Online and adaptive detection of web attacks

Wei Wang, Thomas Guyet, René Quiniou, Marie-Odile Cordier,
Florent Masseglia, Brigitte Trousse

INRIA Sophia Antipolis et
IRISA

Motivation

- A web attacks is one of major threat in current computer networks
  - With over 70% of attacks now carried out over the web application level
- Online detection
  - Unsupervised: no need of labeled data
  - Adaptive: automatically labeled data
- Adaptive detection
  - Deal with concept drift problem
  - Continually update the model

Key Components of Anomaly Intrusion Detection Model

- Data
  - What kind of data used – target to protect
    - Network behaviors - Network traffic, …
    - User behaviors – command history; keystroke; http logs, …
    - Program behaviors – system calls, …
  - What kind of features extracted from the data -How to do data preparation
- Methods (statistic, machine learning, data mining,…)
  - Supervised: precise labeled data is required
  - Unsupervised: low detection accuracy

Data

- Http log data from INRIA Sophia Antipolis
  - Original size: 1.536GB
  - N. of request: 5,700,949
  - Duration: 10 days and 21 hours 26 mins
    - From 01/01/2007 00:00:14
    - To 11/02/2007 21:26:44
- Data filtering
  - Filtered the search robot (e.g., google, msn,…)
  - Filtered most of static request
    - File htm, jpg, gif, pdf, doc, ppt, ico,…
  - Size after filtering:
    - N. of request: 265,717
      - Only remain 4.66% of the original requests

Data Preprocessing

- Original data form

  60.50.99.87 - - [02/Jan/2007:07:47:03 +0100] "GET /acacia/project/edccaeteras/wakka.php?wiki=ActionOrphanedPages/referrers HTTP/1.0" 200 10753 "http://a.js1.bosja.com/index1.htm" "-"

- Computer the character distribution of the request path source
  - Only consider the request source
  - Only compute the distribution of ASCII 33-127
  - Each request is thus represented by a vector with 95 dimensions
  - Vector form after transformation example): 0 0 0 0 0 0 0 0 0 0 0 0 0 1 5 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 9 0 6 2 9 1 1 2 4 1 3 0 0 2 2 4 0 7 3 3 0 0 2 0 0 0 0 0 0 0 0
  - Online Classification is based on the vectors

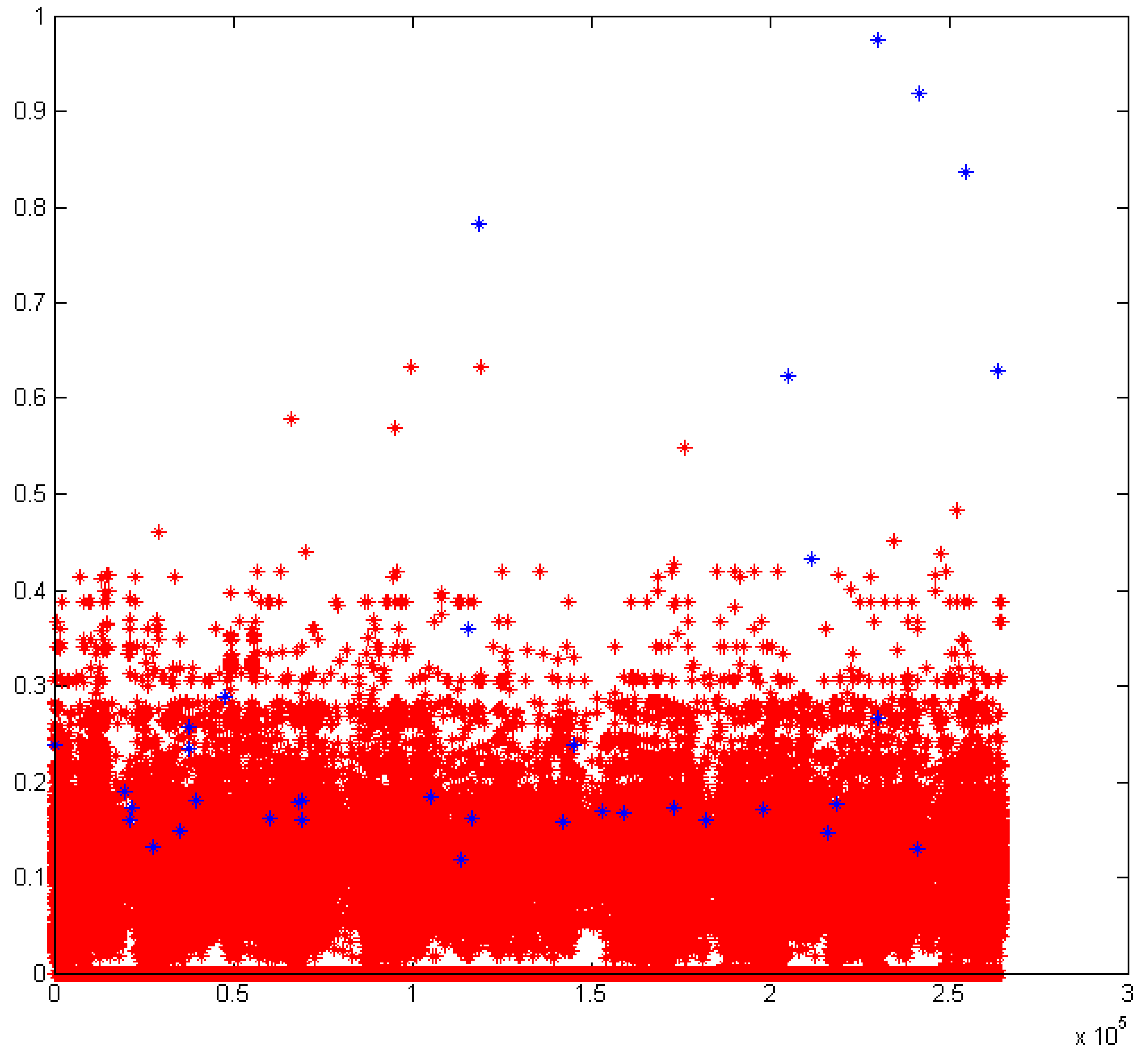
Data labeling (manually)

- Find only 1 attack line
- Add 35 attacks downloaded from website (http://www.i-pi.com/HTTP-attacks-JoCN-2006/)
  - 35 attacks in the same http server (apache) and in the same systems (Linux/Unix)
  - Different kinds of attacks represented different attack situations
    - URL decoding error
    - Buffer overflow
    - Poor memory management
    - Signed interpretation of unsigned value
    - …

Classification (traditional methods)

- Anomaly detection (k-NN)
  - Select the first 800 requests as references (base)
    - Only used normal data (labeled data)
  - Compute distances between each coming request and all the first 800 requests
  - Select the minimal distance as the *anomaly index*

Classification results (Anomaly Detection with anomaly index k-NN)



Classification results(kNN)

- Supervised
- Static model
- Results
  - Detection rates and False positive rates

Threshold	Detection Rate (%)	False position rate (%)
0.200	14/36=38.8	3510/264916=1.3
0.160	80	3.8
0.119	97.2	14.5

Use AP for adaptive online and unsupervised detection

- AP (Affinity Propagation)
  - No need to define how many clusters
    - K-means, k-clusters need to define
  - Can find some representative vectors to represent the cluster center
    - Reference data becomes smaller
  - Suitable for IDS
    - No need to have labeled data
      - Labeled data is very difficult to get

ONE Assumption and THREE states

- Three state of the data points
  - Normal
  - Uncertain
  - Attack
- One assumption: normal data is very large while attack data is rare in practice
  - Identify small size of clusters as uncertain
  - Re-cluster after if a change is detected
    - If the uncertain events remains then uncertain is changed as Attacks
    - Otherwise the uncertain events changed as normal
  - Data streaming environments

Detection Model (first stage Initial)

time

Clustering

Initial model

(e_i, n_i, ui)

If ni< t1 or

ui > t2

Uncertain anomaly detected and all these points will be put into a temporary reservoir

Ei: representative vector

Ni: no. of vector that points to ei

ui : mean distance between vectors and ei

Detection Model (detection stage)

time

Current point **X**

Clustering

Initial model

$(e_i, n_i, ui\quad)$

f=min ({ $d^2(e_i, x)$ })

$$e^* = \arg\min(d^2(e_i, x))$$

If f < threshold, upgrade the model with weight

Else uncertain detected and put f into a temporary buffer

Initial clustering

Detection Model (upgrade)

time

Clustering

Initial model

(e_i,n_i,ui)

Initial clustering

Current point **X**

f=min ({ $d^2(e_i, x)$ })

$$e^* = \arg\min(d^2(e_i, x))$$

<u>Model incremental upgrade</u>

e_i: keep it

n_i: ni+1

$$\mu_i = \frac{\mu_i \times n_i + d(e_t, e_i)}{n_i + 1}$$

Not really upgrade, but compute the weights preparing for model rebuild

Model rebuild

change

Detection Model (second stage)

- Change point detection
  - If # of coming uncertain events exceeds a threshold (e.g., 200)
  - Or if a time period passed (e.g., 2000 points)
- Rebuild the model if a change point is detected

Detection Model (decay)

time

Clustering

Initial model

(e_i, n_i, Σ_i)

Initial clustering

Current point **X**

f=min ({ $d^2(e_i, x)$ })

$$e^* = \arg\min(d^2(e_i, x))$$

Model incremental upgrade

e_i: keep it

n_i: ni+1

$$\mu_i = \frac{\mu_i \times n_i + d(e_t, e_i)}{n_i + 1}$$

Model rebuild

change

If one cluster that has not been visited in,

e.g., 2000 points, it is forgotten

Flow:

Pseudo code of AODIS
Audit data stream $e_1, e_2, \ldots, e_t, \ldots$; **fit threshold** $N_{cluster}$, $D_{cluster}$, ε, $N_{outlier}$.
Clustering ($e_1, e_2, \ldots, e_t, \ldots e_T$) with AP
Reservoir={}
If $n_i \leq N_{cluster}$ or $\mu_i \geq D_{cluster}$
Reservoir $\leftarrow e_t$
$r = 0;\ t_r = T$
For $t > T$ **do**
Compute e_i = nearest exemplar to e_t
If $d(e_t, e_i) < \varepsilon$
Update the model
Else
Reservoir $\leftarrow e_t$
End if
If Restart criterion then
Rebuild the model
$r = r+1;\ t_r = t;$
Consider Reservoir
For $t < t_{r-1}$
If e_t is a exemplar
If $n_i \leq N_{cluster}$ or $\mu_i \geq D_{cluster}$ **then**
e_t is an attack
Else Reservoir $\leftarrow e_t$
End for

Results with AP

- Results (comparison)

Threshold	Detection Rate (%)	False position rate (%)
0.200	14/36=38.8	3510/264916=1.3
0.160	80	3.8
0.119	97.2	14.5

Detection Rate (%)	False position rate (%)
44.4	0.47
86.11	2.86
100	5.62

Paramètres for detection rate 100%

Clustersize=1,meandis=0.06974,

Uncertain=300, time=2000,

forget=2000, p0=0,05*

Work in progress

- In the following week
  - Paper to PAKDD'2009 (deadline is approaching…)
- In the following month
  - General framework for adaptive intrusion detection
  - Other clustering methods
  - Solve frequent attacks detection?
  - Paper to SDM' 2009?
- In the following year?
  - Generally improve the data preprocessing methods
  - Character distribution is effective but not enough
    - LCS distance?
  - The data?
  - Parameters?
  - Practical use?
  - …

Suggestions et Questions

Merci