Systems, Apparatus, Key Device, and Methods for Managing Access to Data Communication Ports

Field

The present invention relates to a data port blocking apparatus, a key device, a method and a system, for managing access to at least one of a plurality of data communication ports of a computing system residing in an interior environment.

Background

Conventional technologies for preventing unauthorized access to data communication ports such as USB ports typically involve a blocker for plugging in a port and means to unlock the blocker so that the port becomes usable. In some cases, the means to unlock the blocker is controlled by software of a computing device comprising the blocked port (e.g. EP2827279B1). However, this requires the computing device to be powered up and fully functional before the port can be unplugged. In other cases, a mechanical key is provided to unlock a blocker mechanically and no software is involved (e.g. US9646179B1 and US20080041125A1). In such cases, it is typical that a limited number of combinations of mechanical keys and blockers are manufactured. There is a security issue in that any of the blockers can be unlocked if the limited number of mechanical keys are obtained and each key is used in turn to try to unlock the blocker.

Summary

The present invention is defined in the independent claims. Optional features of the invention are defined in the dependent claims.

Brief Description of the Drawings

Embodiments of the invention will be better understood and readily apparent to one skilled in the art from the following written description, by way of example only, and in conjunction of the drawings, in which:

Figure 1 is a diagram illustrating an exemplary system for managing access to a plurality of data communication ports according to an example of the present disclosure.

Figure. 2A is a block diagram illustrating main components of an apparatus (also known as blocker) for blocking access to a data communication port according to an example of the present disclosure.

Figure 2B is a diagram illustrating an interlocking mechanism of a blocker for blocking access to a data communication port according to an example of the present disclosure.

Figure 2C is a diagram illustrating an interlocking mechanism of a blocker for blocking access to a data communication port according to another example of the present disclosure.

Figure 3 is a block diagram illustrating communication between a key device and a port blocker according to an example of the present disclosure.

Figure 4A shows an authentication challenge object in an access request object sent to a key device according to an example of the present disclosure.

Figure 4B shows a challenge response object in an authentication response object sent to a port blocker according to an example of the present disclosure.

Figure 5 is a block diagram illustrating components of an apparatus for blocking access to a data communication port according to an example of the present disclosure.

Figure 6A is a block diagram illustrating components of a key for communicating with a port blocker according to an example of the present disclosure.

Figure 6B is a block diagram illustrating locking/unlocking events that will occur at the data communication port according to an example of the present disclosure.

Figure 6C is a diagram illustrating use of magnetic connectors in an example of the present disclosure.

Figure 7 shows a setup of a computing device, a key and a port blocker and an overview of key and blocker provisioning processes according to an example of the present disclosure.

Figure 8 shows an authentication process performed during latching and unlatching of a port blocker to a port and during code reprovisioning for a port blocker according to an example of the present disclosure.

Figure 9 is a diagram showing a graphical user interface for providing codes to a key and a port blocker according to an example of the present disclosure.

Figure 10 is a flowchart showing the steps involved during the provisioning of a code to a key according to an example of the present disclosure.

Figure 11 is a flowchart showing the steps involved during the provisioning of a code to a new port blocker according to an example of the present disclosure.

Figure 12 is a flowchart showing the steps involved to replace an existing code stored in a port blocker with a new code according to an example of the present disclosure.

Figure 13 is a flowchart illustrating a method of managing access to the data communication port using a key device according to an example of the present disclosure.

Figure 14 is a flowchart illustrating a method of managing access to the data communication port using a port blocker according to an example of the present disclosure.

Detailed Description

Examples of the present disclosure aim to address a need for methods, systems, and apparatuses that provide users with a simple way of managing access to data communication ports through an unlimited or increased number of keys and blockers relative to conventional key-blocker systems.

In particular, examples of the present disclosure comprise an apparatus for blocking a data communication port of computing devices, such as USB ports, RJ45 ports, Lightning ports (e.g. for Apple devices) and other known communication Input/Output interface ports. Throughout the present disclosure, the apparatus for blocking also known as “blocker”. The apparatus comprises a plug insertable into the data communication port and the apparatus is configured to such that mechanisms residing at the plug engage or disengage from an interior of the data communication port to block or unblock the data communication port from connection respectively. The apparatus comprises activators such as buttons to trigger the blocking and unblocking. The USB ports can be USB-A, USB-C, and the like. A key external to the apparatus for accessing a blocked data communication port is provided to work with the apparatus electronically to unblock the blocked data communication port when required. The apparatus and/or the key may have power supply to power components requiring electricity to function. Each of the apparatus and the key has a part of an electrical circuit and the electrical circuit is a complete circuit when the apparatus and the key is physically connected to each other.

A technical effect of the blocker to only comprise a first part of an electrical circuit and a key device to only comprise a second part of the electrical circuit is that the blocker must be interfaced with the key device to form a complete electrical circuit for the blocker before the blocker can be released and/or attached to the data communication port. This enhances the security of the data communication port that is being protected by the blocker. In this way, the blocker cannot be non-destructively disengaged/released from the the data communication port to unblock the data communication port without the key device.

In the present disclosure, the same numerals are used for the elements having the same features or function throughout the figures and description.

With reference to Figure 1, there is provided an example of a system 1000 for managing access to a plurality of data communication ports of a plurality of computing devices 200. Reference numeral 200 refers to the plurality of computing devices 200 or each computing device 200. The plurality of computing devices 200 constitute a computer system. Each data communication port is a physical interface on each computing device 200 that allows the computing device 200 to connect with a device to enable transfer of data. In a preferred application of the system 1000, the computing devices 200 are computers and/or servers housed in an interior (or indoor) environment (e.g. a server room).

Each computing device 200 (e.g. mobile phone, server, computer, laptop etc.) has one or more data communication ports, such as USB-C, RJ45, HDMI, Firewire (IEEE 1394) and the like. Selected data communication ports of the computing device 200 can each be inserted with a blocker 100 to block the plugging of cables to the data communication port. This prevents unauthorized access to the data communication port. A key (or key device) 300 has to be used to unblock the blocker 100. Figure 1 shows 9 blockers 100 for blocking 9 data communication ports of 5 computing devices 200. 3 keys 300 programmed to unblock specific blockers 100 are provided. Notably, each key 300 is programmed to unblock more than one blockers 100.

Figure 2A shows an example of the connections between a data communication port 202 of one computing device 200 of Figure 1, one blocker 100 of Figure 1, and one key 300 of Figure, 1. The blocker 100 comprises a plug (not shown in Figure 2A) insertable into the data communication port 202. The plug is configured to be releasably engaged to or disengaged from an interior of the data communication port 202. The blocker 100 is configured to physically connect with the key 300 to allow an authentication process to commence to ascertain whether the key 300 is authorized to unlock or disengage from the interior of the data communication port 202. Each of the blocker 100 and the key 300 comprises a part of an electrical circuit and the electrical circuit is formed completely, i.e. form a complete circuit when the blocker 100 and the key 300 are physically connected. Forming this electrical circuit is a requirement to allow the authentication process to take place. In one example, the authentication process can be based on protocols such as challenge-response authentication (CRA) using for instance, cryptographic hashing like SHA-256. SHA-256 (Secure Hash Algorithm 256-bit) is a cryptographic hash function used for data integrity verification and secure authentication.

In the example of Figure 2A, the blocker 100 is also configured to physically connect with the key 300 if it is a new blocker 100 to allow the blocker 100 to be programmed to store a secret code (or token or password) to be used for disengaging from the data communication port 202.

Furthermore, if the blocker 100 is not a new blocker 100 and already has a secret code (or token or password), and this secret code needs to be replaced with a new secret code, the blocker 100 has to physically connect with the key 300 first. Thereafter, an authentication process to check whether the key 300 is authorized is performed using the old secret code. The key 300 must show that it has knowledge of the old secret code and/or must know how to solve a challenge from the blocker 100 , in order for the authentication to be successful. Such challenge may be configured to have something to do with the old secret code. The blocker 100 can be programmed to replace the old secret code with the new secret code if the key 300 is authenticated successfully. In one example, the authentication process used for provisioning the new secret code to the blocker 100 can also be based on protocols such as challenge-response authentication (CRA) using for instance, cryptographic hashing like SHA-256. SHA-256 (Secure Hash Algorithm 256-bit) is a cryptographic hash function used for data integrity verification and secure authentication.

In one example, with reference to Figure 2B, the blocker 100 described in examples above may use a physical interlocking system 220 for physically lodging the earlier described plug (210 in Figure 2B) in an interior 204 of a data communication port 202. For ports such as USB-A and RJ45, there may be one or more openings 205 in the port 202, which can be exploited for lodging the plug 210 in the port. The plug 210 may be configured to have one or more locking members 206, which are extendable by means of a suitable actuator (not shown in Figure 2B) to insert into the openings 205 in the port to create a physical interlock with the port 202.

In another example of the physical interlocking system 220, with reference to Figure 2C, for ports with limited space such as USB-C, the plug 210 of the blocker 100 described in examples above may be configured to employ a mechanical gasket concept. Such concept involves a gasket 208, which expands to engage an interior 204 of a data communication port (202 in Figure 2C), thereby forming a seal within the port 202, which creates a physical interlock with the port 202. The gasket or seal 208 can be provided by covering one or more locking members (similar to 206 in Figure 2B but not visible in Figure 2C) with a flexible cover. The one or more locking members are configured to extend or retract under the control of an actuator (not shown in Figure 2C). The flexible cover can be manipulated to form one or more protrusions to abut the interior 204 of the port 202 to form a seal. Specifically, when the one or more locking members extends, they push and stretch the flexible cover such that the one or more protrusions are extended or expanded to abut and engage the interior 204 of the data communication port 202. When the

one or more locking members retracts, the flexible cover recovers to its initial form and the protrusions retracts to disengage from the interior 204 of the data communication port 202.

The physical interlocking system 220 of Figure 2B or Figure 2C may comprise an electromechanical lever or switch, which is toggled by an external key via secure authentication involving a token, which makes it a digital security lock system. The electromechanical lever or switch shifts the locking element or gasket in and out of its locking position to provide a physical lock with the port, which differs from other commonly used commercial solutions using only friction (e.g. Lindy, SmartKeeper).

More details of the blocker 100 and the key 300 described earlier are described below.

With reference to Figure 3, the blocker 100 has an input/output interface (i.e. I/O interface) 106 and memory storing a unique apparatus (or blocker) identifier token 110 provided to the blocker 100. This unique blocker identifier token 110 is the secret code described earlier. The key 300 has a key I/O interface 304 and memory storing a key identifier token 308. The token 308 corresponds to the unique blocker identifier token 110 of the blocker 100 that the key 300 can unlock. The key 300 can store one or more of the token 308 to unblock more than one blockers 100. The blocker I/O interface 106 and key I/O interface 304 are capable of interfacing or connecting to each other so that the blocker 100 can be communicatively coupled to the key 300. In the present example, the blocker 100 has to be physically connected to the key 300 to enable the blocker 100 to block and unblock a data communication port (202 in Figure 2A). The blocker 100 has to be physically connected to the key 300 to program the blocker 100 to store or replace the unique blocker identifier token 110. The I/O interfaces 106 and 304 are configured to be connected mechanically or in another example, via magnetic connectors. Each of the blocker 100 and the key 300 comprises a part of an electrical circuit. The electrical circuit is formed as a complete circuit when the I/O interfaces 106 and 304 are connected.

Figure 5 shows a block diagram of the main hardware components of the blocker 100. The blocker 100 comprises the plug 210 described earlier and an electromechanical locking mechanism 104. The blocker 100 may be characterized by a length L1 (mm), a width W1 (mm), and a height H1 (mm). H1 is not shown. The length L1 may be in the range of from 20 mm to 100 mm, from 30 mm to 90 mm, from 50 mm to 90 mm, from 60 mm to 85 mm, or any combinations of upper and lower numerical values described above or combinations of any numerical value in the ranges listed above. The width W1 may be in the range of from 10 to 60 mm, 15 mm to 55 mm, or any combinations of upper and lower numerical values described

above or combinations of any numerical value in the ranges listed above. The height H1 may be in the range of from 1 mm to 30 mm, from 10 mm to 30 mm, from 20 mm to 25 mm, or any combinations of upper and lower numerical values described above or combinations of any numerical value in the ranges listed above. The blocker 100 may be characterized by an aspect ratio of from 1:2 to 3:5 or combinations of any numerical value therebetween, wherein the aspect ratio is defined by a width (W1) to length (L1) ratio of the blocker 100. An advantage of the aspect ratio in the abovementioned range is that the blocker 100 is of a compact design. It will be appreciated that the width W1 of the blocker 100 may be configured based on a form factor of the data communication port for which the blocker 100 is releasably engaged with.

The electromechanical locking mechanism 104 is an actuator for actuating the physical interlocking system 220 described earlier with reference to Figure 2B or 2C to block or unblock a data communication port 202. The electromechanical locking mechanism 104 may comprise a stepper motor and mechanical parts like a movable screw with screw thread. The screw can be moved by the stepper motor to screw or unscrew. In the case of the physical interlocking system 220 of Figure 2B, the screw can be directly or indirectly (via parts for transmitting motion such as gears) connected to one or more locking elements 206 in Figure 2B residing in the plug 210 and such one or more locking elements 206 will protrude or retract when the screw is screw or unscrewed. When protruded, the locking elements 206 will engage one or more openings 205 in the interior 204 of the data communication port 202. In the case of the physical interlocking system 220 of Figure 2C, the one or more locking elements may also be present but they are wrapped or covered by a flexible cover or seal 208. When the one or more locking elements are moved by the actions of the screw, they will expand the seal (or gasket) 208 to lodge the plug 210 in the data communication port 202 or retract the seal 208 to dislodge the plug 210.

With reference to Figures 3 and 5, the blocker 100 comprises a processor 112 connected to the blocker I/O interface 106 for connecting to the key 300. The processor 112 has access to memory 108 for storing the blocker identifier token 110. The memory 108 also stores instructions executable by the processor 112 to operate the functions of the processor 112. In one example, the processor 112 is configured to, upon physical connection with the key device 300, execute instructions in the memory 108 to activate an authentication process. The authentication process starts with the blocker 100 sending an access request object 10 to the key device 300 and thereafter, receiving an authentication response object 20 from the key device 300. The authentication response object 20 has to be deemed “correct” by the processor 112 in order for the key device 300 to be authenticated successfully. If there is a blocker identifier token 110 stored in memory 108, the blocker 100 will engage in further data communication with the key

300 only if authentication is successful. If there is no blocker identifier token 110 in memory 108 and the blocker 100 is a new blocker, the authentication process will not take place and the blocker 100 will just allow data communication with the key device 300 to take place. One example of the authentication process that can be implemented is Challenge Response Authentication involving sending of random data challenge and hash function.

With reference to Figure 4A, an example of the access request object 10 of Figure 3 is a data packet comprising a random (text) string 42 generated by the processor 112 of Figure 5. In one example, the processor 112 of the blocker 100 temporarily stores this random text string 42 in the memory 108 after it is generated by the processor 112. The required headers should be provided in the data packet to indicate to the key 300 that the data packet is the access request object 10 so that the key 300 will know what actions to take after receiving the data packet.

With reference to Figure 4B, an example of the authentication response object 20 of Figure 3 is a data packet comprising a generated text output 44. In one example, this text output 44 may be generated by a processor of the key 300 using an algorithm taking in the key identifier token 308 stored at the blocker 100 and the random string of text 42 received from the blocker 100 as inputs. After receiving the authentication response object 20, the processor 112 uses the same algorithm taking in the blocker identifier token 110 and the temporarily stored random text string 42 as inputs to compute a text result. This text result is then compared against the text output 44 to see whether they match. If they match, the key 300 is authenticated and the blocker 100 will allow further data communication with the key 300 to take place. In the present example, the key identifier token 308 has to be the same as the blocker identifier token 110 for the text result and the text output 44 to match. The further data communication between the blocker 100 and key 300 after key authentication may be, for instance, sending instructions from the key 300 to activate the blocker 100 to unblock or block a data communication port, or sending a new token to replace an existing blocker identifier token 110 stored at the blocker. Similarly, the required headers should be provided in the data packet to indicate to the blocker 100 that the data packet is the authentication response object 20 so that the blocker 100 will know what actions to take after receiving the data packet.

In one example, in the case that the key 300 stores more than one key identifier tokens 308 for unblocking more than one corresponding blockers 100, the authentication response object 20 can include more than one text outputs 44 generated using each stored key identifier token 308 and the random text string 42 received. Upon receiving the more than one text outputs 44, the processor 112 of the blocker 100 checks whether text result computed using the blocker identifier

token 110 and the random text string 42 matches with any one of the text outputs 44. If there is a match, the key 300 is authenticated successfully and the blocker 100 will allow further data communication with the key 300 to take place.

Figure 6A shows a block diagram of the main hardware components of the key 300. The key 300 comprises a processor 310, which has access to a memory 306. The memory 306 stores instructions executable by the processor 310 to operate the functions of the processor 310. The key 300 comprises a key I/O interface 304 for connecting to the blocker I/O interface 106 of Figure 5. The key 300 comprises a data communication interface 302 for connecting to a computing device (e.g. 200 of Figure 1) in a wired or wireless manner. For instance, the data communication interface 302 can be a serial port like the USB port and a USB cable is used to connect the key 300 to the computing device (e.g. 200 of Figure 1). In another example, the data communication interface 302 may be a wireless transceiver for connecting to the computing device for wireless data communication via WiFi, Bluetooth, Near Field Communication Technology, and the like. The key 300 further comprises an “unlock” trigger 312 activatable to cause the processor 310 to send an instruction to the blocker 100 to unblock a data communication port (202 in Figure 2A) and a “lock” trigger 314 activatable to cause the processor 310 to send an instruction to the blocker 100 to block the data communication port 202 in Figure 2A). The triggers 312 and 314 can be used for such unblock and block functions after the key 300 has successfully authenticated itself to the (physically connected) blocker 100 using the authentication process described earlier. The triggers 312 and 314 can be the electromechanical levers or switches described earlier with reference to Figures 2B and 2C. The key device 300 may comprise a power supply or source 301 i.e. a battery wherein the power supply or source 301 is operably connected to a power button for powering the key 300

Referring to Figure 6A, the key device 300 may be characterized by a length L2 (mm), a width W2 (mm), and a height H2 (mm). H2 is not shown. The length L2 may be in the range of from 20 mm to 100 mm, from 30 mm to 95 mm, from 40 mm to 95 mm, from 50 mm to 95 mm, from 70 mm to 95 mm, or any combinations of upper and lower numerical values described above or combinations of any numerical value in the ranges listed above. The width W2 may be in the range of from 10 mm to 60 mm, from 20 mm to 50 mm, from 30 mm to 45 mm, or any combinations of upper and lower numerical values described above or combinations of any numerical value in the ranges listed above. The height H2 may be in the range of from 1 mm to 25 mm, from 10 mm to 25 mm, from 20 mm to 25 mm, or any combinations of upper and lower numerical values described above or combinations of any numerical value in the ranges listed above. The key device 300 may be characterized by an aspect ratio of about 3:10 to 5:6 or

combinations of any numerical value therebetween, wherein the aspect ratio is defined by a width (W2) to length (L2) ratio of the key device 300. An advantage of the aspect ratio in the abovementioned range is that the key device 300 is of a compact design.

It will be appreciated that the dimensions may be configured accordingly to enable interfacing of the blocker 100 and the key device 300.

Figure 6B shows 3 configurations of the data communication port 202, the blocker 100 and the key 300 shown in Figure 2A to unlock (or unblock) a locked (or blocked) port 202. In a first (lock) configuration 602, the blocker 100 is locking or blocking the port 202 and the key 300 is being moved by a user to physically connect with the blocker 100.

Each of the blocker 100 and the key 300 comprises a part of an electrical circuit. The electrical circuit is formed as a complete circuit when the blocker 100 and the key 300 are physically connected. Forming the complete electronic circuit is required to lock or unlock the blocker 100 from the port 202.

Upon connection, as shown in a second (toggling) configuration 604, the electrical circuit is formed by the key 300 and the blocker 100 to define a complete circuit. The blocker 100 can initiate the key authentication process described earlier after the electrical circuit is formed. Once the key is authenticated, a button (i.e. the trigger 312 of Figure 6A) of the key 300 can be pressed by a user to toggle the lock of the blocker 100 to cause the blocker 100 to unblock the port 202. Once unlocked, the blocker 100 can remove and unplug its plug (210 in Figure 5) from the port 202. The scenario is similar for blocking the port 202. Firstly, the key 300 is physically connected to the blocker 100. This physical connection initiates the key authentication process at the blocker 100. Once the key is authenticated, a button (i.e. the trigger 314 of Figure 6A) can be pressed by a user to toggle the lock of the blocker 100 to cause the blocker 100 to block the port 202.

A third (unlock) configuration 606 in Figure 6B shows the blocker 100 is connected to the key 300 and the blocker 100 is unplugged from the port 202. In the present example, the key device 300 comprises a power supply or source 301 i.e. a battery wherein the power supply or source 301 is operably connected to a power button for powering the key 300. The battery can be a rechargeable battery and in this case, a battery charging circuit would be provided. The blocker 100 does not have a power source. When the electrical circuit is formed completely, the blocker 100 is powered up by the power supply of the key 300. The electrical components of the blocker

100 described in Figure 6A can operate after the blocker 100 is powered up. For instance, the processor 310 of the blocker 100 is powered up to execute a program to perform the authentication process. The blocker 100 is only mechanically (and not electrically) locked to the port 202 via for instance, the physical interlocking system 220 of Figure 2B or 2C.

In another example, the electric circuit may be configured to power both the key 300 and the blocker 100 using a power source residing at the blocker 100 and not at the key 300. In a further example, both the key 300 and the blocker 100 have individual power sources. In this case, the electrical circuit does not need to be responsible for channeling power from the blocker 100 to key 300 or vice versa. In another example, the key 300 obtains power from a computing device (e.g. 200 of Figure 1) after the key 300 is connected to the computing device and the obtained power is shared with the blocker 100 when the computing device, the key 300 and the blocker 100 are connected like the second configuration 604 shown in Figure 6B. The above-described power sources may be rechargeable.

As mentioned earlier, in one example, the I/O interface 106 of the blocker 100 in Figure 5 and the I/O interface 304 of the key 300 in Figure 6A are configured to be connected via magnetic connectors. Specifically, magnetic self-mating connectors may be used as an alternative to mechanical mating. Figure 6C shows the key 300 of Figure 6A and the Blocker 100 of Figure 5. Figure 6C also shows an example of a magnetic self-mating connector 608 designed to provide seamless and secure connection for power and data transmission. The magnetic connector 608 comprises 4 pins 610 and magnetic pads 612 for attracting and contacting a corresponding magnetic connector. The magnetic pads 612 enables automatic alignment when the magnetic connector 608 is connected to the corresponding magnetic connector. Two of the pins are dedicated to power transmission i.e. one pin carries positive voltage, while the other serves as the ground or negative connection. This setup ensures that the connected devices receive the necessary power to operate. The remaining two pins are used for data transmission. These data transmission pins facilitate the exchange of information between the connected key 300 and blocker 100, enabling functionalities such as data transfer, communication, and control signals.

Each of the key 300 and the blocker 100 are mounted with a pair of magnetic connectors such as the magnetic connector 608. The polarities of the pair of magnetic connectors have to be configured such that attraction will take place when they are in close proximity. At the blocker 100, the magnetic connector 608 is preferably mounted at an opposite end of the plug 210, which is for plugging into a port 202 to be blocked. At the key 300, the magnetic connector 608 is preferably mounted at an opposite end of the part comprising means for connecting to a

computing device 200. When the magnetic pads 612 of the pair of magnetic connectors are placed sufficiently close to each other, they will be attracted by magnetic force. In the present example, one of the pair of magnetic connectors has protruding pins and the other magnetic connector has corresponding sockets for receiving the protruding pins. The blocker 100 comprises a first part 616 of an electrical circuit 618 and the key device 300 comprises a second part 614 of the electrical circuit 618. When the magnetic pads 612 of the key 300 and blocker 100 are in contact, the protruding pins will slide into the respective sockets, thereby electrically connecting the key 300 and blocker 100 and the electrical circuit 618 partially in the key 300 and partially in the blocker 100 is formed completely. The connected magnetic connectors of the key 300 and the blocker 100 can easily break-away to disconnect the electrical circuit 618 when subjected to an acute tensile force. In this example, the electrical circuit 618 comprises the power lines for powering the blocker 100 and the data transmission lines for data communication between the key 300 and the blocker 100.

The key 300 and the blocker 100 described above may be configured as a digital key-blocker lock system (or key-blocker system) that allows users to digitally configure each key to lock/unlock a high number of blockers through a provisioning process. Figure 7 shows a computing device 200, which is a laptop in the present example. The computing device 200 is running a program (or an application) (i.e. Personal Computer (PCT) Software) with a graphical user interface 702 for performing the provisioning process. Specifically, there are two types of provisioning processes, a key provisioning process 704 and a blocker provisioning process 706, which can be performed separately from each other. In order for the key 300 to lock and unlock one blocker 100, both of them has to be provided with the same password.

The key provisioning process 704 provides the key 300 with one or more tokens (or passwords) to allow it to unlock or unblock one or more respective blockers 100. Each blocker 100 has to carry a matching token stored at the key 300 in order for the key 300 to lock or unlock a blocker 100. The ability to allow a user to configure a large number of key-to-blocker combinations is something that the current commercial solutions lack. In the example of Figure 7, the computing device 200 and the key 300 have to be physically connected, for instance via USB connection, and data communication between the computing device 200 and the key 300 has to be enabled for the key provisioning process 704 to take place.

For better security, it is preferred that the blocker 100 is provided with only one token (or password). The blocker provisioning process 706 is performed when the blocker 100 is either a new blocker and needs to be provided with a password, or that an old password stored at the

blocker 100 needs to be replaced. In the example of Figure 7, for the blocker provisioning process 706 to take place, the key 300 and the computing device 200 have to be physically connected first, and thereafter, the blocker 100 has to be physically connected to the key 300.

Details of the earlier described authentication process initiated by the blocker 100 to authenticate the key 100 are described below with reference to Figure 8.

Figure 8 shows an example of an authentication process 800 specifically using CRA and involving SHA256 algorithm. The authentication process 800 is performed when the key 300 is used to latch (or lock or block) a blocker 100 to a data communication port (e.g. 202 in Figure 2A) or to unlatch (or unlock or unblock) a blocker 100 from a data communication port (e.g. 202 in Figure 2A). The authentication process 800 is also performed when the key 300 is connected to a computing device (e.g. 200 in Figure 2A) running an application to re-provision the blocker 100 with a new password to replace an old password.

In order for a key 300 to authenticate itself successfully to the blocker 100, the key 300 and the blocker 100 have to be provided with the same shared secret code 802. In the present example, the key 300 comprises a power source (i.e. battery) and the blocker 100 is powered up after the key 300 is physically connected with the blocker 100. After the blocker 100 is powered up, the authentication process 900 starts by the blocker 100 generating a random string and sending a random challenge 808 containing the random string to the key 300. Preferably, the blocker 100 generates only one random challenge each time it is powered up. The blocker 100 has to be disconnected from power and powered up again (i.e. undergo a reset) to send another random challenge. The blocker 100 may also be configured to perform a specified maximum number of actions, e.g. not greater than two actions (lock, unlock) per random challenge. Limiting to one random challenge and restricting the number of actions for each random challenge keeps processing to minimum and the software and hardware design of the blocker 100 can be kept simple, thereby incurring lesser resources during manufacturing. The key 300 receives the random challenge 808 and inputs the shared secret code 802 stored at the key 300 and the random string to the SHA256 algorithm 810 to generate a response 806. The response 806 is sent from the key 300 to the blocker 100. At the blocker 100, the blocker 100 also inputs the shared secret code 802 and the random string that the blocker 100 generates to the SHA256 algorithm 804 to generate a response 810. When the response 806 is received by the blocker 100, it is compared with the response 810 at a step 812. If the response 806 matches the response 810, the key's authentication is successful (i.e. a "pass" result) and the blocker 100 confirms that the key 300 has the same shared secret code 802. If the response 806 and the response 810 do not match, the key's authentication is unsuccessful (i.e. a "fail" result) and the

blocker 100 will not allow further data communication with the key 300. After a “fail” result, for the blocker 100 to be able to re-initiate another authentication process 900 with the key 300, the key 300 has to be disconnected from the blocker 100 to switch off the power to the blocker 100. Another attempt to authenticate key 300 can be made by reconnecting the key 300 with the blocker 100. However, before the second attempt, the key 300 should be programmed to store the same shared secret code 802 so that authentication will be successful.

In another example, in the case that the key 300 has more than one secret codes 802 for unlocking or locking more than one blockers 100, a modification to the authentication process 900 above is that the key 300 will compute and send more than one of the response 806 for the blocker 100 for each code the key 300 stores. For instance, after one random challenge is sent, the blocker 100 allows the key 300 to send more than one of the response 806 generated based on different secret codes for authentication. The blocker 100 will compare each of the responses 806 received. The key 300 is authenticated if any one of the response 806 matches the response 810 generated by the blocker 100.

Figure 9 shows an example of the provisioning graphical use interface 702 described above with reference to Figure 7. The graphical use interface 702 is a window comprising a drop down menu 902 to allow a serial port of the computing device 200 of Figure 7 to be selected. A user is required to select a port number where the key 300 of Figure 7 is connected. Once selected, the user clicks an Open Serial Port button 904 to open the selected port with the connected key 300. Provisioning can take place thereafter. If only the key 300 is connected to the selected port, a code can be provided to the key 300 by entering the code in a key code text box 906 provided. The key code will be programmed to store at the key 300 by clicking a key code button 908. In the present example, if a blocker 100 has to be provided with a code, a code can be provided to the blocker 100 by entering the code in a new code text box 910 that is provided. If the blocker 100 has an existing code, the user has to enter the existing code to an old code text box 912 so that the key 300 can successfully authenticate itself when the blocker 100 is connected to the key 300. After successful authentication, the key 300 would be allowed to replace the old or existing code with the new code. If the blocker 100 is a new blocker, only the new code text box 910 has to be completed and the old code text box 912 can be kept empty. A blocker button 908 can be clicked to initiate the provision of a new code to a new blocker or the provision of a new code to replace an old code of a blocker.

In another example, in the case that more than one codes can be stored at the key 300, the graphical user interface 702 may be modified to provide an add multiple codes button, which a

user can click to open up a window to allow the user to select a file containing a list of one or more pre-entered codes. An example of the file format of the file may be Microsoft Notepad, Words or Excel. After selecting the file, the user can click the key code button 908 to initiate the storage of all the codes in the file in the memory (e.g.. 306 in Figure 6A) of the key 300.

In one example, an administrator should keep records of all codes issued to keys and blockers to enable codes to be recovered in the case of loss of codes/keys or in the case that codes are forgotten by users. For security reasons, each blocker may be configured such that if its existing code is forgotten and unrecoverable, the blocker may have to be sent back to the manufacturer to have its memory reset.

The details of the steps of the key provisioning process 704 and the blocker provisioning process of Figure 7 are described below.

Figure 10 shows the computing device 200 and the key 300 of Figure 7 and the steps of the key provisioning process 704 of Figure 7. The key 300 can be a new key without any codes or have existing codes to be replaced. Specifically, the steps of the key provisioning process 704 are:
Step 1: A user plugs in the key 300 to a serial port (e.g. USB port) of the computing device 200 (i.e. personal computer or PC).
Step 1A: Upon plugging the key 300, an instruction to power on or to start the operating process of the key 300 is sent from the computing device 200.
Step 1B: At the computing device 200, the user selects the serial port and supplies one or more new secret code to store at the key 300 via the graphical user interface (e.g. 702 of Figure 9). After supplying the one or more new secret code, the user triggers the sending of the one or more new secret code to the key 300.
Step 2: The key 300 updates its memory (i.e. 306 in Figure 6A) with the one or more new secret code.
Step 2A: After updating its memory, the key 300 sends an acknowledgement to the computing device 200, which indicates that the new secret code is stored.
Step 3: The key provisioning process is completed after the computing device 200 receives the key's acknowledgement.

Figure 11 shows the computing device 200, the key 300 and the blocker 100 of Figure 7 and the steps of the blocker provisioning process 706 of Figure 7 for a new blocker. It is assumed that blocker 100 is the new blocker for this example. Note that new blocker also refers to a blocker that had stored codes previously but the code was reset and/or set to null by the user. For

instance, set to null can be that no code is typed in the new code text box 910 in Figure 9 when setting a code for the blocker 100. Specifically, the steps of the blocker provisioning process 706 to provide a code to a new blocker are:

Step 1: A user plugs in the key 300 to a serial port (e.g. USB port) of the computing device 200 (i.e. personal computer or PC).

Step 1A: Upon plugging the key 300, an instruction to power on the key 300 is sent from the computing device 200.

Step 1B: At the computing device 200, the user selects the serial port and supplies one new secret code to be stored at the new blocker 100 via the graphical user interface (e.g. 702 of Figure 9). After supplying the new secret code, the user triggers the sending of a provisioning request (or command) to provide the blocker 100 with the new secret code to the key 300.

Step 2: The key 300 receives the provisioning request from the computing device 200 to store the new secret code at the blocker 100.

Step 2A: The user physically connects the blocker 100 to the key 300 to power up the blocker 100 using a power source (e.g. battery) of the key 300. In another example, after the user physically connects the blocker 100 to the key 300, the key 300 has to send an instruction to power on or to start the operating process of the blocker 100.

Step 3: The blocker 100 powers up upon physical connection with the key 300 or in another example, the blocker 100 powers up after receiving the instruction from the key 300 to power on or to start its operating process.

Step 3A: The blocker 100 sends data to the key 300 to notify that it is a blocker with no stored code (i.e. new blocker). The blocker 100 does not require the key 300 to authenticate itself since the blocker 100 has no stored code.

Step 4: After the key 300 is notified that the blocker 100 is new, the key 300 proceeds to generate a message (or data packet) containing the new secret code received from the computing device 200.

Step 4A: The key 300 sends the message containing the new secret code to the blocker 100.

Step 5: The blocker 100 updates its memory (i.e. 108 in Figure 5) with the new secret code sent from the key 300.

Step 5A: After updating its memory, the blocker 100 sends an acknowledgement to the key 300, which indicates that the new secret code is stored.

Step 6: The key 300 receives the blocker’s acknowledgement.

Step 6A: The key 300 sends an acknowledgement to the computing device 200, which indicates that the blocker provisioning process is successful.

Step 7: The blocker provisioning process 706 is completed after the computing device 200 receives the key’s acknowledgement.

Figure 12 shows the computing device 200, the key 300 and the blocker 100 of Figure 7 and the steps of the blocker provisioning process 706 of Figure 7 for reprovisioning a blocker i.e. replace an existing code (old secret code) with a new code. The blocker 100 has an existing code stored in its memory (i.e. 108 in Figure 5). Specifically, the steps of the blocker provisioning process 706 to replace an old code of a blocker are:

Step 1: A user plugs in the key 300 to a serial port (e.g. USB port) of the computing device 200 (i.e. personal computer or PC).

Step 1A: Upon plugging the key 300, an instruction to power on the key 300 is sent from the computing device 200.

Step 1B: At the computing device 200, the user selects the serial port and supplies one new secret code to replace the old secret code stored at the new blocker 100 via the graphical user interface (e.g. 702 of Figure 9). The user also supplies the old secret code of the blocker 100 via the graphical user interface (e.g. 702 of Figure 9) to allow the key 300 to be able to authenticate itself. After supplying the new secret code and the old secret code, the user triggers the sending of a provisioning request (or command) to the blocker 100 to replace the old secret code of the blocker 100 with the new secret code.

Step 2: The key 300 receives the provisioning request from the computing device 200 to replace the old secret code of the blocker 100 with the new secret code.

Step 2A: The user physically connects the blocker 100 to the key 300 to power up the blocker 100 using a power source (e.g. battery) of the key 300. In another example, after the user physically connects the blocker 100 to the key 300, the key 300 has to send an instruction to power on or to start the operating process of the blocker 100.

Step 3: The blocker 100 powers up upon physical connection with the key 300 or in another example, the blocker 100 powers up after receiving the instruction from the key 300 to power on or to start its operating process.

Step 3A: As the blocker 100 has the old secret code stored in its memory, the blocker 100 generates a random string and a challenge (data message or packet) containing the random string to the key 300. The blocker 100 sends the generated challenge to the key 300.

Step 4: After receiving the challenge, the key 300 inputs the old secret code and the random string to an algorithm, for instance, a cryptographic hash function like SHA256, to compute a key authentication response.

Step 4A: The key 300 sends the computed key authentication response to the blocker 100.

Step 5: The blocker 100 inputs the old secret code and the random string to the same algorithm used by the key 300, for instance, a cryptographic hash function like SHA256, to compute a blocker authentication response. The blocker 100 compares the key authentication response

received from the key 300 with the blocker authentication response. If the results of the hashing are correct i.e. the key authentication response and the blocker authentication response matches, the key 300 and the blocker 100 are successfully authenticated.

Step 5A: Upon successful authentication, the blocker 100 sends an acknowledgement indicative of successful authentication to the key 300. If authentication is not successful, the blocker 100 will not send the acknowledgement or the blocker 100 will send a message to notify that authentication is unsuccessful. The blocker 100 will take no further action in the case of unsuccessful authentication. The blocker 100 may require powering off and powering on to allow another authentication process to take place, or receive another instruction from the key 300 to commence authentication again.

Step 6: After the key 300 receives the acknowledgement indicative of successful authentication, the key 300 proceeds to generate a message (or data packet) containing the new secret code received from the computing device 200.

Step 6A: The key 300 sends the message containing the new secret code to the blocker 100.

Step 7: The blocker 100 updates its memory (i.e. 108 in Figure 5) by replacing the old secret code with the new secret code sent from the key 300.

Step 7A: After updating its memory, the blocker 100 sends an acknowledgement to the key 300, which indicates that the old secret code is replaced with the new secret code.

Step 8: The key 300 receives the blocker's acknowledgement confirming code replacement.

Step 8A: The key 300 sends an acknowledgement to the computing device 200, which indicates that the blocker provisioning process is successful.

Step 9: The blocker provisioning process 706 is completed after the computing device 200 receives the key's acknowledgement.

An exemplary firmware for operating the key 300 is described hereinafter with respect to Figure 13. In particular, Figure 13 is a flow chart illustrating a method 930 for managing access to at least one of a plurality of data communication ports of a computing system residing in an interior environment. The method 930 may comprise the following steps of:

(a) receiving 932 an activation signal for activating a key device; wherein the key device is configured for interfacing with a data port blocking apparatus, wherein the data port blocking apparatus comprises a first part of an electrical circuit and the key device comprises a second part of the electrical circuit, and the electrical circuit is a complete circuit upon interfacing the key device with the apparatus; wherein the trigger is associated with a request for executing a user-specified task;

(b) when the key device is interfaced with the data port blocking apparatus, performing 934 an authentication process to authenticate the apparatus; and

(c) after successful authentication of the apparatus, allowing 936 data communication between the apparatus and the key device to take place to execute the user-specified task to manage access to the data communication port.

Referring to step 932 of receiving an activation signal for activating the key device, the activation signal may be generated by: powering the key device through pressing a power button comprised in the key device, wherein the power button is operably connected to the processor; and activating a lock/unlock trigger comprised in the key device.

An exemplary firmware for operating the blocker is described hereinafter with reference to Figure 14. In particular, Figure 14 is a flowchart illustrating a method 950 for managing access to at least one of a plurality of data communication ports of a computing system residing in an interior environment in a system comprising a key device and a data port blocking apparatus. The method may comprise the following steps of:

(a) receiving 952 an activation signal for activating the apparatus upon triggering of a key device; wherein the data port blocking apparatus comprises a first part of an electrical circuit and the key device comprises a trigger for triggering the apparatus and a second part of the electrical circuit, and the electrical circuit is a complete circuit upon interfacing the key device with the apparatus; wherein the trigger is associated with a request for executing a user-specified task;

(b) when the key device is interfaced with the data port blocking apparatus and in response to (i) a force applied to a unlock/lock trigger on the key device or (ii) a user input from a computing device, performing 954 an authentication process to authenticate the key device; and

(c) after successful authentication of the key device, allowing 956 data communication between the apparatus and the key device to take place to execute the user-specified task to manage access to the data communication port.

In step 952 above, the triggering of the key device may comprise activating a lock/unlock trigger comprised in the key device to send an activation signal to activate the blocker 100.

The above-described examples of key-blocker systems can be implemented in places where data breaches would lead to potential unauthorized collection, use or disclosure of personal data held by organizations, such as, hospitals, data centers, etc. The above-mentioned systems can also be implemented in places where data breaches would cause disruptions to critical infrastructure and/or means of transportation, such as ships, aero-planes, power plants, transport systems, video surveillance systems and the like.

Preferably, the above-described examples of key-blocker systems should be deployed in an interior environment is selected from the group consisting of: a residential interior environment, a commercial interior environment, a healthcare provider interior environment, a defence agency interior environment, a transport means interior environment, a power plant interior environment, a computer peripheral machine interior environment, a household appliance interior environment.

With regard to the physical interlocking system of key-blocker systems, the type of communication port to be blocked (restrict access) could limit the implementation given the shape and size limitations. Conventional techniques such as friction-only blockers can only deter (but cannot completely block) the unauthorized access to the communication port, and software blockers would be vulnerable to hardware attacks without any form of hardware protection. In addition, conventional techniques provide limited combinations of key-blocker, which exposes the possibility of other users possessing compatible key(s) to unlock blockers alike. This means that a bad actor could purchase all of the limited number of combinations of blocker-key and access the blocked communication port with the keys on hand by brute force experimentation. The examples of the present disclosure aim to address the above-mentioned problems in prior art.

The computing device 200 in Figure 1 may comprise a processing unit (or processor) for processing software including one or more programs for running one or more computer/server/device applications to enable a backend logic flow or the method or methods for carrying out the steps as described with reference to the earlier Figures (e.g. the key provisioning process 704 and the blocker provisioning process 706 of Figure 7). The processor 112 of the blocker 100 of Figure 5 and/or the processor 310 of the key 300 of Figure 6A may be configured for processing software including one or more programs for running one or more applications to enable a backend logic flow or the method or methods for carrying out the steps as described with reference to the earlier Figures (e.g. the key provisioning process 704 and the blocker provisioning process 706 of Figure 7).

With reference to Figure 7, the processing unit of the computing device 200 may include user input modules such as a computer mouse 712 (a laptop mouse control pad is shown in Figure 7), keyboard/keypad 710, and/or a plurality of output devices such as a display device 708. The display device 708 may incorporate technologies like LCD, LED, OLED, and the like. It may also be a touch screen capable of receiving user input. User interface components of the blocker 100 and the blocker 300 are described with reference to Figure 5 and Figure 6A respectively.

The processing unit of the computing device 200 may be connected to a computer network via a suitable transceiver device (i.e. a network interface), to enable access to e.g. the Internet or other network systems such as a wired Local Area Network (LAN) or Wide Area Network (WAN). The processing unit may be connected to one or more external wireless communication enabled devices via a suitable wireless transceiver device, e.g. a WiFi transceiver, Bluetooth module, Mobile telecommunication transceiver suitable for Global System for Mobile Communication (GSM), 3G, 4G, 5G telecommunication systems, and the like. Through the computer network (i.e. internet), the processing unit can gain access to one or more storages i.e. data storages, databases, data servers and the like connectable to the computer network to retrieve and/or store data in the one or more storages.

The processing unit may include a microprocessor, a Random Access Memory (RAM) and a Read Only Memory (ROM). The components of the processing unit typically communicate via an interconnected bus and in a manner known to the person skilled in the relevant art. Likewise, the processor 112 of the blocker 100 of Figure 5 and/or the processor 310 of the key 300 of Figure 6A may also include RAM and ROM.

The programs for execution by the processing unit, the processor 112 and/or the processor 302 may be supplied to the user of the computing device 200 and/or the key 300 and the blocker 100, encoded on a data storage medium such as a CD-ROM, on a flash memory carrier, Solid State Drive, or a Hard Disk Drive, and are to be read using a corresponding data storage medium drive of a data storage device. Such computer or application programs may also be downloaded from the computer network. The application programs are read and controlled in its execution by the microprocessor of the processing unit, the processor 112 and/or the processor 310. Intermediate storage of program data may be accomplished using RAM.

In more detail, one or more of the computer or application programs may be stored on any non-transitory machine- or computer- readable medium. The machine- or computer- readable medium may include storage devices such as magnetic or optical disks, memory chips, or other storage devices suitable for interfacing with a general purpose computer. The machine- or computer- readable medium may also include a hard-wired medium such as that exemplified in the Internet system, or wireless medium such as that exemplified in the Wireless LAN (WLAN) system and the like. The computer program when loaded and executed on such a general-purpose computer effectively results in an apparatus that implements the steps of the computing methods in examples herein described.

Representative embodiments of the present disclosure described above can be described as set out in the following paragraphs. Each paragraph is an embodiment. Reference numerals of the elements in the figures of the present disclosure that are examples of the features described are provided in brackets.

A. A data port blocking apparatus (100) in a system (1000) for managing access to at least one of a plurality of data communication ports of a computing system (200) residing in an interior environment, wherein the apparatus (100) comprises:
a plug (210) insertable into a data communication port (202) to block access to the data communication port (202);
an electromechanical locking mechanism (104) configured to releasably engage the plug (210) with an interior of the data communication port;
an Input/Output (I/O) interface (106) for interfacing the apparatus (100) with a key device (300);
a memory (108) for storing data for authenticating the key device (300); and
a processor (112) coupled to the I/O interface (106),
wherein the processor (112) is configured to, when the key device (300) is interfaced with the apparatus (100), execute instructions in the memory (108) to operate the apparatus (100) to:
perform an authentication process to authenticate the key device (300); and
after successful authentication of the key device (300), allow data communication between the apparatus (100) and the key device (300) to take place to manage access to the data communication port (202),
wherein the apparatus (100) comprises a first part (616) of an electrical circuit (618) and the key device (300) comprises a second part (614) of the electrical circuit (618), and the electrical circuit (618) is a complete circuit when the key device (300) is interfaced with the apparatus (100) via the I/O interface (106).

B. The apparatus (100) of paragraph A wherein the first part (616) of the electrical circuit (618) is configured for supplying power to the apparatus (100) upon forming the complete circuit so as to enable data communication between the apparatus (100) and the key device (300).

C. The apparatus (100) of paragraph B, wherein the apparatus (100) is configured to draw power from a power source residing at the key device (300) upon interfacing with the key device (300).

D. The apparatus (100) of any one of the preceding paragraphs, wherein the apparatus (100) is operable to: after successful authentication of the key device (300), engage in data communication directed to controlling of the electromechanical locking mechanism (104) to engage or disengage the plug (210) from the data communication port (202).

E. The apparatus (100) of any one of the preceding paragraphs, wherein the processor (108) is operable to: during the authentication process,
send an access request object (10) to the key device (300);
obtain an authentication response object (20) from the key device (300);
analyze the authentication response object (20) so as to generate a comparison result; and
determine whether the key device (300) is successfully authenticated based on the comparison result.

F. The apparatus (100) of paragraph E, wherein the data for authenticating the key device (300) stored in the memory comprises a code, and said code is stored at both the key device (300) and the apparatus (100) to be used in computations to authenticate the key device (300), wherein the memory (108) only stores said code for use in computations to authenticate the key device (300).

G. The apparatus (100) of paragraph F, wherein the access request object (10) comprises an authentication challenge object (808) comprising randomly generated data, a challenge response object (806) is generated at the key device (300) using the randomly generated data and the code stored at the key device (300), and the authentication response object (20) comprises the challenge response object.

H. The apparatus (100) of paragraph G, wherein the apparatus is operable to generate a second challenge response object using the randomly generated data and the code for comparison with the challenge response object generated by the key device to determine whether the two challenge response objects match, wherein a match indicates that the key device is successfully authenticated.

I. The apparatus (100) according to any one of the preceding paragraphs, wherein the electromechanical locking mechanism (104) comprises:
one or more locking members (206) configured to:
extend and engage one or more openings in the interior of the data communication port to prevent removal of the plug from the data communication port,

wherein the processor (112) is operable to:
control the electromechanical locking mechanism (104) to extend or retract the one or more locking members (206).

J. The apparatus (100) of any one of the preceding paragraphs, wherein the plug (210) comprises a flexible seal for wrapping at least a part of the electromechanical locking mechanism (104), wherein said part is moveable to expand the seal to abut and engage the interior of the data communication port.

K. The apparatus (100) of any one of the preceding paragraphs, wherein the I/O interface is a magnetic connector for interfacing with a corresponding magnetic connector at the key device.

L. A key device (300) in a system for managing access to at least one of a plurality of data communication ports of a computing system residing in an interior environment, wherein the key device (300) comprises:
an Input/Output (I/O) interface (304) for interfacing the key device (300) with a data port blocking apparatus,
a data communication interface (302) for interfacing the key device (300) with a computing device for configuring the key device (300) or configuring the data port blocking apparatus (100) via the key device;
a memory (306) storing data for authenticating the key device (300); and
a processor (310) coupled to the I/O interface (304),
wherein the processor (310) is configured to, when the key device (300) is interfaced with the data port blocking apparatus (100), execute instructions in the memory (306) to operate the key device (300) to:
proceed with an authentication process to authenticate the blocker (100); and
after successful authentication of the blocker (100), allow data communication between the data port blocking apparatus (100) and the key device (300) to take place to manage access to the data communication port,
wherein the data port blocking apparatus (100) comprises a first part of an electrical circuit and the key device comprises a second part of the electrical circuit, and the electrical circuit is a complete circuit upon interfacing of the key device (300) with the data port blocking apparatus (100) via the I/O interface (304).

M. The key device (300) of paragraph L, wherein the key device (300) comprises a power source and the data port blocking apparatus (100) draws power from the power source when the key device (300) is interfaced with the data port blocking apparatus (100).

N. The key device (300) of paragraph L or M, wherein the data port blocking apparatus (100) comprises a plug (210) insertable into a data communication port to block access to the data communication port; and an electromechanical locking mechanism (104) configured to releasably engage the plug with an interior of the data communication port; wherein the key device (300) is operable to: after successful authentication of the key device (300), engage in data communication directed to controlling of the electromechanical locking mechanism (104) to engage or disengage the plug (210) from the data communication port.

O. The key device (300) of any one of paragraphs L to N, wherein the key device (300) is operable to: during the authentication process,
receive an access request object (10) from the data port blocking apparatus (100); and
compute and send an authentication response object (20) to the data port blocking apparatus (100).

P. The key device (300) of paragraph O, wherein the data for authenticating the key device (300) stored in the memory (306) comprises a code, and said code is stored at both the key device (300) and the apparatus (100) to be used in computations to authenticate the key device.

Q. The key device (300) of paragraph P, wherein the access request object (10) comprises an authentication challenge object comprising randomly generated data, wherein the processor (310) is operable to generate a challenge response object using the randomly generated data and the code stored at the key device (300), and the authentication response object (20) comprises the challenge response object.

R. The key device (300) of paragraph Q, wherein the processor (310) is further configured to receive a second challenge response object from the data port blocking apparatus (100) for comparison with the challenge response object generated by the processor (310) to determine whether the two challenge response objects match, wherein a match indicates that the key device (300) is successfully authenticated.

S. The key device (300) of any one of paragraphs L to R, wherein the I/O interface (304) is a magnetic connector for interfacing with a corresponding magnetic connector at the data port blocking apparatus.

T. The key device (300) of any one of paragraphs L to S, wherein the key device (300) comprises at least one trigger (312, 314) operable to:
after the key device (300) has interfaced with the data port blocking apparatus (100) and the key device (300) has successfully authenticated, activate the data port blocking apparatus (100) to control the electromechanical locking mechanism (104) to engage the plug (210) with the interior of the data communication port or disengage the plug (210) from the interior of the data communication port.

U. The key device (300) of any one of paragraphs L to T, wherein more than one codes is stored in the memory (306), wherein each of the more than one codes is also stored in a memory (108) of a data port blocking apparatus (100) that is configured to store one code, wherein the code is to be used in computations to authenticate the key device (300) and upon successful authentication, allow the key device (300) to commence data communication with the data port blocking apparatus (100) with the stored code to manage access to a data communication port blocked or to be blocked by the data port blocking apparatus (100).

V. A method (930) for managing access to at least one of a plurality of data communication ports of a computing system residing in an interior environment, wherein the method comprises the steps of:
(a) receiving (932) an activation signal for activating a key device; wherein the key device is configured for interfacing with a data port blocking apparatus, wherein the data port blocking apparatus comprises a first part of an electrical circuit and the key device comprises a second part of the electrical circuit, and the electrical circuit is a complete circuit upon interfacing the key device with the apparatus; wherein the trigger is associated with a request for executing a user-specified task;
(b) when the key device is interfaced with the data port blocking apparatus, performing (934) an authentication process to authenticate the apparatus; and
(c) after successful authentication of the apparatus, allowing (936) data communication between the apparatus and the key device to take place to execute the user-specified task to manage access to the data communication port.

W. A method (950) for managing access to at least one of a plurality of data communication ports of a computing system residing in an interior environment in a system comprising a key device and a data port blocking apparatus, wherein the method comprises the steps of:

(a) receiving (952) an activation signal for activating the apparatus upon triggering of a key device; wherein the data port blocking apparatus comprises a first part of an electrical circuit and the key device comprises a trigger for triggering the apparatus and a second part of the electrical circuit, and the electrical circuit is a complete circuit upon interfacing the key device with the apparatus; wherein the trigger is associated with a request for executing a user-specified task;

(b) when the key device is interfaced with the data port blocking apparatus and in response to (i) a force applied to a unlock/lock trigger on the key device or (ii) a user input from a computing device, performing (954) an authentication process to authenticate the key device; and

(c) after successful authentication of the key device, allowing (956) data communication between the apparatus and the key device to take place to execute the user-specified task to manage access to the data communication port.

X. The method of paragraph V or paragraph W, wherein the user-specified task is selected from the group consisting of:

(i) a request for blocking access to a data communication port using an insertable plug of a data port blocking apparatus;

(ii) a request for controlling an electromechanical locking mechanism of the data port blocking apparatus to releasably engage the plug with an interior of the data communication port.

(iii) a request for provisioning the key device or the apparatus;

(iv) a request for re-provisioning the key device or the apparatus.

Y. The method of paragraph W or paragraph X dependent on paragraph W, wherein the authentication process comprises:

sending an access request object to the key device;

obtaining an authentication response object from the key device;

analyzing the authentication response object so as to generate a comparison result; and

determining whether the key device is successfully authenticated based on the comparison result.

Z. The method of paragraph V or paragraph X dependent on paragraph V wherein the authentication process comprises:

receiving an access request object from the apparatus;

sending an authentication response object to the apparatus for generating a comparison result.

AA. The method of paragraph Y or paragraph Z, wherein the access request object comprises an authentication challenge object (808) comprising randomly generated data, wherein the authentication response object comprises a challenge response object (806) generated at the key device using the randomly generated and the code stored at the key device, preferably the access request object consists of the authentication challenge object (808).

BB. The method of any one of paragraphs V to AA, wherein the method comprises:
interfacing the key device with a computing device so that data communication is able to take place between the key device and the computing device; and
storing one or more codes in a memory of the key device supplied by a user through a graphical user interface displayed on a display of the computing device, wherein each of the one or more codes is also stored in a memory of a data port blocking apparatus that is configured to store one code, wherein each of the one or more codes is to be used in computations to authenticate the key device and upon successful authentication, allow the key device to commence data communication with the data port blocking apparatus with the stored code to manage access to a data communication port blocked or to be blocked by the data port blocking apparatus.

CC. The method of any one of paragraphs V to AA, wherein the method comprises:
interfacing the key device with a computing device and with a data port blocking apparatus so that data communication is able to take place between the key device and the computing device and between the key device and the computing device;
sending a request from the computing device to the key device to store a code in a memory of the data port blocking apparatus, wherein the code is supplied by a user through a graphical user interface displayed on a display of the computing device, wherein the code is to be used in computations for authentication of a key device; and
sending the code from the key device to the data port blocking apparatus for storing in the memory of the data port blocking apparatus.

DD. The method of any one of paragraphs V to AA, wherein the method comprises:
interfacing the key device with a computing device and with a data port blocking apparatus so that data communication is able to take place between the key device and the computing device and between the key device and the computing device;
sending a request from the computing device to the key device to replace an existing code stored in the memory of the data port blocking apparatus with a new code, wherein the request comprises the new code and the existing code, which are supplied by a user through a graphical user interface displayed on a display of the computing device;

sending an instruction from the key device to the data port blocking apparatus to power on the data port blocking apparatus;

in response to the instruction, performing the authentication process at the data port blocking apparatus to authenticate the key device, wherein the existing code is used in computations at the key device and the data port bocking apparatus to authenticate the key device; and

upon successful authentication, sending the new code from the key device to the data port blocking apparatus for replacing the existing code.

EE. The method of any one of paragraphs V to DD, wherein the interior environment is selected from the group consisting of: a residential interior environment, a commercial interior environment, a healthcare provider interior environment, a defence agency interior environment, a transport means interior environment, a power plant interior environment, a computer peripheral machine interior environment, a household appliance interior environment.

FF. A system for managing access to at least one of a plurality of data communication ports of a computing system residing in an interior environment, wherein the system comprises:

a plurality of data port blocking apparatuses for blocking access to a plurality of data communication ports, wherein each data port blocking apparatus is the data port blocking apparatus of any one of paragraphs A to K; and

a plurality of key devices, wherein each key device is the key device of any one of paragraphs L to U.

GG. The system of paragraph FF, wherein one or more codes to be used in computations to authenticate one of the plurality of data port blocking apparatuses is stored at said key device.

HH. The system of paragraph FF or 33, wherein a code to be used in computations to authenticate one of the plurality of key devices is stored at one of the plurality of data port blocking apparatus.

II. The system of any one of the preceding paragraphs FF, GG or HH, wherein an existing code in one of the plurality of data port blocking apparatuses to be used in computations to authenticate one of the plurality of key device is replaced by a new code.

JJ. The system of any one of the preceding paragraphs 31 to 35, wherein the interior environment is selected from the group consisting of: a residential interior environment, a commercial interior environment, a healthcare provider interior environment, a defence agency

interior environment, a transport means interior environment, a power plant interior environment, a computer peripheral machine interior environment, a household appliance interior environment.

KK. A graphical user interface (702) for provisioning a key device (300) and/or a data port blocking apparatus (100), wherein the key device (300) is configured for exchanging digital content with the data port blocking apparatus (100) in a system (1000) for managing access of at least one of a plurality of data communication ports (202) in a computing system (200) comprised in an interior environment, , wherein the graphical interface is on a computing device including a touch screen display/display with an input device, the graphical user device comprising:

a first area of the display, a section for providing user input to a key device;

a second area of the display different from the first area, a section for providing user input to a data port blocking apparatus;

wherein the first area and the second area display one or more fields for receiving user input to provide data for authentication to allow data communication between a key device and a data port blocking apparatus;

wherein in response to a force applied to a trigger comprised in the key device upon interfacing the key device with the data port blocking apparatus, a request for the data for authentication to be updated at the key device and/or the data port blocking apparatus is sent to the key device and/or the data port blocking apparatus; wherein the data port blocking apparatus comprises a first part of an electrical circuit and the key device comprises a second part of the electrical circuit, and the electrical circuit is a complete circuit when the key device is interfaced with the apparatus; wherein the trigger is associated with a request for executing a user-specified task.

LL. A method for managing access to at least one of a plurality of data communication ports (202) of a computing system (200) residing in an interior environment, wherein the method (100) comprises the steps of:

blocking access to a data communication port (202) using an insertable plug (210) of a data port blocking apparatus (100);

controlling an electromechanical locking mechanism (104) of the data port blocking apparatus to releasably engage the plug with an interior (204) of the data communication port;

interfacing a key device (300) with the data port blocking apparatus;

when the key device is interfaced with the data port blocking apparatus, performing an authentication process (800) to authenticate the key device; and

after successful authentication of the key device, allowing data communication between the apparatus and the key device to take place to manage access to the data communication port,

wherein the data port blocking apparatus comprises a first part (616) of an electrical circuit (618) and the key device comprises a second part (614) of the electrical circuit, and the electrical circuit is formed completely when the key device is interfaced with the apparatus.

MM. A method (704, 706) for provisioning a data port blocking apparatus or a key device, the method comprising the steps of:

(a) sending a secret code to one of the data port blocking apparatus and a key device;

(b) provisioning the one of the data port blocking apparatus and the key device based on the secret code;

wherein the data port blocking apparatus comprises a first part (616) of an electrical circuit (618) and the key device comprises a second part (614) of the electrical circuit, and the electrical circuit is a complete circuit when the key device is interfaced with the apparatus.

NN. The method according to paragraph MM, wherein the method (706) further comprises:

(c) performing an authentication process for the data port blocking apparatus and the key device prior to step (b).

OO. The method according to paragraph MM or paragraph NN, further comprising repeating step (b).

PP. A system for managing access of a plurality of data communication ports , the system comprising:

a web application capable of being compiled to run on a computing system for receiving a user input request for provisioning or re-provisioning of a data port blocking apparatus and a key device, wherein the computing system is in communication with a content server configured to store the received user input request;

a server different from the content server in communication with the web application through a network, wherein the server comprises a processor configured to, based on computer-executable instructions stored in a memory to execute a method according to paragraphs MM to OO.

The dimensions and values disclosed herein are not to be understood as being strictly limited to the exact numerical values recited. Instead, unless otherwise specified, each such dimension is intended to mean both the recited value and a functionally equivalent range surrounding that value. For example, a dimension disclosed as “40 mm” is intended to mean “about 40 mm”.

In the specification and claims, unless the context clearly indicates otherwise, the term “comprising” has the non-exclusive meaning of the word, in the sense of “including at least” rather than the exclusive meaning in the sense of “consisting only of”. The same applies with corresponding grammatical changes to other forms of the word such as “comprise”, “comprises” and so on.

While the invention has been described in the present disclosure in connection with a number of embodiments and implementations, the invention is not so limited but covers various obvious modifications and equivalent arrangements, which fall within the purview of the appended claims. Although features of the invention are expressed in certain combinations among the claims, it is contemplated that these features can be arranged in any combination and order.

Claims

1. A data port blocking apparatus (100) in a system (1000) for managing access to at least one of a plurality of data communication ports of a computing system (200) residing in an interior environment, wherein the apparatus (100) comprises:

a plug (210) insertable into a data communication port (202) to block access to the data communication port (202);

an electromechanical locking mechanism (104) configured to releasably engage the plug (210) with an interior of the data communication port;

an Input/Output (I/O) interface (106) for interfacing the apparatus (100) with a key device (300);

a memory (108) for storing data for authenticating the key device (300); and

a processor (112) coupled to the I/O interface (106),

wherein the processor (112) is configured to, when the key device (300) is interfaced with the apparatus (100), execute instructions in the memory (108) to operate the apparatus (100) to:

perform an authentication process to authenticate the key device (300); and

after successful authentication of the key device (300), allow data communication between the apparatus (100) and the key device (300) to take place to manage access to the data communication port (202),

wherein the apparatus (100) comprises a first part (616) of an electrical circuit (618) and the key device (300) comprises a second part (614) of the electrical circuit (618), and the electrical circuit (618) is a complete circuit when the key device (300) is interfaced with the apparatus (100) via the I/O interface (106).

2. The apparatus (100) of claim 1, wherein the first part (616) of the electrical circuit (618) is configured for supplying power to the apparatus (100) upon forming the complete circuit so as to enable data communication between the apparatus (100) and the key device (300).

3. The apparatus (100) of claim 2, wherein the apparatus (100) is configured to draw power from a power source residing at the key device (300) upon interfacing with the key device (300).

4. The apparatus (100) of any one of the preceding claims, wherein the apparatus (100) is operable to: after successful authentication of the key device (300), engage in data communication directed to controlling of the electromechanical locking mechanism (104) to engage or disengage the plug (210) from the data communication port (202).

5. The apparatus (100) of any one of the preceding claims, wherein the processor (108) is operable to: during the authentication process,
send an access request object (10) to the key device (300);
obtain an authentication response object (20) from the key device (300);
analyze the authentication response object (20) so as to generate a comparison result; and
determine whether the key device (300) is successfully authenticated based on the comparison result.

6. The apparatus (100) of claim 5, wherein the data for authenticating the key device (300) stored in the memory comprises a code, and said code is stored at both the key device (300) and the apparatus (100) to be used in computations to authenticate the key device (300), wherein the memory (108) only stores said code for use in computations to authenticate the key device (300).

7. The apparatus (100) of claim 6, wherein the access request object (10) comprises an authentication challenge object (808) comprising randomly generated data, a challenge response object (806) is generated at the key device (300) using the randomly generated data and the code stored at the key device (300), and the authentication response object (20) comprises the challenge response object.

8. The apparatus (100) of claim 7, wherein the apparatus is operable to generate a second challenge response object using the randomly generated data and the code for comparison with the challenge response object generated by the key device to determine whether the two challenge response objects match, wherein a match indicates that the key device is successfully authenticated.

9. The apparatus (100) according to any one of the preceding claims, wherein the electromechanical locking mechanism (104) comprises:
one or more locking members (206) configured to:
extend and engage one or more openings in the interior of the data communication port to prevent removal of the plug from the data communication port,
wherein the processor (112) is operable to:
control the electromechanical locking mechanism (104) to extend or retract the one or more locking members (206).

10. The apparatus (100) of any one of the preceding claims, wherein the plug (210) comprises a flexible seal for wrapping at least a part of the electromechanical locking mechanism (104), wherein said part is moveable to expand the seal to abut and engage the interior of the data communication port.

11. The apparatus (100) of any one of the preceding claims, wherein the I/O interface is a magnetic connector for interfacing with a corresponding magnetic connector at the key device.

12. A key device (300) in a system for managing access to at least one of a plurality of data communication ports of a computing system residing in an interior environment, wherein the key device (300) comprises:
an Input/Output (I/O) interface (304) for interfacing the key device (300) with a data port blocking apparatus,
a data communication interface (302) for interfacing the key device (300) with a computing device for configuring the key device (300) or configuring the data port blocking apparatus (100) via the key device;
a memory (306) storing data for authenticating the key device (300); and
a processor (310) coupled to the I/O interface (304),
wherein the processor (310) is configured to, when the key device (300) is interfaced with the data port blocking apparatus (100), execute instructions in the memory (306) to operate the key device (300) to:
proceed with an authentication process to authenticate the blocker (100); and

after successful authentication of the blocker (100), allow data communication between the data port blocking apparatus (100) and the key device (300) to take place to manage access to the data communication port,

wherein the data port blocking apparatus (100) comprises a first part of an electrical circuit and the key device comprises a second part of the electrical circuit, and the electrical circuit is a complete circuit upon interfacing of the key device (300) with the data port blocking apparatus (100) via the I/O interface (304).

13. The key device (300) of claim 12, wherein the key device (300) comprises a power source and the data port blocking apparatus (100) draws power from the power source when the key device (300) is interfaced with the data port blocking apparatus (100).

14. The key device (300) of claims 12 or 13, wherein the data port blocking apparatus (100) comprises a plug (210) insertable into a data communication port to block access to the data communication port; and an electromechanical locking mechanism (104) configured to releasably engage the plug with an interior of the data communication port; wherein the key device (300) is operable to: after successful authentication of the key device (300), engage in data communication directed to controlling of the electromechanical locking mechanism (104) to engage or disengage the plug (210) from the data communication port.

15. The key device (300) of any one of claims 12 to 14, wherein the key device (300) is operable to: during the authentication process,

receive an access request object (10) from the data port blocking apparatus (100); and compute and send an authentication response object (20) to the data port blocking apparatus (100).

16. The key device (300) of claim 15, wherein the data for authenticating the key device (300) stored in the memory (306) comprises a code, and said code is stored at both the key device (300) and the apparatus (100) to be used in computations to authenticate the key device.

17. The key device (300) of claim 16, wherein the access request object (10) comprises an authentication challenge object comprising randomly generated data, wherein the processor (310) is operable to generate a challenge response object using the randomly generated data and the code stored at the key device (300), and the authentication response object (20) comprises the challenge response object.

18. The key device (300) of claim 17, wherein the processor (310) is further configured to receive a second challenge response object from the data port blocking apparatus (100) for comparison with the challenge response object generated by the processor (310) to determine whether the two challenge response objects match, wherein a match indicates that the key device (300) is successfully authenticated.

19. The key device (300) of any one of claims 12 to 18, wherein the I/O interface (304) is a magnetic connector for interfacing with a corresponding magnetic connector at the data port blocking apparatus.

20. The key device (300) of any one of claims 12 to 19, wherein the key device (300) comprises at least one trigger (312, 314) operable to:
after the key device (300) has interfaced with the data port blocking apparatus (100) and the key device (300) has successfully authenticated, activate the data port blocking apparatus (100) to control the electromechanical locking mechanism (104) to engage the plug (210) with the interior of the data communication port or disengage the plug (210) from the interior of the data communication port.

21. The key device (300) of any one of claims 12 to 20, wherein more than one codes is stored in the memory (306), wherein each of the more than one codes is also stored in a memory (108) of a data port blocking apparatus (100) that is configured to store one code, wherein the code is to be used in computations to authenticate the key device (300) and upon successful authentication, allow the key device (300) to commence data communication with the data port blocking apparatus (100) with the stored code to manage access to a data communication port blocked or to be blocked by the data port blocking apparatus (100).

22. A method (930) for managing access to at least one of a plurality of data communication ports of a computing system residing in an interior environment, wherein the method comprises the steps of:

(a) receiving (932) an activation signal for activating a key device; wherein the key device is configured for interfacing with a data port blocking apparatus, wherein the data port blocking apparatus comprises a first part of an electrical circuit and the key device comprises a second part of the electrical circuit, and the electrical circuit is a complete circuit upon interfacing the key device with the apparatus; wherein the trigger is associated with a request for executing a user-specified task;

(b) when the key device is interfaced with the data port blocking apparatus, performing (934) an authentication process to authenticate the apparatus; and

(c) after successful authentication of the apparatus, allowing (936) data communication between the apparatus and the key device to take place to execute the user-specified task to manage access to the data communication port.

23. A method (950) for managing access to at least one of a plurality of data communication ports of a computing system residing in an interior environment in a system comprising a key device and a data port blocking apparatus, wherein the method comprises the steps of:

(a) receiving (952) an activation signal for activating the apparatus upon triggering of a key device; wherein the data port blocking apparatus comprises a first part of an electrical circuit and the key device comprises a trigger for triggering the apparatus and a second part of the electrical circuit, and the electrical circuit is a complete circuit upon interfacing the key device with the apparatus; wherein the trigger is associated with a request for executing a user-specified task;

(b) when the key device is interfaced with the data port blocking apparatus and in response to (i) a force applied to a unlock/lock trigger on the key device or (ii) a user input from a computing device, performing (954) an authentication process to authenticate the key device; and

(c) after successful authentication of the key device, allowing (956) data communication between the apparatus and the key device to take place to execute the user-specified task to manage access to the data communication port.

24. The method of claim 22 or claim 23, wherein the user-specified task is selected from the group consisting of:

(i) a request for blocking access to a data communication port using an insertable plug of a data port blocking apparatus;

(ii) a request for controlling an electromechanical locking mechanism of the data port blocking apparatus to releasably engage the plug with an interior of the data communication port.

(iii) a request for provisioning the key device or the apparatus;

(iv) a request for re-provisioning the key device or the apparatus.

25. The method of claim 23 or claim 24 dependent on claim 23, wherein the authentication process comprises:

sending an access request object to the key device;

obtaining an authentication response object from the key device;

analyzing the authentication response object so as to generate a comparison result; and determining whether the key device is successfully authenticated based on the comparison result.

26. The method of claim 22 or claim 24 dependent on claim 22 wherein the authentication process comprises:

receiving an access request object from the apparatus;

sending an authentication response object to the apparatus for generating a comparison result.

27. The method of claim 25 or claim 26, wherein the access request object comprises an authentication challenge object (808) comprising randomly generated data, wherein the authentication response object comprises a challenge response object (806) generated at the key device using the randomly generated and the code stored at the key device, preferably the access request object consists of the authentication challenge object (808).

28. The method of any one of claims 22 to 27, wherein the method comprises:

interfacing the key device with a computing device so that data communication is able to take place between the key device and the computing device; and

storing one or more codes in a memory of the key device supplied by a user through a graphical user interface displayed on a display of the computing device, wherein each of the one or more codes is also stored in a memory of a data port blocking apparatus that is configured to store one code, wherein each of the one or more codes is to be used in computations to authenticate the key device and upon successful authentication, allow the key device to commence data communication with the data port blocking apparatus with the stored code to manage access to a data communication port blocked or to be blocked by the data port blocking apparatus.

29. The method of any one of claims 22 to 27, wherein the method comprises:

interfacing the key device with a computing device and with a data port blocking apparatus so that data communication is able to take place between the key device and the computing device and between the key device and the computing device;

sending a request from the computing device to the key device to store a code in a memory of the data port blocking apparatus, wherein the code is supplied by a user through a graphical user interface displayed on a display of the computing device, wherein the code is to be used in computations for authentication of a key device; and

sending the code from the key device to the data port blocking apparatus for storing in the memory of the data port blocking apparatus.

30. The method of any one of claims 22 to 27, wherein the method comprises:

interfacing the key device with a computing device and with a data port blocking apparatus so that data communication is able to take place between the key device and the computing device and between the key device and the computing device;

sending a request from the computing device to the key device to replace an existing code stored in the memory of the data port blocking apparatus with a new code, wherein the request comprises the new code and the existing code, which are supplied by a user through a graphical user interface displayed on a display of the computing device;

sending an instruction from the key device to the data port blocking apparatus to power on the data port blocking apparatus;

in response to the instruction, performing the authentication process at the data port blocking apparatus to authenticate the key device, wherein the existing code is used in computations at the key device and the data port bocking apparatus to authenticate the key device; and

upon successful authentication, sending the new code from the key device to the data port blocking apparatus for replacing the existing code.

31. The method of any one of claims 22 to 30, wherein the interior environment is selected from the group consisting of: a residential interior environment, a commercial interior environment, a healthcare provider interior environment, a defence agency interior environment, a transport means interior environment, a power plant interior environment, a computer peripheral machine interior environment, a household appliance interior environment.

32. A system for managing access to at least one of a plurality of data communication ports of a computing system residing in an interior environment, wherein the system comprises:
a plurality of data port blocking apparatuses for blocking access to a plurality of data communication ports, wherein each data port blocking apparatus is the data port blocking apparatus of any one of claims 1 to 11; and
a plurality of key devices, wherein each key device is the key device of any one of claims 12 to 21.

33. The system of claim 32, wherein one or more codes to be used in computations to authenticate one of the plurality of data port blocking apparatuses is stored at said key device.

34. The system of claim 32 or 33, wherein a code to be used in computations to authenticate one of the plurality of key devices is stored at one of the plurality of data port blocking apparatus.

35. The system of any one of the preceding claims 32, 33 or 34, wherein an existing code in one of the plurality of data port blocking apparatuses to be used in computations to authenticate one of the plurality of key device is replaced by a new code.

36. The system of any one of the preceding claims 31 to 35, wherein the interior environment is selected from the group consisting of: a residential interior environment, a commercial interior environment, a healthcare provider interior environment, a defence agency interior environment, a transport means interior environment, a power plant interior environment, a computer peripheral machine interior environment, a household appliance interior environment.

37. A graphical user interface (702) for provisioning a key device (300) and/or a data port blocking apparatus (100), wherein the key device (300) is configured for exchanging digital content with the data port blocking apparatus (100) in a system (1000) for managing access of at least one of a plurality of data communication ports (202) in a computing

system (200) comprised in an interior environment, , wherein the graphical interface is on a computing device including a touch screen display/display with an input device, the graphical user device comprising:

a first area of the display, a section for providing user input to a key device;

a second area of the display different from the first area, a section for providing user input to a data port blocking apparatus;

wherein the first area and the second area display one or more fields for receiving user input to provide data for authentication to allow data communication between a key device and a data port blocking apparatus;

wherein in response to a force applied to a trigger comprised in the key device upon interfacing the key device with the data port blocking apparatus, a request for the data for authentication to be updated at the key device and/or the data port blocking apparatus is sent to the key device and/or the data port blocking apparatus; wherein the data port blocking apparatus comprises a first part of an electrical circuit and the key device comprises a second part of the electrical circuit, and the electrical circuit is a complete circuit when the key device is interfaced with the apparatus; wherein the trigger is associated with a request for executing a user-specified task.

38. A method for managing access to at least one of a plurality of data communication ports (202) of a computing system (200) residing in an interior environment, wherein the method (100) comprises the steps of:

blocking access to a data communication port (202) using an insertable plug (210) of a data port blocking apparatus (100);

controlling an electromechanical locking mechanism (104) of the data port blocking apparatus to releasably engage the plug with an interior (204) of the data communication port;

interfacing a key device (300) with the data port blocking apparatus;

when the key device is interfaced with the data port blocking apparatus, performing an authentication process (800) to authenticate the key device; and

after successful authentication of the key device, allowing data communication between the apparatus and the key device to take place to manage access to the data communication port,

wherein the data port blocking apparatus comprises a first part (616) of an electrical circuit (618) and the key device comprises a second part (614) of the electrical circuit, and the electrical circuit is a complete circuit when the key device is interfaced with the apparatus.

39. A method (704, 706) for provisioning a data port blocking apparatus or a key device, the method comprising the steps of:

(a) sending a secret code to one of the data port blocking apparatus and a key device;

(b) provisioning the one of the data port blocking apparatus and the key device based on the secret code;

wherein the data port blocking apparatus comprises a first part (616) of an electrical circuit (618) and the key device comprises a second part (614) of the electrical circuit, and the electrical circuit is a complete circuit when the key device is interfaced with the apparatus.

40. The method according to claim 39, wherein the method (706) further comprises:

(c) performing an authentication process for the data port blocking apparatus and the key device prior to step (b).

41. The method according to claim 39 or claim 40, further comprising repeating step (b).

42. A system for managing access of a plurality of data communication ports , the system comprising:

a web application capable of being compiled to run on a computing system for receiving a user input request for provisioning or re-provisioning of a data port blocking apparatus and a key device, wherein the computing system is in communication with a content server configured to store the received user input request;

a server different from the content server in communication with the web application through a network, wherein the server comprises a processor configured to, based on computer-executable instructions stored in a memory to execute a method according to claims 39 to 41.