Include Security Monitoring Terms into SLA

The SLA verification phase

Amir Teshome, Louis Rilling, Christine Morin

September 7, 2017

1Inria, 2DGA
Context & Objective
• Clients outsource their information system
IaaS Cloud

- Clients outsource their information system
- Loss of full control
  - Trust issue
- Clients outsource their information system
- Loss of full control
  - Trust issue
- Service providers, guarantee through SLA
- Clients outsource their information system
- Loss of full control
  - Trust issue
- Service providers, guarantee through SLA
- *Security monitoring* not included
Security Monitoring

Definition

Security Monitoring is the collection, analysis, and escalation of indications and warnings to detect and respond to intrusions.

1 "Tao of Network Security Monitoring, Beyond Intrusion Detection" by Richard Bejtlich
## Definition

Security Monitoring is the collection, analysis, and escalation of indications and warnings to detect and respond to intrusions.

- Detect suspicious behaviors and take action before severe damage

---

1. "Tao of Network Security Monitoring, Beyond Intrusion Detection" by Richard Bejtlich
Security Monitoring

Definition

Security Monitoring is the collection, analysis, and escalation of indications and warnings to detect and respond to intrusions.

- Detect suspicious behaviors and take action before severe damage
- Intrusion Detection Systems (IDS) and logs from firewalls are used as monitoring systems

1 “Tao of Network Security Monitoring, Beyond Intrusion Detection” by Richard Bejtlich
Example of Monitoring Device

- Example: *Intrusion Detection System*

Network IDS (NIDS) connected through mirror port
Service Level Agreement (SLA)

SLA: An agreement between cloud providers and customers

It describes:
Service Level Agreement (SLA)

SLA: An agreement between cloud providers and customers

It describes:

- Provided service
Service Level Agreement (SLA)

SLA: An agreement between cloud providers and customers

It describes:

- Provided service
- Rights and obligations
Service Level Agreement (SLA)

SLA: An agreement between cloud providers and customers

It describes:

- Provided service
- Rights and obligations
- Penalties
SLA: An agreement between cloud providers and customers

It describes:

- Provided service
- Rights and obligations
- Penalties

To our knowledge it **doesn’t include** the security monitoring aspect of an information system
Goal: Including security monitoring in Service Level Agreements
Challenges

Goal: Including security monitoring in Service Level Agreements

- Lack of standard to express security monitoring properties
Goal: Including security monitoring in Service Level Agreements

- Lack of standard to express security monitoring properties
- SLA enforcement is done at low level
Challenges

**Goal: Including security monitoring in Service Level Agreements**

- Lack of standard to express security monitoring properties
- SLA enforcement is done at low level
- Malleability of virtual infrastructures (elasticity and migration)
Challenges

Goal: Including security monitoring in Service Level Agreements

- Lack of standard to express security monitoring properties
- SLA enforcement is done at low level
- Malleability of virtual infrastructures (elasticity and migration)
- Lack of methods to evaluate security monitoring setups
Challenges

Goal: Including security monitoring in Service Level Agreements

- Lack of standard to express security monitoring properties
- SLA enforcement is done at low level
- Malleability of virtual infrastructures (elasticity and migration)
- Lack of methods to evaluate security monitoring setups
Proposed Directions

Definition and Negotiation

- Tenant
- Provider
- SLA
- services
- Requirements

1. Specify security monitoring requirements/services
Proposed Directions

Definition and Negotiation

Tenant
Provider
SLA
Deployservices
Requirements

1. Automatically configure and deploy
- Verify if the specified SLA is respected or not
Proposed Directions

- Definition and Negotiation
- Enforcement
- Verification
- Tenant
- Provider
- SLA
- Deploy
- OK?
- NO
- Take action
- Verify
- services
- Requirements

- If not take action
Proposed Directions

- Definition and Negotiation
- Enforcement
- Verification

Tenant
Provider
SLA
Deploy
OK?
NO
Take action

Verifiication and Metrics
• Focus on a specific monitoring probe (NIDS)

Network IDS (NIDS) connected through mirror port

IDS

Alerts

Host 1

Host 2

VM1

VM2

vSwitch

vSwitch

mirrored traffic

Network IDS (NIDS) connected through mirror port
Monitoring Probe

- Focus on a specific monitoring probe (NIDS)

Network IDS (NIDS) connected through mirror port

- Evaluation method for an NIDS configuration, used for the verification of compliance to an SLA
State of the art in NIDS Configuration Evaluation

[T. Probst et al] two phase approach:

1. Analysis of network access control phase
2. IDS evaluation phase

Approche d’audit automatisé
3 phases
2. Détermination des accessibilités par différentes méthodes et analyse des déviances dans les résultats.
3. Exécution de campagnes d’attaque et analyse de la réaction des NIDS.

Cloning is required & the cloned infrastructure is evaluated on basic metrics (TPR, FPR)

$$\text{True Positive Rate} = \frac{TP}{TP + FN}$$
$$\text{FP Rate} = \frac{FP}{FP + TN}$$
[T. Probst et al] two phase approach:

- Analysis of network access control phase
- IDS evaluation phase

Clone the infrastructure

- Analyse statique
- Analyse dynamique
- Analyse configurées
- Accessibilités observées
- Accessibilités définies

Analysis of network access control

- Analyse des déviances
- Automates
- Dictionnaire d’attaques
- Exécution de campagnes d’attaque
- Résultats des campagnes
- Calcul des définitions

NIDS evaluation

- Alertes des NIDS
- Rapport
[T. Probst et al] two phase approach:

- Analysis of network access control phase
- IDS evaluation phase

- Cloning is required & the cloned infrastructure is evaluated
State of the art in NIDS Configuration Evaluation

[T. Probst et al] two phase approach:
- Analysis of network access control phase
- IDS evaluation phase

- Cloning is required & the cloned infrastructure is evaluated
- Uses only basic metrics (TPR, FPR)
  - True Positive Rate $= \frac{TP}{TP+FN}$, FP Rate $= \frac{FP}{FP+TN}$
NIDS Evaluation
Evaluation Metrics

- Basic metrics: TPR and FPR,
  - other metrics were proposed based on these

$P(I)$: Base Rate

$C_{ID}$: Intrusion Detection Capability

$^{11}$: Presence of Intrusion
Evaluation Metrics

- Basic metrics: TPR and FPR,
  - other metrics were proposed based on these
- Issues with most NIDS evaluation metrics
  - Lack of a single unified metric
    - e.g. TPR and FPR

\[ \text{Base Rate (B)} = \frac{1}{P(I)} \]

\( P(I) \): Presence of Intrusion
Evaluation Metrics

- Basic metrics: TPR and FPR,
  - other metrics were proposed based on these
- Issues with most NIDS evaluation metrics
  - Lack of a single unified metric
    - e.g. TPR and FPR
  - Not taking base rate into account
    - Base Rate (B) = \( P(I) \), rate of occurrence of an attack

\[1\]: Presence of Intrusion
Evaluation Metrics

- Basic metrics: TPR and FPR,
  - other metrics where proposed based these

- Issues with most NIDS evaluation metrics
  - Lack of a single unified metrics
    - e.g. TPR and FPR
  - Not taking base rate into account
    - Base Rate (B) = \( P(I) \), rate of occurrence of an attack

- Single unified metric: *Intrusion Detection Capability* (*C*\(_{ID} \))
  \[
  C_{ID} = \frac{I(i;a)}{H(i)}
  \]
  *I*: Mutual Information & *H*: Entropy
  *i*: input packets: part of attack or legitimate packet
  *a*: IDS output: detected as intrusive or nonintrusive

\(^{1}\): Presence of Intrusion
Evaluation Metrics

- Basic metrics: TPR and FPR,
  - other metrics where proposed based these
- Issues with most NIDS evaluation metrics
  - Lack of a single unified metrics
    - e.g. TPR and FPR
  - Not taking base rate into account
    - Base Rate (B) = $P(I)^1$, rate of occurrence of an attack

- Single unified metric: Intrusion Detection Capability ($C_{ID}$)

$$C_{ID} = \frac{I(i;a)}{H(i)}$$

$I$: Mutual Information & $H$: Entropy

$i$: input packets: part of attack or legitimate packet
$a = IDS$ output: detected as intrusive or nonintrusive

$^1I$: Presence of Intrusion
Attack Running Method

Controller

Host Node

NIDS

VM1

VM2

VM3

mirrored traffic

IDS Node

vSwitch

vSwitch

vSwitch

Controller

Host Node

NIDS

VM1 VM2 VM3

mirrored traffic

IDS Node

vSwitch
Attack Running Method

- VM1
- VM2
- VM3
- Attacker
- mirrored traffic
- IDS Node
- Host Node
- Controller
- vSwitch
- NIDS
- vSwitch
- IDS Node
Attack Running Method

- Controller
- Host Node
- NIDS
- VM1, VM2, VM3
- Mirrored traffic
- IDS Node
- vSwitch
- Target VM
- Attack Injector
- Target VM
Attack Running Method

Controller
Host Node
NIDS
VM1 VM2 VM3
mirrored traffic
IDS Node
vSwitch
vSwitch
vSwitch
Attack Injector
Target VM
Attack Running Method

- Controller
- Host Node
- NIDS
- VM1
- VM2
- VM3
- Target VM
- mirrored traffic
- IDS Node
- vSwitch
- vSwitch
- vSwitch
- Attack Injector
- Target VM
- logging
Packet Injection Algorithm

- Set of running services with applications providing the service

<table>
<thead>
<tr>
<th>Application</th>
<th>Version</th>
<th>Attack Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache</td>
<td>Apache/2.4.7 (Ubuntu)</td>
<td>Denial of Service (DOS), Port Scanning</td>
</tr>
<tr>
<td>Mysql</td>
<td>14.14 Distrib 5.6.31</td>
<td>Brute Force</td>
</tr>
<tr>
<td>WordPress</td>
<td>V. 4.4.5</td>
<td>Base for other apps</td>
</tr>
<tr>
<td>Instalinker (WordPress plugin)</td>
<td>V. 1.1.1</td>
<td>Cross site scripting (XSS)</td>
</tr>
<tr>
<td>Custom Contact Forms</td>
<td>V. 5.1.0.2</td>
<td>Sql Injection (SQLi)</td>
</tr>
</tbody>
</table>
Packet Injection Algorithm

- Set of running services with applications providing the service

<table>
<thead>
<tr>
<th>Application</th>
<th>Version</th>
<th>Attack Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache</td>
<td>Apache/2.4.7 (Ubuntu)</td>
<td>Denial of Service (DOS), Port Scanning</td>
</tr>
<tr>
<td>Mysql</td>
<td>14.14 Distrib 5.6.31</td>
<td>Brute Force</td>
</tr>
<tr>
<td>WordPress</td>
<td>V. 4.4.5</td>
<td>Base for other apps</td>
</tr>
<tr>
<td>Instalinker (WordPress plugin)</td>
<td>V. 1.1.1</td>
<td>Cross site scripting (XSS)</td>
</tr>
<tr>
<td>Custom Contact Forms</td>
<td>V. 5.1.0.2</td>
<td>Sql Injection (SQLi)</td>
</tr>
</tbody>
</table>

- Set of legitimate requests and attacks with #packets sent for each of them
Packet Injection Algorithm

- Set of running services with applications providing the service

<table>
<thead>
<tr>
<th>Application</th>
<th>Version</th>
<th>Attack Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache</td>
<td>Apache/2.4.7 (Ubuntu)</td>
<td>Denial of Service (DOS), Port Scanning</td>
</tr>
<tr>
<td>Mysql</td>
<td>14.14 Distrib 5.6.31</td>
<td>Brute Force</td>
</tr>
<tr>
<td>WordPress</td>
<td>V. 4.4.5</td>
<td>Base for other apps</td>
</tr>
<tr>
<td>Instalinker (WordPress plugin)</td>
<td>V. 1.1.1</td>
<td>Cross site scripting (XSS)</td>
</tr>
<tr>
<td>Custom Contact Forms</td>
<td>V. 5.1.0.2</td>
<td>Sql Injection (SQLi)</td>
</tr>
</tbody>
</table>

- Set of legitimate requests and attacks with \( \# \) packets sent for each of them
- Base rate (\( B \), picked from observed statistics)
  - In practice its very small \( (10^{-2} - 10^{-6}) \)
Packet Injection Algorithm

- Attacks are interlaced with legitimate requests
Packet Injection Algorithm

- Attacks are interlaced with legitimate requests
- Two processes running in parallel (legitimate requests and attacks)
Packet Injection Algorithm

- Attacks are interlaced with legitimate requests
- Two processes running in parallel (legitimate requests and attacks)
- Legitimate requests follow some probability distribution
• Attacks are interlaced with legitimate requests
• Two processes running in parallel (legitimate requests and attacks)
• Legitimate requests follow some probability distribution
• Attacks injected with average number of packets sent matching the chosen base rate
• Metrics computation process

19:39:03.684834 IP 10.0.0.3.35396 > 10.0.0.5.80:......
19:39:03.684834 IP 10.0.0.3.35396 > 10.0.0.5.80:.......
19:39:03.684834 IP 10.0.0.3.35396 > 10.0.0.5.80:.....
Overhead on production environment

- Only network overhead
Overhead on production environment

- Only network overhead
- Computation could be performed outside the production environment
Experimental Evaluation of the Method
• Services, legitimate requests and attacks with number of packets

<table>
<thead>
<tr>
<th>Application</th>
<th>Version</th>
<th>Attack Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache</td>
<td>Apache/2.4.7 (Ubuntu)</td>
<td>Denial of Service (DOS), Port Scanning</td>
</tr>
<tr>
<td>Mysql</td>
<td>14.14 Distrib 5.6.31</td>
<td>Brute Force</td>
</tr>
<tr>
<td>WordPress</td>
<td>V. 4.4.5</td>
<td>Base for other apps</td>
</tr>
<tr>
<td>Instalinker (WordPress plugin)</td>
<td>V. 1.1.1</td>
<td>Cross site scripting (XSS)</td>
</tr>
<tr>
<td>Custom Contact Forms</td>
<td>V. 5.1.0.2</td>
<td>Sql Injection (SQLi)</td>
</tr>
</tbody>
</table>

**Legitimate Requests**

<table>
<thead>
<tr>
<th>Name</th>
<th>View site</th>
<th>WP login</th>
<th>Mysql login</th>
<th>Upload file</th>
</tr>
</thead>
<tbody>
<tr>
<td># of packets</td>
<td>6</td>
<td>12</td>
<td>5</td>
<td>3111</td>
</tr>
</tbody>
</table>

**Attacks**

<table>
<thead>
<tr>
<th>Name</th>
<th>DOS</th>
<th>Brute Force</th>
<th>XSS</th>
<th>SQLi</th>
<th>Port Scanning</th>
</tr>
</thead>
<tbody>
<tr>
<td># of packets</td>
<td>1916</td>
<td>2000</td>
<td>6</td>
<td>25</td>
<td>1012</td>
</tr>
</tbody>
</table>

• NIDS rules to detect the attacks
Setup

OpenStack on Grid'5000

VM1 VM2 VM3 Target VM

Open vSwitch (OvS)

Compute Node 1

Cloud Controller

Snort Node

Snort

OvS

logging

mirrored traffic

Attack Injector
Network Overhead

- Base rate $B = 10^{-2}$
Network Overhead

- Base rate $B = 10^{-2}$
Network Overhead

- Base rate $B = 10^{-2}$

![Graph showing network overhead with different response times and actions like WP login, SQL login, View Blog, and Upload File at verification points t_0, t_1, t_2, and t_3.]
Network Overhead

- Base rate $B = 10^{-2}$
Network Overhead

max: 6.9% for “SQL Login” and min: 2.2% for “WP Login”
Trade-off between time required for verification and overhead on the production environment
Conclusion & Future Work
Conclusion

- Studied different NIDS evaluation metrics
Conclusion

- Studied different NIDS evaluation metrics
- Proposed an in situ SLA verification mechanism for NIDS
  - Using attack injection without damaging the production VMs
  - Dynamically respecting a given base rate
Conclusion

- Studied different NIDS evaluation metrics
- Proposed an in situ SLA verification mechanism for NIDS
  - Using attack injection without damaging the production VMs
  - Dynamically respecting a given base rate
- Experimental evaluation
  - shows reasonable overhead (less than 10%)
Future Work

- Definition of security monitoring SLA and

```
Client

Provider

SLA

Deploy

Requirements

Services

Verify

Take Action

OK ?

NO
```

Diagram: Flowchart showing interaction between Client and Provider with SLA, Deploy, Verify, and Take Action steps.
Future Work

- Definition of security monitoring SLA and
- Enforcement of security monitoring SLA terms
Future Work

- Definition of security monitoring SLA and
- Enforcement of security monitoring SLA terms
- Extend this work to other monitoring probes