Høgskolen i Gjøvik

Software Security

Obligatory exercise #1

Trusted Path in Linux

Author:
120915 - Halvor Mydske Thoresen

September 3, 2014

1 Task

- Choose an operating system that does not have "Microsoft Windows" in its name, e.g. Mac OS, Linux, iOS, Android, Multics, Plan 9, Turaya, Solaris
- Find out and describe how the OS implements the concept of a trusted path (if it does)
- Include references to the sources that support your findings (scientific articles, developer documentation)

I chose to look into how Linux implements the concept of a trusted path.

2 Trusted Path

Trusted Path is the term used for a mechanic that guarantees legit communication between the user and what the user intended to communicate with and that the communication can't be intercepted or modified by an attacker. A common problem where the trusted path is needed is the "fake login screen". This is a program that simulates a login promt for the user and captures the users information. The program can then initiate the legit login prompt and ask the user to type the information again to eliminate suspicion. To provide access to a trusted path we use something called Secure Attention Key. [4]

3 Secure Attention Key (SAK)

Secure Attention Key is a command in form og a key or a combination of keys that will guarantee a legit login prompt and login program, and not a simulator that are trying to get hold of the users information. On Linux kernels, this has been supported since verson 2.2, however it is not implemented by default. [3]. The SAK provides this security by suspending all programs and logs you out before starting the legit login operation after the kernal detects that the right combination of keys has been pressed. [1]

4 Trusted Path Execution (TPE)

Trusted Path Execution provides protection against execution of files by adding restrictions based on their path. This means that even if an attacker successfully manages to escalate his privileges, he will not be able to execute custom binaries if they are not in the trusted path. TPE does not only protect against targeted attacks, but also provides a restriction for the users of the system. TPE does this by checking if either the user or the path is trusted. If true then the file wil execute, if not, an error message will be returned.[2]

References

[1] Andrew Morton. Secure attention key handling. https://www.kernel.org/doc/Documentation/SAK.txt, 2001. [Online; accessed 01-September-2014].

[2] Niki A. Rahimi. Trusted path execution for the linux 2.6 kernel as a linux security module. https://www.usenix.org/legacy/event/usenix04/tech/freenix/full_papers/rahimi/rahimi_html/index.html, 2004. [Online; accessed 3-September-2014].

[3] Bob Toxen. *Real World Linux Security: Intrusion Prevention, Detection, and Recovery*. Pearson Education, Inc, 2002.

[4] David A. Wheeler. Secure programming for linux and unix howto. http://www.tldp.org/HOWTO/Secure-Programs-HOWTO/trusted-path.html, 2003. [Online; accessed 29-August-2014].