PROTECTION OF DOWNLOADABLE SOFTWARE ON SDR DEVICES

(6.4 SW Download / 6.3 System Security & Authentication)

Eimear Gallery (Information Security Group, Royal Holloway, University of London: Egham, Surrey TW20 0EX, UK; e.m.gallery@rhul.ac.uk; Tel.: +44 (1784) 41 4345); and
Allan Tomlinson (Information Security Group, Royal Holloway, University of London: allan.tomlinson@rhul.ac.uk).

Introduction

It is envisaged that mobile communications systems that will emerge after the current 3G devices will be sufficiently advanced to make use of SDR techniques. These techniques will be used to reconfigure the air interface, providing the consumer with greater flexibility in the choice of access networks. While the concept of a reconfigurable air interface holds considerable promise, there are several security issues to consider.

Regulatory bodies will expect the device to conform to appropriate standards and safety regulations. Type approval will be required, and the network operators may require the device to conform to their requirements before network access is permitted. It is therefore important that any downloaded code is protected from malicious activities. It is also important that the host, where the code will execute, provides a stable platform to support the downloaded application.

The security of downloaded code has been considered by the SDR forum, who have produced a set of requirements for software downloads [ref], and described the security considerations for SDR [ref]. This paper addresses some of the security issues identified by the SDR, and describes two protocols that may be used to meet several of the requirements described.

Security requirements

Regarding radio software download, security issues arise in relation to both the SDR terminal host on which the software will execute and also in relation to the RF reconfiguration software itself. Both the host and indeed the application need to be protected.

Our previous work [citation], presented to the SDR forum, focused on protecting the SDR terminal from malicious applications through the deployment of a policy-based authorization framework for implementation within the mobile environment, with the objective of providing both mobile devices with the ability to assign appropriate privileges to software, according to both where it originates from and the attributes it possesses.

This paper, however, aims to fulfil requirements associated with application as defined in [citation]. These application-oriented requirements are all associated with the download process.

1. The download process shall employ means, such as encryption, of protecting proprietary radio software and data during download to prevent unauthorized parties from gaining access to or altering this proprietary data or software. It is important to ensure that when this proprietary intellectual property is being downloaded to the SDR device, it is adequately protected from unauthorized access.

2. The software shall be downloaded to a buffer area in the SDR device and verified for integrity and authenticity. Upon successfully receiving the radio software download, the SDR device shall transmit an Acknowledge Safe Receipt message to the download server, which shall then terminate the download procedure. Upon successfully receiving the software download, the SDR device shall also release the network connections/resources that were utilized for the download. If for some reason the radio software download fails or is interrupted before completion, the download server shall have the ability to mark the exact point of interruption and complete the download at a later time instead of starting over from the beginning.

3. The download process shall employ an effective authorization procedure to verify that the entity requesting the radio software download has the authority to receive, install, and utilize the radio software download.

4. A capability exchange shall take place between the network and the SDR device prior to radio software download to enable the network to select appropriate software entities and parameters sets for the SDR device. If the network server finds no matching software entities and parameter sets, the download process shall be terminated with a failure message back to the SDR device. The network shall also inform the SDR device of its own capabilities, including supported modes of operation and available services. The capability response from the SDR device may include: current SDR device configuration; type approval data; API revisions supported; resident hardware resources; resident software profile; resident compilers and operating systems; resident licenses; memory capability; processing power; display capability; and user interface information.

5. Finally, with respect to the final installation of the downloaded radio software onto the SDR device the following requirements must be fulfilled: it must be determined whether the configuration of the SDR device remains acceptable for correct operation of the downloaded radio software. If a mismatch occurs, the installation process shall be terminated with feedback to an appropriate network entity.

While many envisage the SDR environment to become open, with software widely and freely available, in the short to medium term, it is envisaged that IPR associated with radio software will remain a protected commodity. In our proposed architecture and associated protocols for download, both the confidentiality and integrity of the RF

reconfiguration software will be protected while the software is in transit from the software provider to the SDR terminal, as well as on the terminal itself, while in storage and when executing.

While SDR will open the system up, it will still have to remain under some degree of control. In order to comply with user safety controls and regulator guidelines, it may be in the interest of the software or network provider to validate that the device has a specific set of OS controls in place before allowing the release and execution of SDR software to prevent damage occurring to either the user or the network.

It is also important to ensure that the capabilities reported by the SDR device are indeed accurate such that the most suitable SDR software can be transmitted to the terminal for execution. Mechanisms fulfilling these requirements will help to thwart against any legal issues arising with respect to user devices radiating dangerous emissions. In conjunction with this, have there are commercial benefits associated with protecting consumers in this way.

Closely associated with requirements 3 and 4, it is of little help if properties about the terminal sent to the software provider may be changed directly afterwards such that software, which may work as intended with the presumed environment on the terminal, causes the terminal to function incorrectly or maliciously due to the new environment actually present. These requirements associated with the integrity verification of the SDR terminal are met by our proposed protocols.

Finally in relation to the download of malicious code, which has emerged as one on the most prominent security issues surrounding SDR, where in the long term scenario where a device may be reconfigured in an ad hoc manner and RF reconfiguration software made freely available, there can be no guarantees that when a device leaves a somewhat controlled network, malicious code will not be downloaded either accidentally or maliciously.

In order to prevent malicious damage that may be caused by rogue terminals, which request to rejoin a commercial network controlled by operators, it is in the operators interest to ensure that rogue terminals can be detected prior to their access to the required SDR software and entry back onto the network. In order to achieve this, it must be ensured that only uncompromised terminals are permitted access to the required SDR software.

Protocols

The Trusted Computing Group[1] is an industrial forum who is defining standards for trusted platform technology. This forum has a Mobile Phone working group who are developing trusted computing standards specifically for mobile devices. It is therefore reasonable to expect that future mobile devices will be able to make use of trusted computing technology, and the protocols we describe leverage this technology to provide solutions to the security requirements identified above.

The protocols described in the paper apply the concepts of trusted computing to protect the downloaded application during transport to, and execution on, the mobile receiver.

[1] www.trustedcomputinggroup.org

The protocols use nonce based challenge-response mechanisms together with trusted computing technology to provide assurance that the mobile receiver is configured in a manner which is acceptable to the software provider. Shared keys are also established during this stage, which are used to protect the integrity and confidentiality of the downloaded application. Once the application has been downloaded, the protocols ensure that the receiver will not be able to access the downloaded application if it is configured in any other way.

Similar solutions have been described to provide security for broadcast video reception by mobile devices. [Reference to RHUL-MA-2005-8]. This paper takes these concepts and adapts them specifically to provide the above solutions for SDR

Approvals

The authors agree that this paper may be reviewed by the SDR Forum paper evaluation committee for purposes of inclusion in the 2005 Software Defined Radio Technical Conference and Product Exposition.

The authors agree to allow inclusion of this paper in SDR Forum conference publications, and other SDR Forum publications, if the paper is accepted