https://doi.org/10.33778/kcsa.2022.22.2.003

Linux 환경에서 사용자 행위 모니터링 기법 연구

한 성 화*

요 약

보안 위협은 외부에서 발생하기도 하지만 내부에서 발생하는 비율이 더 높다. 특히 내부 사용자는 정보 서비스에 대한 정보를 인지하고 있기에 보안 위협에 의한 피해는 더욱 커진다. 이러한 환경에서 중요 정보 서비스에 접근하는 모든 사용자의 행위는 실시간으로 모니터링되고 기록되어야 한다. 그러나 현재 운영체제는 시스템과 Application 실행에 대한 로그만을 기록하고 있어, 사용자의 행위를 실시간으로 모니터링하기에는 한계가 있다. 이러한 보안 환경에서는 사용자의 비인가 행위로 인한 피해가 발생할 수 있다. 본 연구는 이러한 문제점을 해결하기 위하여, Linux 환경에서 사용자의 행위를 실시간으로 모니터링하는 아키텍처를 제안한다. 제안하는 아키텍처의 실효성을 확인하기 위하여 기능을 검증한 결과, 운영체제에 접근한 모든 사용자의 console 입력값과 출력값을 모두 실시간으로 모니터링하고 이를 저장한다. 제안한 아키텍처의 성능은 운영체제에서 제공하는 식별 및 인증 기능보다는 다소 늦지만, 사용자가 인지할 수준은 아닌 것으로 확인되어, 충분히 실효적이라고 판단되었다.

Real-time user behavior monitoring technique in Linux environment

Sung-Hwa Han*

ABSTRACT

Security threats occur from the outside, but more often from the inside. In particular, since the internal user knows about the information service, the security threat damage caused by the internal user is greater. In this environment, the actions of all users accessing information services should be monitored and recorded in real-time. However, the current operating system records only the logs of system and application execution, so there is a limit to monitoring user behavior in real-time. In such a security environment, damage may occur due to user's unauthorized actions. To solve this problem, this study proposes an architecture that monitors user behavior in real-time in a Linux environment. As a result of verifying the function to confirm the effectiveness of the proposed architecture, the console input values and output angles of all users who have access to the operating system are monitored in real-time and stored. Although the performance of the proposed architecture is somewhat slower than the identification and authentication functions provided by the operating system, it was confirmed that the performance was not at a level that users would recognize, and thus it was judged to be sufficiently effective. However, since this study focuses on monitoring the console behavior, it is impossible to monitor the behavior of user applications running in the background, so additional research is needed.

Key words : Real-time monitoring, User Authentication, Console In/Output, TTY Session, Background Application

접수일(2022년 02월 24일), 수정일(2022년 03월 28일), 게재확정일(2022년 05월 06일)

* 동명대학교/정보보호학과(주저자, 교신저자)

1. 서 론

최근 Smartwork나 재택근무 환경 등이 활성화 되면서, 보안 위협의 범위는 더욱 커지고 있다[1]. 보안 위협의 시작은 외부에서 발생할 수 있으나, 그 방법이나 과정, 주체는 내부 사용자에 의해 발생하는 경우가 많다.

내부 사용자는 보안 위협의 대상인 정보 서비스에 대해 상당한 지식을 가지고 있는 만큼, 보안 위협은 체계적이며 그 피해도 상대적으로 크며 계속 발생할 수 있다[2]. 일부 보안 솔루션이 명령어 통제 기능을 제공하고 있다. 그러나 이 명령어 통제 기능은 확인된 시스템 명령을 통제할 수 있을 뿐, 공격자가 생성한 system command나 malware C&C message는 차단할 수 없는 한계가 있다[3].

이러한 보안 환경에 대비하여, 사용자의 정보 서비스 접근 이후의 행위를 모니터링하는 보안 기능이 요구되고 있다. 현재 대다수 운영체제는 시스템 로그나 사용자의 Application 실행 로그 등의 제한적인 모니터링 기능을 지원하고 있다[4]. 그러나 엔터프라이즈 환경에서 요구되고 있는 실시간 모니터링 기능은 제공하지 않고 있다.

이러한 문제점을 해결하기 위하여, 엔터프라이즈 환경에서는 모든 정보 시스템에 접근하는 사용자의 모든 행위를 모니터링할 수 있는 보안 기능을 요구하고 있다[5].

본 연구는 이러한 요구사항을 만족하기 위하여, 많은 정보 시스템 중 Linux 환경에 접근한 사용자의 행위를 실시간으로 모니터링할 수 있는 아키텍처를 제안한다. 제안한 사용자 실시간 모니터링 기능의 기능과 성능을 검증하여 그 실효성을 검증한다.

2. 관련 연구

2.1 엔터프라이즈 환경 보안 요구사항

IT융합 트렌드를 바탕으로, 정보 서비스는 갈수록 그 범위가 확대되고 있으며 그 종류도 많아지고 있다. 특히, Legacy 정보 서비스와 연동하는 정보서비스가 많아지면서, 최근 개발되는 정보 서비스의 복잡도도 크게 증가하고 있다[6].

이러한 환경에서 정보 서비스에 접근하는 내부 사용자는, 연동하는 다른 정보 서비스에도 쉽게 접근할 수 있게 된다.

이러한 보안 환경은, 내부 사용자에 의한 보안 위협 발생 가능성과 그 피해 규모를 크게 증가시키고 있다.

내부 사용자는 조직 내부에서 운영하는 정보 서비스의 목적과 범위, 동작 방식에 대한 정보를 쉽게 취득할 수 있어, 해당 정보 서비스에 접근하여 비인가 정보를 유출하거나 변조할 수 있다[7].

이러한 문제점을 해결하기 위하여, 엔터프라이즈 환경에서는 내부 사용자가 정보 시스템에 접근하였을 때, 이를 실시간으로 모니터링하는 보안 기능을 요구하고 있다[8].

2.2 운영체제 모니터링 기술 현황

기본적으로 현존하는 모든 운영체제는, 운영체제에 접근하는 사용자에 대한 식별 및 인증 기능을 지원하며, <표 1>과 같은 다양한 시스템 로그를 기록한다. 또 설정에 따라, 사용자의 Application 실행과 종료 이력도 기록할 수 있다[9,10,11,12].

<표 1> Linux 로그 종류

Log File	Daemon	설명
/dev/console	kernel	Console Message Log
/var/log /messages	syslogd	Linux kernel system/application log
/var/log/secure	xinetd	User authentication log
/var/log/cron	crond	crond log
/var/log/boot.log	kernel	System booting log
/var/dmesg	kernel	Booting system messages
/var/log/wtmp	kernel	User login log
/var/log/utmp	kernel	Current user login log

다만, 운영체제에서 제공하는 로그 기록 기능은 시스템 이벤트에 대한 주체와 객체, 발생 시간 등의 최소한의 정보만을 기록한다. 또 조회 방식도 보안 관리

자에 의한 수동적인 조회만 가능하다.

3. 사용자 행위 실시간 모니터링 시스템

3.1 사용자 console 모니터링 시스템 구조

본 연구는, 엔터프라이즈 환경에서 요구되는 실시간 모니터링 기능 중, Linux 환경에 접근한 사용자의 console In/Output을 실시간으로 모니터링할 수 있는 사용자 console 실시간 모니터링 시스템을 제안한다. 사용자 console 실시간 모니터링 시스템은 정보 시스템에 접근한 사용자의 console 행위를 실시간으로 모니터링하고 필요한 경우 이를 로그로 기록한다.

(그림 1)은 본 연구에서 제안하는 사용자 console 실시간 모니터링 시스템 구조이다. 이 구조는 크게 3개의 컴포넌트로 구성된다. Console Log Viewer는 정보 시스템에 접속한 사용자의 행위를 모니터링한다. Console Log Server는 보안 관리자가 요청한 사용자의 TTY Session Log를 Console Viewer에 전달한다. Console Log Collector는 PAM Module을 통해 Loading 되어 사용자의 TTY Session을 복제한 후, 해당 Session에서 발생하는 사용자의 Text input과 output 내용을 수집하여 이를 로그 파일에 저장한다.

3.2 사용자 console 모니터링 동작 방식

console Log Collector는 사용자가 정보 시스템에 접속할 때 생성되는 TTY Session을 확인하여, 해당 Session을 Mirroring 하는 방식으로 모니터링을 시작한다. 사용자의 TTY Session에 In/Output이 발생하면 이를 단위 시간 단위로 로그를 생성하여 Console Log Server에 전달한다.

console Log Server는 사용자의 console 로그를 저장한다. 만약 console Log Viewer가 console Log Server에 접속했을 때는, 실시간 console 로그를 전달한다.

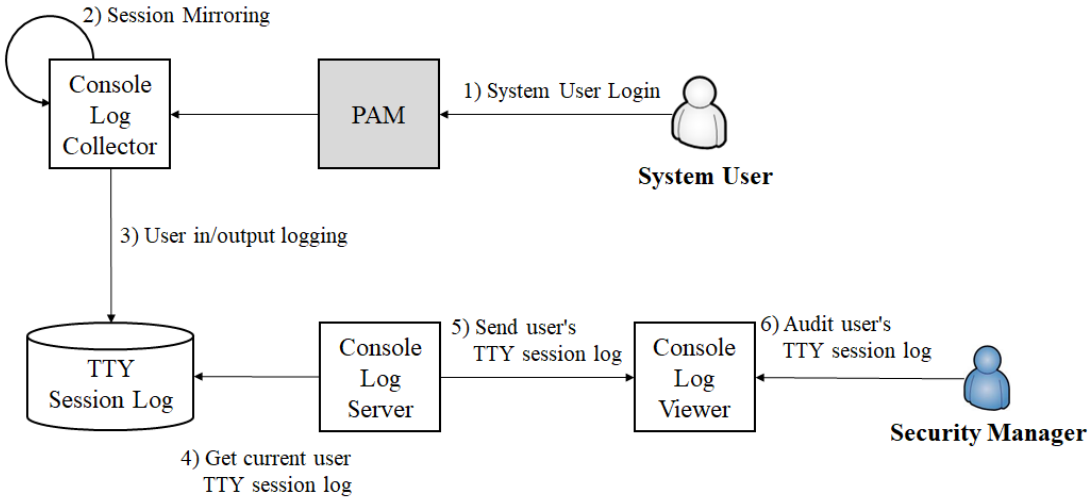
보안 관리자는 console Log Viewer를 실행하면, console Log Viewer는 Cosole Log Server에 접속하여 실시간 console 모니터링 시작 명령을 전달한다. 이후 console Log Server가 전달하는 실시간 로그를 조회한다.

4. 사용자 행위 실시간 모니터링 시스템 검증

4.1 보안 기능 검증

4.1.1 기능 검증 환경 및 검증 항목

본 연구에서 제시하는 사용자 Console 실시간 모니터링 시스템이 유효한지를 확인하기 위하여 기능을



(그림 1) 사용자 console 실시간 모니터링 시스템 구조

검사한다. 기능 검증을 위한 시험 환경은 i5-6500 CPU와 8Gbyte Memory, 256 Gbyte SSD Hardware, Redhat Enterprise Linux 8.4 운영체제 환경에서 진행하였다. 기능 검증 항목은 <표 2>와 같다.

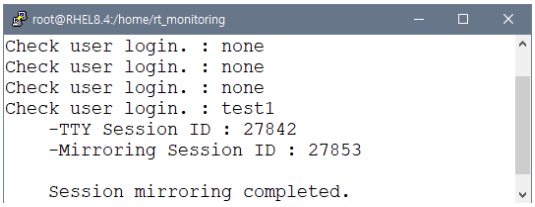
<표 2> 기능 검증 항목

Func_ID	검증 내용	검증 대상 컴포넌트
Pst_F_1	사용자가 Linux 환경에 TTY 접근을 하였을 때, 이를 모니터링 여부 확인	Console Log Collector
Pst_F_2	사용자의 TTY Session에서 발생하는 In/Output에 대한 Log를 생성하는지 확인	Console Log Server
Pst_F_3	보안 관리자가 실시간 로그 조회를 할 수 있는지를 확인	Console Log Viewer

4.1.2 기능 검증 결과

■ Pst_F_1

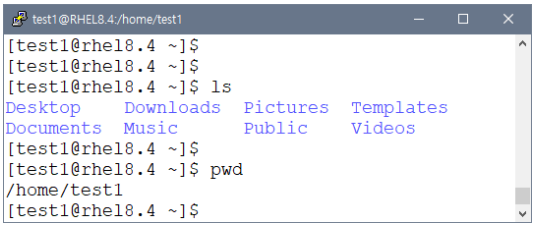
(그림 2)와 같이, 사용자가 Linux 환경에 접속하였을 때 Console Log Collector는 접속 Session을 확인하고 이를 정상 Mirroring 하는 것을 확인하였다.



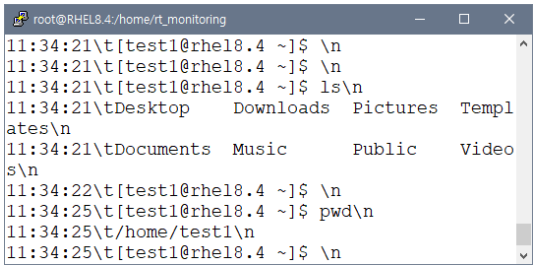
(그림 2) 사용자 TTY session mirroring 확인

■ Pst_F_2

사용자가 TTY Session에서 CLI Input/Output을 발생하는 경우, (그림 3)과 같이 Console Log Server에서는 단위 시간 별로 입출력된 Text에 대한 Log가 정상 생성됨을 확인하였다.



(a) 사용자 TTY session in/output

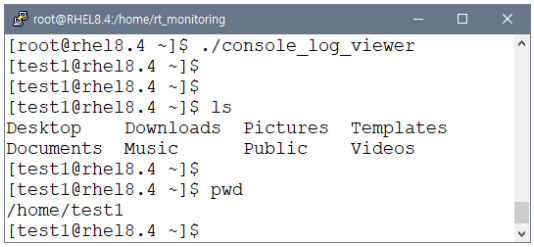


(b) 사용자 TTY mirror session log

(그림 3) 사용자 TTY session log 생성 확인

■ Pst_F_3

보안 관리자가 Console Log Viewer를 실행하면, (그림 4)와 같이 시스템에 접속한 사용자 TTY Session에서 발생하는 현재 In/Output Text를 실시간으로 조회할 수 있는 것을 확인하였다.

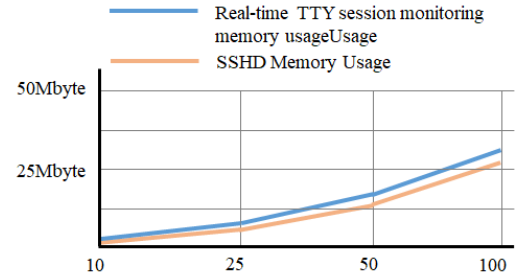


(그림 4) 사용자 TTY session 실시간 조회 확인

4.2 실시간 모니터링 성능 검증

본 연구에서 제안하는 실시간 모니터링 시스템이 동작할 때, 과도한 리소스를 소모하는지를 확인하기 위하여 리소스 사용 수준을 측정하였다.

성능 검증 환경은 기능 검증 환경과 같으며, TTY Session 별로 1분을 유지하면서 임의의 Random Text를 출력하였을 때, Memory 사용 정도를 측정하였다.



(그림 4) 실시간 TTY session 실시간 모니터링 기능 Memory 사용 확인

(그림 5)은 10, 25, 50, 100명이 동시 접속하였을 때의 메모리 사용량 측정 결과이다. 여기서 볼 수 있듯이 정보 시스템에 접속한 사용자 TTY Session을 실시간 모니터링하는 기능이 소요하는 Memory의 사용량은 사용자 접속 수 만큼 증가하고 있다. 그러나 전체 시스템 Memory 사용률을 크게 점유하고 있지는 않다.

5. 결 론

엔터프라이즈 환경에서 내부 사용자는 목표 대상 정보 서비스에 대한 정보를 알고 있다. 이러한 보안 환경으로 내부 사용자는 다양한 보안 위협을 가할 수 있으며, 이러한 보안 위협을 확인하기 위해서는 정보 시스템에 접근한 사용자의 행위를 모니터링 할 수 있는 보안 기능이 필요하다.

본 연구는 이러한 보안 요구사항을 만족하기 위하여 사용자 Console 실시간 모니터링 시스템을 제안하였으며, 목표하는 보안 기능과 성능을 검증하여 그 실효성을 입증하였다.

다만, 본 연구는 TTY Session에서 이루어지는 사용자의 In/Output Text만을 모니터링 하는 것에만 집중하였기 때문에, 사용자 Session의 백그라운드에서 발생하는 행위의 모니터링은 불가능하다. 그러므로 사용자 Application의 백그라운드에서 이루어지는 행위 모니터링에 대한 추가 연구가 필요하다.

참고문헌

[1] Eun-byol Koh, Joo-hyung-Oh and Chaete Im, “A study on security threats and dynamic access control technology for BYOD, smart-work environment”, Proceedings of the International MultiConference of Engineers and Computer Scientists. p.1-6, 2014.

[2] François Amigorena, “The threat from within: how to start taking internal security more seriously”, Computer Fraud & Security, vol.7, pp.5-7, 2014.

[3] Fedorov, V. K., Balenko, E. G., Shterenberg, S. I., and Krasov, A. V. “Development of a Method for Building a Trusted Environment by Using Hidden Software Agent Steganography”, In Journal of Physics: Conference Series, IOP Publishing, vol.2096, no.1, p.012047, 2021.

[4] Zeng, D., Wu, G., Pang, S., Chen, L. and Chen, X., “Research and implementation of campus network mass log collection platform based on elastic stack”, Computer, p.1216724, 2022.

[5] Salem, R. M., Saraya, M. S. and Ali-Eldin, A. M., “An Industrial Cloud-based IoT System for Real-time Monitoring and Controlling of Wastewater”, IEEE Access, 2022.

[6] Keun-Ho Lee, “Analysis of Threats Factor in IT Convergence Security”, Journal of the Korea Convergence Society, vol.1, no.1, pp.49-55, 2010.

[7] Malik Nadeem Anwar, Mohammed Nazir and Adeeb Mansoor Ansari, “Modeling security threats for smart cities: A stride-based approach”, Smart Cities—Opportunities and Challenges, Springer, Singapore, p.387-396, 2020.

[8] Li, R., Wang, Q., Wang, Q., Galindo, D. and Ryan, M., “SoK: TEE-assisted Confidential Smart Contract”, arXiv preprint, arXiv:2203.08548, 2022.

[9] Sohail, S. S., Khan, M. M., Arsalan, M., Khan, A., Siddiqui, J., Hasan, S. H.,, and Alam, M. A., “Crawling Twitter data through API: A technical/legal perspective”, arXiv preprint arXiv:2105.10724, 2021.

[10] Jia, Z., Shen, C., Yi, X., Chen, Y., Yu, T. and Guan, X., “Big-data analysis of

multi-source logs for anomaly detection on network-based system", 2017 13th IEEE conference on automation science and engineering (CASE) pp. 1136-1141, IEEE, Aug, 2017.

[11] Sakti, B., Aziz, A. and Doewes, A, "Uji Kelayakan Implementasi SSH sebagai Pengaman FTP Server dengan Penetration Testing", ITSMART: Jurnal Teknologi dan Informasi, vol.2, no.1, pp.44-51, 2013.

[12] Pawlikowski, K., "Log Parsing and Template Extraction Using Neueral Sequence-To-Sequence Models", Doctoral dissertation, Southern Connecticut State University, 2021.

[저 자 소 개]

한 성 화 (Sung-Hwa Han)
동명대학교 정보보호학과 교수
숭실대학교 공학박사
SW영향평가 전문위원
관심분야 : IT융합보안, 시스템보안, 인공지능, 악성코드 탐지, 제로트러스트 보안
email :shhan@tu.ac.kr