“Kerberos: a real-time Fraud Detection System for IMS-enabled VoIP Networks”

S. P. Romano and S. Marseglia
spromano@unina.it
INW2016, San Candido
January 13-15 2016
Context & Contribution

Context:

- Fraud Detection Systems for Next Generation VoIP Networks
- Ongoing collaboration with tiscali:

Contribution:

- Design and implementation of a scalable architecture for:
  - on-line call monitoring
  - automated production of Call Detail Records (CDRs)
  - real-time detection of anomalous behaviors
Call monitoring

- **On-line Charging Systems (OCS):**
  - systems which monitor calls in real-time by generating charging events

- **Call termination:**
  - production of a *Call Detail Record (CDR)*
    - summary information about the call:
      - caller, callee, call duration, cost of the call, etc.

- **CDR utilization:**
  - performance analysis
  - accounting, charging, billing
  - *fraud detection*...
Offline Fraud detection

Classical approach:

- Analysis is performed only after a CDR has been generated
- Supported by widely deployed software systems like CDR-stats

A novel approach:

- Analysis performed while a call is in progress
- Made possible by Complex Event Processing (CEP) techniques

Kerberos adopts the CEP-based, on-line detection approach
Kerberos: architecture
The OCS pushes messages onto the queue. Messages are exchanged in an asynchronous fashion. Kerberos pulls messages from the queue.
Collaboration with Tiscali

- Kerberos has been designed and implemented based on specifications provided by Tiscali
- Tiscali’s OTT (Over The Top) service portfolio:
  - indoono platform (http://www.indoona.com)
    - unified access (web, mobile, PSTN, etc.) to Tiscali’s cloud-based IMS (IP Multimedia Subsystem) infrastructure
- Tiscali has provided us with:
  - the overall guidelines for the system
  - detailed information about the basic fraud templates to be taken into account
  - a rich dataset containing real-world OCS events collected on-the-field
Collaboration with Tiscali

- Tiscali’s Indoona “free calls” offer

Call and Videocall for free

With indoona you can call and video call for free your friends, from a smartphone or personal computer wherever you are. Indoona Out is the cheapest way to call landlines and mobile anywhere in the world and thanks to the promo, each month, the first 100 minutes are free to call landlines in Europe and to landlines and mobiles in USA, Canada and China, forever. See details
Collaboration with Tiscali

Upon reception of an event with `req_type = 2`, a CDR has to be produced

### Rule templates derived from system’s specification:

1. “A PSTN number is called for free by at least \([X]\) different Indoona users in the past \([Y]\) hours”
2. “A PSTN number receives at least \([X]\) minutes of free calls in the past \([Y]\) hours”
3. “An Indoona caller consumes more than \([X]\) minutes of free calls towards just \([Y]\) PSTN numbers in the past \([Z]\) hours”
4. “An Indoona caller consumes more than \([X]\)€ of non-free calls in the past \([Y]\) hours”

Kerberos also allows for the definition of customized rule patterns!
User-defined rules

- **Caller**: A
  - Free seconds constraint: > 500
- **Called**: B
  - Duration constraint: > 1000
  - Time range: 01:00:00

**Rule type**: 101

**Add rule!**
SELECT caller, sum(freeTime), count(distinct called) FROM ClosedCallEvent.win:time(1 hour) WHERE destDomain LIKE '%ngn%' GROUP BY caller
Performance testing

Long-run test:
- Application’s behavior in the long run

≈ 9 hours

Stress testing:
- Increased event reception frequency
  - 100 evt/s
  - 0.1 evt/s
  - ≈ 1 hour
- Increased number of predefined rule instances
  - 5000 regole
  - 5 regole
  - ≈ 1 hour
- Increased number of user-defined rule instances
  - 50 regole
  - 5 regole
  - ≈ 1 hour
**Long-run test**

Server’s features:
- CPU: 8 Intel(R) core(TM) i7-4770s a 3.10GHz;
- RAM: 16 GB;
- OS: Ubuntu 14.10 LTS

Traffic data provided by Tiscali:
- 562,533 events, corresponding to:
  - 119,034 calls;
  - A time frame of 7 days:
    - 01/12/2014-07/12/2014
Stress tests

0.1 req/sec, 5 predefined rules

100 req/sec, 5 user-defined rules

10 req/sec, 50 predefined rules

10 req/sec, 50 user-defined rules
Conclusions & Future Work

Conclusions:

• Design and development of a scalable architecture for the real-time processing of “call-tracking” events

• The Kerberos system is currently being tested “on stage” at Tiscali’s premises

Future work:

• Refining the ‘remediation’ component of the system:
  • implementation of the actions to be undertaken ‘after’ a fraud has been detected (black-listing, call diversion, etc.)

• Refining the test plan with the aim of arriving at a complete picture of system’s performance in a number of operational scenarios

• Making the system “production-ready”
Any questions?