISDN, Terminated IMSI |

For traffic generation environment have 980 subscribers (490 virtual numbers for each carrier). Number of sessions in network can vary from 7 to 8 thousand within 24 hours. Table 4 contain number of files and session records generated within 1 day for each source type (Rhino CDR, NRTRDE, TAP3).

**Table 4**
Number of files and records for each source type

| Source name | Number of files | Number of records |
|---|---|---|
| Rhino CDR | 7472 | 7472 |
| NRTRDE | 101 | 6227 |
| TAP3 | 13 | 8605 |

## 5. Algorithm approbation results

The formulas defined in [12] will be used to calculate algorithm efficiency. For each record recognized as fraud by FMS, $T_{coll}$ will be calculated. For better scaling each result is divided into intervals of 10 minutes and weighted average value of $T_{coll}$ is calculated for NRTRE and TAP3.

Table 5 contain summary for fraud detection based on NRTRDE and TAP3 files detected within 24 hours.

**Table 5**
Number of $T_{coll}$ fraudulent records for NRTRDE and TAP3 source types

| NRTRDE | | TAP3 | |
|---|---|---|---|
| Time interval | Number of $T_{coll}$ | Time interval | Number of $T_{coll}$ |
| 00:10:00 | 4 | 03:30:00 | 2 |
| 00:20:00 | 2 | 11:50:00 | 2 |
| 00:30:00 | 2 | 15:50:00 | 2 |
| 00:40:00 | 2 | 19:00:00 | 2 |
| 00:50:00 | 2 | 24:40:00 | 2 |
| 01:10:00 | 2 | 24:50:00 | 2 |
| 01:40:00 | 2 | 30:40:00 | 2 |
| 02:10:00 | 2 | 32:50:00 | 2 |
| 02:20:00 | 2 | 33:10:00 | 2 |
| 02:30:00 | 2 | 35:30:00 | 2 |
| 03:10:00 | 2 | 37:20:00 | 2 |
| 03:20:00 | 2 | | |
| 04:00:00 | 12 | | |
| Overall number of alerted records | 38 | Overall number of alerted records | 22 |
| Weighted average value of $T_{coll}$ | 02:15:47 | Weighted average value of $T_{coll}$ | 24:28:11 |

In percentage distribution 31% of alerts generated based on NRTRDE files were uploaded in system within one hour, while 36% of alerts generated based on TAP3 files were uploaded in system within 24 hours.

Table 6 provide combined statistics with Rhino for NRTRDE and TAP3 data sources. In percentage distribution 94% of alerts created based on combination of NRTRDE and Rhino CDR files were uploaded in system within one hour, while 95% of alerts created based on combination of TAP3 and Rhino CDR files that were uploaded in system within one hour.

**Table 6**
Number of $T_{coll}$ fraudulent records for NRTRDE+Rhino CDR and TAP3+Rhino CDR source types

| NRTRDE+Rhino CDR | | TAP3+Rhino CDR | |
|---|---|---|---|
| Time interval | Number of $T_{coll}$ | Time interval | Number of $T_{coll}$ |
| 00:10:00 | 32 | 00:10:00 | 28 |
| 00:20:00 | 146 | 00:20:00 | 144 |
| 00:30:00 | 149 | 00:30:00 | 147 |
| 00:40:00 | 70 | 00:40:00 | 68 |
| 00:50:00 | 2 | 03:30:00 | 2 |
| 01:10:00 | 2 | 11:50:00 | 2 |
| 01:40:00 | 2 | 15:50:00 | 2 |
| 02:10:00 | 2 | 19:00:00 | 2 |
| 02:20:00 | 2 | 24:40:00 | 2 |
| 02:30:00 | 2 | 24:50:00 | 2 |
| 03:10:00 | 2 | 30:40:00 | 2 |
| 03:20:00 | 2 | 32:50:00 | 2 |
| 04:00:00 | 12 | 33:10:00 | 2 |
| | | 35:30:00 | 2 |
| | | 37:20:00 | 2 |
| Overall number of alerted records | 425 | Overall number of alerted records | 409 |
| Weighted average value of $T_{coll}$ | 00:36:21 | Weighted average value of $T_{coll}$ | 01:44:08 |

Comparing weighted average value of $T_{coll}$ for NRTRDE and TAP3, can be noticed that for NRTRDE latency was reduced by 3,7 times and for TAP3 it was reduced by 14 times (table 7).

**Table 7**
Difference of Weighted average value for NRTRDE, NRTRDE+Rhino CDR and TAP3, TAP3+Rhino CDR

| NRTRDE and NRTRDE+Rhino CDR | | | TAP3 and TAP3+Rhino CDR | |
|---|---|---|---|---|
| | NRTRDE | NRTRDE+Rhino CDR | TAP3 | TAP3+Rhino CDR |
| Weighted average value of $T_{coll}$ | 02:15:47 | 00:36:21 | 24:28:11 | 01:44:08 |
| Difference | 3,7 | | 14 | |

## 6. Conclusions

Approbation results of developed algorithm showed improvement in fraud detection speed, that decrease reaction time from security on fraudulent traffic. Algorithm allows to directly upload, transform time and calculate transaction cost of detail record in FMS database. Results proved that data delivery time have significant impact on fraud detection systems and should be considered during development and maintenance in test environment before system delivery on actual carrier network.

## Declaration on Generative AI

The authors have not employed any Generative AI tools.

## References


[1] V. A. Lucas, Authenticated caller ID plus regulatory changes, Document Presented to the FTC Robocall Challenge, May 2013.

[2] SoK: Fraud in Telephony Networks / M. Sahin et al. 2017 IEEE European Symposium on Security and Privacy (EuroS&P), Paris, 26–28 April 2017. 2017. doi: 10.1109/eurosp.2017.40.

[3] Macia-Fernandez G., Garcia-Teodoro P., Diaz-Verdejo J. Fraud in roaming scenarios: an overview. IEEE Wireless Communications. 2009. Vol. 16, no. 6. P. 88–94. doi: 10.1109/mwc.2009.5361183.

[4] Sahaidak V. A., Lysenko M. M., Senkov O. V., «Telecom fraud and it's impact on mobile carrier business». Connectivity, № 6(160), p. 17–20, Q4 2022. doi: 10.31673/2412-9070.2022.061720.

[5] Telecom Fraud Detection via Hawkes-enhanced Sequence Model / Y. Jiang et al. IEEE Transactions on Knowledge and Data Engineering. 2022. P. 1. doi: 10.1109/tkde.2022.3150803.

[6] Djizanne Toukem Joel, Fotsa Jounda Boris, Manfouo Kennedy Armel, Tchouatcha Deumaga Michel, Yuhala Peterson Jr. Dohbila (C), Alain Tchana, Mise en place d'un dispositif de détection des fraudeurs à la Simbox au Cameroun, Department OF Computer engineering, National advanced school of engineering, University OF Yaounde I, Année academique 2017-2018.

[7] Kehelwala Gamaralalage Dasun Chamara Kehelwala, REAL-TIME FRAUD DETECTION IN TELECOMMUNICATION NETWORK USING CALL PATTERN ANALYSIS, Dissertation submitted in partial fulfillment of the requirements for the degree Master of Science, Department of Computer Science and Engineering, University of Moratuwa, Sri Lanka, December 2017. URL: https://dl.lib.uom.lk/items/d5f9f86a-2ab3-40d9-bfce-5dc203ae31fe.

[8] GSM Association, Official Document TD.104 - Use of TAP for the Single IMSI Wholesale Billing Interface, Version 1.1, 15 May 2014.

[9] GSM Association, Official Document TD.106 - Use of NRTRDE for the Single IMSI Fraud Interface, Version 1.0, 27 November 2013.

[10] ETSI TS 132 299 V17.1.0 (2024-05), Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; 5G; Telecommunication management; Charging management; Diameter charging applications (3GPP TS 32.299 version 17.1.0 Release 17).

[11] Richard Johnson, Oracle Data Integrator Essentials: Definitive Reference for Developers and Engineers, HiTeX Press, June 20, 2025. ASIN: B0FDYS4V1T.

[12] Viktor Sahaidak, «Overview of fraud detection systems and performance KPI development». Cybersecurity: Education, Science, Technique, № 3 (23), p. 274-283, Q1 2024. doi: 10.28925/2663-4023.2024.23.274283.