Platform-agnostic Low-intrusion Optical Data Exfiltration

Arthur Costa Lopes and Diego F. Aranha
Institute of Computing, University of Campinas, Campinas, Brazil
arthur.lopes@students.ic.unicamp.br, dfaranha@ic.unicamp.br

Keywords: Covert Channel, Data Exfiltration, Information Hiding, Air-gapped Machines, Error-correcting Codes.

Abstract: Information leakage through covert channels is a growing and persistent threat, even for physical perimeters considered as highly secure. We study a new approach for data exfiltration using a malicious storage device which subtly transmits data through blinking infrared LEDs. This approach could be used by an attacker trying to leak sensitive data stored in the device, such as credentials, cryptographic keys or a small classified document. An ideal application for this approach is when an attacker is capable of sneaking a malicious device inside a protected perimeter and has remote control over a camera inside such perimeter. The device can then collect information and transmit directly to the attacker, without the need of recovering the device to obtain the captured information, erase evidence or prevent a forensic investigation. We discuss techniques for improving communication efficiency up to 15 bits per second per LED, and possible countermeasures for mitigation.

1 INTRODUCTION

With the major advances in computing observed in the last decades and widespread availability of computer systems, all kinds of organizations make extensive use of computers to store and exchange information, sometimes of extremely sensitive nature. Techniques developed by the information security field are critical to protect and prevent such data from leaking in several ways, for example by data breaches and malicious *data exfiltration* activities. The concept of data exfiltration consists in the extraction of data from a closed network after a malicious software has infiltrated such network. A *convert channel* is typically employed for the task, posing a challenge to current monitoring capabilities.

The Snowden leaks (Verble, 2014) were among the first of a series of high-profile leakage cases in recent years, after which security processes have seriously started to consider the threat of data exfiltration. Various forms of prevention have been recommended and adopted, including compartmentalization of storage, data encryption at rest, and machine isolation from networks, in a so-called air-gapped system (Maass, 2013). For these reasons, this paper aims to explore new low-intrusive optical methods for data exfiltration and their efficiency and limitations. The goals are two-fold: by anticipating the impact of this threat and proposing countermeasures, organizations become able to protect data from exfiltration and design mitigation strategies; by improving low-intrusive exfiltration mechanisms, activists and whistleblowers can be informed about plausible ways to subtly and efficiently collect evidence of misbehavior or leak sensitive information.

In this paper, we study an efficient and ingenious approach for data exfiltration, involving a malicious storage device that leaks its critical contents. It assumes a scenario where the attacker is able to insert the malicious device in the security perimeter and wait until it is used to copy data from or between air-gapped machines. After the device detects the storage of critical data, it starts transmitting information through blinking infrared LEDs to a nearby camera under the control of the attacker. We claim the assumption is realistic, since there is growing evidence that users will simply plug in or even use USB drives they find (Tischer et al., 2016). This can happen even in tight security perimeters (Langner, 2011). Since no modifications are required in any of the air-gapped attacked machines, the approach requires a low level of intrusion and is agnostic to choices of operating system and software security mechanisms.

The approach has several advantages when compared to recent results in the research literature. For example, a recent paper (Sepetnitsky et al., 2014) builds an optical covert channel from hard-to-detect luminosity changes in a monitor LED. Similarly, the exfiltrated data can then be captured by a receiving device controlled by the attacker in the same environ-

Lopes, A. and Aranha, D.
Platform-agnostic Low-intrusion Optical Data Exfiltration.
DOI: 10.5220/0006211504740480
In *Proceedings of the 3rd International Conference on Information Systems Security and Privacy (ICISSP 2017)*, pages 474-480
ISBN: 978-989-758-209-7

ment. However, this and other approaches (Zaddach et al., 2013; Guri et al., 2015b; Guri et al., 2016a; Guri et al., 2016d; Guri et al., 2016c) require a vulnerability to be exploited and malicious software to be installed in the air-gapped machine, because hardware control typically requires local privileges. Requiring a target-specific exploit to be crafted may not be always under the capabilities of the attacker; and the installed software component can be found and dissected afterwards in case of detection, representing obstacles in practice. In comparison, our approach requires only a flash drive to be inserted in the target environment, perhaps by employing social engineering skills (Abraham and Chengalur-Smith, 2010). After the flash drive is connected to the air-gapped machine and data is received, the malicious device can start transmitting without requiring privileges or interfering in any way with the attacked system. The attacker does not need to retrieve the flash drive later, since data can then be captured by a networked camera or a smartphone inside the security perimeter. Because no actual changes have to be made in the attacked system, this imposes further obstacles to forensic investigations for collecting evidence of the malicious activities. Optical data exfiltration through blinking LEDs also offers higher bandwidth compared to heat-based methods (Loughry and Umphress, 2002), allowing the attacker to leak larger pieces of data. The additional flexibility allows the idea to be used in several scenarios.

The paper is organized as follows. Section 2 discusses data exfiltration techniques and related work available in the research literature. Section 3 proposes a threat model and presents our prototype device and protocol design. Section 4 collects preliminary results, Section 5 discusses proposed countermeasures and Section 6 concludes the paper.

2 RELATED WORK

Data exfiltration can be defined as an unauthorized way of obtaining and transmitting data from a closed or private network. To prevent detection, the transmission commonly employs a process not meant to transfer data, such as covert channels or steganographic techniques (Cheddad et al., 2010). This type of attack creates a subliminal channel to transmit sensitive data from inside a security perimeter directly to the attacker. Performance is usually low, in the order of a few bits per second, due to low signal-to-noise ratios or limited bandwidth.

Electromagnetic emanations (Kuhn and Anderson, 1998) and keyboard LEDs (Loughry and Umphress, 2002) were amid the first covert channels applied to data exfiltration, but there has been significant interest in other types of covert channels in recent years. Several studies focus on using optical effects, heat, sound, radio and have been demonstrated to work in machines not connected to external networks. Low-intrusive approaches may employ optical channels and a receiver device infected with software under remote control of the attacker, but there are several other ways to accomplish this, with different trade-offs in terms of efficiency and intrusiveness.

Techniques exploiting optical effects include LEDs (Loughry and Umphress, 2002; Sepetnitsky et al., 2014) or projecting whole images on a reflecting LCD monitor (Guri et al., 2016a). Transmission rate is generally low, compared to other means of transmission, due to limitations in both software and hardware. Frequently, there is some inherent limitation in the components that imposes an upper bound on transmission rate, such as the number of times a LED can be blinked in a time unit (25Hz) and its luminosity, or the screen brightness. This can make the attacks unreliable and hard to port across different machines. The low bandwidth restricts the approach to transmit cryptographic keys and other short sensitive bit strings. Heat can be exploited in an extremely subtle way (Guri et al., 2015b), by increasing computing load until a nearby machine can capture changes in temperature. A severe limitation is inefficient communication due to the time required to heat up and cool down a computer just by running software in user space, thus resulting in extremely restricted bandwidth and transmission rates. Covert channels can also be built from sounds emitted by mechanical hardware components, such as the computer fan (Guri et al., 2016d) or the hard drive (Guri et al., 2016c); or inaudible sound transmitted through the speakers (Hanspach and Goetz, 2014), with various transmission rates. The most efficient methods are based on radio transmission emanating from display cables (Guri et al., 2014), the USB bus (Guri et al., 2016b) and CPU instructions for multi-rate data transfers (Guri et al., 2015a); which can be then captured by a conventional cell phone or dedicated equipment. In practice, a lot of variables can disturb the transmission and efficient reception mechanisms are often difficult to design or even impractical. There could be also unexpected obstacles between the machine and receiver using optical channels or interference from heat or sound emitting objects nearby. The exfiltration effort can also be detected and disrupted if operational security is sophisticated enough to block radio signals or other means of communication.

Another disadvantage of those methods is the high

level of intrusiveness regarding the air-gapped machine, requiring an internal component to be tampered with (like in Fansmitter (Guri et al., 2016d) or DiskFiltration (Guri et al., 2016c)) before it is installed inside the target machine. This may be hard to manage in a real attack and perhaps can be detected. In terms of software, in some cases low-level access to operating system data is necessary for fine-grained status information such as CPU temperature(Guri et al., 2015b) or to control LEDs in peripherals (Loughry and Umphress, 2002; Sepetnitsky et al., 2014). These drawbacks limit the the applicability of these ideas, since the target machine necessarily needs to be infected with malicious software before transmission, making the attack unreliable across different operating systems or work against multiple software security mechanisms. The software artifact can also be discovered after the attack and inspected in a later forensic investigation, even if attribution may be difficult to resolve (Tsagourias, 2012).

Table 1 summarizes some aspects of different approaches for data exfiltration proposed in the literature. Approaches requiring intrusive access to the target air-gapped machine for introducing malicious components or local execution privileges are marked *high*. Approaches requiring user space software execution in the target machine are marked as *medium*, and the introduction of a malicious device in the security perimeter are marked *low*. Although there are previous works about data exfiltration using flash drives and hard drives (Clark et al., 2009; Zaddach et al., 2013), the optical approach we pursue is new and untested. The proposed method was designed to circumvent many of restrictions in other approaches and obtain a better trade-off between speed and level of intrusiveness. In our case, no malware infection, hardware tampering or any general modifications are required, since the malicious device needs only to be attached to the target machine. Additionally, no software execution or privileges are needed in the air-gapped machine. These features should allow the method to be hard to detect in practice or during a forensic investigation.

3 OPTICAL DATA EXFILTRATION

In this section, we describe in detail our approach for data exfiltration. First, we define a threat model and a possible application scenario. Then, we present the design of the prototype transmitter and receiver devices, and the transmission protocol.

3.1 Threat Model and Scenario

In our scenario, communication is established between a transmitter and receiver. The transmitter is a malicious thumb drive connected to the target air-gapped machine from which the attacker wants to leak information from. We assume the device was previously introduced in the environment using social engineering skills. The malicious device is equipped with blinking infrared LEDs and looks just like a simple USB flash drive, although the firmware is modified to check stored files for critical information and subsequent transmission. The file extensions or patterns for locating critical information are programmed in the firmware before the device is deployed.

For reception, we assume the attacker as capable of infecting a connected machine with malware that controls the camera. This machine can be either a networked computer or a smartphone inside the security perimeter. The camera monitors its view and waits for LEDs to start blinking in a predetermined specific way, as a handshake sequence to synchronize the transmitter and receiver. After the camera reads the LED and completes the handshake, it starts transmissting the captured data to an attacker-controlled remote server. The diagram in Figure 1 illustrates how the studied optical data exfiltration method works.

3.2 Transmitter

The main component of our prototype transmitting device is a Teensy2 board[1] to simulate the mass storage device. The LUFA framework (acronym for Lightweight Framework for AVRs) (Camera, 2013) and additional software [2] were employed for assembling the firmware portion. An SD card attached to the board works as the storage memory of the device. Two infrared LEDs finish the design.

Figure 2 presents the latest version of the transmitter. It is connected to a computer using a common USB connector and can be used as a regular flash drive, communicating to the SD card normally. Although the prototype is clearly different than a regular flash drive, we claim that an improved version can be built to be much closer in appearance than a typical USB stick; and the LEDs can also be made much smaller. This is very important for our approach, because targeted users must find the device convincing enough for daily use. With a realistic appearance, the device can be easily used in a real situation, effortlessly deceiving the target.

[1] https://www.pjrc.com/store/teensy.html

[2] http://elasticsheep.com/2010/04/teensy2-usb-mass-storage-with-an-sd-card/

Table 1: Comparison of different data exfiltration methods presented in the recent literature using various covert channels, resulting in different performance characteristics and requiring different levels of intrusiveness. Our approach tries to maximize transmission speed without requiring highly intrusive mechanisms (malware infection and local execution privileges in the target machine).

Work	Type	Speed	Intrusiveness	Transmission rate
Monitor LED (Sepetnitsky et al., 2014)	Light	Low	High	$<$ 25Hz
VisiSploit (Guri et al., 2016a)	Light	High	High	–
BitWhisper (Guri et al., 2015b)	Heat	Low	Medium	1-8 bits/h ($<$ 1 bps)
Fansmitter (Guri et al., 2016d)	Sound	Low	High	900 bits/h ($<$ 1 bps)
Diskfiltration (Guri et al., 2016c)	Sound	Low	High	180 bits/min (3 bps)
Ultrasound (Hanspach and Goetz, 2014)	Sound	Medium	High	20 bps
AirHopper (Guri et al., 2014)	Radio	High	High	104-408 bps
USBee (Guri et al., 2016b)	Radio	High	High	1200-4800 bps
GSMem (Guri et al., 2015a)	Radio	Low	High	1-2 or 100-1000 bps
This work	*Light*	*Medium*	*Low*	*30 bps*

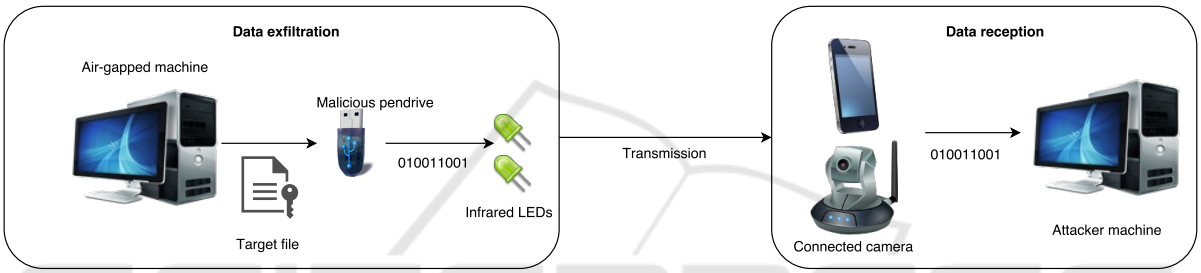


Figure 1: Diagram illustrating the process of data exfiltration. The malicious device is attached to the air-gapped machine, while the camera under control of the attacker is part of the receiver. A simple transmission protocol ensures synchronized communication between the two.

Our first attempt at building the transmitter consisted in repurposing the open source project BadUSB (Caudill, 2014) to control a Phison microcontroller in a way that the LED would flash in a controlled way. Unfortunately, modifying the firmware in the way we wanted was problematic. We also started using a single activity LED, commonly found in flash drives, but limited bandwidth suggested multiple LEDs as an alternative. Using multiple LEDs introduced a new problem: the device may now look unusual and suspicious during high activity, risking detection. We mitigated this threat by using light invisible to the human eye, augmenting the device with infrared LEDs. The current version of the transmitter can send multiple bits at a time, substantially increasing the bandwidth, without facilitating means of detection. We argue that such a device would be used by unsuspicious users, or even security-aware professionals.

Figure 2: The final prototype of our malicious data exfiltration device, with focus given to two infrared LEDs in the rear part of the device.

3.3 Receiver

A webcam or a smartphone camera are ideal choices at the receiving end. Both of these devices do not have the best resolution available in the market, but reproduce realistic conditions with a capture rate of no more than 30 frames per second and a resolution up to 720p. Another requirement for the receiver is a sta-

tionary position, so we consider that a smartphone is being held steadily or that a fixed webcam is already targeting the LEDs to simplify the capture.

We implemented the software component using the OpenCV 3.0 library (Itseez, 2015). The software monitors the captured images and identifies the source of data being transmitted, keeping track of each individual LED and its behaviour. After the handshake, the received bit string can then be stored in the remote-controlled machine or be sent to a web server through the smartphone mobile connection or the computer network. This web server is assumed to be under adversarial control, providing a safe haven for the data do be finally extracted.

The algorithm for LED detection and location is simple. Basically, it looks for differences between two consecutive frames and determines what portions of the image have the blinking LEDs. The infrared light produces a very distinctive color when captured, allowing for background noise to be detected and isolated. After the initial handshake protocol has finished, the receiver knows the location of each LED and can monitor the states to receive incoming messages. The bits are transmitted in the simplest way possible: a value 1 is represented by turning the LED on, and a value 0 contrariwise.

Figure 3 shows the reception process after the handshake has been completed. The image is a frame captured by a standard webcam. The distance between the device and the camera in this experiment was about one meter.

Figure 3: A screenshot of the exfiltration process after the handshake, as per the receiving view. The red rectangles show the location of each LED, only one is turned on in the picture, showcasing the distinctive color of infrared light.

3.4 Protocol

The transmission protocol starts from the handshake sequence transmitted at the beginning of the message, implementing a simple kind of synchronization. This is used to mark when transmission is starting again so the receiver can stop decoding new bits and start looking for any missing piece of information that has been lost in the last attempt. Another purpose for the handshake is to provide additional time for the camera to locate the LEDs in the start phase.

After the handshake protocol is finished, the message bits are transmitted through the blinking LEDs. However, this kind of noisy transmission introduces some errors in the data being sent. Because of the time the LED takes to light up or turn off, sometimes the bit is incorrectly read and ends up flipped at the receiving end. The easiest way we found to cope with this problem was to employ a Hamming(7,4)-code (Hamming, 1986) that allows for the correction of one bit per word. This choice was apparently sufficient in our tests with one and multiple LEDs and has solved the cases in which the error occurs when the LED starts to fade and the camera captures this transitory moment, resulting in a flipped bit.

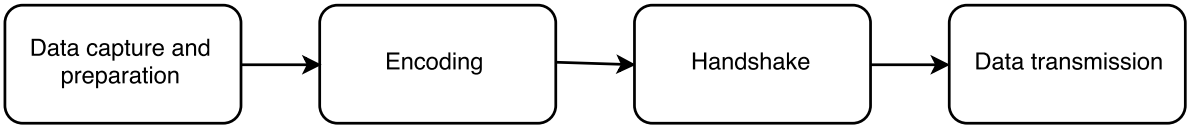
One important feature about the designed protocol was the low quantity of parity bits, decreasing the available bandwidth just a little. More sophisticated types of error-correcting codes were considered, but discarded due to a much higher number of parity bits. Another positive aspect about Hamming codes is their classical and well-understood nature, allowing simple implementation and usage.

Figure 4 presents an overview of our approach, with short descriptions of the process for transmission and reception.

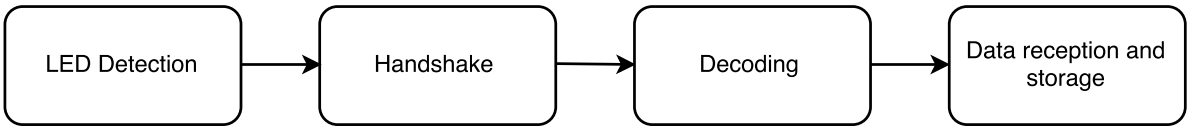
4 RESULTS AND DISCUSSION

We experimentally evaluated our approach with the main goal of transmitting specific small pieces of information from the attached SD card, such as credentials and cryptographic keys. The data is transmitted using two infrared LEDs, each one representing a different bit.

The biggest observed limitation is the camera frame rate at the receiving end, because each LED is capable of transmitting more than 30 bits per second. In terms of speed, we achieved rates of 10-15 bits per second per LED, which is efficient enough for our purposes. The main restriction in the transmission is the time that the LED takes to light up or turn completely off. Our results are not so far from some



(a) Overview of the transmission process. Data is first captured and encoded for transmission using an error-correcting code. After the handshake sequence is sent, data can be transmitted.



(b) Overview of the reception process. The receiver camera first locates the LEDs and waits for the synchronization sequence (handshake). After the handshake, data can be decoded and stored for retrieval by the attacker.

Figure 4: Description of the steps required by our studied approach for data exfiltration, encompassing the transmission and reception of data.

published related works (Sepetnitsky et al., 2014) and represent an improvement in terms of level of intrusion and stealthiness. Given the limitations of the optical approach, the achieved speed appears promising, making it possible to send data like passwords and cryptographic keys in just a few seconds, and some small files in minutes.

We observed three critical factors from the environment that may interfere with the transmission: (i) level of brightness, (ii) distance between the transmitter and receiver, and (iii) device positioning. Because the infrared light is weaker than regular light from a common LED, the infrared LED will fade if the environment is too bright, which may incur in additional loss during the transmission, since the capture would lose track of the device more easily. The distance between transmitter and receiver interferes in a similar way. If the distance between the two increases, the amount of noise in the environment that interferes with the transmission will increase too, since other external LEDs and lights may appear in the camera's view. Another important factor to the transmission is the positioning of the device regarding the camera. The data transmitting LEDs are located in the opposite side from the USB port, and the camera needs to be able to capture the image of the LEDs directly, otherwise the entire process will fail.

Our preliminary results were obtained in an ambient light room and with a distance of about one meter between the device and the camera. During the experiment we tried to minimize the amount of noise introduced from the environment while still keeping it realistic. Improving the quality of the camera at the receiving end allows both the distance and transmission rate to be increased, but this may not be compatible with real scenarios.

5 COUNTERMEASURES

Countermeasures for defensive purposes are certainly possible and mostly require improved operational procedures (Zander et al., 2007). Ensuring that USB ports are not visible in the air-gapped machines (e.g. using the machine in a closed environment with restricted access and tinted windows) and sanitizing external devices by default should mitigate the exfiltration threats posed by our approach. The main suggested point of entrance for the malicious device in the security perimeter is through an employee targeted by social engineering. Raising security awareness of the staff through user training is thus essential to prevent data exfiltration efforts.

6 CONCLUSION

We explored a new data exfiltration approach using an optical covert channel. The approach is platform-agnostic and does not require malware infection or any other modifications to be performed in the target air-gapped machine. It involves a malicious flash drive equipped with infrared LEDs for transmission and a connected camera for reception. Compared to related work, it achieves moderate speed with a low level of intrusiveness. As a result, it can be used to transmit small pieces of sensitive information, such as credentials, cryptographic keys or a short confidential document. A prototype was built and experimentally evaluated, reaching transmission rates of 30 bps of exfiltrated data using two LEDs. This can be easily improved by including more LEDs and adjusting the reception software accordingly.

ACKNOWLEDGEMENTS

We thank FAPESP (São Paulo Research Foundation) for financial support through process 2015/13876-7.

REFERENCES

Abraham, S. and Chengalur-Smith, I. (2010). An overview of social engineering malware: Trends, tactics, and implications. *Technology in Society*, 32(3):183–196.

Camera, D. (2013). LUFA - Lightweight USB Framework for AVRs. http://www.fourwalledcubicle.com.

Caudill, A. (2014). Phison 2251-03 (2303) Custom Firmware and Existing Firmware Patches (BadUSB). https://github.com/adamcaudill/Psychson.

Cheddad, A., Condell, J., Curran, K., and McKevitt, P. (2010). Digital image steganography: Survey and analysis of current methods. *Signal Processing*, 90(3):727–752.

Clark, J., Leblanc, S., and Knight, S. (2009). Hardware trojan horse device based on unintended USB channels. In *NSS*, pages 1–8. IEEE.

Guri, M., Hasson, O., Kedma, G., and Elovici, Y. (2016a). VisiSploit: An Optical Covert-Channel to Leak Data through an Air-Gap. *CoRR*, abs/1607.03946.

Guri, M., Kachlon, A., Hasson, O., Kedma, G., Mirsky, Y., and Elovici, Y. (2015a). Gsmem: Data exfiltration from air-gapped computers over GSM frequencies. In *USENIX Security Symposium*, pages 849–864. USENIX Association.

Guri, M., Kedma, G., Kachlon, A., and Elovici, Y. (2014). Airhopper: Bridging the air-gap between isolated networks and mobile phones using radio frequencies. In *MALWARE*, pages 58–67. IEEE.

Guri, M., Monitz, M., and Elovici, Y. (2016b). USBee: Air-Gap Covert-Channel via Electromagnetic Emission from USB. *CoRR*, abs/1608.08397.

Guri, M., Monitz, M., Mirski, Y., and Elovici, Y. (2015b). BitWhisper: Covert Signaling Channel between Air-Gapped Computers Using Thermal Manipulations. In *CSF*, pages 276–289. IEEE.

Guri, M., Solewicz, Y. A., Daidakulov, A., and Elovici, Y. (2016c). DiskFiltration: Data Exfiltration from Speakerless Air-Gapped Computers via Covert Hard Drive Noise. *CoRR*, abs/1608.03431.

Guri, M., Solewicz, Y. A., Daidakulov, A., and Elovici, Y. (2016d). Fansmitter: Acoustic Data Exfiltration from (Speakerless) Air-Gapped Computers. *CoRR*, abs/1606.05915.

Hamming, R. W. (1986). *Coding and information theory (2. ed.)*. Prentice Hall.

Hanspach, M. and Goetz, M. (2014). Recent developments in covert acoustical communications. In *Sicherheit*, volume 228 of *LNI*, pages 243–254. GI.

Itseez (2015). Open source computer vision library, version 3.0. https://github.com/itseez/opencv.

Kuhn, M. G. and Anderson, R. J. (1998). Soft tempest: Hidden data transmission using electromagnetic emanations. In *Information Hiding*, volume 1525 of *Lecture Notes in Computer Science*, pages 124–142. Springer.

Langner, R. (2011). Stuxnet: Dissecting a cyberwarfare weapon. *IEEE Security & Privacy*, 9(3):49–51.

Loughry, J. and Umphress, D. A. (2002). Information leakage from optical emanations. *ACM Trans. Inf. Syst. Secur.*, 5(3):262–289.

Maass, P. (2013). How Laura Poitras Helped Snowden Spill His Secrets. New York Times. http://www.nytimes.com/2013/08/18/magazine/laura-poitras-snowden.html.

Sepetnitsky, V., Guri, M., and Elovici, Y. (2014). Exfiltration of information from air-gapped machines using monitor's LED indicator. In *JISIC*, pages 264–267. IEEE.

Tischer, M., Durumeric, Z., Foster, S., Duan, S., Mori, A., Bursztein, E., and Bailey, M. (2016). Users really do plug in USB drives they find. In *IEEE Symposium on Security and Privacy*, pages 306–319. IEEE Computer Society.

Tsagourias, N. (2012). Cyber attacks, self-defence and the problem of attribution. *Journal of Conflict and Security Law*.

Verble, J. (2014). The NSA and Edward Snowden: surveillance in the 21st century. *SIGCAS Computers and Society*, 44(3):14–20.

Zaddach, J., Kurmus, A., Balzarotti, D., Blass, E., Francillon, A., Goodspeed, T., Gupta, M., and Koltsidas, I. (2013). Implementation and implications of a stealth hard-drive backdoor. In *ACSAC*, pages 279–288. ACM.

Zander, S., Armitage, G. J., and Branch, P. (2007). A survey of covert channels and countermeasures in computer network protocols. *IEEE Communications Surveys and Tutorials*, 9(1-4):44–57.