Collaborative Anomaly Detection Framework for handling Big Data of Cloud Computing

Nour Moustafa, Gideon Creech, Elena Sitnikova, Marwa Keshk

MilCIS conference, Nov 2017
Contribution

• With the ubiquitous computing of providing services and applications at anywhere and anytime, cloud computing is the best option as it offers flexible and pay-per-use based services to its customers.

• Nevertheless, security and privacy are the main challenges to its success due to its dynamic and distributed architecture, resulting in generating big data that should be carefully analysed for detecting network’s vulnerabilities.

• We propose a Collaborative Anomaly Detection Framework (CADF) for detecting cyber attacks from cloud computing environments. We provide the technical functions and deployment of the framework to illustrate its methodology of implementation and installation.
Cloud computing and Big data

- The term ‘cloud computing’ denotes a network of networks interconnected using internet services in which virtual shared servers offer the software, infrastructure, platform, services and other resources to customers anywhere and anytime.
Cloud computing and Big data (cont.)

Cloud Clients:
- Thin Client
- Terminal Emulator
- Web Browser
- Mobile App
- Etc.

IAAS:
- Virtual Machines
- Servers
- Network
- Storage
- Load Balancers

PAAS:
- Database
- Web Server
- Dev Tools
- Execution Runtime

SAAS:
- Email
- CRM
- Virtual Desktop
- Communications
- Gaming
Cloud computing and Big data (cont.)

- Because of the dynamic configurations of cloud computing, numerous vulnerabilities attempt to penetrate its architecture, leaving loopholes in which attackers exploit cloud’s services and its big data.
- The analysis of cloud data should consider the inspection of big data properties, i.e., volume, velocity, variety, veracity, for efficiently detecting malicious activities.
Intrusion Detection system (IDS)

• An IDS is widely used to detect intrusive activities from cloud’s big data, but it still faces the challenge of successfully recognizing zero-day/new malicious activities.

• The IDS detection approaches are classified into three categories: misuse- (MDS), anomaly- (ADS) and hybrid-based IDS, merging the first two types.
Proposed Collaborative Anomaly Detection Framework (CADF)
Proposed Collaborative Anomaly Detection Framework (CADF) (Cont.)

• **Capturing and logging module**
  – This module includes the steps of sniffing network data and storing them to be processed by the DE technique, like the steps of designing the UNSW-NB15 dataset.

• **Pre-processing module**
  – **feature conversion** replaces non-numeric features with numeric ones.
  – **feature reduction** uses the PCA technique to select a small number of uncorrelated features.
  – **feature normalisation** arranges the value of each feature in a certain range to remove any bias from raw data and easily process it. We apply the z-score function.
Finite Mixture Model using Gaussian distribution

- As a finite mixture model is defined as a convex combination of two or more Probability Density Functions (PDFs), the joint properties of these functions can approximate any arbitrary distribution.
Algorithm 1 Generation of normal profile in training phase.

**Input:** normal vectors \( r_{1:n}^{normal} \)

**Output:** normal profile \( \text{pro} \)

1. **for** (record in \( r_{1:n}^{normal} \) **do**
2. estimate the parameters \((\alpha, \mu, \delta)\) of the GMM
3. compute the PDFs using equations (2) to (6) based on the parameters estimated in Step 2
4. **end for**
5. calculate lower = quartile (PDFs, 1)
6. calculate upper = quartile (PDFs, 3)
7. calculate IQR = upper - lower
8. \( \text{pro} \leftarrow \{ \theta = (\alpha, \mu, \delta), (\text{lower-upper IQR}) \} \)
9. **return** \( \text{pro} \)
Algorithm 2 Testing phase and decision-making method.

**Input:** observed record \( r^{testing} \), normal profile (pro) \( \{ \theta = (\alpha, \mu, \delta), \text{(lower-upper IQR)} \} \)

**Output:** normal or abnormal record

1: compute \( PDF^{testing} \) using equations 2 to 6 with parameters \( \theta = (\alpha, \mu, \delta) \)

2: if \( PDF^{testing} < (\text{lower} - w.IQR) \) \( \mid \) \( PDF^{testing} > (\text{upper} + w.IQR) \) then

3: arrack

4: else

5: normal

6: end if
Deployment of Proposed framework for Cloud Computing environments

- The deployment of this framework is described for three nodes (A, B and C) depicted in Figure 3 in order to be executed for cloud computing systems.

- Unlike traditional IDSs, the CADF is deployed on each network node and each CADF connected simultaneously with the shared module of capturing and logging.
The evaluation of the proposed framework is conducted using the UNSW-NB15 dataset which has a hybrid of authentic contemporary normal and attack vectors.

Table I. FEATURES SELECTED FROM UNSW-NB15 DATASET using The PCA

<table>
<thead>
<tr>
<th>Feature set</th>
</tr>
</thead>
<tbody>
<tr>
<td>ct_dst_sport_ltm, tcprrt, dwin, ct_src_dport_ltm, ct_dst Src_ltm, ct_dst_ltm,</td>
</tr>
<tr>
<td>smean, dmean, service, proto</td>
</tr>
</tbody>
</table>
Experimental Results

<table>
<thead>
<tr>
<th>w value</th>
<th>DR</th>
<th>Accuracy</th>
<th>FPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.5</td>
<td>86.3%</td>
<td>88.2%</td>
<td>8.4%</td>
</tr>
<tr>
<td>2</td>
<td>89.1%</td>
<td>90.1%</td>
<td>5.5%</td>
</tr>
<tr>
<td>2.5</td>
<td>93.4%</td>
<td>94.8%</td>
<td>4.4%</td>
</tr>
<tr>
<td>3</td>
<td>95.6%</td>
<td>96.7%</td>
<td>3.5%</td>
</tr>
</tbody>
</table>

UNSW-NB15 dataset

![Graph showing detection rate vs false positive rate for different w values]
## Experimental Results (Cont.)

<table>
<thead>
<tr>
<th>Technique</th>
<th>DR</th>
<th>FPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>TANN (Zhiyuan et al.)</td>
<td>88.2%</td>
<td>12.3%</td>
</tr>
<tr>
<td>EDM (Tan et al.)</td>
<td>89.4%</td>
<td>10.6%</td>
</tr>
<tr>
<td>MCA (Tsai et al.)</td>
<td>91.4%</td>
<td>8.9%</td>
</tr>
<tr>
<td>Proposed CADF</td>
<td>95.6%</td>
<td>3.5%</td>
</tr>
</tbody>
</table>
Experimental Results (Cont.)

- The key reasons for the CADF performing better than the other peer techniques discussed below are that the GMM can perfectly fit the boundaries of each feature as it accurately estimates the mixing weights of network features in order to model normal data.

- Moreover, the lower-upper IQR method can successfully specify the boundaries between normal and outlier observations.

- Since the shared module as SaaS collects important network observations from different network nodes, the DE as SaaS does not consume high processing time to inspect the observations, either normal or attacks, for these nodes.
Reference


Thanks for your attention!

Any questions?