EVENT DRIVEN SECURITY

Nenad Stojanovski
Makedonski Telekom
Skopje, Macedonia

Marjan Gushev
Institute of Informatics, Faculty of Sciences,
St. Cyril and Methodious University,
Skopje, Macedonia

Abstract

This paper will present the new trend in computer security which is based upon system events and their use in new security technologies. The paper will first give some outlook on system events with a main focus on security. Later on we will give a detailed use of logs and logging technologies in event driven security and event driven security systems and a short outlook of a system that is capable of event driven security. As a conclusion, one possible use of event security will be given with focus to the main purposes of that system.

Introduction

System events and their logging is a measure that is used in IT as a multi purpose driver. The main use of the event logging is the need for troubleshooting system failure, but in the last decade it is also used for computer security purposes. Since its begining computer security has widden its domains of operations and now it is concerned with defense and offense methodologies. The main purpose of the event logs is system forensics.
In the last last years, the trend shows that the event logging can be used also for automatization of the defense technology of the systems. By knowing the pattern of the attack, it is possible to make a service that will defend the system from attacks for which it has pattern definitions. On the other side it is possible to use this logging information to automize procceses is such way that user interaction isn't needed and all the administrative costs can be erased.

In this paper we will present a new trend in computer security that is driven by security events. This concept will be analyzed and used to describe different methods for using it and different technologies where it can be implemented.

System logs and their purpose

With the advances of computer science and development of the operating system, the need of logging the states of the operating system was introduces. The whole logging concept was introduces to log the failure states of the operating system and the applications that were running on it.
Historically, Syslog was the first ever service that logged events and it was developed as part of the Sendmail project in the 1980s. Today, Syslog is the de facto standard for logging Unix and Linux events. On the other side, when we speak of Microsoft Windows, then the logging concept on this operating system is slightly different. The Windows logging concept is different than the Unix concept, because it logs the events in three or more different files. Depending on the type of event, Microsoft Windows writes the event in the appropriate file. By default Microsoft Windows uses three files, which are named Application Log, System Log and Security Log. Some applications that are developed for Microsoft Windows use their own Event Long files to log events that concern them.
Generally, we can group events based on two philosophies. The first one is based on the severity of the event that occurred. The second classification is by using the type of the event that occurred, whether it is a security, system or application event.
As an example, to explain the classifications we will use the following two events:

- Kernel crash
- User log in

The first event represents a crash of the Operating System kernel, which would mean that the whole system will become unusable. If we use the first classification, then this event would be a critical event. On the other side of we use the second classification; we can classify the event as a System event, since it concerns the system.
The second event represents a user log into the Operating System. This means that the user has entered his user/password correctly and that he is allowed to log into the system. According to the first classification, the event can be classified as an event with a low to normal severity. The second classification would classify the event as a security event.
One of the most interesting part is the comparison between the Unix event logging and the Windows event logging. The comparison between this two logging technologies is shown in table 1.

	Severity logging	Event type logging
Unix Syslog	Yes	No
Windows Event Long	No	Yes

Figure 1: Logging technologies.

Security events

When logging system events, depending on the system, we can have different types of event logging. As described before, the most common type of event model is either the Unix model or the Microsoft Windows model. By using either of them we can define some kind of event nomenclature that can be used for different purposes according to its needs.
When speaking of system events it is important to define the meaning of a security event according to Unix Syslog model and according to the Microsoft Windows model. The security

logging in Unix is done in such way that the logging is done in a centralized repository together with the system logs, kernel logs and boot logs. Although there is a never syslog concept with log segmentation in different files, most of the unix systems use the old logging concept. The following is a unix security log example:

```
Jan  8 09:42:57 unix_server sshd[1683]: Server listening on 0.0.0.0 port 22.
Jan  8 09:42:57 unix_server sshd[1683]: Server listening on :: port 22.
Jan  8 09:43:20 unix_server pam: gdm-password[2010]: pam_unix(gdm-password:session): session opened for user user1 by (uid=0)
Jan  8 11:32:37 unix_server sudo: user1 : TTY=pts/0 ; PWD=/home/user1 ; USER=root ; COMMAND=/bin/bash
Jan  8 12:27:34 unix_server su: pam_unix(su-l:session): session opened for user root by user1(uid=500)
Jan  8 12:33:12 unix_server su: pam_unix(su-l:session): session closed for user root
Jan  8 12:33:19 unix_server su: pam_unix(su-l:session): session opened for user root by user1(uid=500)
```

Microsoft Windows on the other side has a logging model that segments the security logs in a single file. Another difference from the syslog logging model is that every event in Microsoft Windows has its own Event ID and event description. By using this concept, Microsoft has given the security engineers a way to read and understand the security logs much easier. In this way the logs can be filtered by using Event IDs and they can be understood very easily by reading the Event description. The following is an example of Event IDs and their descriptions:

```
Event ID 529 : Unknown user name or bad password
Event ID 530 : Logon time restriction violation
Event ID 531 : Account disabled
Event ID 532 : Account expired
Event ID 533 : Workstation restriction, the user is not allowed to logon at this computer
Event ID 534 : Inadequate rights for console login.
```

When speaking of security events and logging models it's very hard to chose a model that is better than the other one, because they both do the jobs, and they both have their pros and cons.

System events that are connected with security occur almost all the time, but depending on the user actions it can happen that several security events occur for that single action. It's very hard to project the number of security events that will occur on one system, or how they will be related to it. For example, when a user executes a user action that logs him into the system with his domain account, the system writes 3 events to the event logs. On the other side, when a user tries to log into an FTP server only one event appears in the logs, the one from the FTP service.

It is very important, from security perspective, to know the occurrence of security events and to know how these events appear and under which circumstances. This is very important mostly for forensics, because the audit trail is the most relevant source for conclusions regarding security incidents.

Event Driven Security

Previously we've seen that almost every change in the system is logged. Depending on the definition in the logging system the Event is sometimes classified in a category or it's written as an error in the logs. By having the knowledge that everything is logged, it is possible to define a way how we will defend the systems based on the events that arise in those systems. By definition, event driven security is a security system that responds input from the user (mouse movement, keystrokes, menu choices, etc.) or from messages from other applications. This is in contrast to a batch operation that continuously processes the next item from a group.

By following the definition of event driven security, we can conclude that in order to have event driven security system, we have to monitor the events that arise in the monitored systems. Since we know that every event is written in some log file we would have to monitor those log files for some type of events. The main purpose of the event driven security system is to protect the assets by responding to malicious events. Because it is an event driven security system, when it detects any of the events that are specified as malicious it will execute some kind of action of the destination systems.

For example, let's say that we have an Event Driven security defense system that protects the user accounts from being locked. That system will monitor the event log on the domain controller for events of the type “Event ID 529: Login failed”. In order to protect it, the event driven system will start acting after the third failed login where it will send a command to the domain controller to block the IP address that had 3 failed logins.

Sometimes attacks are much more complicated and they don't appear straight forward as described in the example above. The anomaly that happens when they happen is connected with more than one event, and those events appear in different log files. In order to detect and defend against this type of attacks the event driven system should use event correlation. By using event correlation we are able to define sequential list of events that occur during some attacks. This technique will allow us to stop the attack before the last step of the attack occurs.

For example, the attack against our system happens in three steps where the steps are:

1. login to the operating system
2. warning in the system log that service X is failing
3. error in the system log that service X has stopped responding

In order for the event driven system to protect against this type of attack we should define a 2 step event correlation where the rule will be:

1. login to the system
2. warning in the system log that service X is failing after x seconds from the login event

When this rule is detected by the event driven system it should send an action to the attacked system to block the access from the attacking IP address.

I. Event driven security systems

One of the most comprehensive and notable systems that is capable of event driven security is ArcSight Enterprise

Security Manager. Althoug the main purpose of ArcSight ESM isn't event driven security, it is capable of executing actions upon certain events. ArcSight ESM is a systems that does log collection and monitoring in real-time. The concept of event driven securityu lies in defining rules which are based upon events that are collected from the logs. By doing this, users of this system are capable of creating triggered actions upon certain events or upon a set of correlated events. This rules also allow the users to define actions that should be executed when the conditions are satisfied. The commands can be executed via a script which can also be able to connect to the target systems and execute certain commands.

II. Conclusion and future work

In this paper we showed how it is possible to use system logs for automated defense. By doing this, we also showed that by using the pattern idea it is possible to build defense strategies against more complicated types of attacks. The main concept that was used was the event driven methodology, which is also the core of the event driven security. By using this core methodology, we are able to build or define different type of systems that can work by using the event driver.

The future work concerning this topic will be the use of the security driven methodology in the firewall rule technology. Mainly, the event driven methodology will be used in defining dynamic firewall rules. The purpose of the dynamic firewall rules will be full automation of the process of ip address changes that occur in networks.

Future work, will be based on event driven security and its integration into firewall systems. By integrating event driven security into the firewall system, we will be able to build an identity based firewall system.

References

[1] S. Garfinkel, G. Spafford, "Practical UNIX & Internet Security" *O'Reilly*, April 1996.

[2] "*Windows Event Log"*, http://msdn.microsoft.com/en-us/library/aa385780(VS.85).aspx .

[3] "ArcSight User manuals", http://www.arcsight.com, 2010.