ATMitch: remote administration of ATMs

By Sergey Golovanov

Published: 2017-04-04 · Archived: 2026-04-05 19:10:22 UTC

In February 2017, we published research on fileless attacks against enterprise networks. We described the data collected during incident response in several financial institutions around the world, exploring how attackers moved through enterprise networks leaving no traces on the hard drives. The goal of these attackers was money, and the best way to cash out and leave no record of transactions is through the remote administration of ATMs. This second paper is about the methods and techniques that were used by the attackers in the second stage of their attacks against financial organizations – basically enabling remote administration of ATMs.

In June 2016, Kaspersky Lab received a report from a Russian bank that had been the victim of a targeted attack. During the heist, the criminals were able to gain control of the ATMs and upload malware to them. After cashing out, the malware was removed. The bank's forensics specialists were unable to recover the malicious executables because of the fragmentation of a hard drive after the attack, but they were able to restore the malware's logs and some file names.

The bank's forensic team were able, after careful forensic analysis of the ATM's hard drive, to recover the following files containing logs:

- *C:\Windows\Temp\kl.txt*
- *C:\logfile.txt*

In addition, they were able to find the names of two deleted executables. Unfortunately, they were not able to recover any of the contents:

- *C:\ATM\!A.EXE*
- *C:\ATM\IJ.EXE*

Within the log files, the following pieces of plain text were found:

[Date – Time]
[%d %m %Y – %H : %M : %S] > Entering process dispense.
[%d %m %Y – %H : %M : %S] > Items from parameters converted successfully. 4 40
[%d %m %Y – %H : %M : %S] > Unlocking dispenser, result is 0
[%d %m %Y – %H : %M : %S] > Catch some money, bitch! 4000000
[%d %m %Y – %H : %M : %S] > Dispense success, code is 0

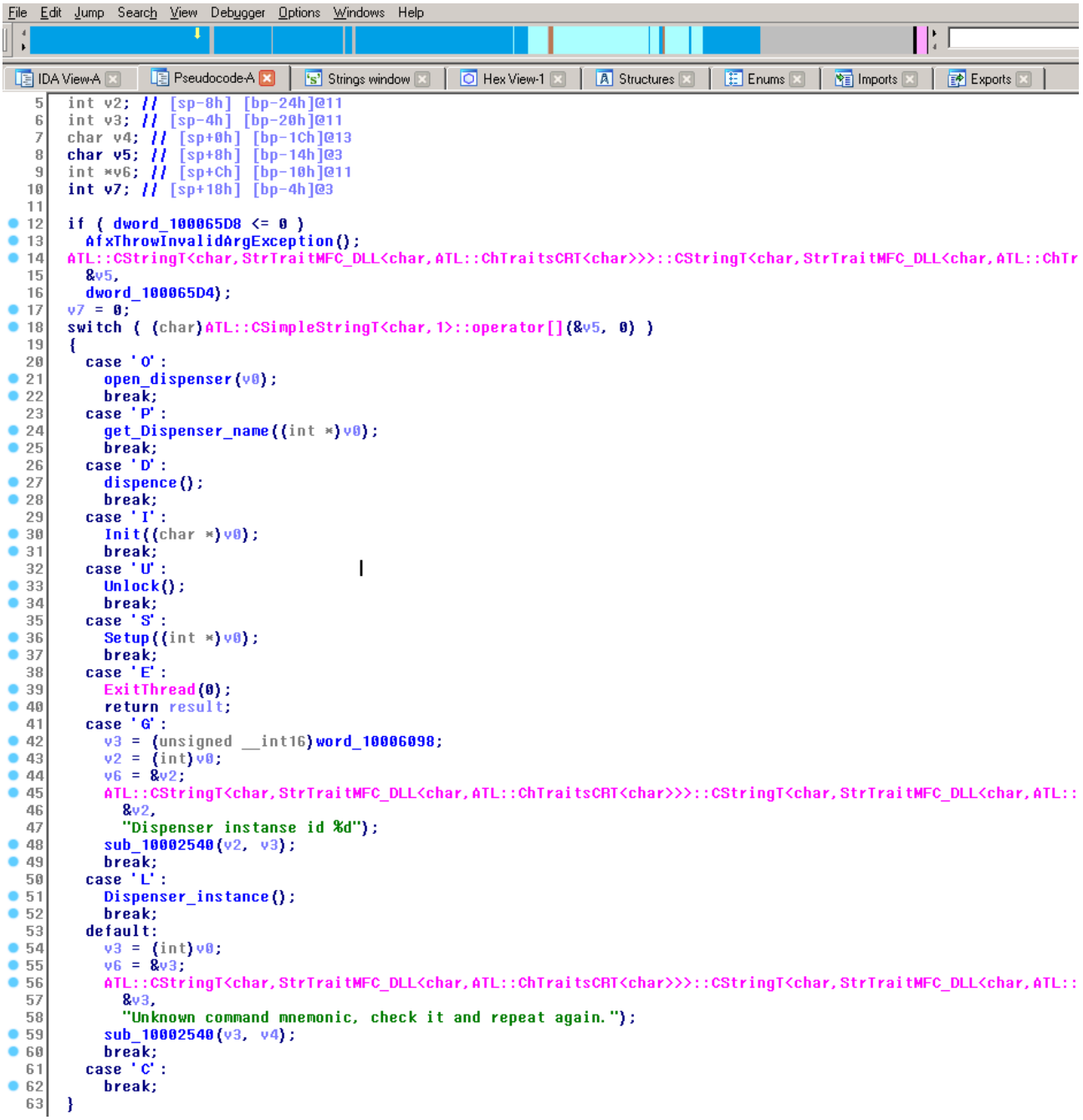
As mentioned in the previous paper, based on the information from the log file we created a YARA rule to find a sample, in this case: MD5 cef6c2aa78ff69d894903e41a3308452. And we've found one. This sample was uploaded twice (from Kazakhstan and Russia) as "tv.dll".

The malware, which we have dubbed ATMitch, is fairly straightforward. Once remotely installed and executed via Remote Desktop Connection (RDP) access to the ATM from within the bank, the malware looks for the "command.txt" file that should be located in the same directory as the malware and created by the attacker. If found, the malware reads the one character content from the file and executes the respective command:

- *'O'* – Open dispenser
- *'D'* – Dispense
- *'I'* – Init XFS
- *'U'* – Unlock XFS
- *'S'* – Setup
- *'E'* – Exit
- *'G'* – Get Dispenser id
- *'L'* – Set Dispenser id
- *'C'* – Cancel

After execution, ATMitch writes the results of this command to the log file and removes "command.txt" from the ATM's hard drive.

The sample "tv.dll" successfully retrieved in this case does not try to conceal itself within the system.



The malware's command parser

The malware uses the standard XFS library to control the ATM. It should be noted that it works on every ATM that supports the XFS library (which is the vast majority).

Unfortunately, we were unable to retrieve the executables (!A.exe and IJ.exe, located in C:\ATM) from the ATM; only the file names were found as artefacts during the forensic analysis. We assume that these are the installer and uninstaller of the malware. It should also be noted that "tv.dll" contained one Russian-language resource.

Kaspersky Lab continues to monitor and track these kinds of threats and reiterates the need for allowlisting in ATMs as well as the use of anti-APT solutions in banking networks.

Source: https://securelist.com/atmitch-remote-administration-of-atms/77918/