Defining a simple metric for real-time security level evaluation of multi-sites networks

A. K. Ganame and J. Bourgeois
LIFC, University of Franche-Comte
Montbeliard, France
{Julien.Bourgeois,Abdoul.Ganame}@univ-fcomte.fr
http://rma.pu-pm.univ-fcomte.fr

Abstract

In previous research work, we have developed a centralized Security Operation Center (SOC) [2] and a distributed SOC [4]. These environments are very useful to react to intrusions or to analyze security problem because they provide a global view of the network without adding any kinds of software on network components. They therefore lack the possibility to have a real-time metric which measures the security health of the different sites. The idea is to have, in one look, an indication of the security level of all the sites of the network. In this article, we propose to define such a metric which gives the user 3 states for a given network.

1 Introduction

In our last research works we have presented a centralized Security Operation Center (SOC) called SOCBox [2] which is commercialized by IV2 Technologies and its evaluation [5] [3]. Based on this centralized SOC, we have designed and developed a distributed SOC named DSOC[1][4], for intrusion detection on multi-sites networks. DSOC will be commercialized in a few months by a new company.

With DSOC, in any network site, a local detection engine analyzes data collected by one or several collection boxes to find intrusion patterns. Afterward, all the generated alerts are processed by a global intrusion detection engine to find more complex intrusions and to give a global view of the security of the network. This leads to generate multi-views reports for the different categories of people who manage the network. The idea of our SOCs was to give a global view of the network and to simplifying the security management by eliminating non pertinent security alerts. The idea of this article is to push this concept a bit further and to provide to security managers the most simple control panel as possible. Indeed, managing several sites requires to be aware of too many information, so simplify the information is necessary to be understandable by users. But this simplification must not hide important events in the network and that is the complicated point that we have to deal with: simplifying without missing important events.

The rest of the paper is structured as follows: Section 2 defines the security level metric and in section 3, the decision making process. In section 4, we present an evaluation of our system which is followed by a conclusion.

2 Security level evaluation of multi-sites networks

2.1 Objectives

The aim of the security level evaluation of multi-sites network is to give an understandable real-time security level evaluation of the different sites of a network. This will allow the network security manager to react as fast as possible to a threat by understanding quickly that it is occurring.

To do so, a graphical dashboard represents all the sites of a network and their security level. This security level represents the vulnerability state of site due to threat activities that are may occur. A color is defined for each security level:

[1]This project was partially funded by a CAPM, CRFC, government and EU programme under the STIC pole.

- *Green*: there is no threat occurring in the network or the threats which occur cannot open security holes into the network like script kiddies activities. No action is required for this level of security.
- *Orange*: threats are occurring in the network, they are not critical at this time, but if they succeed they can lead to security holes. The security of the site has to be reinforced in order to stop these intrusions.
- *Red*: intrusions are in progress and they can lead to critical security problems. If they succeed, the impact on integrity, confidentiality and stability of the information system could be very high. Immediate countermeasures have to be taken to stop these threats.

According to the malicious activities at each site, site indicators go from green to orange and then to red. As alarms are associated with the critical level, this allows the security administrator to be very reactive in managing security incidents. The logical consequence of this increased responsiveness is that information system is kept in a stable and coherent state. When an attack like DOS is detected in a site, if the source of this attack is blocked in time, total paralysis of the entire network is avoided.

2.2 Defining metrics for security level evaluation

Let X_i, Y_j and Z_k be respectively a site of a multi-sites network of n_x sites ($n \geq 1$), a sensor and a security alert and:

$S^t_{(X_i)}$: Reprensents the security level evaluation of the site X_i at time t. It is the main metric. $S^t_{(X_i)} \in \{R, O, G\}$, R=Red, O=Orange and G=Green.

$Ssl^t_{(X_i)}$: Represents the static security level of site X_i at time t.

$Rsl^t_{(x)}$: Represents the real security level of x at time t, where x can be either a site or a sensor.

$Asl^t_{(X_i)}$: Represents the approximate security level of site X_i at time t.

$C_{(Z_k,Y_j)}$: Represents the criticity level of the security alert Z_k on sensor Y_j, with $k \in [0...n_z]$, $\forall Y_j \in X_i$ and $C_{(Z_k,Y_j)} \in \{L, M, H\}$. L, M and H are respectively Low (0), Medium (1) and High (2).

$Esl_{(x)}$: Represents the expected security level of x, where x can be either a site or a sensor.

$A_{(Y_j,X_i)}$: Represents the capacity for the sensor Y_j to access to the others elements of the site X_i, with $Y_j \in X_i$.

$A_{(Y_j,X_l)}$: Represents the capacity for the sensor Y_j to access to the others elements of the site X_l, with $Y_j \in X_i$ and $\forall l, X_l \neq X_i$

The evaluation of the security level of a multi-sites network is composed of three phases.

In a first phase, a static security level of X_i noted $Ssl_{(X_i)}$ is realized, $\forall i \in [1...n_x]$.

The second phase is composed of two operations which take place simultaneously on all the supervised networks ($\forall i \in [1...n_x]$). They are repeated according to time t parameter (t, $t+1$, etc.)

- *Real security level evaluation of X_i*: At each time t, the local analyzer of X_i, LA_i evaluates the real security level of X_i, that is noted $Rsl^t_{(X_i)}$. This operation is realized by taking into account all the generated security alerts by all the sensors presents in X_i. The security level obtained is then transmitted to the global intrusion database (gidb).
- *Logs collected by the RCBox are transferred to a remote site and the approximate security level of X_i is calculated.* While LA_j calculates $Asl^t_{(X_i)}$ ($i \neq j$), logs from sensors are sent to R-CBox_i which takes care to transfer them to the local database intrusion of a remote site. These data will be used to determine the approximate security level of X_i.

In a third phase, real ($Rsl^t_{(X_i)}$) and approximate ($Asl^t_{(X_i)}$) security levels of each X_i are compared according to the value of the static security level ($Ssl_{(X_i)}$). Depending on this evaluation a decision is taken.

2.3 Static security level of a site X_i

The static security level of a site X_i ($Ssl_{(X_i)}$) is the basic security level of all the equipment and services available on this site. Also known as "security level viewed from the firewall", it is determined by the combination of the security level of the site viewed from the inside ($Insl_{(X_i)}$) and its security level viewed from the outside ($Osl_{(X_i)}$). $Ssl_{(X_i)}$ is given by:

$$Ssl_{(X_i)} = \begin{cases} H & \text{if } Insl_{(X_i)} \geq 3 \ \& \ Osl_{(X_i)} \geq 3 \\ M & \text{if } 2.5 \leq Insl_{(X_i)} < 3 \ \& \\ & 2.5 \leq Osl_{(X_i)} < 3 \\ L & \text{in all other cases.} \end{cases}$$

2.3.1 Security level of a site viewed from the inside

The security level of a site X_i viewed from the inside noted $Insl_{(X_i)}$ is determined by a security expert. It depends on the following parameters:

The presence of security administrators, their experience, their level of skill. If administrators have experience and expertise in security, the network is considered as properly secured. Security certifications, or any other recognized computer security training coupled with a number of years of experience, can be a criterion for assessing the level of competence of security administrators. This value must be determined without complaisance to avoid bias the results.

Firewalls inside the site, their features and their levels of configuration also occupy an important place in the process of security level evaluation of a site from the inside. In fact, they give an idea of the degree of protection of each sensor of the site. This data is closely linked to the level of knowledge in the security of those who have installed and configured the firewall.

The security strategy implementation is also a valuable indicator in the process of evaluating the level of security of a site. It allows checking whether the company has set up several lines of defense to secure its information system. Partitioning the network as well as the implementation of deep device defenses are relevant solutions to enhance the security of a network.

The availability of a security policy in a network plays a huge role in assessing its security level. Indeed, this document summarizes all standard and security guidelines to strengthen and better managed the security of its information system. Its absence in a network leads to a rough and expeditious security administration of its information system and increases the vulnerability of the network to security incidents. This document must be constantly updated during the life of the information system to be adapted to changes.

The availability and implementation of an awareness program on computer security for network users are of great importance in the implementation of an effective security policy. Confidentiality of passwords is stronger, the risk of downloading malicious software will be reduced, and so on.

The availability of a plan for service continuity that retraces all the processes and mechanisms to be put in place to ensure service continuity in the case of a major security incident, participates to secure the network. Its availability allows administrators to be very reactive and to avoid a prolonged crash of the information system when an incident has occured. This document must be constantly updated during the life of the information system, especially after a major incident.

The availability of a plan for disaster recovery and data-loss is vital for a company. This document contains a set of policies, procedures and processes to be put in place in case of major incident, to get an information system back in a stable and functional status corresponding to its normal working condition.

To each preceding basic parameter noted p_j, is associated to a mark $M(p_j) = m$ with $m \in [1...5]$. According to the security policy of the network, other parameters can be added to the basic ones.

The security level of a site X_i viewed from the inside is determined like the average of the set of parameters:

$$Insl_{(X_i)} = \frac{1}{n_p} \sum_{j=1}^{n_p} M(p_j)$$

Even if $Insl$ is static in regards to Rsl and Asl, it must be recalculated depending on the modifications made to the information system.

2.3.2 Security level of a site viewed from the outside

This security level viewed from the outside of a site X_i noted $Osl_{(X_i)}$ is determined by the capacity of security devices located to the edge of the site to contain attacks from outside noted $Csd_{(X_i)}$ and by the level of security of X_i obtained after an audit, $Au_{(X_i)}$.

$$Osl_{(X_i)} = \begin{cases} H & \text{if } Csd_{(X_i)} \geq 3 \ \& \ Au_{(X_i)} \geq 3 \\ L & \text{if } Csd_{(X_i)} \leq 3 \ \& \ Au_{(X_i)} \leq 3 \\ M & \text{in all other cases.} \end{cases}$$

Capacity of security devices located to the edge of the site to contain attacks from outside is assessed by performing a penetration test on firewalls of the site and

by checking their compliance with the security policy of the network. Following the penetration test, a score between 1 and 5 is attributed to the level of security from the outside.

$$Csd_{(X_i)} = x,\ x \in [1...5]$$

The level of security of X_i obtained after an audit is determined by a security audit realized by an external security expert. Depending on the network size, the security audit can be a vulnerability scan, a security scan or a penetration test. Tests carried out by the site's internal security team can be added to the audit. It is defined as:

$$Au_{(X_i)} = \frac{1}{n}\sum_{j=1}^{n_y^{X_i}} \frac{Cl_{(Y_j)} * (Isl_{(Y_j)} + 1)}{5 * 3}$$

$Isl_{(Y_j)}$ represents the intrinsic security level of the sensor Y_j. It is defined as the security level in terms of OS, applications and services on the sensor. It is determined according to a security audit conducted on Y_j and after having implemented the proper recommendations that can be corrective or palliative. Depending on the network size, the audit can be a deep vulnerability scan, a security scan or a pen test. $Isl_{(Y_j)}$ must be re-evaluated after any major operation carried out on the sensor in order to reflect reality. Examples of major interventions are: an system update, an OS change or a new service deployed like a web server. $Isl_{(Y_j)} \in \{L, M, H\}$.

The criticity level of the sensor Y_j, noted $Cl_{(Y_j)}$, is defined according to its importance in the information system, with $Cl_{(Y_j)} \in [1...5]$.

2.4 Real security level evaluation of X_i

The evaluation of $Rsl^t_{(X_i)}$ is an operation conducted by the local analyzer (LA_i) installed on the site X_i. It consists in determining at regular intervals, a value that corresponds to the average levels of criticity of real intrusion activities that takes place on the site X_i. To determine this value, a criticity level is assigned to each generated alert (H, M and L). This level is defined according to the real impact of the intrusion on the information system. For example, an attempted attack that may lead to total paralysis of the network will lead to the generation of a high level of the generated alert.

The real security level of a site X_i at time t, has the inverted value of the criticity level of an alert raised on a sensor:

$$Rsl^t_{(X_i)} = \overline{C_{(Z^t_{max}, Y_j)}} \Leftrightarrow C_{(x)} \leq C_{(Z^t_{max})}$$

$$\forall x \in \left[Z^t_0 ... Z^t_{n_z}\right], \forall Y_j \in X_i$$

The real security level of a site is then high when all the sensors have no critical security alert activities. It is medium when there is one or several medium security alerts in one or several sensors and it is low when one or several sensors have high critical alerts.

2.4.1 Criticity level of an alert raised on a sensor

According to the intrinsic criticity level of the alert Z_k, $Ci_{(Z_k)}$, and to the theoritical security level of the sensor Y_j, $Esl_{(Y_j)}$, the criticity level of the alert Z_k on the sensor Y_j will variate. This security criticity level is given by:

$$C_{(Z_k,Y_j)} = \begin{cases} L & \text{si } Ci_{(Z_k)} = L \ \forall\ Esl_{(Y_j)} \text{ and } A^*_{(Y_j)} \\ M & \text{si } Ci_{(Z_k)} = M \ \forall\ Esl_{(Y_j)} \text{ and } A^*_{(Y_j)} \ || \\ & \text{si } Ci_{(Z_k)} = H\ \&\ Esl_{(Y_j)} = H\ \& \\ & A^*_{(Y_j)} = \left| \begin{matrix} M \\ L \end{matrix} \right. \\ H & \text{si } Ci_{(Z_k)} = H\ \&\ Esl_{(Y_j)} = \left| \begin{matrix} L \\ M \end{matrix} \right. \\ & \forall\ A^*_{(Y_j)} || \\ H & \text{si } Ci_{(Z_k)} = H\ \&\ Esl_{(Y_j)} = H\ \& \\ & A^*_{(Y_j)} = H \end{cases}$$

2.4.2 Expected security level of a sensor

The expected of security level of a sensor, $Esl_{(Y_j)}$, is the result of two parameters: its intrinsic security level $Isl_{(Y_j)}$ and its capacity to access to the others network components of the whole network, noted $A^*_{(Y_j)}$.

The expected security level of a sensor ($Esl_{(Y_j)}$) is defined by:

$$Esl_{(Y_j)} = \begin{cases} Isl_{(Y_j)} & \text{si } A^*_{(Y_j)} = H. \\ L & \text{si } Isl_{(Y_j)} = L \ \forall\ A^*_{(Y_j)} \\ M & \text{si } Isl_{(Y_j)} = M \ \forall\ A^*_{(Y_j)} \\ M & \text{si } Isl_{(Y_j)} = H\ \&\ A^*_{(Y_j)} = H \\ H & \text{si } Isl_{(Y_j)} = H\ \&\ A^*_{(Y_j)} = \left| \begin{matrix} L \\ M \end{matrix} \right. \\ M & \textit{in all other cases.} \end{cases}$$

$Isl_{(Y_j)}$ is defined as the security level in terms of OS, applications and services on the sensor as defined in part 2.3.2.

The capacity of Y_j to access to all network components noted $A^*_{(Y_j)}$ is determined by two parameters:

- $A_{(Y_j,X_i)}$ with $Y_j \in X_i$.
-

- $A_{(Y_j,X_l)}$ with $Y_j \in X_i$ and $\forall l, X_l \neq X_i$.

These values are determined according to the permissions the sensor has on the other components.

$A^*_{(Y_j)}$ is then given by:

$$A^*_{(Y_j)} = \begin{cases} L & \text{if } A_{(Y_j,X_i)} = L \text{ with } Y_j \in X_i \ \&\& \\ & A_{(Y_j,X_l} = L \text{ with } Y_j \in X_i \text{ and } \forall l, X_l \neq X_i \\ H & \text{if } A_{(Y_j,X_i)} = H \text{ with } Y_j \in X_i \ \&\& \\ & A_{(Y_j,X_l)} = H \text{ with } Y_j \in X_i \text{ and } \forall l, X_l \neq X_i \\ M & \textit{in the other cases.} \end{cases}$$

2.4.3 Criticity level of a security alert

The criticity level of a security alert, also known as $Ci_{(Z_k)}$, is given by the impact of the events which led to its generation on the integrity and consistency of data available on the network. The databases of known vulnerabilities (CVE, CERT, Snort, etc.) link criticity levels to the listed signatures of attacks. These criticity levels are then used to define $Ci_{(Z_k)}$. For intrusions based on a behavior deviation or a violation of security policy, it is the site administrator who should associate criticity levels to $Ci_{(Z_k)}$ according to the security policy of the company and the importance of data handled.

2.5 Approximate security level evaluation of X_i

The method for calculating the approximate security level of X_i at time t, $Asl^t_{(X_i)}$, if the same as for the real security level:

$$Asl^t_{(X_i)} = \overline{Ca_{(Z^t_{max},Y_j)}} \Leftrightarrow Ca_{(x)} \leq Ca_{(Z^t_{max})} \quad \forall x \in \left[Z^t_0 ... Z^t_{n_z}\right], \forall Y_j \in X_i$$

The only difference lies in the function Ca which evaluates the criticity of the alerts based on selected sensors and no more on all the sensors. Each RCBox must indeed collect data on sensors that have the highest probability of being targeted by attacks so that its analysis on a site X_i gives an approximate result of its security level. The methodology for selecting the sensor is as follows:

For each sensor, the approximate security level is determined. Then sensors with the lowest levels are monitored as a priority, followed by those with an average rating. If the selected site is not targeted by intrusive activities, it is also possible to monitor the nodes with a high security level.

The approximate security level of a sensor Y_j, noted $Asl_{(Y_j)}$, is calculated with three other functions:

- *The expected security level of a sensor Y_j ($Esl_{(Y_j)}$).* This function has already been defined in the previous part and remains the same.
- *The importance of the data contained on the sensor Y_j ($Idc_{(Y_j)}$).* This value is determined by the network administrator in terms of data sensibility and confidentiality contained by the sensor. This value is dynamic and must be constantly updated at each major operation on a node.
- *The importance of the sensor Y_j in the information system ($Is_{(Y_j)}$)* represents its value in the stability and availability of the information system of the company. The importance of a node in a network is a fluctuating value that must be renewed at every notable intervention made on the node. A sensor with a high Is value should have in normal conditions, a high $Esl_(Y_j))$.

The approximate security level of a sensor Y_j is given by:

$$Asl_{(Y_j)} = \begin{cases} L & \text{si } Esl_{(Y_j)} = L \\ L & \text{si } (Idc_{(Y_j)} = H \textit{ ou } Is_{(Y_j)} = H) \ \& \\ & Esl_{(Y_j)} = M \\ H & \text{si } (Idc_{(Y_j)} = H \textit{ ou } Is_{(Y_j)} = H) \ \& \\ & Esl_{(Y_j)} = H \\ H & \text{si } (Idc_{(Y_j)} = M \textit{ ou } Is_{(Y_j)} = M) \ \& \\ & Esl_{(Y_j)} = H. \\ M & \textit{in all other cases.} \end{cases}$$

2.5.1 Choice of the sensors which transmit their logs

The chosen sensors which transmit their logs to the RCBox, are primarily those with high or low approximate security level, has seen in previous section. They are mainly:

- *Sensors with low approximate security level (L).* These types of sensors are the weakest links in the network and have to be supervised carefully. Their corruption can facilitate a rapid expansion of the intrusion to the whole network, this leading to a total unavailability or an overall corruption of the information system.
- *Sensors with high approximate security level (H).* These types of sensors are generally security devices (firewalls, IDS, IPS, anti-virus servers, etc.), components of a security system (agents, distributed analyzers), the production servers or active networks equipment (switches, routers, etc.). The confidentiality, integrity and availability of services offered by the information system of the company are conditioned by their proper functioning.

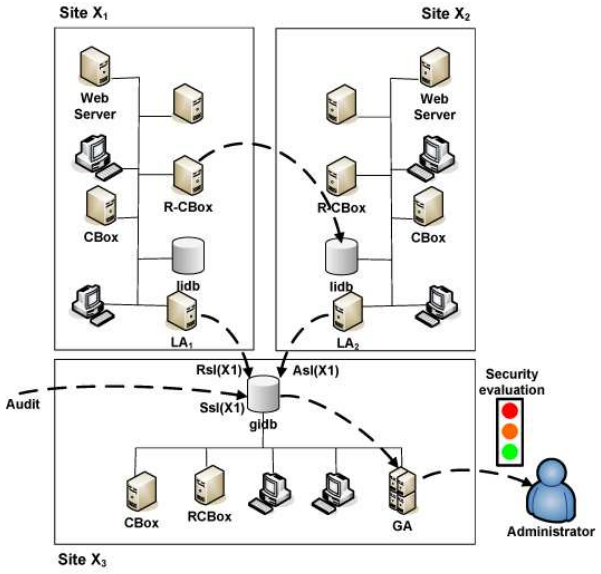

Figure 1. Decision making process

On a site with low intrusive activities, it is possible to collect in addition to the above-mentioned sensors, data from sensors with a medium approximate security level.

3 Rules for decision making

On each site X_i, the local analyzer LA_i periodically calculates $Rsl^t_{(X_i)}$. This information is then stored in the global intrusion database (gidb). A the same time, LA_j, $i \neq j$ receives logs from RCBox of X_i and calculates $Asl^t_{(X_i)}$ and sends this information to the gidb.

The global analyzer (GA) compares this two values with $Ssl_{(X_i)}$ for each X_i and takes a decision. Figure 1 sums up this decision making process. In normal condition, when integrity of components of X_i is preserved, we have:

$$Rsl^t_{(X_i)} \approx Asl^t_{(X_i)}$$

GA calculates $Dev^t_{(X_i)} = Rsl^t_{(X_i)} - Asl^t_{(X_i)}$ and defines the importance of the deviation as:

$$\Delta^t_{ra\ (X_i)} = \begin{cases} H & \text{if } Dev^t_{(X_i)} = -2 \\ M & \text{if } Dev^t_{(X_i)} = +2 \ ||Dev^t_{(X_i)} = -1 \\ L & \text{if } Dev^t_{(X_i)} = +1 \ ||Dev^t_{(X_i)} = 0 \end{cases}$$

The meaning of these three cases for a network which have a medium level of security ($Ssl^t_{(X_i)}$=M) is:

$\Delta^t_{ra\ (X_i)} = L$ A low $\Delta^t_{ra\ (X_i)}$ is a sign of that the components of the sites have not been compromised. The state of network is *Green*.

$\Delta^t_{ra\ (X_i)} = M$ A medium $\Delta^t_{ra\ (X_i)}$ means that attacks are carried out against moderately secure nodes of X_i. These types of attacks have relatively low spreading since attacked nodes have a fairly limited access to other nodes. If they are compromised, attacked nodes can be used as rebounds to launch attacks to critical systems in the network.
These attacks rarely lead to a total unavailability of the services provided by the information system because they are not directed against vital components. The state of network is *Orange*.

$\Delta^t_{ra\ (X_i)} = M$ This situation reflects the compromise or the denial of service of one or more security devices on X_i. More generally, it may be a sign of corruption of the local database intrusion, a compromise of the local analyzer or the local knowledge base of DSOC (commonly known as (KBox)). This situation is the most dangerous and requires the implementation of effective and pragmatic countermeasures to avoid the compromise of the entire information system. One of the phases to eradicate the incident could be the total isolation of X_i from the rest of the network. The investigations must be conducted to understand the progress of the intrusion so that the appropriate corrective action can be applied.

If we want to take into account all the cases, the following results are obtained:

$$S^t_{(X_i)} = \begin{cases} R & \text{if } Ssl^t_{(X_i)} = \begin{cases} M & \& \ \Delta^t_{ra\ (X_i)} = H \\ L & \& \ \Delta^t_{ra\ (X_i)} = \begin{vmatrix} H \\ M \end{vmatrix} \end{cases} \\ O & \text{if } Ssl^t_{(X_i)} = \begin{cases} H & \& \ \Delta^t_{ra\ (X_i)} = H \\ M & \& \ \Delta^t_{ra\ (X_i)} = M \end{cases} \\ G & \text{if } Ssl^t_{(X_i)} = \begin{cases} H & \& \ \Delta^t_{ra\ (X_i)} = \begin{vmatrix} M \\ L \end{vmatrix} \\ M & \& \ \Delta^t_{ra\ (X_i)} = L \\ L & \& \ \Delta^t_{ra\ (X_i)} = L \end{cases} \end{cases}$$

4 Case study

This example aims at showing the efficiency and the pertinence of our network security evaluation method.

To do this, we developed the scenario described below. The network architecture evaluation is presented in figure 2):

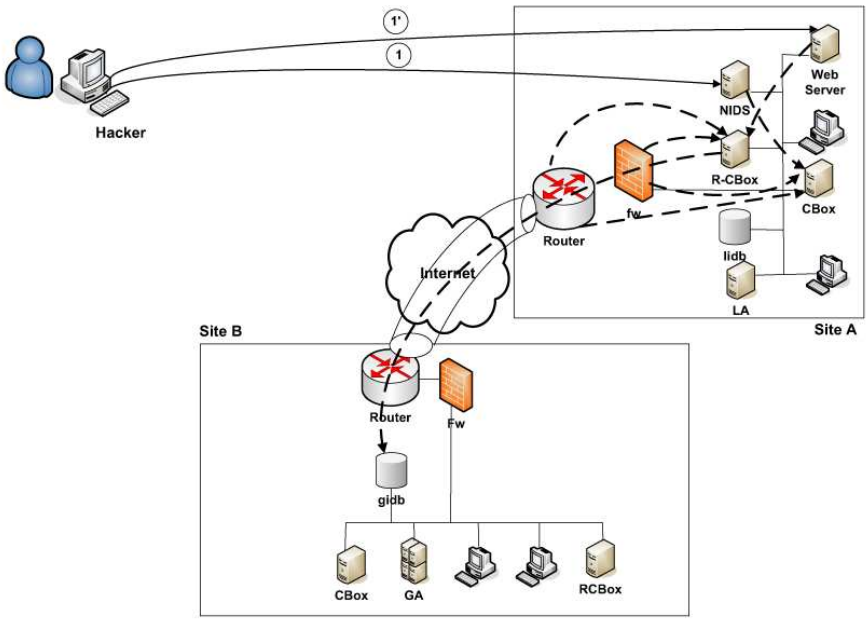

Figure 2. Network architecture for our experiment

A network intrusion detection system (NIDS) is installed on the network of the site A. This NIDS captures and analyzes all network traffic which destination is the sensors.

To meet its expansion needs, the company merges with another one. This leads to the interconnection of its network (site A) with that of the new company (site B).

In order to have a system to supervise the security of this new network, it is planned to install a DSOC. However, the company believes that its NIDS gives good results and it did not want to replace it. The DSOC must be installed in the new network (site B) to take into account this parameter.

Two local analyzers are then installed on the two sites and the site B acts as a global analyzer. The CBox of site A intercepts security alerts from NIDS, the events from firewall and router logs. The RCBox intercepts logs from the web server, the firewall and the router and transfers them to the site B.

A hacker decides to attack this multi-sites network. After several attempts, he compromises the NIDS of site A, stop its logging and alerts generation systems. The next step is to attack the web server in order to stop it.

Observations The compromise of the NIDS automatically stops security supervision of the web server by the local analyzer of the site A. Only events from firewall and router logs are intercepted by the CBox of site A. Since no intrusive activity is directed against its supervised components, the local analyzer determines that the real security level $Rsl^0_{(A)}$ of site A is acceptable. At the same time, the RCBox of site A collects logs from the web server, router and firewall and transfers them to site B. The local analyzer of site B determines the approximate security level of site A $Asl^0_{(A)}$ and sends it to the global analyzer which compares $Rsl^0_{(A)}$ and $Asl^0_{(A)}$. In this case, $Asl^0_{(A)}$ is far greater than $Rsl^0_{(A)}$, so the indicator of this network turns in orange if $Ssl_{(A)} = H$ or in red in all other cases. The network administrator is then quickly warned that there is something wrong with the site A. By looking at the log of DSOC, he will quickly find that the NIDS did not send logs.

This example has shown the possibilities offered by our security level evaluation. Real tests will be needed to better stress the system and to learn its qualities and drawbacks.

5 Related work

To overcome the limitations of the traditional IDS which are unable to detect complex attacks, several types of IDS have been proposed and tested. None of the work mentioned here has the same specificity as our

work to define a simple metric for security evaluation.
The first type of IDS are the distributed ones [1, 9, 8]. DSCIDS [1] is a distributed IDS composed of two elements: intelligent agents which collect data on each host and send them to a central unit called Analyzer/Controller. The Analyzers/Controllers are built hierarchically and each one manages several collection agents. They are also responsible for data analysis and alert correlation. To detect complex intrusions, a collaboration of several intelligent agents is needed. Lee, Chung, Kim, Younho, Park and Yoon proposed a distributed intrusion detection system [8] which correlates alerts in real time. On each network, a sensor collects data and eliminates redundant information. Afterwards data is analyzed for intrusion detection. Correlation is carried out to detect complex intrusions like distributed ones and if there is a great similarity between several alerts, they are merged.
Other methods based on a P2P approach were proposed for scalability purpose. One of the best-known methods is INDRA [7]. With INDRA, each host runs a daemon which analyzes local intrusions and provides controlled access to resources. When a host detects an intrusion, a multicast message is sent to the other hosts which check the integrity of the message and blacklist the source of the intrusion if it needs to be.
Cooperation of IDSs is still an ongoing work [6, 10]. PAID [6] is a cooperative agent-based intrusion detection system. In PAID architecture, each agent is autonomous. It detects intrusions and collaborates with the other agents to detect complex intrusions. With TRINETR [10], an intelligent agent collects data on each host of a network and sends it to a coordination agent which analyzes data to find intrusion patterns. When the coordination agent needs information to deduce whether or not there is an intrusion, it can request a particular collection agent.

6 Conclusion and future work

In this article, we have proposed a simple metric based on three values represented by colors (red, orange and green) to evaluate the security level of sites in a multi-sites network. This metric is based on our experience in security operation centers design and it is integrated in our new SOC dedicated to multi-site networks: DSOC. A simple example has shown the possibilities of such a metric: allowing fast countermeasures to intrusions which we were unable to detect without it.
Future works will include real experiments in a real network to define some unknown parameters like t and to test the efficiency of our method in real conditions. The ideal case is to have t expressed in seconds, but it is very difficult to know if this option is realistic or not, depending on the number of sensors, the power of the different boxes and so on.

References

[1] J. T. Ajith Abraham, Ravi Jain and S. Y. Han. D-scids: Distributed soft computing intrusion detection systems. *Journal of Network and Computer Applications*, 30:81–98, 2007.

[2] R. Bidou, J. Bourgeois, and F. Spies. Towards a global security architecture for intrusion detection and reaction management. In *4th Int. workshop on information security applications*, pages 111–123, 2003.

[3] A. Ganame, J. Bourgeois, R. Bidou, and F. Spies. A High Performance System for Intrusion Detection and Reaction Management. *Journal of Information Assurance and Security*, 3:181–194, sep 2006.

[4] A. Ganame, J. Bourgeois, R. Bidou, and F. Spies. A Global Security Architecture for Intrusion Detection in Computer Networks. In *IPDPS'07, Proc. of the ACM/IEEE Int. Parallel and Distributed Processing Symposium*, Long Beach, California USA, mar 2007. IEEE computer society press.

[5] A. K. Ganame, J. Bourgeois, R. Bidou, and F. Spies. Evaluation of the intrusion detection capabilities and performance of a security operation center. In I. Press, editor, *International Conference on Security and Cryptography*, pages 48–55, August 2006.

[6] V. Gowadia, C. Farkas, and M. Valtorta. Paid: A probabilistic agent-based intrusion detection system. *Computers & Security*, 24(7):529–545, 2005.

[7] R. Janakiraman, M. Waldvogel, and Q. Zhang. Indra: A peer-to-peer approach to network intrusion detection and prevention. In *Proceedings of IEEE WETICE*, June 2003.

[8] S. Lee, B. Chung, H. Kim, Y. Lee, C. Park, and H. Yoon. Real-time analysis of intrusion detection alerts via correlation. *Computers & Security*, 25(3):169–183, 2006.

[9] C. Li, Q. Song, and C. Zhang. Ma-ids architecture for distributed intruction detection using mobile agents. In *In Proc. of the 2nd International Conference on Information Technology for Application (ICITA)*, pages 451–455, May 2004.

[10] J. Yu, Y. V. Reddy, S. Selliah, S. Reddy, V. Bharadwaj, and S. Kankanahalli. TRINETR: An architecture for collaborative intrusion detection and knowledge-based alert evaluation. *Advanced Engineering Informatics*, 19(2):93–101, 2005.