J. Irvine Secure Software Development
Initial Post

**Collaborative Discussion 2: Cryptography Case Study: TrueCrypt**

The cryptanalysis in the article does not appear to prove or disprove that TrueCrypt contains unfixed security issues. This has information based around an audit of TrueCrypt, which is an open-source encryption software. The audit was carried out by a small team of experts who manually audited the code. This audit did not find any evidence of backdoors, which was backed up in a study by de Carné de Carnavalet and Mannan (2014), or that there were any intentional vulnerabilities. It did find some vulnerabilities could be exploited by attackers. The recommendation was that TrueCrypt users should migrate to other types of encryption software.

TrueCrypt is no longer maintained and the company was dissolved in 2014. Ongers (2020) stated that the TrueCrypt website recommended that users migrate to other encryption software, VeraCrypt is an example of this. Junestam and Guigo (2014) state that TrueCrypt is not necessarily insecure, it is vulnerable to known attacks, for instance Mojžiš and Balogh (2020) found it could be susceptible to a Memory dump attack. The documentation distributed with TrueCrypt requires users to follow various security precautions to protect yourself from this.

As TrueCrypt is no longer maintained and not audited for many years, I would recommend using other encryption software that is maintained and has had more recent security audits. Some alternatives include VeraCrypt, BitLocker, and FileVault.

**References**

de Carné de Carnavalet, X. and Mannan, M., 2014, December. Challenges and implications of verifiable builds for security-critical open-source software. In Proceedings of the 30th Annual Computer Security Applications Conference (pp. 16-25).

Junestam, A. & Guigo, N. (2014) Open Crypto Audit Project Truecrypt Security Assessment

Ongers, G., 2020. Securing software development using developer access control.

Mojžiš, J. and Balogh, Š., 2020. Breaking microsoft azure information protection viewer using memory dump. In Software Engineering Perspectives in Intelligent Systems: Proceedings of 4th Computational Methods in Systems and Software 2020, Vol. 1 4 (pp. 913-920). Springer International Publishing.

https://truecrypt.sourceforge.net/

J. Irvine Secure Software Development
Initial Post

## Ontology Design on the Weaknesses of TrueCrypt

I had a go at producing an ontology prior to the lecture.

| Severity / Class | High | Medium | Low | Undetermined |
|---|---|---|---|---|
| Access Controls | Privilege Elevation Holes: James Forshaw of Google's Project Zero found flaws that could allow attackers to obtain elevated privileges on a system if they have access to a limited user account.<br>Physical Security: Unable to secure data on a computer if an attacker physically accessed.<br>Malware: According TrueCrypt documentation it cannot secure data on a computer if it has any kind of malware installed. | | | |
| Cryptography | CryptAcquireContext may silently fail in unusual scenarios.<br>AES implementation susceptible to cache-timing attacks. | Weak Volume Header key derivation algorithm. | | Unauthenticated ciphertext in volume headers. |
| Patching | No updates since 2014. | | | |
| Data Exposure | | Sensitive information might be paged out from kernel stacks.<br>Windows kernel driver uses memset() to clear sensitive data. | | |
| Data Validation | | Multiple issues in the bootloader decompressor. | IOCTL_DISK_VERIFY integer overflow.<br>MountVolume() device check bypass. | |
| Denial of Service | | | MainThreadProc() integer overflow.<br>GetWipePassCount() / WipeBuffer() can cause BSOD. | |
| Error Reporting | | | EncryptDataUnits() lacks error handling. | |

I was given feedback by the Dr. Peoples about the ontology diagram and so carried out an update.