Monitoring and redirecting system calls on a Client Workstation

Fabio Oesch - Supervised by: Konstantinos Markantonakis and Raja Naeem Akram

Information Security Group, Smart Card and IoT Security Center

The Smart Card and Internet of Things Security Centre

Objectives

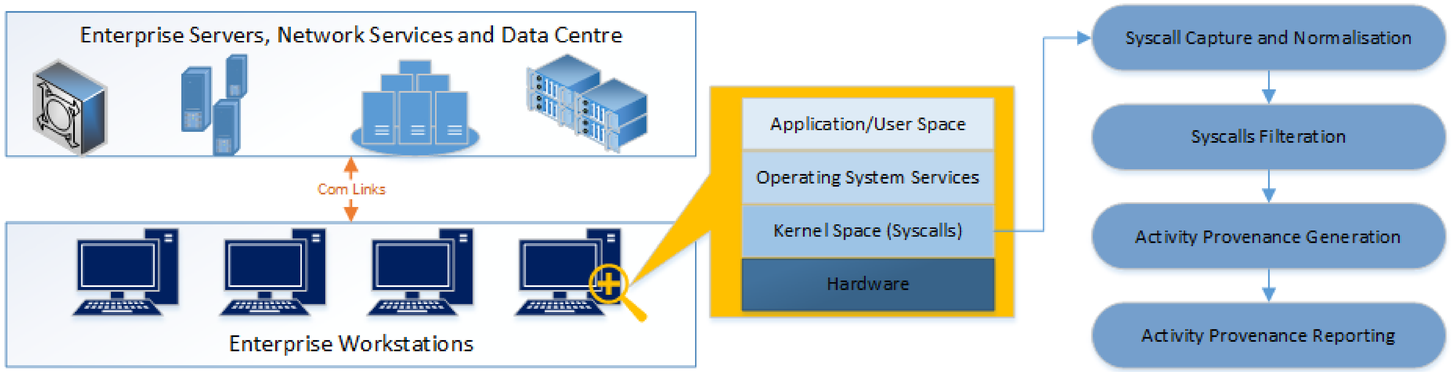
The main objectives for this projects are the following

- Collect activities on a selected Operating System at a granular level (Kernel Level)
- Filter the activity information and report only the most crucial events that has an impact on the overall security and privacy aspects
- The reporting of the events will be structured as a causality chain. The structure of causality chain will be explain later on during the project

Introduction

Monitoring events of a workstation is a difficult task, since we have to decide on what level we want to monitor it. On one hand if the monitoring is on a superficial level the system could be exploited by the user. Whereas when everything is monitored the amount of information gathered is on a scale which is not feasible to save. So a tradeoff between the vast amount of data monitored and what is saved is necessary. After the decision has been made about what is relevant to the privacy and security to the system this data is stored on a server which records these chains of events.

The project is still in its infancy. This means there are parts which still needs researching. The way system calls are saved on the server has yet to be touched. At this moment in time monitoring the system calls has priority. During the research multiple ways to approach the problem of system call monitoring have been considered.



Approaches

One is to create a **proxy layer** for the system calls. This means that the kernel of a Linux system is modified and instead of calling the system call itself, the proxy would be called first. This proxy will be able to save the system call and redirect the call to the actual location of the system call.

The current approach is using **eBPF (extended Berkley Package Filter)** which has functionality which could be of great use. For one it can monitor system calls without the need to modify a kernel. Another advantage is the number of tools which already exist can be useful and extended.

Known Issues

Some of the issues which are already known and have to be researched in further detail are

1. Saving system calls to a server will need to call a system call. This creates an infinite loop of saving and creating system calls. A solution could be to ignore system calls from specific pid's.
2. The amount of data to deal with is extremly large and has to be reduced. Otherwise the normal usage of the system could slow down to an unusable state. No solution for this issue has been suggested.

Future Research

The next steps will be to monitor and log these system calls. After this first hurdle of being able to monitor system calls and being able to store them. The next step would be to filter system calls and create event chains.

Additional Information

This project is related to the EPSRC funded project "Data to Improve Customer Experience (DICE)". The project is particularly interested in personal data, and is using rail passengers as a specific focus of interest. The overall aims of the project are:

- Understand the role that personal data plays in enhancing the user experience of rail passengers
- To develop technical solutions to data privacy
- To develop an evaluation framework that can be implemented so passengers can understand how their data is used and

how they can control and verify its use. The project started in October 2016, and runs for three years to September 2019. For more information about the project, please visit http://www.dice-project.org.

References

[1] Maximiliano Caceres and maximiliano Caceres@corest Com. Syscall proxying-simulating remote execution. 06 2018.

Acknowledgements

We acknowledge the support of the ISG-SCC for the summer internship program and EPSRC funded project. The views and opinions expressed in this poster are those of the authors and do not necessarily reflect the position of DICE project or any of partners associated with this project.

Contact Information

- Email: Fabio.Oesch.2016@rhul.ac.uk
- Phone: +44 (0)7449 162794