A Scalable SIEM Correlation Engine and its Application to the Olympic Games IT Infrastructure

Valerio Vianello, Vincenzo Gulisano, Ricardo Jimenez-Peris, Marta Patiño-Martínez
Facultad de Informática
Universidad Politécnica de Madrid
Madrid, Spain
{vvianello,vgulisano,rjimenez,mpatino}@fi.upm.es

Rubén Torres, Rodrigo Díaz, Elsa Prieto
Atos Research & Innovation
Atos
Madrid, Spain
{ruben.torres, rodrigo.diaz, elsa.prieto}@atos.net

***Abstract*— The security event correlation scalability has become a major concern for security analysts and IT administrators when considering complex IT infrastructures that need to handle gargantuan amounts of events or wide correlation window spans. The current correlation capabilities of Security Information and Event Management (SIEM), based on a single node in centralized servers, have proved to be insufficient to process large event streams. This paper introduces a step forward in the current state of the art to address the aforementioned problems. The proposed model takes into account the two main aspects of this field: distributed correlation and query parallelization. We present a case study of a multiple-step attack on the Olympic Games IT infrastructure to illustrate the applicability of our approach.**

Keywords—SIEM; CEP; Complex Event Processing; Scalability; Olympic Games; low and slow, brute force

I. INTRODUCTION

Current Security Information and Event Management (SIEM) technology is not scalable. Basically, it relies on centralized servers, and when the load for a server is too high, either the number of events processed by the SIEM server is reduced artificially (by filtering events) or the load is split across different servers in disjoint manner [1] [2] . The latter approach results in the inability to correlate events processed at different SIEM servers. However, there is an increasing need to use the SIEM technology to protect large scale Information Technology (IT) infrastructures by taking into account even more hard security requirements.

In this paper we describe how we dealt with scalability issues of SIEMs in the MASSIF project [3] , and in particular we focus on a new technology based on parallel-distributed complex event processing (CEP) [4] . We demonstrate that it is suitable to improve the capabilities of the SIEM correlation servers and satisfy the requirements above. We defined the operational requirements of the correlation engine based on the challenging use case of the Olympic Games IT infrastructure [5] , whose security is managed by one of the project partners, namely the Atos Company. Indeed, the success of the Olympic Games events highly depend on the IT infrastructure, and due to its particular nature of being a non-repeatable event, it requires very high levels of security. On the other hand, the Olympic Games IT infrastructure has grown significantly and it is posing difficult challenges to cope with the security events that the SIEM has to process. The very nature of the Games' IT security operations, which involve initial planning and build up until the last-minute massive deployment, causes huge sudden peaks in the number of events produced and highlights the need to correlate distant and potentially related events. Security events produced during the preparation and build-up stage, in which testing of the whole IT infrastructure implies many configuration changes and temporary decreases in the security levels, need to be able to be correlated with others produced live during the Games, weeks or months later. In this sense, there is a clear need for a versatile, elastic, and scalable correlation engine.

The MASSIF SIEM is a next generation SIEM system that provides a number of novelties. In this paper we focus on one of its most innovative features: the scalability of the correlation engine. Specifically, a number of security directives or rules are typically defined and deployed on the correlation servers in order to detect the suspicious activities preceding an attack; the directives describe how to identify the most significant security events occurring in the infrastructure. The SIEM workflow operates by collecting events from the monitored systems and by correlating them in order to check for conditions defined in the rules above. Finally, the SIEM raises alarms when such conditions are verified. These rules are basically continuous queries over streaming events. When the stream of events is too large to be handled by a single node, then a centralized complex event processing (CEP) system cannot simply process the stream of incoming events. Parallelization into different multiple nodes also signifies improvements in another event correlation dimension: increased memory for storing past events to effectively extend the time window in which isolated but related events can be correlated, and enable detection of "low and slow" attacks, among others. As "low and slow" attacks we understand those where an attacker tries to tamper the system, but the stages of the attack are distant in time to prevent detection by the SIEM system.

In MASSIF, in order to scale this complex event processing we have designed and implemented a CEP engine that is able to process CEP queries in a parallel-distributed manner. To this aim, the CEP queries can be split into subqueries deployable on different nodes. This is

978-0-7695-5008-4/13 $26.00 © 2013 IEEE
DOI 10.1109/ARES.2013.82

what is known as intra-query parallelism. However, this kind of parallelism might not be enough for massive event streams since the processing of the events for a subquery might require more resources than a single node. For this reason, in MASSIF, we have also implemented intra-operator parallelism. Intra-operator parallelism enables to run a subquery on a set of nodes to share the load of a single subquery across several nodes and, so, to scale the processing of massive event streams.

Operators are the building blocks of CEP queries. The CEP operators are used to filter (filter operator), transform (map operator), aggregate (aggregate operator) and correlate (join operator) events. The logic of the intra-operator parallelism is also encapsulated as special CEP operators.

The paper follows the following structure: in Section II the Olympic Games use case scenario as motivation of the work, taking a particular security rule as example. Then, in Section III, we map a security rule extracted from the scenario to a CEP query. This security rule is meant to reflect and detect a low and slow brute force attack, in which the events related to the attack can be numerous but separated by long periods of time and also be clumped together in high numbers at a time. In this section, the focus is more on the low and slow aspect of the attack. In Section IV, we present the MASSIF distributed processing event correlation engine and show how to translate a security rule into a CEP query and how this query is parallelized to reach the processing capabilities required to detect a brute force attack. This enables us to show how the implemented correlation engine is prepared to detect two extreme examples of attacks which can be part of the same attack (variable number of events very distant in time versus high peak numbers at a time). Finally, in Section V we present our conclusions.

II. OLYMPIC GAMES USE CASE SCENARIO

One of the use case scenarios used to demonstrate the MASSIF SIEM key features is the Olympic Games IT infrastructure. In this scenario, the main task for the MASSIF SIEM is to protect the IT infrastructure from any threat which can impact any part of the result chain and associated services. The Olympic Games, supported by the IT team of Atos, must successfully issue and activate more than 200,000 accreditations for Games that comprise around 300 events representing over 4,500 hours of live competition. Live commentator services are delivered for around 26 sports. More than 15 million information pages are viewed, with peaks of 1 million pages viewed on specific days. Over 3 GB of live results are provided in around 800,000 messages to the Olympic website, broadcasters and sports federations. This colossal event easily spans over more than 60 competition and non-competition venues, involving more than 10,500 athletes, 27,000 members of the accredited media or 70,000 volunteers. The intensity and complexity of this kind of sporting event presents a big challenge to ordinary SIEM infrastructure, mainly, due to two very characteristic features: the number of security event types (about 20,000), and the volume of generated events to be handled (around 11 million alerts per day).

In MASSIF project a testbed deployment has been created to simulate a simplified environment of the Olympic Games IT infrastructure in order to reproduce some of the misuse cases defined in this scenario, such as low and slow attacks, vulnerability scanning, and privilege escalation. Some Games services are accessible over the Internet. This includes the accreditation and sport entries applications that national Olympic committees use to register their athletes and dignitaries. Being one of the few external entry points to the Games infrastructure, the attacker can start from there and scan for well-known vulnerabilities. Figure 1. depicts the different servers considered in the simulation, as well as the misuse cases and their attack targets. The solid line represents the attack flow, whereas the dashed line represents the result of a successful attack.

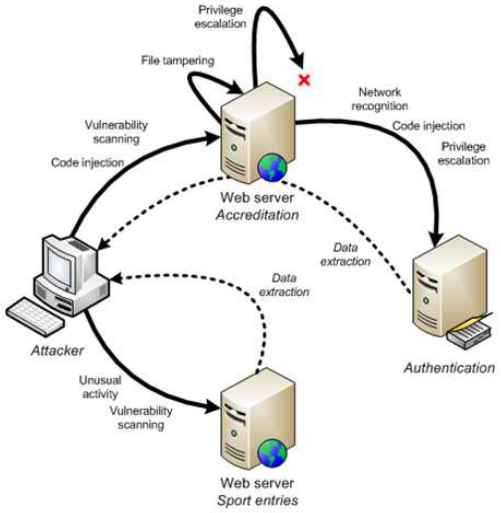

Figure 1. Olympic Games testbed

This scenario has highlighted the limits of current SIEM technologies. In most of the misuse cases defined in the Olympic Games scenario the malicious activities of an attacker can span over several days and even weeks. Correlation rules that are used in current SIEMs apply to short periods of time (e.g. numbers of recurrent events happening during an hour) and it is impossible to correlate isolated, but related, events that happen during long periods of time. Indeed, correlating events produced during the celebration of the games with others from weeks or months before represents itself a big challenge as well as a significant potential improvement to the security of the Games. This would enable the detection of potential attacks initiated during the preparation of the Games by internal Atos staff (as well as outside attackers), and meant to reach their climax once the Games are underway for maximum impact.

III. MAPPING THE USE CASE TO A CEP QUERY

Among the possible misuse cases defined into the Olympic Games scenario, we have focused on the one named “data tampering and privilege escalation on web server”, which reflects the first stage of a complex multi-step attack meant to eventually access accreditation data of the Games’ staff and competitors (among others), as depicted in the above figure. A malicious user decides to attack the Accreditation web server of the Olympic Games infrastructure by taking advantage of reported remote

execution vulnerabilities in the software stack used [6] [7] , in order to guess an admin password ("privilege escalation"). The attacker successfully deploys a malicious Java Server Page (JSP) implementing a shell into a web application run by the JBoss application server [8] ; we call this the "data tampering" step. To deploy the shell, the attacker leverages the remote execution to deploy a new JSP in the JBoss work folder. Upon normal invocation using a web browser, the JSP gets dynamically compiled and executed. Once the malicious JSP is deployed, the attacker follows a "low and slow" approach to brute force the password of a local Windows administrative account over two weeks; this is the called "privilege escalation" step.

In order to detect this attack on the MASSIF CEP correlation engine, the data format of the event stream has to be defined. An event stream is characterized by a schema that defines all the relevant fields. In this particular misuse case all the information about the attack is collected by a Windows System iNtrusion Analysis and Reporting Environment (SNARE) agent, which becomes the only data feed for the CEP engine. The SNARE agent feeds a MASSIF component called "Generic Event Translator"[5] , which is in charge of extracting the data fields from the messages and forward them to the CEP. TABLE I. lists the fields extracted from the syslog messages used for the CEP input stream.

The attack is detected in two steps. First, events related to the creation of files into a specific server folder are identified. Once the first event has been detected, we look for a certain number of login failed events on the same server using an event window of 2 weeks. If the number of failed login attempts detected in that period of time exceeds a defined threshold, an alarm is generated.

Figure 2. shows graphically the query used to detect the "data tampering and privilege escalation on web server" attack.

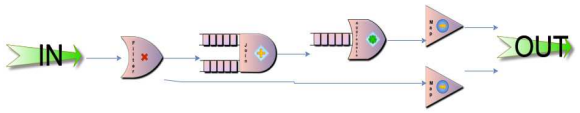

Figure 2. Privilege escalation: CEP query.

Five CEP operators compose the query. The semantic of each operator is listed sequentially below:

- Filter: it filters out all the input events but the one with EventType 4663, an object was created in a given folder path, or 4625, login failed.
- Join: it joins all the login failed events received after the reception of a 4663 event in a time window of 2 weeks.
- Aggregate: this operator counts the events coming from the Join and it generates a new output event when the number of received events in a time window of 2 weeks is greater than a predefined threshold.
- Map: the two map operators are used to generate alarms. The first one creates a low priority alarm any time a new file is created on the server (event 4663). The second map is used to create a high priority alarm any time the aggregate outputs a new event.

TABLE I. EVENT FIELDS EXTRACTED FOR CEP SCHEMA

Field Name	**Windows SNARE**	
	Description	***Example***
DeviceEventTime	Date when the corresponding event was detected.	Jan 24 07:35:59
InitUserName	Either the user name or the application name (account name) that generated the event.	jboss
TargetUserName	Target hostname and user name for the event generated.	MASSIF-1\jboss | MASSIF-1\ma
EventType	The four digit number to identify the Windows event type	4663 | 4625
EventShortMessage	Human readable version of the event message	An attempt was made to access an object. | An account failed to log on
InitAssetID	Caller process name that generated the event	cmd.exe
FullTargetDataName	Object name provided as target data by the InitAssetId	C:\JBoss\server\SEQ\tmp\a02f-dq04er-hcc1yfg9-1-hcc1zflw-9m\SEQ.war\About.jsp
TargetDataName	Folder where the object / path FullTargetDataName above is located	C:\JBoss\server\SEQ\tmp\
StatusCode	Failure/Status code, in an 8 digit HEX format	0x0000006d
AccessMask	A write access attempt increases the severity of the event	0x6
EventLongMessage	The full log entry	Jan 24 07:36:00 MASSIF-1 MSWinEventLog 4 Security 22203 Thu Jan 24 07:35:59 2013 4663 Microsoft-Windows-Security-Auditing MASSIF-1\jboss N/A Success Audit MASSIF-1 File System An attempt was made to access an object. Subject: Security ID: S-1-5-21-1171625486-3579515180-3666504820-1002 Account Name: jboss Account Domain: MASSIF-1 Logon ID: 0x6d8dab Object: Object Server: Security Object Type: File Object Name: C:\JBoss\server\SEQ\tmp\a02f-dq04er-hcc1yfg9-1-hcc1zflw-9m\SEQ.war\About.jsp Handle ID: 0x5c Process Information: Process ID: 0x7cc Process Name: C:\Windows\System32\cmd.exe Access Request Information: Accesses: WriteData (or AddFile) AppendData (or AddSubdirectory or CreatePipeInstance)

Finally, Figure 3. shows a screenshot of the test environment where the experiment was run. It shows the

messages related to the events that pass the filter and also the messages corresponding to the alarms. The first correlation rule detects the write access to the JBoss application server folder and generates a first low-level alarm.Once this alarm is produced, the second rule detects 10 (in this case) failed login attempts distant in time and reports a higher-level alarm.

Figure 3. "Privilege escalation on web server" detection

IV. MASSIF SIEM PARALLEL EVENT CORRELATION ENGINE

MASSIF SIEM aims at protecting large and complex distributed systems. Attackers can exploit vulnerabilities at several points of these systems, from the network layer to the application layer. Furthermore, attacks cannot be detected just by monitoring a single system layer but it needs to collect and correlate all the data produced by sensors distributed across the system. These heterogeneous sensor data must be filtered, aggregated and correlated in order to detect and report potential attacks. The event correlation engine of the MASSIF SIEM is the component in charge of processing these data and sending alarms to other MASSIF components that will apply the countermeasures.

The MASSIF SIEM correlation engine is a parallel-distributed Complex Event Processing (CEP) system. The streaming operators available in the engine can be parallelized across an arbitrary number of nodes allowing the engine to scale with respect the input data volume. Queries are specified in XML schemas telling the engine which event patterns must be detected in the incoming event stream. A query is an acyclic direct graph of operators where each operator takes input events on their incoming edges and route output events on their outgoing edges. Queries are split into sub-queries and each sub-query can be deployed in an arbitrary set of nodes. The operators are organized into four categories:

- Stateless operators: these operators process one event at a time and the produced output events, if any, are based only on the information contained by the processed event. Operators such as Filter and Map are examples in this category.
- Stateful operators: these operators are equipped with a window in which they store a certain number of events. The computation of the output events is based on all the events stored in the window. Join and Aggregate are the most important examples of stateful operators because they allow performing operations such as aggregation and correlation.
- Database operators: these operators allow the CEP engine to correlate events with stored information in databases as well as to store events in a database.
- Parallelization operators: the Semantic Router and the Event Merger are the operators that encapsulate the parallelization logic. In particular these operators are in charge of managing the event routing among sub-queries when one or more operators are parallelized.

A. Parallelization Strategy

The parallelization strategy of the MASSIF correlation engine is completely encapsulated into the parallelization operators. These operators are deployed on the outgoing and incoming edges of each sub-query. Specifically, Semantic Router operators are in charge of routing the output events produced by a particular sub-query to the nodes running in the subsequent sub-queries. While Semantic Router operators are deployed on the outgoing edges of a sub-query, Event Merger operators are always deployed on the incoming edges because they merge the incoming streams from all the upstream nodes and feed the local sub-query with a single, timestamp-ordered stream.

Event Merger operators are independent from the operator categories deployed in the local sub-query, this means that their behavior is independent of the kind of operator and they perform a merge sort based on the timestamp field and feed the sub-query with a new single timestamp ordered stream. The Semantic Router operators have knowledge about the semantics of the operators deployed in the following sub-queries because they must guarantee that all events that need to be aggregated or correlated together are actually received by the same sub-query instance.

Let us consider a simple query composed by two sub-queries A and B. Consider that the query is parallelized deploying the sub-query A on m nodes, and the sub-query B on n nodes as shown in Figure 4. The Semantic Routers of sub-cluster A make sure that all the tuples that have to be correlated and aggregated together are routed to the same downstream node. The input mergers take care of providing a single merged stream of events that is timestamp ordered.

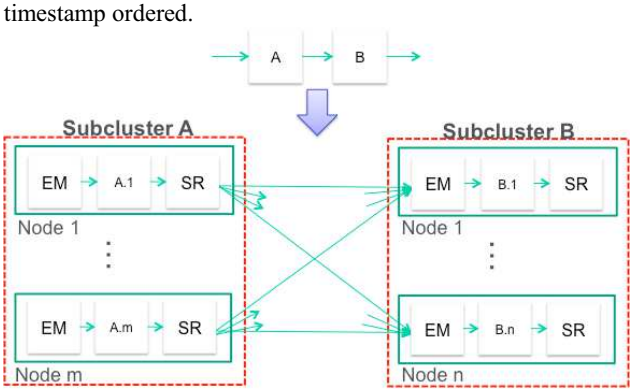

Figure 4. Simple query parallelization.

B. Scalability Evaluation

The scalability of the MASSIF CEP engine will be shown using the query from the Olympic Games scenario depicted in Figure 5. , the one about brute force attacks. The query is divided into two sub-queries, the first sub-

query contains one filter operator that let pass only events of failed login attempts, and the second sub-query contains the other three operators that are used to count how many login failed attempts happened in a certain time interval. We run two experiments, one experiment deploying the full query on a single node and another one parallelizing the first sub-query (the heaviest one) on 10 nodes and deploying the second sub-query on a different node. The data set containing the brute force attacks were created using an ad hoc data generator.

The evaluation is being made at UPM testbed. UPM testbed is a shared-nothing cluster of 100 nodes (blades) with 320 cores. All blades are Supermicro SYS-5015M-MF+ equipped with 8GB of RAM and 1Gbit Ethernet and a directly attached 0.5TB hard disk. Blades are distributed into 4 racks: Rack 1 has 20 blades, with a dual-core Intel PentiumD@2.8GHz. Rack 2 has 20 blades, with a dual-core Intel Xeon 3040@1.86GHz. Rack 3 and rack 4 have 30 blades, each with a quad-core Intel Xeon X3220@2.40GHz.

Figure 6. and Figure 7. show same statistic graphs related to the filter operator of the first sub-query during the two experiments. Each figure has six graphs, namely: Input Stream Rates that reports the number of tuple processed per second, Output Stream Rates that reports the number of tuples per second produced in output, Cost that indicates the percentage of the total CPU consumed by a particular operator of the sub-query, Queue length that indicates the size of the operator buffers, CPU that reports the CPU utilization in the node where the sub-query (containing the selected operator) is running, and finally Size that reports the number of nodes on which is deployed the operator. It is worth noting that the statistics reported on Figure 7. are the aggregated values of all the nodes employed for the operator.

Focusing on the size graphs, it is possible to see that, as we said before, in the first case the filter is deployed on 1 node only and in the second case on 10 nodes. Furthermore, comparing the two Input Stream Rates graphs we can see how moving from 1 node deployment to a 10 node deployment, the maximum supported load is increased from 20,000 tuples per second to 200,000 tuples per second. As can be seen the MASSIF CEP scales almost linearly because moving from 1 node deployment to an 11 node deployment the maximum supported load has increased by 10 times.

V. CONCLUSIONS

We have presented the scalable event correlation engine of the MASSIF SIEM and motivated it through a real scenario such as the one of the Olympic Games IT infrastructure. This event correlation engine is able to parallelize arbitrary SIEM security rules that are expressed as CEP queries that detect potential attacks. The CEP queries can be deployed on a large number of nodes to scale with respect to i) the volume of the security event streams (increased processing capabilities) as well as to ii) the timeframe in which two related events might take place (increased memory and extended correlation window capabilities).

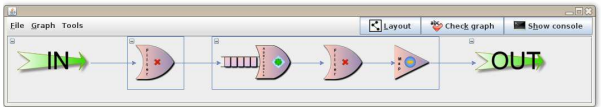

Figure 5. Brute force attack query.

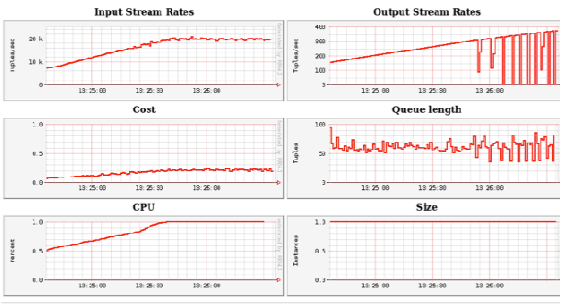

Figure 6. Statistics of the centralized deployment (1 node)

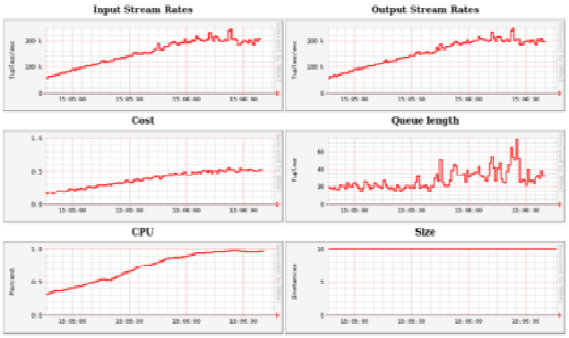

Figure 7. Statistics of the distributed deployment (11 nodes)

ACKNOWLEDGMENT

This work has been partially funded by the European Commission under Project MASSIF (FP7-257495), the Madrid Research Council under project CLOUDS (S2009TIC-1692) with funding from ESF and ERDF and the project CloudStorm funded by the Spanish Science Foundation (TIN2010-19077).

REFERENCES

[1] OSSIM Home page. http://www.alienvault.com/

[2] Prelude Home page. http://www.prelude-ids.com/en/

[3] MASSIF project Home page. http://www.massif-project.eu/

[4] Gulisano, V., Jiménez-Peris, R., Patiño-Martínez, M., Soriente, C., Valduriez, P., "StreamCloud: An Elastic and Scalable Data Streaming System". [Online]. Available: http://lsd.ls.fi.upm.es/lsd/papers/2012/2012_streamcloud.pdf

[5] Prieto, E., Diaz, R., Romano, L., Rieke, R., Achemlal, M., "MASSIF: A promising solution to enhance olympic games IT security". In: International Conference on Global Security, Safety and Sustainability (ICGS3 2011) (2011)

[6] Apache Struts 2, "About Apache Struts 2," 2013. [Online]. Available: http://struts.apache.org/2.3.8/index.html

[7] Struts 2, "S2-009 Security Bulletin - OGNL injection vulnerability," 2011. [Online]. Available: http://struts.apache.org/2.x/docs/s2-009.html

[8] Redhat , "JBoss Application Platform," [Online]. Available: https://www.redhat.com/products/jbossenterprisemiddleware/application-platform/