Risk Scoring Engine to Detect Anomalous Service Accesses

Daisuke Mashima and Mustaque Ahamad

Motivation

- Growing use of online identity credentials
  - Passwords, certificates, SSN, etc.
  - Loss and theft due to phishing, malware, etc.
- Consequence of online identity theft
  - Impersonation
  - Disclosure of sensitive information
  - Financial loss for both users and service providers
- Many large companies rely on manual review.
  - Huge amount of log records
  - Non-real time processing

Goals

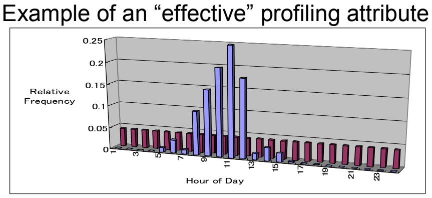
- Secure monitoring of login (service access) requests in an automated and real-time manner
- Computation of Risk Score based on suspiciousness of each access to help reduce burden on human experts
- Broad applicability by supporting general access log records

Challenges

- Limited amount of information in access logs
  - E.g., user ID, timestamp, IP address, etc.
- Limited types of events
  - Only a login event in an extreme case
- Real-time score calculation
- Reasonably high accuracy to reduce human effort

Anomaly-based Scoring

- Extract (categorical) **profiling attributes** from an individual log record
  - Timestamp (day-of–week etc.), IP address, etc.
- Construct a user profile as frequency distribution over categories of an profiling attribute
- Support multiple profiles per user
  - E.g., Day-of-week profile and hour-of-day profile etc.
- Implement data aging
  - Aiming at reducing the impact of older observations
  - Multiplying a decay factor with all frequency counts
- Calculate *Base Score* based on "**unlikeliness**" of an observed attribute value
  - *Base Score = -log (Relative Freq. of Attribute Value)*
- Determine *Weight* based on "**effectiveness**" of the corresponding profiling attribute.
  - Use "**distance**" between the frequency distribution and uniform distribution
    - Bhattacharyya Distance etc.
    - Example of an "effective" profiling attribute



- *Sub Score = Base Score * Weight*
- Aggregate *Sub Scores* to output *Risk Score*

Future Work

- Investigate other profiling attributes
  - Session duration, access frequency / interval, etc
- Implement in production environment
  - White / Black list for score adjustment
  - Interaction with human operators
- Conduct detailed experiments and evaluation
- Integrate into other security mechanisms
  - Risk-based authentication systems
  - Other fraud / intrusion detection systems

Preliminary Experiments

- Data set 1: University portal site
  - Profiling attributes:
    - Week of month, day of week, and hour of day
  - Decay factor for data aging:
    - 0 (without data aging) and 0.5 (with data aging)
- Data set 2: E-commerce company portal
  - Profiling attributes:
    - Week of month, day of week, and hour of day
    - Country, region (state), city
    - Organization name / ISP name
  - Decay factor for data aging: 0.5
- Methodology
  - Scale scores in [0,100)
  - Pick the max of sub scores
  - Determine thresholds based on past scores
- False positive / True positive for Data set 1

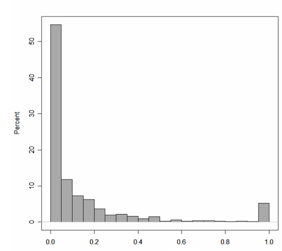
Table 1: Results of False Positive / True Positive Evaluation (without data aging)

User	Access Freq. [per Month]	Tentative Threshold	FP Rate	TP Rate
A	4	40	16.6 %	96.0 %
B	25	20	0.0 %	90.0 %
C	62	30	0.0 %	90.0 %
D	172	20	18.6 %	100.0 %

Table 2: Results of False Positive / True Positive Evaluation (with data aging)

User	Access Freq. [per Month]	Tentative Threshold	FP Rate	TP Rate
A	4	40	61.0 %	90.0 %
B	25	30	1.0 %	90.0 %
C	62	40	6.0 %	90.0 %
D	172	30	3.0 %	100.0 %

- False positive rate distribution for Data set 2



- 9,500 users
- Mean: 0.14
- Std dev.:0.25
- 80 percentile: 0.20
- 90 percentile: 0.43

Integration of Domain Knowledge

- Rule-based scoring module
  - Define scoring criteria tailored for each domain and setting, such as
    - Consecutive login failure
    - Simultaneous login with distant location
    - Speed contradiction
      - Access interval against distance moved
- Rule-based scores can be combined with anomaly-based scores.
  - Sum, max, weighted average etc.