AlmightyKeyLogger - Daniel Ayzenshteyn

Video demonstration: https://www.youtube.com/watch?v=FEbFMjtARMY

Summary

AlmightyKeyLogger is a user-level keylogger. It is a multi-desktop and multi-session keylogger. It bypasses Microsoft's secure desktop feature, thus it can log **EVERY** key press on the system, even on the secured Winlogon screen.

The AlmightyKeyLogger keylogs all sessions and desktops on the computer, "virtual" desktops (Win+Tab on Windows 11) are keylogged as well.
On screen keyboard is keylogged as well as it's creating the same messages as the regular keyboard.

The keylogger is persistent via the SCM and starts on boot time.
Log files are stored encrypted in the specified directory with the specified encryption key.

It requires Administrative privileges for installation (Runs as service).

Keylogger Usage:

The keylogger is installed via an installer app. After installation it is persistent on the system and keylogs all users and all desktops.

```
.\AlmightyKeyLoggerInstaller.exe –-install --f "logs_directory_path" --p
"password"
.\AlmightyKeyLoggerInstaller.exe --uninstall
.\AlmightyKeyLoggerInstaller.exe --start
.\AlmightyKeyLoggerInstaller.exe --stop
```

–install - Requires log directory to which the logs would be saved and an encryption key. The service is installed and saves the parameters in the registry. The service starts after installation.
–uninstall - Stops the service if running, uninstalls the service from the SCM and clears the registry.
–start - Start the service (Only if already installed)
–stop - Stops the service (Registry and persistence remains - will be loaded on boot time)

Decryptor Usage:

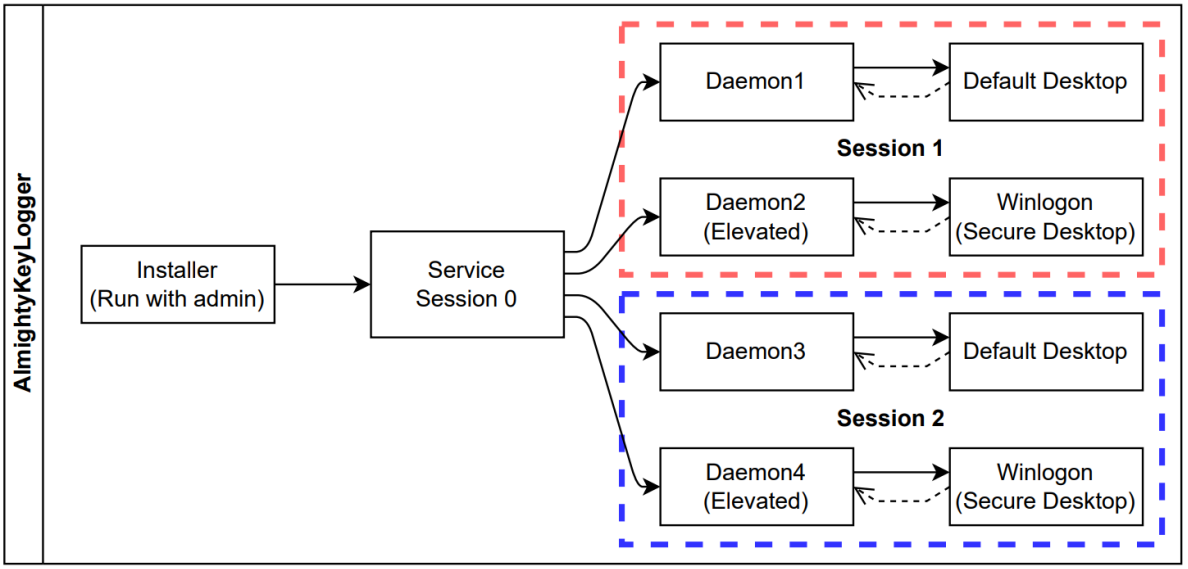
Decryptor.exe is an auxiliary tool, it decrypts and merges the log files to one file.

```
.\Decryptor.exe "logs_directory_path" "password"
"full_path_for_decrypted_log_file"
```

Architecture

The AlmightyKeyLogger consists of 3 parts:

1. The installer - provides a CLI to operate and manage the keylogger. It provides functionality to set the log directory, password for encryption, service installation, starting the service, stopping the service and uninstalling and clearing the registry.
2. The daemon - the keylogger itself, operates with a hook to WH_KEYBOARD_LL and keylogs everything on the desktop it's attached to, writes to an in memory buffer. runs a helper thread which encrypts and writes the buffer to the log file every T time.
3. The service - operates at session 0 and is responsible for detection of new sessions and injection of a daemon to the sessions. Starts a process with a user impersonated token in every default desktop in any session that is created. Additionally, it spots new winlogon desktops (Secured desktops) and starts an elevated daemon inside those desktops to secure critical credentials for Windows, Active Directory, UAC, KeePass, etc…



In the diagram above we can see the flow and relationship between those parts. The installer starts the service which itself starts the daemons, each of them is responsible for a specific desktop.

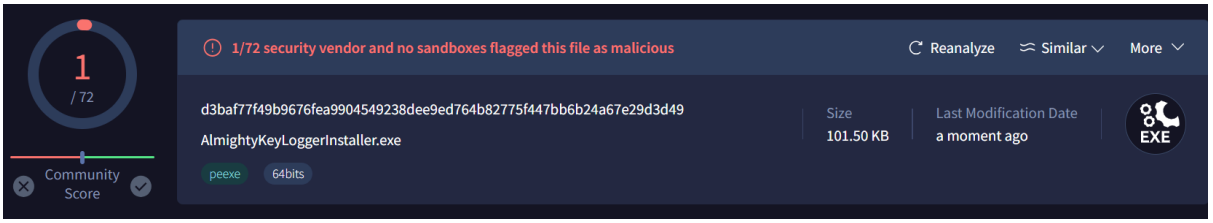
Virus Total - Detection Rate Evaluation

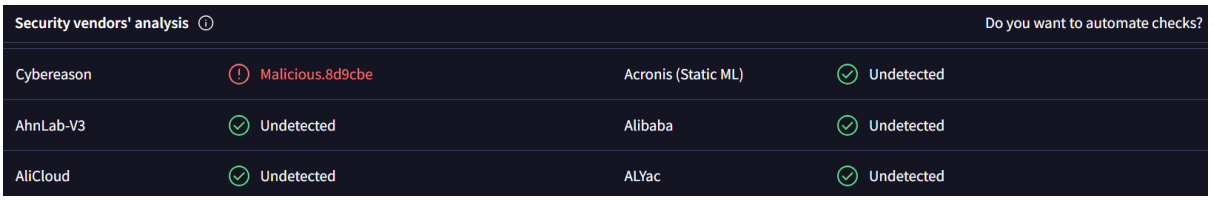
In this section I show the scan on VirusTotal.com results on the **debug** version, **non** obfuscated, with **no** evasion techniques applied.

The results show **1%** detection rate on the "Installer" part, **2%** on the "Service" and **9%** on the "Desktop Keylogger" itself. Combined, in the worst case we achieve around a **12%** detection rate.

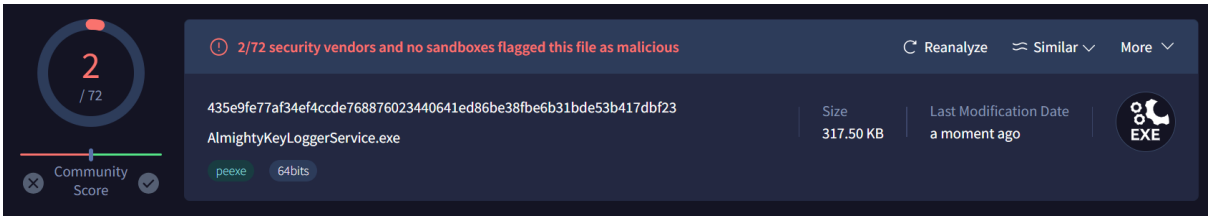
Most of the detections I would guess are made by static ML analysis of strings, mainly the use of string "KeyLogger" in the PE. I would argue that with some variable renaming and some obfuscation, those detections would be cleared.

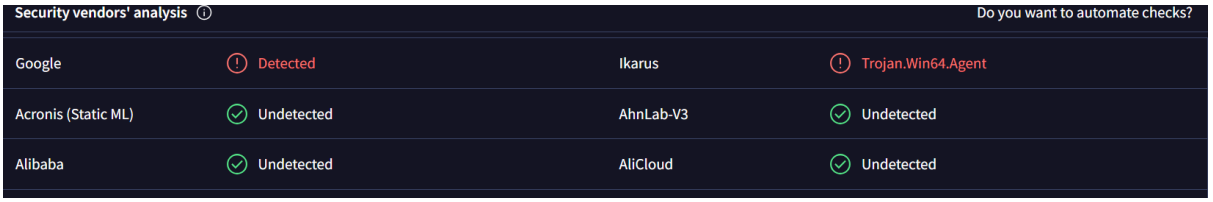
Installer - AlmightyKeyLoggerInstaller.exe



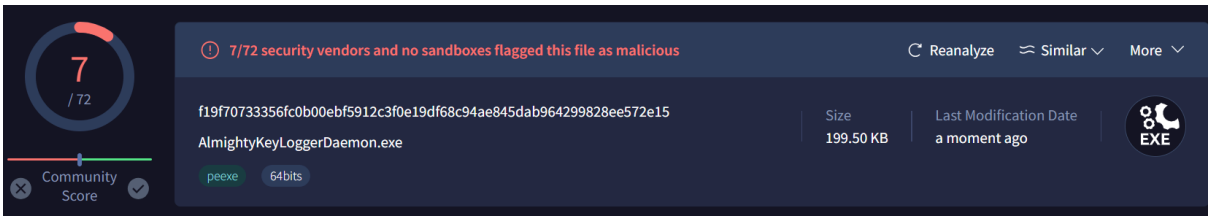


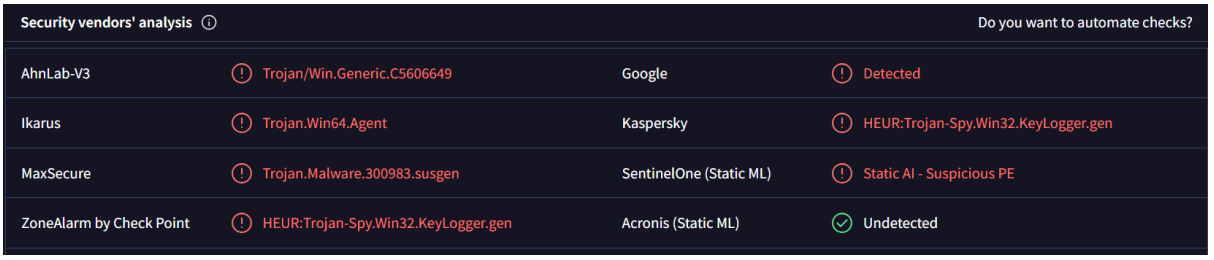
Service - AlmightyKeyLoggerService.exe





Desktop keylogger - AlmightyKeyLoggerDaemon.exe





Compilation Requirements:

- Decryptor.exe requires C++ 17. Rest of the programs were compiled with C++ 14.

Notes:

- The current encryption is a basic XOR encryption. Should be changed to AES in production.