Chapter 5

Addressing leaks through networking

Assuming the platform might be compromised with sophisticated rootkits, e.g. running in SMM or ME, that are actively trying to steal e.g. GPG private keys from the host memory, it is important to ensure the malware cannot leak the data using networking. It should be realized that for malware running in ME or SMM it might be possible to leak data using networking irrespectively of what specific networking hardware is in use by the host OS. It should be just enough for the malware to (asynchronously) find pages containing what looks like specific data structures (e.g. Linux sk_buff structures) and modify just a few fields there in order to implement some form of covert channel for exfiltration (see e.g.[8]).

On the other hand, such advanced malware (e.g. especially when running in the ME) might be reluctant to (somehow blindly) modify outgoing networking packets without fully understanding the bigger picture of the specifics of the environment and the user setup. This is because such modifications might easily be detected by more inquisitive users or admins, using more or less standard network analysis tools, risking detection of the malware. Again, for malware located that deep in the hardware, in the processor itself, this might not be acceptable. Nevertheless, let's discuss what we could do to prevent such leaks anyway. We will do that starting from the simplified scenario of an air-gapped system, then move on to increasingly more connected scenarios.