Secure Peripherals in a Converged Mobile Environment

Jaco du Toit(✉) and Ian Ellefsen

University of Johannesburg, Johannesburg, South Africa
jacodt@uj.ac.za

Abstract. Users of computing devices have evolved from a fixed desktop computer environment to a situation where a user not only use one personal computer, but also one or more mobile device for both personal and business purposes. As soon as a user uses multiple computing devices for personal and business purposes, an exchange of data is necessary to ensure that the data is available from the various devices. The mechanisms used to enable data exchange may include potentially insecure cloud storage. Business data stored on cloud storage may also be against company policy. To minimize the dependency on multiple devices, this paper describes a computing environment where a user uses only one computing device, but with multiple input\output peripherals. The communication between the IO peripherals and the Neo device is described in terms of a communications model. The communications model highlight the information security aspects of confidentiality, integrity and authorization. The implementation viability of the model is reviewed against existing technologies. The paper concludes that existing technologies can fulfil most of the requirements for the mdoel, but may require customization to ensure fine-grained access control.

Keywords: Mobile · Wireless · Security

1 Introduction

The history of computing has moved from a centralized, mainframe, computing model to a decentralized, distributed, personal computer and Internet model. This model expanded to include mobile devices in the last few years. It has been observed that many employees own multiple mobile devices for their computing requirements.

Arguably, one of the reasons, why people own and use more than one mobile device is because of the devices' ergonomic nature. Some people prefer the smaller smartphone sized mobile devices in certain environments, while they prefer to use the larger tablet-sized mobile devices in other environments and situations.

As soon as users start using more than one mobile device, one of the ways in which users exchange data between devices is by using potentially insecure cloud storage areas. Previous published work [1, 2] argues that in order to minimize the cloud based storage risk to interchange data between devices, it should be possible to design a mobile based ecosystem where a user has one mobile computing device, called the Neo device, with multiple separate input\output peripherals. These peripherals can be any

T. Tryfonas (Ed.): HAS 2017, LNCS 10292, pp. 296–308, 2017.
DOI: 10.1007/978-3-319-58460-7_20

type of input\output device, such as a smart-phone sized touch screens, or even bigger desktop monitors, with physical keyboard and mouse. An important aspect to understand is that these peripherals do not store any apps or data on them directly.

When a Neo device is used in a mobile environment, with multiple peripherals, the issue of secure communication between the peripherals and the computing device comes into play.

This article describes how input\output (IO) peripherals using various communication channels can securely communicate with a Neo device.

This article is organized as follow: Sect. 2 provides an overview of the Neo device and the basic properties of the Neo device. Section 3 describes existing communication technologies used by peripherals, with the security mechanisms they employ. Section 4 describes the proposed communication model for the Neo device with Sect. 5 providing a bit more detail on how this may be implemented in a prototype or real world device. Section 6 concludes by highlighting the advantages of using the Neo devices with their secure peripherals.

2 Overview of the Neo Device

A short overview is provided at this stage in order to better understand the architecture and ecosystem of the Neo device. [2] introduces the Neo model. The Neo model describes two important properties of a hypothetical mobile operating system. The two properties are a Secure Container Property (SCP) and a Mutual Authentication Property (MAP). These two properties ensure data confidentiality and multiple IO peripheral connections. The Neo model defines a hypothetical mobile device that implements these two properties.

The Neo device is defined as a mobile device. The Neo device does not have any built-in input\output peripherals. The Neo device itself is a hypothetical black box that can easily be carried by a person or docked in a docking station. Any interaction with the device is done using the various peripherals available for the Neo device. These peripherals can be any form factor that the user requires. This means that a peripheral can be as small as a wristwatch, and can be as big as a data projector screen.

The secure communication between the IO peripheral and the Neo device is defined as the MAP. The MAP describes and define firstly the identification of peripherals and Neo devices with each other, and secondly the authorisation of peripherals with the Neo device.

The other important aspect of the Neo device is that it must support usage for both personal and business data and applications. This multi-domain usage requirement is addressed using a SCP.

The SCP ensures personal and business data and apps are isolated and controlled separately in specific secure containers. The boundary definition of a secure container is loosely defined by the ownership of the applications and data. All data and applications owned by the user is grouped in the same container, whereas the data and applications owned by a specific company is grouped inside the same container.

Figure 1 shows the model that describes the MAP and the SCP. The MAP encloses all communications to and from the device. When an IO peripheral communicates with the Neo device, the MAP ensures the identification and authorisation of the peripheral. The property ensures not only identification and authorisation to the device, but also to the various secure containers. A more complete description of the identification and authorisation property is found in section four of this document.

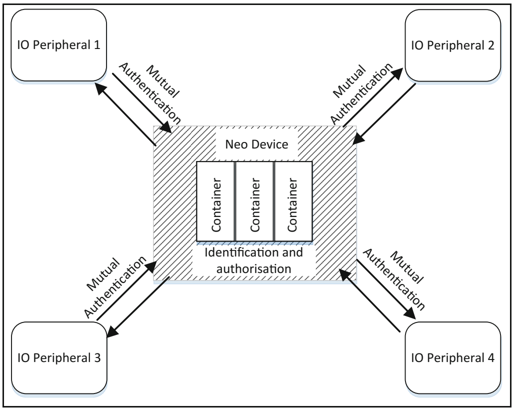

Fig. 1. Neo device with secure containers and identification and authorization service [2]

The SCP describes an isolated computing environment belonging to a specific data and application owner. This property is similar to today's isolation model found in popular mobile operating systems like Android and Apple iOS, except that is it is not based on a per app property, but is applied on a group of applications and their data. The model does not specify the isolation property of apps inside the secure container.

The hypothetical Neo device contains both a MAP and a SCP. The focus of this paper is on how the MAP can be established. The next section provides a literature review on the existing technologies for both wired and wireless IO peripheral security.

3 Security in Today's IO Peripherals

The assurance of Confidentiality, Integrity and Availability of data is an important aspect of data security. Data is provided as input or output through the use of IO peripherals. In both wired and wireless communication models of IO peripherals, a certain amount of risk is associated with the data as it flows between the IO peripheral and the device. This section provides a short discussion on the risks of wired IO peripherals and some of the research and solutions to address some of these risks.

The section then discuss the risks of wireless IO peripherals and in turn summarizes some of the security implementations and research around wireless communications for IO peripherals.

3.1 Risks and Security for Wired IO Peripherals

Wired IO peripherals supply either input to the computer or output from the computer. Input peripherals include typical computer keyboards, while output peripherals include computer screens. Unfortunately, there are risks for both input and output peripherals. The risks for input peripherals are the recording and capturing of keystrokes or mouse movements. The risks for output peripherals are the recording of video while the computer is in use.

The recording of either keystrokes or video can be implemented in either software or hardware. The next two sub-sections provide a brief overview of the risks of software and hardware recorders.

Software Recorders

Software recorders are typically categorized as spyware and many anti-malware products try to detect these types of recorders [3, 4]. These recorders are well documented and described. This paper will not focus on these recorders, but mention them because of the risk that they still provide to Internet connected computers.

The biggest disadvantage for attackers of software recorders are that they can be detected using anti-malware products. The biggest advantage of software recorders for attackers are that an attack is not limit to physical proximity or contact. This proximity negation allows an attacker to attack thousands of Internet connected computers increasing the likelihood that one of the victims will not have the necessary defenses to protect against software recorders.

Hardware Recorders

There are a number of sites that advertise hardware keyloggers [5, 6]. These sites advertise the hardware keyloggers for legitimate purposes that include [5, 6]:

- Monitoring staff productivity.
- Monitor inappropriate use of computers.
- Backing up typed information for authors.
- Computer forensic investigations.

The above purposes may be legitimate, but keyloggers can arguably also be used for malicious purposes.

Hardware keyloggers can be either active or passive [7]. Active keyloggers intercepts a key stroke and retransmits the keystroke to the computer. They are normally connected in-line with the physical keyboard. Figure 2 is an example of an active, in-line, USB keylogger [6].

Passive keyloggers observes the data line or data bus of the keyboard and computer to capture keystrokes. Figure 3 is an example of a passive keylogger that is installed inside a laptop computer [6]. This keylogger observes the data on the PCI bus.

Attackers use hardware recorders to target a specific victim. Hardware recorders are more difficult to distribute and the attacker must have physical access to the hardware of the victim. The challenge to victims of hardware recorders are that you cannot detect them using commonly available software like anti-malware programs.

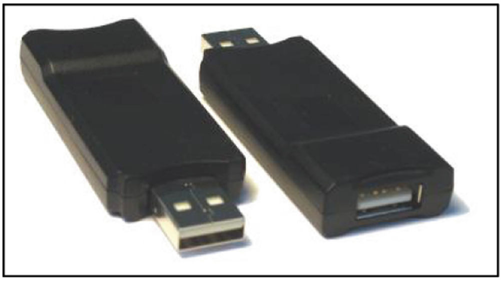

Fig. 2. Active USB keylogger [6]

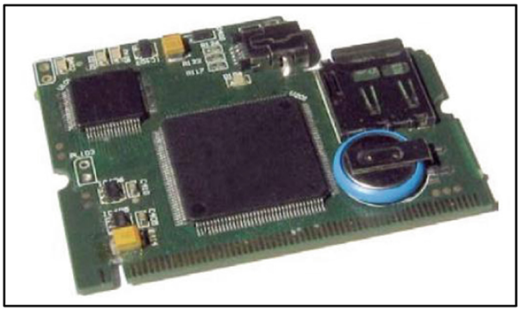

Fig. 3. Passive keylogger [6]

Research has shown that hardware keyloggers can be detected by the effect that they have on the data line of the keyboard. Both active and passive keyloggers change the electrical current of the keyboard through their acts of observation. These current fluctuations can indicate the presence of a hardware keylogger [7].

Even though it is theoretically possible to detect hardware keyloggers, the detection mechanisms must be implemented on the computer hardware, with the associated software before a computer can detect a hardware keylogger. None of today's commercial computers has these hardware and software mechanisms.

Wireless IO peripherals exchange data not over a physical bus, but over the air. This increases the likelihood that someone can listen in or intercept the data to or from the IO peripheral. The next section describes these risks and the implementation technology used by today's wireless IO peripherals.

3.2 Risks and Security of Wireless IO Peripherals

IO peripherals that use a wireless signal typically use either Bluetooth [8] or Wi-Fi [9] technology to transmit and receive wireless signals. Consumer electronics usually rely on Bluetooth for input peripherals and limited output peripherals like Bluetooth speakers [10].

Wi-Fi however can easily transmit either input or output signals. Wi-Fi has a specific standard called Miracast [11], which specifies the use of Wi-Fi to stream high definition audio and video between consumer devices easily. The advantage that Wi-Fi has over Bluetooth is that Wi-Fi supports must faster connection speeds and longer ranges [12].

By using the well-known confidentiality, integrity and availability goals for information security as a guideline. Table 1 summarizes the threat model for wireless IO peripherals. The threats do not take into account threats to the many wireless devices, instead the focus is on the wireless network established between IO peripherals.

Table 1. Wireless network threats.

Security goal	Threat
Confidentiality	Unauthorized parties can intercept and disseminate wireless data The wireless interface allows access to applications and data
Integrity	Tampering of wireless data by unauthorized parties Wireless data sent by unauthorized sources
Availability	The denial of wireless services and data

In order to mitigate these risks both Bluetooth and Wi-Fi standards have implemented certain security mechanisms. The mechanisms aim to minimize the threats during the initial identification and authentication phase, but it also creates a secure channel that ensures data confidentiality and integrity between the sender and receiver.

Bluetooth specification 2.1 introduced simple secure pairing (SSP) that minimized the chances of attackers gaining access to the wireless session, unfortunately it has been proven that even in Bluetooth specification 4.0 that SSP are still vulnerable to man-in-the-middle attacks. SSP uses Elliptic Curve Diffie Hellman cryptography to provide confidentiality of data [13, 14].

Wi-Fi introduced Wireless Equivalent Privacy (WEP) and Wi-Fi Protected Access (WPA and WPA2) as security protocols [15]. Industry and security researchers recommend the use of WPA2 as the more secure protocol [15]. WPA2 uses the concept of pre-shared keys or an authentication server. In situations where WPA2 sessions are established using pre-shared keys, the environment is vulnerable to attacks by getting the handshaking packets and then using brute force or dictionary attacks on the packet data to extract the pre-shared key.

As discussed earlier the risks for wireless communication is significant, because attackers do not have to be physically connected to the peripherals or devices. The security mechanisms in both Bluetooth and Wi-Fi minimizes the risk of attackers gaining access to wireless data or gaining unauthorized access to the devices through the wireless channel. In some cases, IO peripheral data travels across a network, but do not rely on network layer security, but on application layer security. Application-layer security implementations are described next.

3.3 Terminal-Based IO Peripheral Data

In terminal based networking, where users establish a terminal session to a server, the screen, keyboard and mouse traffic is transferred between terminal and server. Some examples of terminal based implementations that was developed for both keyboard and graphical user interfaces include:

- **X Windows System**. The X Windows System has a client, running on a remote system that displays information on the local system using an X Server. Mouse and keyboard input interacts with the X Server and the resulting commands are sent to the client running on the remote system. In this case, the mouse and keyboard settings do not travel across the network, rather the specific action that they cause, travels over the network [16].
- Virtual network computing (**VNC**). VNC is a platform independent open source protocol that allows a VNC Server to run on a computer that is being controlled by another computer that runs a VNC Viewer. Screen information (Output) is sent to the VNC Viewer and Keyboard and Mouse information is sent from the Viewer to the Server (Input) [17].
- Citrix Independent Computing Architecture (**ICA**). As part of the Citrix Terminal services product, Citrix developed ICA. The protocol is proprietary, but is available on multiple operating system platforms [18].
- Microsoft Remote Desktop Protocol (**RDP**). The RDP protocol works in the same principle as VNC and ICA, but is developed to allow remote sessions to Windows operating systems from a number of platforms [19].
- **ITU-T T.128**. The ITU-T T.128 is a standard defined by the International Telecommunications Union that defines multipoint application sharing. The standard does not define any security requirements, instead it relies on the application to implement the security required by the application [20].

These terminal based protocols all run on top of an underlying network protocol, which in turn runs on top of a network architecture, like Ethernet or Wi-Fi. Some of the protocols have built-in encryption protocols, but some also include network level authentication to occur before a session is established. Without network-level authentication, man-in-the-middle attacks are possible [19].

Terminal-based IO peripheral protocols have been developed and has undergone significant upgrades over the years to address network security issues, like confidentiality of data and data integrity. This means that terminal-based IO protocols can allow some level of security even using potential insecure wired or wireless communication channels.

4 Proposed Communications Model

The previous section provided an overview of some of the risks and solutions used by IO peripherals in wired, wireless or terminal based communications. Section 2 described the Neo device, which does not have any built-in IO peripheral, but allows the user to connect different form-factor IO peripherals to the device, either wirelessly or through a wired docking station.

This section discusses the communication model of the Neo device with its IO peripherals in order to address confidentiality of data and integrity. Confidentiality means that the data transferred between the Neo device and IO peripheral must be obfuscated so that only the peripheral and the Neo device understands the data. Integrity means that communication between the IO peripheral and the Neo device has not been changed.

This section breaks up the model by describing how the IO peripheral and Neo device address the following aspects:

- Identification and Authentication.
- Authorization.
- Confidentiality.

4.1 Identification and Authentication

The identification and authentication phase (**Phase 1**) is the initial phase required by any IO peripheral and Neo device that would like to communicate with each other. The outcome of this phase is to establish a shared key between the IO peripheral and the Neo device.

In order to accomplish the above-mentioned goal, both the Neo device and IO Peripheral consists of the following components, as shown in Fig. 4:

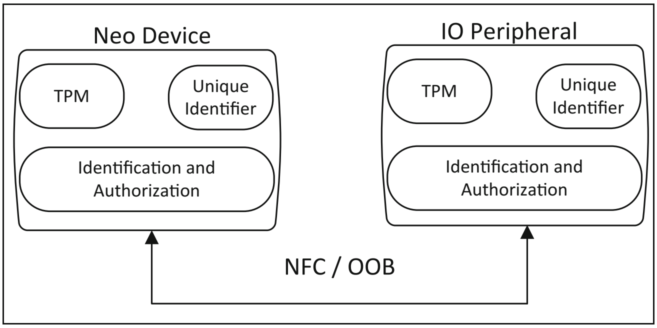

Fig. 4. Components during identification and authentication

- Trusted Platform Module (TPM). The TPM ensures the safe generation and access to the public\private key pairs required for secure communication [21].
- Unique Identifier. The unique identifier is used to ensure that each device is identified and assists in the generation of the shared key.
- Identification and Authorization service. The Identification and Authorization service is the software layer on the device that ensures the successful identification and authentication of devices.

- Near field communication (NFC) [22] or out-of-band (OOB) communication. The initial identification and authorization process occurs only via NFC or OOB. The physical act of either plugging in a device or bringing the devices in close proximity provides another level of authentication.

The most important aspect to understand during Phase 1 is that the secret key that is created between the IO peripheral and the Neo device can only be used between these two devices. Another important aspect is that both devices authenticate with each other. This means the IO peripheral is assured of the identity of the Neo device and the Neo device can be assured of the identity of the IO peripheral, which minimizes the chances of man-in-the-middle attacks.

4.2 Authorization

As soon as both the IO peripheral and the Neo device has been identified, the Neo device authorizes access to the IO peripheral in the Neo device (**Phase 2**). The authorization of the IO peripheral also occurs in the identification and authorization layer (Fig. 4).

Authorization occurs using two specific components:

- **Access control module (ACM)**. The ACM lists the rights of the IO peripherals on the secure containers, as defined by the container owners. The ACM is a complex data structure that defines the low-level rights that the IO peripheral has in both the Neo device and specific container\s. It consists of fine-grained control, which can include settings such as time of day usage and even sensor permissions. These controls is explained in more detail below.
- **Policy management module (PMM)**. The PMM modifies the authorization settings of the various secure containers. Owners of containers interface with the PMM to modify rights. The PMM can accept modification through a custom user interface, or through networked policy commands. This allows the device owner, and thereby the owner of the initial personal container the ability to allow more peripherals to access the personal container, but it allows corporate owners the ability to control access to corporate specific secure containers.

The ACM controls all aspects of the Neo device, IO peripherals and secure containers. The ACM controls whether a specific IO peripheral has access to the Neo device and secondly whether the peripheral has access to a specific secure container. Furthermore, it enforces certain policy requirements that some of the secure containers may have.

These policy requirements defines whether some of the sensors or device features is available to a specific secure container. Example: It may be possible for a specific corporate secure container to not allow access to the microphone or camera.

In addition, a specific policy element controls whether it allows access to other secure containers while some secure container is active or while the device is in a specific area. This allows container owners to ensure only certain containers can be used while in specific areas of a company.

In cases where more than one container requires exclusive access, only the first container starts. More information on the specific access control policy elements can be found in [1].

4.3 Confidentiality

IO peripheral confidentiality is ensured using the secret key that is established during the initial identification and authentication phase. The secret key is established between the Neo device and IO peripheral. This ensures that as peripheral information is transmitted between Neo device and peripheral that the information is encrypted. This type of communication has been briefly introduced in Sect. 3.2.

There is however a concern between confidentiality requirements of the secure containers. The question arises how the system can guarantee confidentiality of peripheral data between containers. How do we ensure that one container cannot access the IO peripheral data of another container?

The simplest solution to the above problem is to implement mutual exclusion for IO peripherals. This means that only one secure container can have access to an IO peripheral at a time.

The communication information between IO peripheral and Neo device is secured using specific identification and authentication phase that establishes a secret key for the peripheral\device combination. This key is used to ensure confidentiality of communication information. Access of peripherals to secure containers are ensured using the ACM. The next section discusses the viability of these model elements using today's known technologies.

5 Viability of Proposed Communications Model Using Existing Technologies

Even though the Neo device is a hypothetical device, it may be possible to implement the requirements defined in the communications model using existing technologies. This section discuss the possibility of using today's communication technology to implement the communications model described for the Neo device.

Table 2 lists the requirements that were identified in Sect. 4 and comments on whether the chosen technologies can be used to fulfill the requirement.

Bluetooth 4.0 fulfills nearly all the requirements in the protocol specification, but it does not specify how the Bluetooth devices should manage access control and how the access control can be modified. Bluetooth also allows a number of SSP association models, which include the insecure "Just works"-association model [14]. The Neo device implementation will not allow this association model, and only allows OOB association to occur.

Bluetooth 4.0 specifies a number of profiles that define the profile specific communication. **Example**: The Video Distribution Profile (VDP) specifies how video is streamed from a master to a slave. The VDP specifications are very different from the Health Device Profile (HDP), which defines how medical devices communicate.

Table 2. Model requirements vs technology

Requirement	Bluetooth 4.0	WPA2
Secure key generation (using TPM)	Yes	Possible using virtual smartcard
Unique identifiers for both IO peripheral and neo device	Yes	Not required
Identification and authorization service	Yes	Yes
Out-of-band initial identification and authentication	Yes	N/A
Access control module	N/A	N/A
Policy management module	N/A	N/A
Encrypted device-to-device communication	Yes	Yes

Bluetooth profiles fit nicely into the principle that the Neo device can also receive not just input signal from an IO device, but potentially also sensor information, like temperature readings, GPS or camera images. When an IO peripheral has a combination of Bluetooth profiles, the access control module can control which profile data is allowed at a specific time, depending on which secure container is currently active.

Bluetooth on its own completes a number of the requirements, but needs extra management software that manages the access control of specific Bluetooth profiles. Some Bluetooth specifications may also be limited to ensure that the OOB requirement is met.

WPA2 defines how devices should identify and authenticate with a WiFi access point and once authenticated ensure encrypted session is established between the device and access point. WPA2 can be extended to ensure that encryption keys are generated using TPM, but relies on higher-level protocols that run over WiFi to ensure authorization. By default, WPA2 does not specify the initial identification to occur using OOB, but technically the implementation stack can be extended to only allow identification and authentication to occur initially OOB.

WPA2 fulfils only some of the requirements for the Neo communication model, but is still a viable option to use for those items that it can fulfill. The other items would need specific implementation extension using higher-level management software. Sensor specific communication would need special application specific implementation.

6 Conclusion

People use multiple mobile devices for the computing needs because of the various form factors available to them. Multiple mobile devices introduce a challenge to ensure that data can safely be transmitted between devices. One solution that addresses the potential insecure data transmission is to use only one device, but with multiple IO peripherals of different form factors.

The use of one device with multiple IO peripherals is not the way in which mobile devices are designed and implemented. The Neo device defines an IO peripheral communication model that ensures IO peripherals and Neo devices are mutually identified, authorized and can securely communicate.

Technologies like Bluetooth or WPA2 can be used as a starting point to implement the communications model that addresses the security requirements for IO peripheral communication.

The challenge with the principle of using one device with multiple IO peripherals allows for many interesting implementation scenarios, but the fact that a user will always need at least two devices may be a big detractor. A possible mitigating action, can be for the Neo device to have at least a built-in touch screen, like today's smartphones. The Neo device will then allow the built-in screen to be switched off in cases where other IO peripherals are communicating with the Neo device.

There are a number of risks when using wired or wireless IO peripherals. Wireless peripherals are particularly at risk because of availability of the wireless channel to potential attackers. One computing device that allows multiple IO peripherals opens up opportunities where the same device can be used for both personal and business use.

To ensure proper isolation and access control of business and personal data a communication model is required that ensures that a data owner has full control over not only who can access the data but must also have the ability to control which peripherals can access the data. The proposed communication model addresses fine-grained access control to not only the computing device, but also the individual secure containers.

References

1. du Toit, J., Ellefsen, I.: Location aware mobile device management. In: 2015 Information Security for South Africa, Rosebank, pp. 1–8 (2015)
2. du Toit, J., Ellefsen, I.: A model for secure mobile computing. In: Science and Information Conference (SAI), London, pp. 1213–1221 (2015)
3. Kaspersky Lab. Spyware Definition and Prevention. http://usa.kaspersky.com/internet-security-center/threats/spyware#.WIxoWUQ2u00. Accessed 28 Jan 2017
4. Trend Micro. Spyware - Threat Encyclopedia. http://www.trendmicro.com/vinfo/us/threat-encyclopedia/malware/spyware. Accessed 28 Jan 2017
5. KeeLog. KeyGrabber - Hardware Keylogger - WiFi USB Hardware Keyloggers. https://www.keelog.com. Accessed 28 Jan 2017
6. KeyCarbon LLC. KeyCarbon Computer Security Hardware. http://www.keycarbon.com. Accessed 28 Jan 2017
7. Gerdes, R.M., Mallick, S.: Physical-layer detection of hardware keyloggers. In: Bos, H., Monrose, F., Blanc, G. (eds.) RAID 2015. LNCS, vol. 9404, pp. 26–47. Springer, Cham (2015). doi:10.1007/978-3-319-26362-5_2
8. Bluetooth SIG, Inc. Bluetooth. https://www.bluetooth.com. Accessed 28 Jan 2017
9. IEEE. In: IEEE-SA -IEEE Get 802 Program - 802.11: Wireless LANs. http://standards.ieee.org/about/get/802/802.11.html. Accessed 28 Jan 2017
10. Bluetooth SIG, Inc. How it Works. Bluetooth Technology Website. https://www.bluetooth.com/what-is-bluetooth-technology/how-it-works. Accessed 2017
11. WiFi Alliance. WiFi Certified Miracast. http://www.wi-fi.org/discover-wi-fi/wi-fi-certified-miracast. Accessed 28 Jan 2017

12. WiFi Alliance. WiFi Certified ac. http://www.wi-fi.org/discover-wi-fi/wi-fi-certified-ac. Accessed 28 Jan 2017
13. Alfaiate, J., Fonseca, J.: Bluetooth security analysis for mobile phones. In: 7th Iberian Conference on Information Systems and Technologies (CISTI 2012), Madrid, pp. 1–6 (2012)
14. Haataja, K., Hyppönen, K., Pasanen, S., Toivanen, P.: Bluetooth Security Attacks: Comparative Analysis, Attacks, and Countermeasures. Springer, Berlin (2013)
15. Khasawneh, M., Kajman, I., Alkhudaidy, R., Althubyani, A.: A survey on Wi-Fi protocols: WPA and WPA2. In: Martínez, P., Thampi, S., Ko, R., Shu, L. (eds.) Recent Trends in Computer Networks and Distributed Systems Security: Proceedings of the Second International Conference, SNDS 2014, Trivandrum, India, 13–14 March 2014, pp. 496–511. Springer, Berlin, Heidelberg (2014)
16. Ts, J.: X window system administration. Linux J. **1998** (1998)
17. Richardson, T., Stafford-Fraser, Q., Wood, K.R., Hopper, A.: Virtual network computing. IEEE Internet Comput. **2**(1), 33–38 (1998)
18. Citrix. In: Citrix. http://www.citrix.com. Accessed 29 Jan 2017
19. Microsoft Corporation. Microsoft Remote Desktop Clients. https://technet.microsoft.com/en-us/library/dn473009(v=ws.11).aspx. Accessed 29 Jan 2017
20. International Telecomunications Union. T.128: Multipoint Application Sharing. https://www.itu.int/rec/T-REC-T.128-200806-I/en. Accessed 2 Feb 2017
21. Trusted Computing Group. TPM Main Specification. https://trustedcomputinggroup.org/tpm-main-specification/. Accessed 2 Feb 2017
22. International Standards Organization: Information Technology – Telecommunications and Information Exchange Between Systems – Near Field Communication – Interface and Protocol (NFCIP-1) ISO/IEC 18092:2013 (2013)