February 19, 2016

APPLICATION PRIVACY LOCK

Nimit Patel

APPLICATION PRIVACY LOCK

ABSTRACT

An application lock system can lock user interface elements associated with an application upon receiving a request from a user or when the system determines a session as being inactive at least for a predetermined time period. Applications may include, for example, a web browser application, a word processor application, or a file manager application. When the system receives a request to lock the session at the application, the system locks one or more user interface elements associated with the session. If a request is not received, then the system determines whether the session at the application is inactive for the predetermined time period. If the session at the application is inactive for the predetermined time period, the system locks one or more user interface elements associated with the session. If the session at the application is inactive for less than the predetermined time period, the system keeps the one or more user interface elements associated with the session unlocked.

PROBLEM STATEMENT

A user locks an electronic device for security and privacy reasons. An electronic device may include, for example, a computer, a laptop, a tablet, and a mobile device. The device may lock based on, for example, a timeout (e.g., display sleep mode, screen saver lock), a physical or soft key sequence (e.g., a lock button or OFF button), or a pointer sequence or location (e.g., “hot corner”). After a device is locked, the user unlocks the electronic device using a password or a pin or biometric data entry (e.g., a fingerprint). If the user does not lock the device, it may be easy for another person to use the user’s signed in credentials on an application to perform unauthorized activities using that user’s account. In an example, a user may have signed into a web-browser application, for example, a chrome browser application, and another person might access that web-browser application while the user is away from the unlocked device.

In other scenarios, multiple users share a common electronic device, which raises other types of security and privacy risks. The current user of the electronic device may sign out of his account associated with an application and then allow other, subsequent users to access the unlocked device. Such possible solutions are cumbersome and inconvenient for users as they pass the device from one person to another. One possible alternative may be to auto-sign out of the electronic device after a set period of time. A disadvantage with this alternative is that it does not secure non-signed in application sessions. Therefore, there exist opportunities to provide users with secure and private application sessions at electronic devices without locking the entire electronic device.

DETAILED DESCRIPTION

The systems and techniques described in this disclosure relate to an application lock system that locks one or more user interface elements associated with an application running on an electronic device. The system can be implemented for use in an Internet, an intranet, or another client and server environment. The system can be implemented locally on a client electronic device device or implemented across a client device and server environment. The client device can be any electronic device such as a mobile device, a smartphone, a tablet, a handheld electronic device, etc.

Fig. 1 illustrates an example method 100 to lock one or more user interface elements associated with a software application upon receiving a request from a user or when a session is inactive for a predetermined time period. The method 100 may be performed by an application lock system of an electronic device or an administrator device. Alternatively or additionally, the method 100 can be implemented at an electronic device as a policy enforced by an administrator device.

The system determines 102 whether the system receives a request to lock a session at the application. The system may receive the request from a user to immediately lock the session at the application using an user interface element like a drop-down menu. On receiving the request, the system locks 106 the user interface elements associated with the session without traversing a timeout block 104, as shown in Fig. 1. In an example, the user or an administrator may lock or wipe the session using a remote, networked electronic device. In another example, if a mobile device owned by the user gets lost, the user may lock the session opened at the mobile device using a security application opened at another user device, such as a tablet, that has network connectivity to the session. The security application may be a security dashboard accessible through any of the user's electronic devices.

If the system does not receive the request, the system determines 104 whether the session at the application is inactive for a predetermined time period. The predetermined time period may be set automatically by the system or may be manually set by the user. For example, the predetermined time period may be 60 seconds, 5 minutes, 15 minutes, etc. The user may be signed into the application using user credentials like username and password associated with a user profile maintained at the application. Alternatively, some applications do not require or request a sign-in.

In implementing step 104, the system may detect activity at the application during the session and start a counter when the session becomes inactive. The system may increment the counter until the predetermined time period lapses and reset the counter if the system detects any session activity.

If the predetermined time period lapses and no activity is detected at the application, the system locks 106 one or more user interface elements associated with the session. The user interface elements include windows, tabs, dialog boxes, etc. The system locks the user interface elements such that the user cannot access the elements unless the user provides the system with user credentials, such as password, biometric identification, etc. Alternatively or additionally, the system may unlock the session when the system detects that the user is near the electronic device. For example, the system may have a proximity sensor at the electronic device to detect when the user is near the electronic device.

However, the user can still use the device's hardware elements and other applications' user interface elements even when the particular application session has been locked. Further, if the session becomes active before the predetermined time period lapses, the system again determines 102 if a request is received to lock the session at the application and executes further steps as per the method 100.

Fig. 2 illustrates example Graphical User Interface (GUI) of a web browser application 202. The user has two application tabs 204 and 206 opened at a user's mobile device. The system determines if a request is received at the system for locking the session that is active at the application. If no request is received, the system initiates monitoring the activity of the session. After the system detects that the session is inactive, the system initiates a counter. The system determines whether the session at the application 202 is inactive for a predetermined time period. The system locks the tabs 204 and 206 if the session remains inactive for the predetermined time period. Further, when the user tries to access the locked tabs, the system interrupts with a pop-up password box 208, as shown in Fig. 2. The system receives a password from the user and provides access to the user if the system receives the correct password. Other sessions of the same application and other applications may remain unlocked while the particular session is locked. Because the locking is performed on a session basis, the system supports finer-grained security and privacy controls.

The subject matter described herein can be implemented in software and/or hardware (for example, computers, circuits, or processors). The subject matter can be implemented on a single device or across multiple devices (for example, a client device and a server device). Devices implementing the subject matter can be connected through a wired and/or wireless network. Such devices can receive inputs from a user (for example, from a mouse, keyboard, or touchscreen) and produce an output to a user (for example, through a display and/or a speaker). Specific examples disclosed are provided for illustrative purposes and do not limit the scope of the disclosure.

DRAWINGS

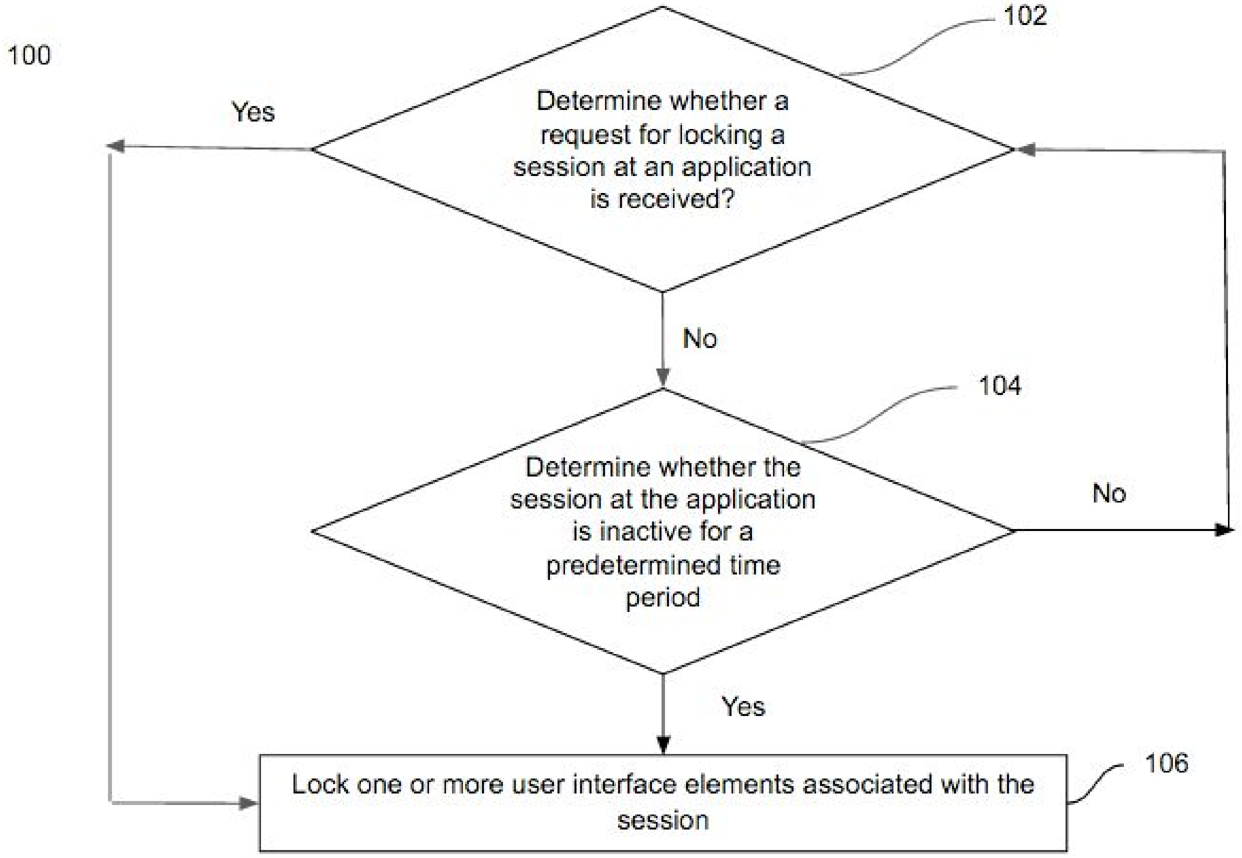
100
102
Determine whether a request for locking a session at an application is received?
Yes
No
104
Determine whether the session at the application is inactive for a predetermined time period
No
Yes
Lock one or more user interface elements associated with the session
106

Fig. 1

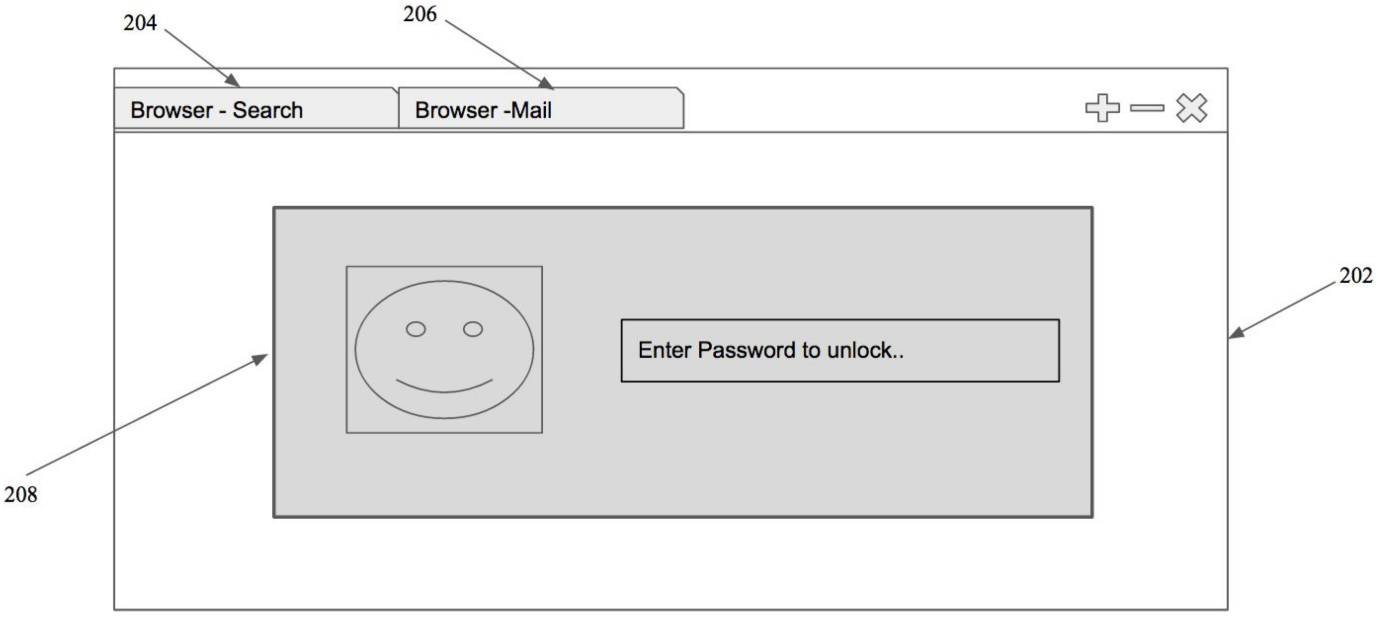
204
206
Browser - Search
Browser -Mail
202
Enter Password to unlock..
208

Fig. 2