A Global Security Architecture for Intrusion Detection on Computer Networks

Abdoul Karim Ganame, Julien Bourgeois, Renaud Bidou, Francois Spies
LIFC, Universit de Franche Comt
4, place Tharradin, 25211 Montbeliard, France
Email: {ganame, bourgeois, bidou, spies}@lifc.univ-fcomte.fr

***Abstract*— Detecting all kinds of intrusions efficiently requires a global view of the monitored network. Built to increase the security of computer networks, traditional IDS are unfortunately unable to give a global view of the security of a network. To overcome this situation, we are developing a distributed SOC (Security Operation Center) which is able to detect attacks occurring simultaneously on several sites in a network and to give a global view of the security of that network. In this article, we present the global architecture of our system, called DSOC as well as several methods used to test its accuracy and performance.**

***Index Terms*— IDS, Distributed intrusion detection, SOC, network security, global view**

I. Introduction

Two aspects are to be taken into account when ensuring network security: protection and supervision. Protection is composed of hardware, software and security policies that must be followed. Even the best protection is always vulnerable to attacks due to unknown security bugs. The network configuration is constantly changing, thus increasing the possibility of creating security holes.
In order to help security administrators, intrusion detection systems have been developed, but they have several drawbacks: insufficient detection rates, too many intrusions are detected or missed [3]. Moreover, basic IDSs have insufficient information to detect complex intrusions like distributed or coordinated attacks.
On the basis of our last research work which present a centralized SOC called SOCBox [2] and its evaluation [4] [5], we are developing a distributed one named DSOC[1], for intrusion detection on wide networks.

With DSOC, in any network site, a local detection engine analyzes data collected by one or several collection boxes to find intrusion patterns. Afterwards, 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.

The rest of the paper is structured as follows: Section 2 describes some drawbacks of centralized security systems collecting data from remote sites and introduces why distributed systems are needed. Section 3 focuses on the description of the global architecture of our distributed system (the DSOC). Then, we focus on data collection, data analysis and correlation operations in sections 4, 5 and 6. Site security level assessment mechanisms are presented in section 7. In section 8, we verify the ability of the DSOC to detect an attack using a relay. Section 9 describes the behavior of the DSOC when a strong attack occurs in a site of a multi-site network. In section 10, we compare the SOCBox and the DSOC bandwidth usage. Section 11 will be devoted to the related work, and will be followed by a conclusion.

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

II. Why a distributed system is needed?

Our centralized SOCBox has some limitations: in addition to the fact that it was designed around a unique analyzer, giving a single point of failure, its performance evaluation [4] [5] showed that its detection capability can decrease when a strong attack occurs or under a high traffic. Another drawback of the SOCBox is its inability to detect intrusions in a remote site in case of the failure of the communication link between the site where the SOCBox is located and a remote site. Moreover, when the monitored network grows and we have to add several new sensors, the performance of the SOCBox can decrease.

These problems are common to all centralized security systems collecting data from remote sites.

A. Single point of failure

One of the principal drawbacks of a centralized SOC is its centralized architecture which induces a single point of failure. This situation increases the probability of denial of service which can decrease the global performance of the SOC.
In order to ensure continuous operation, two or more SOCs can be used in failover mode, but that does not resolve neither the scalability problem nor the performance decreasing during strong attacks.

B. Communication link breaking in a multi-site network

For a centralized SOC be able to manage sensors located on several sites, it is necessary to install VPN links between these sites and the one where the centralized SOC is located. One of the major drawbacks of this approach is that when a strong attack occurs in a site, the redirection of the logs (or

the alerts) towards the centralized SOC can generate a large data flow which can break the communication link between the concerned site and the SOC. This prevents intrusion detection on the attacked site.
The scenario to verify the behavior of a centralized SOC when it manages several sites and when a strong attack occurs in one of them is described below. We named this attack **"isolation attack of a site".**

A centralized SOC (the SOCBox in this test) manages the security of some critical sensors located on a network composed of 3 sites A, B and C (Fig. 1).

The SOCBox is installed on the site A and the sensors located on the other sites send their logs to it through a VPN link.
After a scan in the network, a hacker sees opened ports on sensors located in the site B and decides to hack this site (his purpose is to steal data). Using a traffic generator, he floods the sensors with data containing signatures of real attacks in order to break down any possible IDS installed on the site. The goal of this operation is to camouflage his intrusion.
After a few minutes of flood, the hacker launches an attack on the site B. He compromises a sensor and steals data. After that, he erases his actions on the logs of the compromised sensor.
During this attack, when the flood occurs, the attacked sensors generate large quantity of logs which are redirected towards the SOCBox . Due to the high data flow sent towards the SOCBox, the VPN link is saturated and the communication link between the site B and the site A goes down.

The network administrator notices the interruption of the VPN link and restores it. But, because the SOCBox does not receive all the logs coming from the compromised site, it can not determine if the intrusion in the site B was successfully or not. Thus, the security administrator can not conclude that data is stolen.

C. Limitation of a centralized SOC when operating in a single site

The goal of this test ((Fig. 2) is to check the limitation of centralized SOCs when they manages a single site where a strong attack occurs. In other words, we try to answer the question "Is a centralized SOC able to continue to detect intrusions when it receives a high data flow?".

We use the SOCBox as a centralized SOC in this test which consists in flooding some sensors on a network with a high data flow composed of Ping with large ICMP data (50000 bytes each one) and to initiate a Nikto attack [12] against one of the sensor (a web server). The goal is to check if the SOCBox is able to detect the Nikto intrusion.

The hosts characteristics are:
Victims: PIII, 450 MHz, 256 MB of RAM
The SOCBox and Snort: PIII, 450 MHz, 256 MB of RAM
Attacker: PIV, 1.73 Ghz, 512 MB of RAM.

When the SOCBox receives events coming from the sensors (Fig. 2), it automatically analyzes them and records them on the hard disk only when they match security rules defined by the security administrator. Because we configured the SOCBox to ignore the Pings, it only records events about the Nikto attack. Thus, from 0 to 1.8 millions Pings, the SOCBox records 500 KB of data each time the Nikto attack is launched. After 1.8 millions Pings, the SOCBox is unable to detect the intrusions, due to the fact that it uses a great quantity of memory.

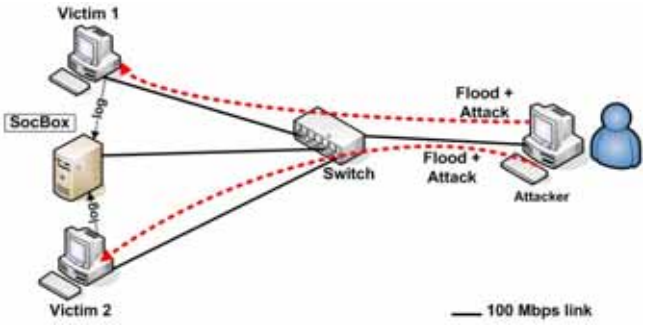

Fig. 2. Using Syslog to send data to the SOCBox

D. Bandwidth usage when the SOCBox gathers data from sensors located in a remote site

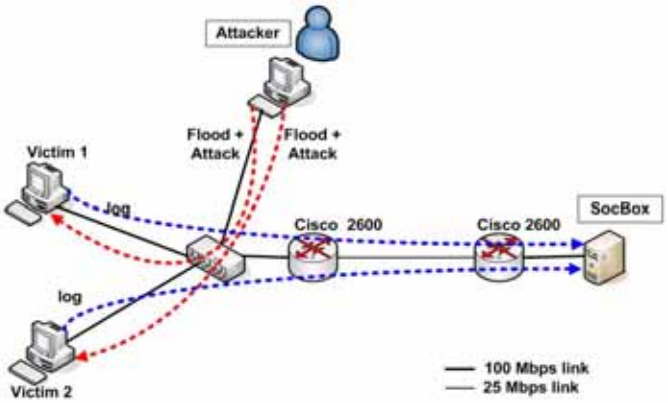

Fig. 3. Remote SOCBox receiving data via syslog

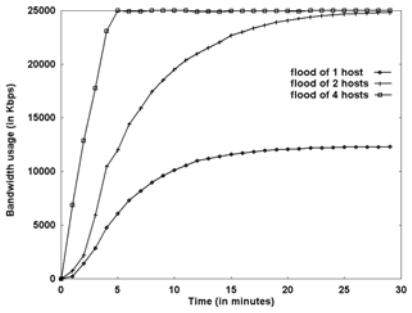

Fig. 4. Bandwidth usage when flooding 1, 2 and 4 hosts.

In this test (Fig. 3), we flood successively one sensor, two sensors, and four sensors installed in a remote site and which forward their logs to a centralized SOCBox located in another site. The goal is to measure the bandwidth usage induced by

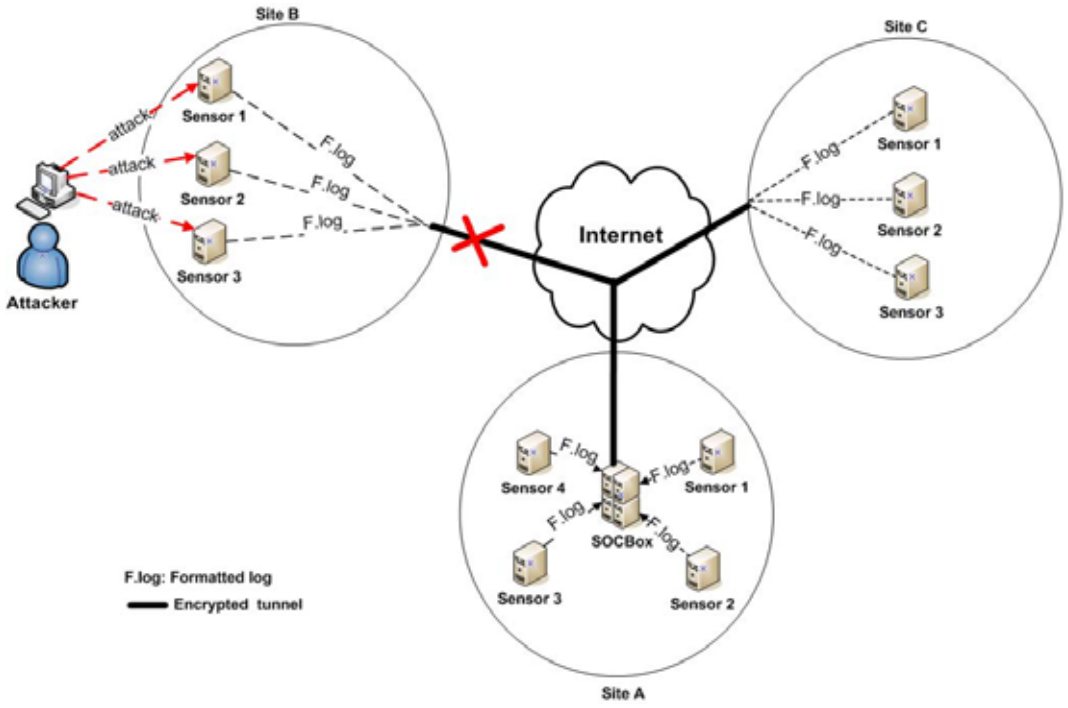

Fig. 1. Isolation attack of a site managed by a centralized SOCBox

data redirection towards the SOCBox when a strong attack occurs.

In the first part of the test, we flood a sensor by sending to it 2 millions Pings with packets of 1460 bytes each one (via IP-Traffic [17]). Then we observe the bandwidth usage when logs are forwarded to the SOCBox located on the remote site. The two sites are connected using two Cisco 2600 routers with a 25 Mbps link.

In the second part of this test, two sensors installed in the same site are flooded simultaneously by sending to each one 2 millions Ping with 1460 bytes packets. These sensors forward their logs to the SOCBox located in another site.

In the third part of this test, 4 sensors are used with the same conditions than the second part of the test.

When flooding the sensors during 30 mins (Fig. 3), we notice a stabilization of the bandwidth usage around 12 Mbps (for 1 sensor) and 24.8 Mbps (for 2 sensors). For 4 sensors, the bandwidth is saturated very quickly after 5 mins (Fig. 4) and that causes losses of packets in the first time, and the breaking of the communication link between the two sites in a second time.

E. Why a new architecture?

To overcome the limitations of the centralized SOCs, we propose a new distributed architecture called DSOC. This architecture is designed to be scalable, to support wide networks and to be highly available.

III. THE DSOC GLOBAL ARCHITECTURE

The DSOC is composed of four components based on the CIDF specifications [13]: data collectors (CBoxes), remote data collectors (R-CBoxes), Local Analyzers (LAs) and a Global Analyzer (GA). Its global and simplified types of architecture are shown in Fig. 5 and Fig. 6.

A. Data Collection Box

A CBox collects data from sensors located on the same segment of a network. A sensor can be a host, a server, a firewall, an IDS or any system that generates logs. The advantage of our log collection approach is that no software has to be installed on the sensors. Moreover, our system is compatible with a wide range of hardware and software [14]. A CBox formats logs and sends them to a local intrusion database (lidb). In each site we have one or several CBoxes and one of them acts as a Master CBox (M-CBox). The M-CBox is responsible for the management of all the CBoxes located on the same site. It polls regularly the other CBoxes and when a CBox is down, the M-CBox will collect data on the segment of the failed CBox. Each Master CBox also has a backup which polls it regularly and will become Master if need be.

B. Remote Data Collector

An R-CBox is a special CBox which collects data coming from some critical sensors and from sensors hosting security tools in any site. Afterwards, data is forwarded to the local intrusion database of another site and is analyzed to give in real time the approximate security level of the concerned site. This helps to anticipate a reaction when a critical intrusion

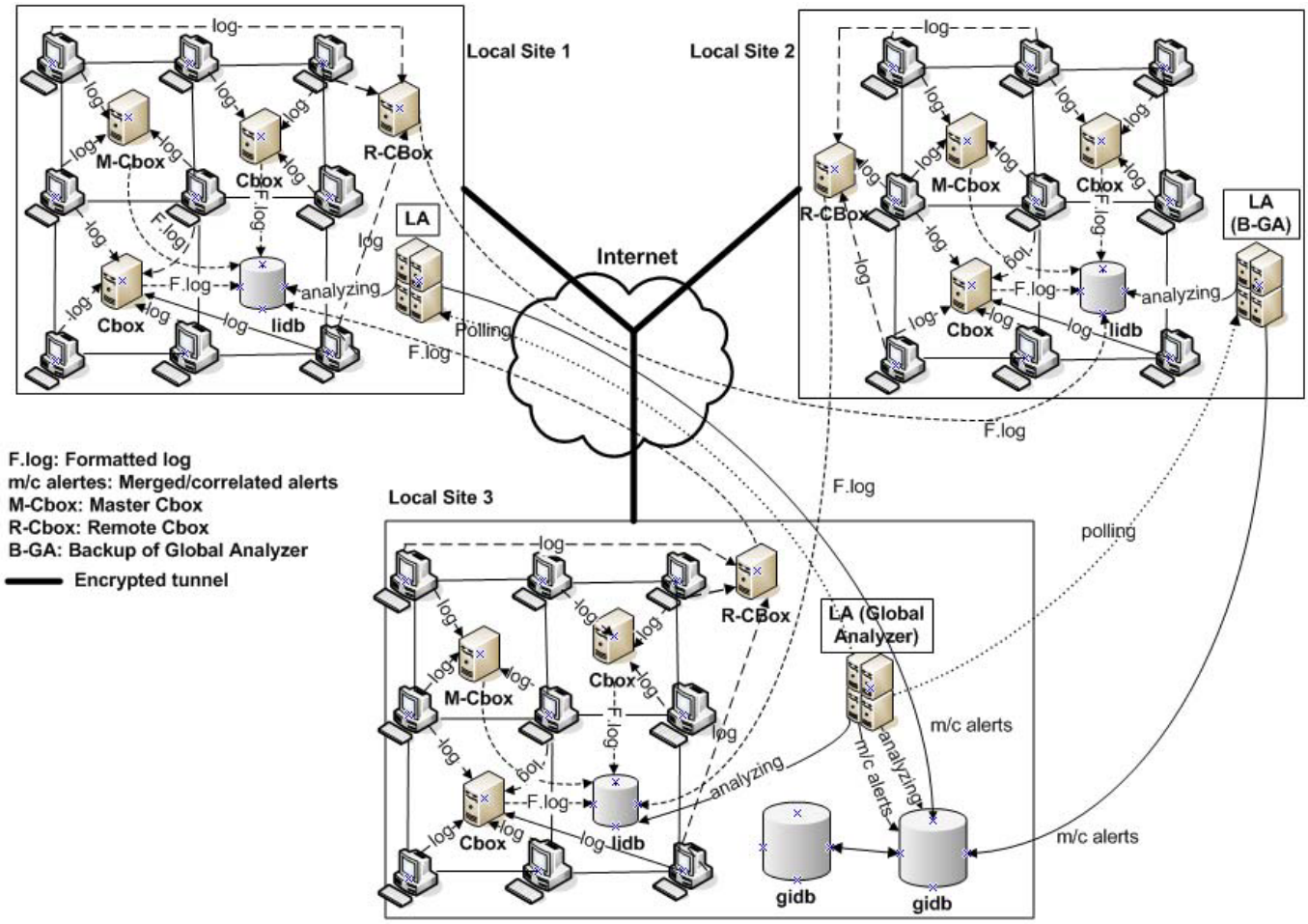

Fig. 5. Global architecture of the DSOC

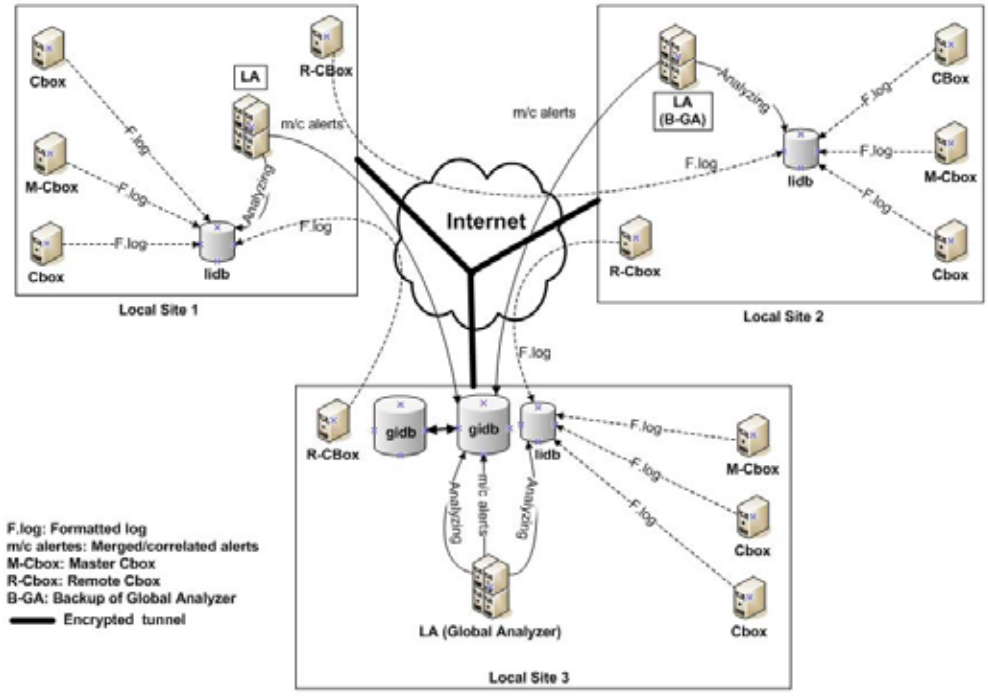

Fig. 6. A simplified view of the DSOC

occurs or to investigate and troubleshoot a site that could be compromised, even if a hacker erases the logs on the compromised sensors (including the security tools).

C. The Local Analyzer

A Local Analyzer (LA) is responsible for intrusion detection at any site of a network. It analyzes formatted logs located in a local intrusion database (lidb) and generates alerts. Afterwards, it correlates the alerts to find more complex intrusions (intrusions composed of several events, distributed intrusions, intrusions directed to many sensors, etc.). The LA also compacts alerts by merging similar ones. All the alerts generated by an LA are sent to the global intrusion database (gidb). The gidb can have a mirror of itself for high availability purpose.

D. The Global Analyzer

The Global Analyzer (GA) is a chosen LA responsible for the global intrusion detection in a network. It analyzes alerts from the gidb, correlates and merges them if possible to generate optimized outputs. It is also able to detect more sophisticated intrusions that are directed to several sites. The GA regularly polls the other LAs and when one of them is down, the GA detects the occurring intrusion into the concerned site.
Another LA acts as the backup of the GA and polls it regularly. When the GA is down, the backup becomes the GA and another backup is elected.
The DSOC architecture is designed bearing in mind that the data flow processed in the different sites of the network is not always homogeneous. Indeed, in some sites, a large amount of data is processed and in this kind of situation, several CBoxes are needed for data gathering.
Even though a single CBox has to be installed on each segment, it is not excluded installing several CBoxes on the same segment when the sensors located in this part of the network are operating under high workloads.
In quieter sites, only one CBox can be used to collect data coming from all the sensors.

The DSOC also implements the different types of boxes defined for network intrusion detection systems in [10]. However, beside the pure technical aspects involved in such implementations, it is necessary to consider the supervision of an IT infrastructure as a full operational project.

IV. Monitoring the security of a network

Ensuring that all goes well in any site is essential to the monitoring of the security activity on a multi-site network. The R-CBoxes are built for this purpose. While a CBox sends data to the local intrusion database (lidb) of the site where it is located, an R-CBox forwards data to the lidb of a remote site. The analysis of forwarded data gives an approximate view of the security level of the concerned site. When an incident occurs, this can help with troubleshooting.

Operations related to a site security level assessment are the following:

- In each site S, the R-CBox collects data from the LA and some critical sensors and sends it to the lidb of another site.
- The LA of the remote site which receives data from the R-CBox analyzes it and generates alerts (each alert has a level of criticality). Afterwards, an approximate security level of the site S (*Sasl*) is determined.
- The LA of the site S analyzes data gathered by CBoxes, finds intrusion patterns on it and detects suspicious behavior. It also determines the real security level of the site (*Srsl*).
- The GA compares these two security levels and when there is a significant deviation between them, a suspicious behavior alert is generated. A significant deviation can be a sign of the R-CBox or the LA compromise. This can also be due to the fact that an intruder attempts to hide the compromise of one or several sensors. In this case, an alarm is sent to the security manager for advanced investigations.

A. Protecting the communications between the DSOC components

One of the key points here is to make sure that no illegitimate computer will act as an LA, a CBox or an R-CBox in order to get privileged access to the system. To ensure the security of our system, any LA or any R-CBox that needs to exchange information with another LA has to use a certificate to prove its identity. Moreover, all the communications between the LAs (also including the GA) and the communications between the R-CBoxes and the LAs will have to pass through an encrypted tunnel, available via the SSL protocol.

V. Detection of an Intrusion which uses a relay

The intrusion detection capabilities and the performance of the Local Analyzers were studied in our recent works [4] [5]. The goal of the present evaluation is to check the aptitude of the DSOC for detecting an intrusion which uses a relay (Fig. 7). The scenario of this attack is the following:
Attacker wants to hack a host (called *Victim*) located in the ISP site and hosting information about subscribers. His idea is to gain access to *Victim* by brute force attack and to steal data about subscribers. The ISP network is well secured and *Victim* can be accessed only from special hosts in the Management LAN (for maintenance purpose) and in our lab site. *Attacker* tries to compromise *Victim* and unfortunately for him, all his actions are refused. On second thoughts, he infers that it would be easier for him to try to hack *Victim* from hosts located in another site of the network. After multiple attempts, he compromises a host (*Victim1*) in our lab site (less secured for the evaluation purpose). Afterwards, he launches the attack consisting in the following actions:

- From *Victim1*, he executes a quick scan with nmap to detect open ports on *Victim*. He sees that ssh and mysql are open on *Victim*.

- From *Victim1* he launches a brute force attack with THC-Hydra [15] to gain access to the mysql database of *Victim*.

During this test, the behavior of the DSOC is the following:

1) The LA of the ISP site generates alerts about the brute force attack on *Victim* (multiple "authentication failed" messages are sent to the local intrusion database and a unique alert "brute force attack" is sent to the global intrusion database).
2)
   - The LA of our lab site generates alerts about the multiple attempts at access to *Victim1*. These alerts are merged and sent to the global intrusion database.
   - The R-CBox of our lab site gathers data from the LA, the Firewall and *Victim1* and sends it to the local intrusion database of the ISP site.
   - The LA of the ISP site generates alerts by analyzing data gathered by the R-CBox of our lab site and gives an approximate security level of our lab site.
3) The LA of the ISP site generates alerts about the scan (detected by the firewall) and the brute force attack (detected by analyzing the logs of *Victim*). These alerts are sent to the global intrusion database.

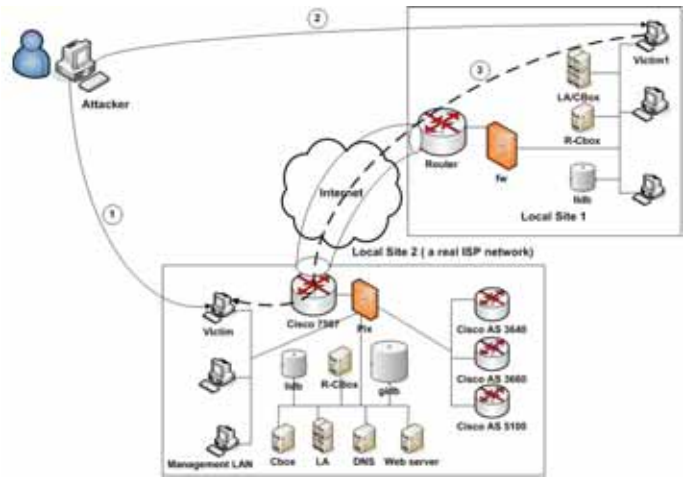

Fig. 7. Evaluation of the DSOC aptitude for detecting an intrusion which uses a relay

Comments

Because the scan and the brute force attack have the same target and they take place approximately at the same time, the LA of the ISP site matches them with the same context (time based correlation) and generates a unique alert.

Due to the fact that one of the steps of the attack (the brute force attack) is executed by 2 hosts (*Attacker* and *Victim1*) against the same target (*Victim*), a second correlation is performed by the LA of the ISP site (acting as the GA). The purpose of this correlation is to verify if there exists a relation between both attempts at brute force attack. The GA sees that *Attacker* attempted to access at *Victim1*. Then, it generates a "suspicious behavior" alert about a "probable compromission of *Victim1*". An information alarm is also sent to the security manager for advanced investigation on *Victim1*.

In short, we can say that our system is able to detect an attack which occurs simultaneously in several sites on a network. Most IDs cannot detect this kind of attacks.

VI. Behavior of the DSOC when a strong attack occurs in a site

To verify the DSOC behavior when a strong attack occurs in a site of a network, the following test is carried out:

A hacker initiates an attack against a network composed of 3 sites connected by VPN links (Fig. 8).

The Global Analyzer (GA) is installed in the site A, a local analyzer (LA) is installed in the site B, another LA is installed in the site C and Scanlogd [11] is installed on the sensors to detect the portscans.

After a scan against the network, the hacker sees opened ports on sensors located in the site B and decides to hack this site. Using a traffic generator, he floods the site B. The goal of this operation is to camouflage his intrusion in a high data flow. After that, he launches an attack against the site B to steal data.

The behavior of the DSOC

During the scan of the site B, the LA of this site detects the scan (data about the scan are collected by Scanlogd) and generates alerts. The R-CBox receives data from some critical sensors and forwards it to the LA of the site A (the GA). The GA analyzes data coming from the R-CBox of the site B and detects that a scan occurs in this site. After that, it evaluates the approximate level of security of the site B and concludes that there is no significant intrusion activity in the site B.

Operations performed in each site during this attack are the followings:

A. In the site B

- The LA generates the alerts, merges and correlates them. Then, it assesses the real level of security of the site B and forwards this information to the GA. Afterwards, it forwards the alerts to the global intrusion database (gidb) located in the site A. Only merged and correlated alerts are transmitted to the gidb via the VPN link. This minimizes the communication through the VPN and optimises the link usage.
- The R-CBox of the site B gathers data coming from some critical sensors and sends it to GA.

B. In the site A

- The GA analyzes data coming from the R-CBox of the site B and generates alerts. Then, it determines the approximate level of security of the site B. It sees that an intrusive activity occurs on the site B. For this reason, The GA sends an information alarm to the security administrator. Afterwards, it compares the real level of security of the site B to the approximate one and sees that they are similar. The GA concludes that both the LA and the R-CBox on the site B are not compromised.

The security administrator being informed by the GA that an intrusion is in progress in the site B, he blacklists the source of the attack and analyzes the alerts to see actions carried out by the hacker.

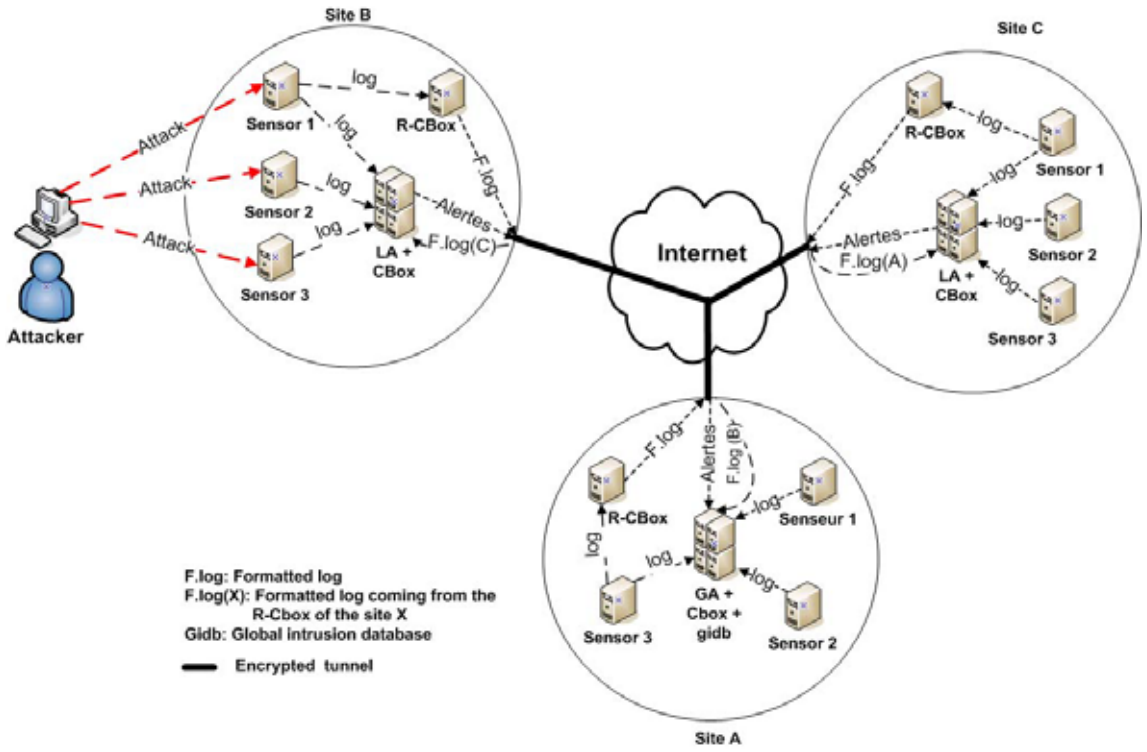

Fig. 8. Isolation attack of a site with the DSOC

Comments

The bandwidth usage during this test is shown in Fig. 9. The graph on this figure can be divided in 3 parts:
- In the first part, the alerts generated by the LA of the site B about the portscan are forwarded to the GA located in a remote site. This transfer of data uses 9 Kbps of bandwidth.
- In the second part of the graph, the R-CBox of the site B gathers data about the portscan and sends it to the GA. This transfer of data uses 14 Kbps of bandwidth.
- The third part of the graph presents the bandwidth usage when the LA of the site B transfers data about the attacks executed on the sensors to the Global Analyzer (GA).

We notice a variation of the bandwidth usage during the test with peaks around 55 Kbps when both the LA and the R-CBox forward data to the GA.

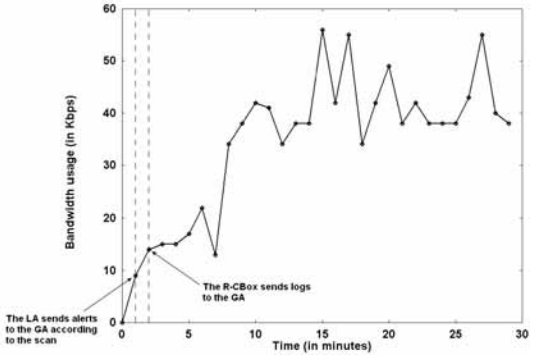

Fig. 9. Bandwidth usage when an LA forwards alerts to the GA

This test also shows that the DSOC makes it possible to see in real time that an intrusion is in progress in the site B. This permits to blacklist the source of the attack and to prevent the steal of data. A centralized SOC is unable to detect this kind of attack.

VII. Comparison between the SOCBox and the DSOC bandwidth usage

The comparison between the SOCBox and the DSOC bandwidth usage is shown on Fig. 10.
The first curve (at the top) shows the bandwidth usage when 2 sensors located in the same site send their log to a centralized SOC located in another site (for more details, sees Fig. 1). The results are obtained by flooding the sensors by Pings with packets of 1460 bytes during 30 mins. The sites are connected with a 25 Mbps link.
The second curve (at the bottom) shows the bandwidth usage when a DSOC monitors the security of 2 sites A and B (more details about the test are given in Fig. 8). The GA is located in the site A and an LA is installed in the site B. The sensors are flooded in the same conditions than the centralized SOC (the first curve) and some attacks are executed against then sensors during the flood.

In this test, any centralized SOC would give the same result than the SOCBox.

Conclusion

With the DSOC, we use around 443 time less bandwidth than with a centralized SOC. This is explained by the follow-

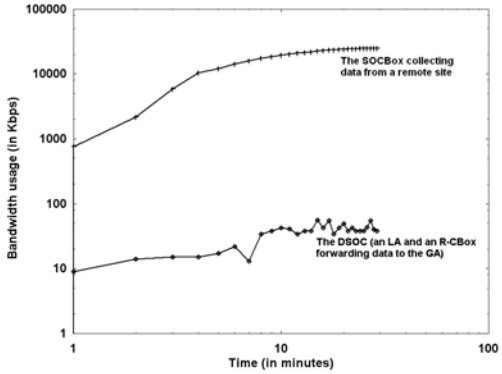

Fig. 10. Comparison between the SOCBox and the DSOC bandwidth usage

ing facts:
- When a centralized SOC gathers data from a remote site, data are forwarded to it in a raw format. The more an attack is strong, the greater forwarded data is.
- With the DSOC, intrusions are detected by the LA locally in the site B. Afterwards, the alerts are merged and correlated before transmitting to the GA. This reduces the quantity of data to forward to the GA.

VIII. 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 like distributed ones [1] [8] [9]. 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 multi-cast message is sent to the other hosts which check the integrity of the message and blacklist the source of the intrusion if need be.
Cooperation of IDSs is still an ongoing work [6] [16]. 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 [16], 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.

IX. Conclusion

Intrusions are clearly taking place and there is thus a need for operational supervision systems today. Our DSOC demonstrates its capability to detect intrusions and to present their status clearly. It also proves its ability to compact similar alerts and to correlate alerts coming from heterogenous platforms in several sites to detect more complex intrusions.
However, the development of some functionalities of the Global Analyzer (the management of the LAs) must be accomplished to make our system entirely operational. This will ensure the system scalability and messages will be processed better.

References

[1] A. Ajith, R. Jain, T. Johnson, Y. H. Sang, and S. Sanyal. D-SCIDS: Distributed soft computing intrusion detection system. *Journal of Network and Computer Applications, Elsevier Science Direct*, 2005.
[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] F. Cuppens. Managing alerts in a multi-intrusion detection environment. In *17th Annual Computer Security Applications Conference 2001*, New-Orleans, December 2001.
[4] A. K. Ganame, J. Bourgeois, R. Bidou, and F. Spies. Evaluation of the intrusion detection capabilities and performance of a security operation center. In INSTICC Press, editor, *International Conference on Security and Cryptography*, pages 48–55, August 2006.
[5] A.K. 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.
[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*, May 2005.
[9] C. Li, Q. Song, and Zhang C. MA-IDS architecture for distributed intrusion detection using mobile agents. In *Proc. of the 2nd International Conference on Information Technology for Application (ICITA)*, pages 451–455, May 2004.
[10] Stephen Northcutt and Judy Novak. *Network Intrusion Detection*. ISBN: 0-73571-265-4. New Riders, third edition edition, 2002. September.
[11] Openwall-Project. Scanlogd 2.2.6: A port scan detection tool http://www.openwall.com/scanlogd, 2006.
[12] R. F. Puppy. Nikto 1.35: An open source web server scanner. http://www.cirt.net/code/nikto.html, 2006.
[13] S. Staniford-Chen, B. Tung, and D. Schnackenberg. The common intrusion detection framework (cidf). In *Information Survivability Workshop*, Orlando, October 1998.
[14] Iv2 Technologies. http://www.iv2-technologies.com.
[15] THC. The hacker's choice, thc releases, thc-hydra v5.2. http://www.thc.org/releases.php, 2006.
[16] 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.
[17] Zti-Telecom. Ip traffic (2.3), a test and mesure tool. http://www.zti-telecom.com/fr/pages/iptraffic-test-measure.htm, 2005.