Trusted Computing vs. Advanced Persistent Threats

Nikos Virvilis, George Stergiopoulos

{nvir, geostergiop}@aueb.gr

Information Security & Critical Infrastructure Protection Research Laboratory

Dept. of Informatics, Athens University of Economics & Business (AUEB)

Introduction

- Traditional security mechanisms have a hard time against sophisticated threats.
- Presentation and analysis of five (5) Advanced Persistent Threats, (APT).
- Work highlights characteristics and identifies common patterns and techniques.
- Issues that enabled the malware to evade detection from security solutions .
- Proposition of technical countermeasures for strengthening defenses against similar threats.

Stuxnet

- ✓ Targeted the Iranian Nuclear Program.
- ✓ Interfered with Industrial Control Systems (ICS) on Windows systems.
- ✓ "Fire and forget weapon".
- ✓ Infected removable drives. Zero-day vulnerability (MS10-046).
- ✓ Modular malware, using compromised certificates to sign components.
- ✓ Rootkit code to hide its binaries on windows systems.
- ✓ XOR encryption with a static key to decrypt payload.
- ✓ Rootkit module to hide its files - Payload injection to evade detection.

MiniDuke

- ✓ Targeted government bodies in 23 countries, mainly in Europe.
- ✓ Combined exploitation of Adobe's PDF sandbox and assembly code for its payload.
- ✓ Sophisticated, layered communication: Twitter accounts with encrypted URL's.
- ✓ Payload obfuscated as GIF images.
- ✓ Evasion techniques using process detection and idle state.

Duqu

- ✓ Developed for espionage: Key logging .
- ✓ Modular malware, using compromised certificates to sign components.
- ✓ Infection and propagation using Microsoft Word files.
- ✓ Rootkit module to hide its files - Payload injection to evade detection.
- ✓ AES-CBC and XOR encryption.
- ✓ Strong connections to Stuxnet development.

Flame

- ✓ Uncommon size: 20 megabytes, including all modules.
- ✓ Widespread Information stealing malware.
- ✓ USB infection but did not replicate on its own – Used two zero-day vulnerabilities.
- ✓ Complex cryptanalytic attack against Microsoft's Terminal Services.
- ✓ Rootkit functionality - .ocx binaries to avoid detection.

Red October

- ✓ Information gathering, targeting diplomatic, governmental and scientific agencies.
- ✓ Minimalistic architecture allowed it to remain undetected: One component downloaded modules (at least 1000 different modules have been identified).
- ✓ Targeted emails containing malicious Word and Excel documents
- ✓ Plugin for Office and Adobe reader.

COMMON CHARACTERISTICS

- Targeted system and architecture.
- Initial attack vectors.
- Escalation of privileges.
- Network Access.
- Network IDS and endpoint antivirus products.
- Use of Encryption / Obfuscation.
- Exploitation of digital signatures.

SUGGESTED COUNTERMEASURES

- **Patch Management**: *Stop further exploitation.*
- **Network Segregation**: *Workstation isolation blocks malware.*
- **Relaxed Whitelisting**: *Block malicious connection attempts.*
- **Dynamic content execution**: *Filter dynamic content in traffic .*

- We provide:
  - ✓ Technical comparison of malware, focusing on behavioral/dynamic analysis of the samples in a controlled environment, as well as on published technical reports
  - ✓ Identification oaf common attack patterns in samples to identify why current security solutions failed
  - ✓ A proposition of effective countermeasures for strengthening our defenses against similar threats.

	Stuxnet	Duqu	Flame	Red October	Mini Duke
Active since	June 2009 (2005)	Nov. 2010	May 2012 (2006)	May 2007	June 2011
Detected	June 2010	Sept. 2011	May 2012	Oct. 2012	Feb. 2013
PE Type	DLL		OCX	EXE	EXE
Initial infection	Unknown	MS Word	Unknown	MS Excel / Word, Java	PDF
Replication	Removable drives, network	Manual replication only			
Rootkit module	Yes		No		
Key logging	No	Yes			No
Evasion	Yes			No	Yes
Encryption	XOR	XOR, AES-CBC	XOR, Substitution, RC4	XOR	Unique per victim, XOR, ROL
Target	Sabotage	Information gathering			

Table 1: Advanced Persistent Threats Comparison

Conclusions

- Widely accepted best countermeasures would have reduced the impact of APT malware.
- Even in critical infrastructures, a small subset of protection mechanisms was enforced.
- Need to shift our focus on more robust and transparent solutions, since traditional security solutions, have failed repeatedly.
- Limiting and controlling the software that is allowed to be installed and executed on a system, would significantly reduce the impact of APT attacks.
- Combines with the research carried out by our Laboratory., in particular the one focused on security-critical information systems (e.g. health information systems).

References

1. Bencsath, B., Pek, G., Buttyan, L., Felegyhazi, M., "Duqu: Analysis, detection, and lessons learned", Proc. of ACM European Workshop on System Security, 2012.
2. Lekkas, D., Gritzalis, D., "Long-term verifiability of healthcare records' authenticity", International Journal of Medical Informatics, Vol. 76, Nos. 5-6, pp. 442-448, 2006.
3. Gritzalis, D., "Embedding privacy in IT applications development", Information Management and Computer Security Journal, Vol. 12, no. 1, pp. 8-26, MCB University Press, 2004.
4. Gritzalis, D., "Enhancing security and improving interoperability in healthcare information systems" , Informatics for Health and Social Care , Vol. 23, No. 4, pp. 309-324, 1998.
5. Kandias, M., Virvilis, N., Gritzalis, D., "The Insider Threat in Cloud Computing", Proc. of the 6th International Conference on Critical Infrastructure Security, pp. 93-103, Springer (LNCS 6983), 2013.
6. Kaspersky Labs. [Online] 28 May 2012 [Cited: 01 January 2013] http://www.securelist.com/en/blog?weblogid=208193522.
7. Litty, L., Lie, D., "Manitou: a layer-below approach to fighting malware", Proc. of the 1st Workshop on architectural and system support for improving software dependability, pp. 6-11, ACM, 2006.
8. Mylonas, A., Dritsas, S., Tsoumas, B., Gritzalis, D., "On the feasibility of malware attacks in smartphone platforms", Proc. of the 9th International Conference on Security and Cryptography, pp. 217-232, Springer (CCIS 314), 2012.
9. Tivadar, M., Balazs, B., Istrate, C.:"A closer look at Mini-duke". [Online] [Cited: 20 05 2013] http://labs. bitdefend-er.com/wp-content/uploads/downloads/2013/04/ MiniDuke_ Paper_Final.pdf.
10. Virvilis, N., Gritzalis, D., "Trusted Computing vs. Advanced Persistent Threats: Can a defender win this game?", Proc. of the 10th IEEE International Conference on Autonomous and Trusted Computing, IEEE Press, Italy, 2013.
11. Virvilis, N., Gritzalis, D., "The Big Four - What we did wrong in Advanced Persistent Threat detection?", in Proc. of the 8th International Conference on Availability, Reliability and Security, pp. 248-254, IEEE, Germany, 2013.