AutoRun-Logger (ARL)

Background

The majority of malicious software wants to have a persistance mechanism (Autoruns) so that it can survive reboots. The go to tool for extracting **autoruns** is the SysInternals Autoruns software, written by Mark Russinovich (Microsoft).

AutoRun-Logger is a three component system that allows organisations to have a network wide view of all of the autorun data of their Windows hosts.

The first component is a simple Windows service that extracts the autoruns data the first time it is run (e.g. boot up), and then periodically from that point. The service sends the data to the second component; the analysis server. The analysis server parses the data and imports it into a database. The server compares the new dataset with the previous, if any differences are identified then an **alert** is generated.

The final component in the system is the user interface (UI) server, that displays the alerts and allows the analyst to drill down into the alert. The UI server allows the analyst to view the current autorun data for a specific host. From the UI server, various data can be exported such as files containing the SHA256/MD5 hashes of all current autoruns. There is also a simple search facility to allow searches across the alert/autorun data for specific values.

Implementation

The Windows service is written using the Microsoft .Net framework. The service uses a local copy of the SysInternals Autoruns command line software to extract the autoruns data in XML format. The autoruns data is compressed using GZIP and is sent via TLS to the server.

The analysis server is written in the Go Programming Language (golang) and is designed to run on a Linux host. The server uses a PostgreSQL database to store the autorun data.

The UI server is again written using golang. The server implements its own HTTPS server so no other software is required.