EURASIP Journal on Wireless Communications and Networking 2005:3, 401–412

Towards a Fraud-Prevention Framework for Software Defined Radio Mobile Devices

Alessandro Brawerman
School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30318, USA
Email: ale@ece.gatech.edu

John A. Copeland
School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30318, USA
Email: copeland@ece.gatech.edu

Received 29 September 2004; Revised 8 March 2005

The superior reconfigurability of software defined radio mobile devices has made it the most promising technology on the wireless network and in the communication industry. Despite several advantages, there are still a lot to discuss regarding security, for instance, the radio configuration data download, storage and installation, user's privacy, and cloning. The objective of this paper is to present a fraud-prevention framework for software defined radio mobile devices that enhances overall security through the use of new pieces of hardware, modules, and protocols. The framework offers security monitoring against malicious attacks and viruses, protects sensitive information, creates and protects an identity for the system, employs a secure protocol for radio configuration download, and finally, establishes an anticloning scheme, which besides guaranteeing that no units can be cloned over the air, also elevates the level of difficulty to clone units if the attacker has physical access to the mobile device. Even if cloned units exist, the anticloning scheme is able to identify and deny services to those units. Preliminary experiments and proofs that analyze the correctness of the fraud-prevention framework are also presented.

Keywords and phrases: cellular frauds, cloning, security and privacy issues, security protocols, software defined radio mobile devices.

1. INTRODUCTION

Software defined radio [1] allows multiple radio standards to operate on common radio frequency hardware, thereby ensuring compatibility among legacy, current, and evolving wireless communication technologies.

A software defined radio mobile device (SDR-MD) is capable of having its operation changed by dynamically loading radio reconfiguration data (R-CFG files) over the air. With different R-CFGs, the device can operate using different wireless communication technologies while having a single transceiver. A typical SDR-MD can manage communication via satellite, over different cellular technologies, VoIP (voice over internet protocol), and operations over the internet.

One of the key issues in SDR wireless communication involves security. According to the SDR Forum [2], some of the concerns are the R-CFG download, storage, and installation; user's privacy, that is, protection of the user's identity, location, and communication with other devices; and finally, SDR-MD cloning, that is, illegally using services that are billed to someone else's device.

To address the SDR Forum concerns and greatly enhance the overall security of SDR-MDs, a fraud-prevention framework is proposed. The proposed framework offers security monitoring against malicious attacks and viruses that may affect the configuration data, protects sensitive information through the use of protected storage, creates and protects an identity for the system, employs a secure protocol for R-CFG download, and finally, establishes an anticloning scheme which guarantees that no units can be cloned over the air, and elevates the level of difficulty to clone units if the attacker has physical access to the SDR-MD. Even if cloned units exist, the anticloning scheme is able to identify and deny services to those units.

Preliminary practical experiments using java 2 micro-edition (J2ME) [3] and proofs that analyze the correctness of the fraud-prevention framework are also presented.

2. BACKGROUND

Research work has been done for each of the SDR concerns previously described; however, no published work has developed a solution that encompasses more than one of the concerns at once. This section is divided according to the SDR Forum concerns. For each subsection, some of the relevant related research is presented.

2.1. R-CFG download, storage, and installation

In [4], the authors discuss a model for securing the R-CFG download and installation that involves the use of secret device keys and signatures. All security operations take place within tamper-proof hardware that also contains the programmable components of the transceiver. This approach provides good security for the radio software that lies within the tamper-proof hardware, but leads to some drawbacks such as the use of nonstandard security methods, lack of a means for third-party vendors to provide R-CFGs, and, most important, lack of a means for securing radio software that resides outside the tamper-proof hardware.

2.2. User's privacy

Some efforts, called privacy extension to Mobile IPv6, deal with user's privacy. The basic idea of these efforts is to replace the MAC address of a mobile device with a random one, called a temporal mobile identifier (TMI) [5] or pseudorandom interface identifier (PII) [6].

In those schemes, personal mobile location privacy control relies on either the home administration, the foreign administration, or both. Moreover, the home administration is required to share some secrets with the foreign administration to prevent eavesdroppers from having any knowledge about the binding users temporal identifiers and real identifiers. These efforts cannot completely control mobile location privacy by a mobile user since the administration can associate any identifier (PII or TMI) with the corresponding real ID of the mobile device.

2.3. SDR-MD cloning

The advanced mobile phone system (AMPS) [7] is the analog mobile phone system standard introduced in the Americas during the early 1980s. Despite the fact that it was a great advance in its time, the AMPS presented several security flaws, and multiple copies of cloned mobile stations were created with little difficulty.

The global system for mobile communication (GSM) [8] is a globally accepted standard for digital cellular communication. The GSM authentication framework relies on special cryptographic codes to authenticate customers and bill them appropriately. A personalized smart card, called a SIM card, stores a secret key that is used to authenticate the customer; knowledge of the key is sufficient to make calls billed to that customer.

The SIM card is easily removable so that the user can use other cell phones. The drawback is that someone who has physical access to the SIM card can copy the information to another card, thereby cloning the authentication information of the user.

Cloning the SIM card is a relevant flaw, however a much more serious flaw was discovered. In [9] it is shown that the cryptographic codes used for authentication are not strong enough to resist attacks. To exploit this vulnerability, an individual would interact with the SIM card repeatedly to learn the secret key and would then be able to clone the phone without having to clone the SIM card. Although it was considered that the attacker had physical access to the SIM card, it was mentioned that over-the-air attacks are possible, making cloning on GSM cellphones a more serious threat.

The Universal Mobile Telecommunications System (UMTS) [10] is an open air-interface standard for third-generation wireless telecommunications. It provides higher data rates and fixes several security flaws encountered in the GSM standard. Despite several advantages that the UMTS standard provides, it also stores vital information in the SIM card. Thus, like the GSM, someone might be able to copy the authentication information from one SIM card to another.

Another drawback concerns the KASUMI block cipher, which is at the core of the integrity and confidentiality mechanisms in the UMTS network. Hardware implementations are required to use at most 10 000 gates and must achieve encryption rates in the order of 2 Mbps (maximum data rate). Thus, a considerable effort must be performed in order to implement a high-performance hardware component that carries out the operations of the KASUMI block cipher.

As a final remark, UMTS devices are not capable of reconfiguring their radio parameters via software. Thus, dual mode or tri-mode expensive cell phones are necessary to guarantee backward compatibility with other standards.

Simpler schemes that only detect cloned units and do not try to prevent cloning have also been proposed. They can be found in [11, 12].

2.4. Trusted computing group

The trusted computing group (TCG) [13] is an industry standards body comprising computer and device manufacturers, software vendors, and others with an interest in enhancing the security of the computing environment across multiple platforms and devices.

The TCG claims that it will develop and promote open industry standard specifications for trusted computing hardware building blocks and software interfaces across multiple platforms, including personal computers (PCs), servers, personal digital assistants (PDAs), and digital phones.

So far the TCG has only presented specification for the PC environment [14]. Some of the benefits include more secure local data storage, a lower risk of identity theft, and the deployment of more secure systems and solutions based on open industry standards.

Despite the fact that the TCG specification for the PC does point out and solve several security flaws, this specification would not achieve a satisfactory performance if employed by constrained SDR-MDs.

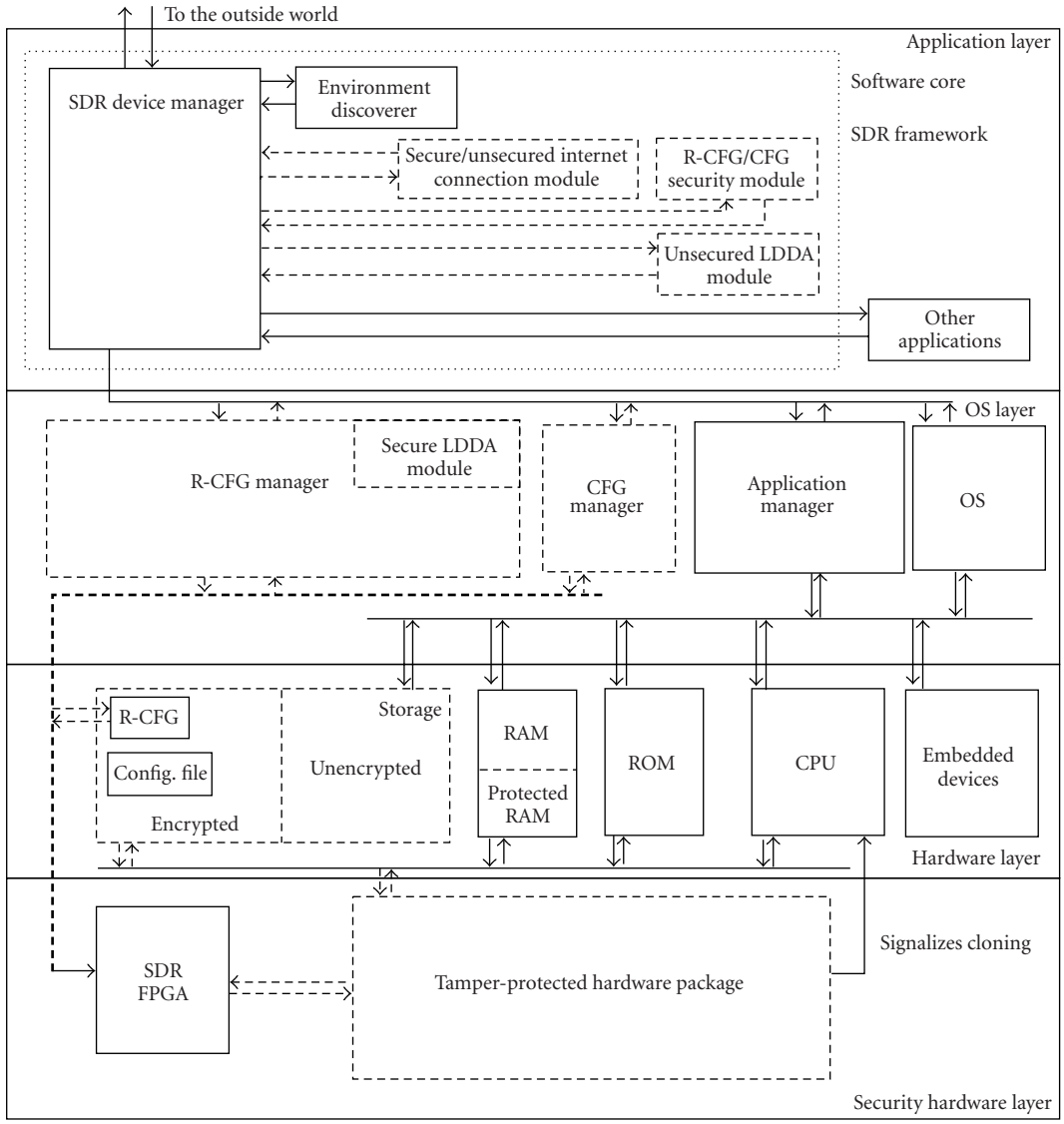

Figure 1: The preliminary design of the fraud-prevention framework.

3. THE FRAUD-PREVENTION FRAMEWORK SPECIFICATION

The fraud-prevention framework is composed of new pieces of hardware, new modules, and new protocols. Figure 1 depicts the preliminary design of the framework. The dashed squares are the main contributions of this work.

Note that the SDR device manager (SDR-DM) is responsible for managing all the communication with the outside world and for requesting the services of each module when needed. Also, the environment discoverer module is responsible for detecting which wireless communication technologies are available in the current SDR-MD's environment. This module is assumed to be present in the software core SDR framework and is outside the scope of this work.

The R-CFG manager is responsible for managing the R-CFG files currently stored in the device and the R-CFG currently installed. It also informs the SDR-DM when a different R-CFG is needed. The CFG manager is responsible for managing the configuration (CFG) file. The CFG file is provided by the wireless operator (WO) and is used to set the device's phone number. Note that both the R-CFG and CFG files are stored in an encrypted storage. Standard encryption algorithms such as RC5 [15] and RSA [16] can be used to provide the encryption storage. Other modules as well as basic definitions are discussed in separate subsections below.

3.1. Basic definitions

This section presents definitions, components, and entities that participate in the fraud-prevention framework. The nomenclature used to specify the framework is presented in Table 1.

The entities that participate in the framework as well as their responsibilities are defined in Table 2.

Table 1: Basic definitions.

C	A 48-bit random number (nonce)
$K_Y\{C\}$	C is cryptographically transformed, somehow, with a key Y
MD(Z)	Hash of Z
$[C]_{\text{Alice}}$	C is transformed using the private key of Alice
$\{C\}_{\text{Alice}}$	C is transformed using the public key of Alice
Attestation	It is used to check integrity status of a certain component. It is defined as the function Att(X), which results in the hash of component X
Attestation key pair (AK)	It is used to obtain the attestation credential. Composed by the 2048-bit attestation private key (AK_{priv}) and public key (AK_{pub})
Attestation credential (AC)	It is used to identify the SDR-MD. It is signed by the privacy credential authority (Privacy CA) and it is presented whenever the user tries to use the network services. AC = $[AK_{pub}]_{\text{Privacy CA}}$
Null AC	When the SDR-MD discovers it is a cloned unit, it sets its AC to null. Every bit in the AC is equal to 0
Endorsement key (EK)	It is used to uniquely identify the SDR-MD. It is never disclosed by the device. Its size is also 2048 bits
R-CFG	It is used to configure the radio of the SDR-MD
Valid R-CFG	An R-CFG that has been approved by the regulatory agency
Invalid R-CFG	An R-CFG that has not been approved by the regulatory agency or it has been modified after been approved by the regulatory agency
CFG	It is used to set up the phone number of the SDR-MD. It is signed by the WO. CFG = $[\text{Phone no.}]_{\text{WO}}$

Table 2: Entities and responsibilities.

Manufacturer (manuf.)	Produces the SDR-MD. Generates the R-CFGs. Generates the SDR-MD's EK and informs the Privacy CA about the EK. Calculates and stores the Att(EK) in the SDR-MD Installs the initial R-CFG and stores the Att(R-CFG) in the SDR-MD
Regulatory agency (RA)	Tests, approves, and licenses the R-CFG. Basically, the RA tests the R-CFG in the specific hardware to ensure that the device does not cause interference or function out of its defined spectrum, as defined in [17]
WO	Sells the SDR-MD. Provides communication services. Generates the CFG Authenticates the SDR-MD to use the network. Detects cloned SDR-MDs
Privacy CA	Provides the SDR-MD with an AK pair, the AC, and the WO public key
SDR-MD	Utilizes the network services. Downloads R-CFGs and CFGs files. Detects if it is a cloned or valid unit

3.2. The tamper-protected hardware package

The TPHP must be physically protected from tampering. This includes physically binding it to the other physical parts of the SDR-MD such that it cannot be easily disassembled and transferred to other devices. These mechanisms are intended to resist tampering. Tamper evidence measures are to be employed. Such measures enable detection of tampering upon physical inspection. The package must limit pin probing and EMR scanning. Similar tamper-protected hardware is the trusted platform module of [13] and the Intel wireless trusted platform processor [18].

The TPHP is composed of two tamper resistant chips (TRCs): TRC1, which is read only, and TRC2, which is read/write. The TRC1 contains the EK, the attestation engines responsible for measuring, reporting, and comparing integrity values, and a specialized hardware to generate 48-bit random numbers. The TRC2 contains the attestation engine responsible for storing integrity values and protected nonvolatile memory to store the necessary keys. Notice that the TPHP comes from the manufacturer with the RA's public key already stored.

The attestation engines are divided into the attestation measurement engine (AMEng), attestation store engine (AS Eng), attestation report engine (AR Eng), and attestation comparison engine (AC Eng). Table 3 presents the functions of each attestation engine. Figure 2 depicts the components of the TPHP as it comes from the manufacturer.

3.3. The secure SDR R-CFG download protocol

To install only valid R-CFGs, a secure SDR R-CFG download protocol is defined as part of the fraud-prevention framework. The secure protocol employs the mutual authentication and R-CFG validation and verification steps described by the R-CFG/CFG security module.

Table 3: Attestation engines and functions.

AM Eng	Measures Att(EK), Att(R-CFG), and Att(CFG), and writes the results into $R0$, $R1$ and $R2$
AS Eng	Stores the Att(EK) in register 0($R0$), the Att(R-CFG) in register 1($R1$), and the Att(CFG) in register 2($R2$)
AR Eng	Reads and reports the values of the registers
AC Eng	Compares the values of $R0$, $R1$, and $R2$, reported by the AR Eng, with the values measured by the AM Eng

Whenever a manufacturer generates a new R-CFG, it has to send the R-CFG to be approved and licensed by the RA. This is called R-CFG validation.

To perform R-CFG validation, the protocol employs a public-private key mechanism. The manufacturer sends to the RA a combination of a header, which contains manufacturer, model, serial number range, and possibly some other information; the new R-CFG; and the hardware in which the R-CFG is to be tested and used.

The RA installs the R-CFG in the specified device and tests the device's behavior. If no malfunction is observed, the RA approves the R-CFG and assigns it a license number. During the test, the RA computes $h = \text{MD}(\text{header}\|\text{R-CFG})$. The value h is then signed with the RA's private key, $[h]_{\text{RA}}$. Figure 3 depicts the signing step. The signed hash value, $[h]_{\text{RA}}$, is sent back to the manufacturer along with the assigned license number.

Once the R-CFG has been licensed, signed, and placed on a server, the SDR-MDs can contact the server at any time to download the combination of header, R-CFG, and $[h]_{\text{RA}}$.

After an SDR-MD has connected to the manufacturer's server, mutual authentication is performed. The mutual authentication step avoids masquerade and replay attacks. When using an unsecured connection, this is done by exchanging random challenges (nonces) or by certificates, while when using a secure connection, the protocol that provides the secure connection is assumed to take care of the mutual authentication.

After the mutual authentication step has been successfully completed, the SDR-MD requests and downloads the new R-CFG. Upon download completion, R-CFG verification is necessary to guarantee that the R-CFG has been approved by the RA and properly signed. The verification step also tests whether the R-CFG is appropriate for the device (Figure 4).

However, to guarantee that the R-CFG has not been modified after being approved and signed by the RA, the following steps are performed:

(1) a new hash value $h' = \text{MD}(\text{header}\|\text{ R-CFG})$ is calculated;
(2) the received $[h]_{\text{RA}}$ is decrypted to obtain h;
(3) h and h' are compared: if $h = h'$, the received R-CFG is accepted. However, if $h \neq h'$, the R-CFG is rejected.

Figure 4 also shows the data integrity check. If the new R-CFG has passed all the tests, it is then installed and the value of Att(R-CFG) is stored in $R1$.

The steps of the secure SDR R-CFG download protocol when using an unsecured connection, such as HTTP, are depicted in Figure 5. Dashed arrows indicate communication inside the SDR-MD.

Although the protocol is specified using an unsecured connection, the R-CFG is still protected since it is encrypted with the EK, thus only that specific device which has initiated the connection can correctly decrypt and install the R-CFG. Details on how to obtain a lightweight secure connection using the Light SSL (LSSL) protocol, specified in the secure/unsecured internet connection module, can be found in [19].

The SDR R-CFG download protocol initiates with the SDR-MD contacting the manufacturer's server and establishing an unsecured connection. Next, the SDR-MD sends MD(EK) and a nonce C encrypted by the EK. The manufacturer maintains a database of all available EKs (M_EKDB), indexed by MD(EK). The database has all information that the manufacturer needs about each SDR-MD it has produced.

When the manufacturer receives the MD(EK), it searches in its M_EKDB for that value. If it does not find the MD(EK), the manufacturer ends the connection. On the other hand, if MD(EK) is in M_EKDB, then the manufacturer obtains the EK of that device and generates a new nonce C'. The C' is then encrypted by the EK and sent, along with C, to the SDR-MD.

Upon receiving C and C', the SDR-MD authenticates the manufacturer if the received C is equal to the one that the SDR-MD has previously generated. If authentication fails, the SDR-MD terminates the connection; otherwise, it obtains C', sends it back to the manufacturer, and requests the necessary R-CFG.

The manufacturer then authenticates the device. If authentication fails, the manufacturer terminates the connection; otherwise, it sends the requested R-CFG encrypted by the EK. The SDR-MD receives the R-CFG, verifies it, and checks the R-CFG data integrity. If the R-CFG tests show no negative results, the SDR-MD installs the R-CFG and acknowledges the manufacturer. The connection is then released.

After releasing the connection, the SDR-MD installs the R-CFG and stores the Att(R-CFG) value in $R1$. Whenever the SDR-MD is booting up, the AM Eng calculates a new Att(R-CFG) value, which is then passed to the AC Eng to be compared with $R1$. If Att(R-CFG) $= R1$, the current radio configuration is trusted. On the other hand, if Att(R-CFG) $\neq R1$, the SDR-DM blocks the use of any service.

3.4. The anticloning scheme

One of the more dangerous threats in SDR wireless communication is cloning. SDR-MD cloning is considered a federal crime. According to [20], telecommunication fraud losses are estimated at more than a billion dollars yearly. A large amount of this loss is due to cloning. Besides illegal billing, cloned units increase the competition of shared resources, which increases network congestion and degrades network services. Furthermore, the impact of overload traffic from

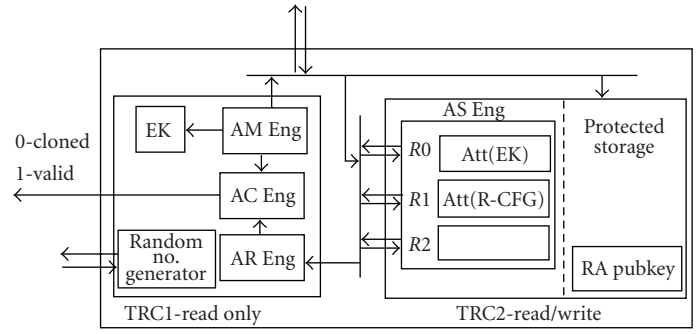

Figure 2: The tamper-protected hardware package in an invalid state.

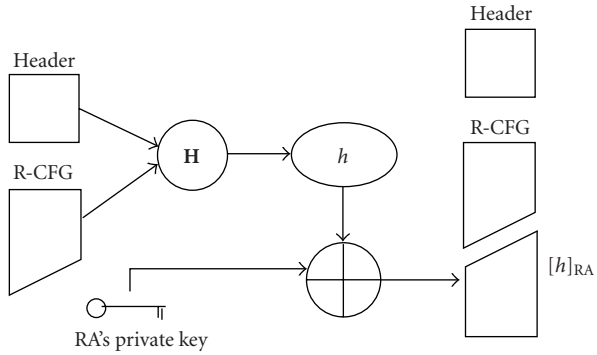

Figure 3: R-CFG validation.

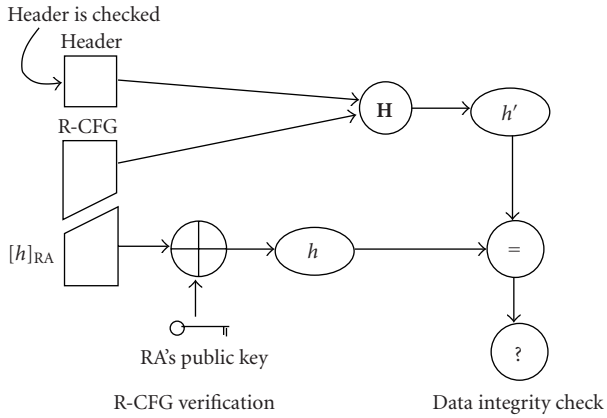

Figure 4: R-CGF verification and data integrity check.

cloned units is unpredictable. Thus, the estimation of traffic patterns is imprecise for network planning.

The anticloning scheme, which is part of the proposed fraud-prevention framework, is designed to provide a core set of hardware and software technologies that provide the basis for a wireless network environment free of cloned units.

Unlike other cloning detection schemes, the proposed anticloning scheme not only detects cloned units, but also elevates the level of difficulty to clone a valid unit. Also, as a new feature, the SDR-MD is aware of cloning, that is, an SDR-MD is able to discover if it is a cloned unit and take the necessary steps to block the use of the network services. Another advantage is that the anticloning framework is independent of technology, working well for different wireless technologies.

3.4.1. Entering a valid state

The SDR-MD comes from the manufacturer in an invalid state, that is, it does not have the AC, therefore, it cannot identify itself to the network. After obtaining the AC, the SDR-MD enters a temporary state, that is, it is able to prove its identity, however, it does not have a phone number yet, it does not have the CFG file installed. After obtaining the CFG, the SDR-MD finally reaches a valid state. It is able to identify itself and use the network services.

Figure 6 depicts the transition states that the SDR-MD has to go through in order to reach a valid state. Note that anytime after the SDR-MD has reached the valid state, it may need a new R-CFG file or a new CFG file. While obtaining any of those files, the SDR-MD goes to a temporary state. With the new data locally stored, the security checks are executed and the SDR-MD goes back to the valid state.

To obtain a valid AC, the SDR-MD has to execute the attestation credential protocol (ACP) depicted in Figure 7. The ACP is a communication process between the SDR-MD and the Privacy CA and it is executed only one time per each EK.

Whenever the manufacturer generates a new EK, it informs the Privacy CA, in a safe way, about that EK. The Privacy CA, like the manufacturer, maintains a database of all available EKs (CA_EKDB), indexed by MD(EK). This database has all information that the Privacy CA needs to know about each SDR-MD produced and links each SDR-MD to its AC.

The ACP steps are defined as follows. First, the SDR-MD contacts the Privacy CA and sends the value $R0$ = Att(EK). The Privacy CA looks for a matching MD(EK) in the CA_EKDB. If it finds a match, the Privacy CA obtains the EK of that unit and acknowledges the unit. If no equivalent MD(EK) is found, either the manufacturer failed to inform the Privacy CA about this unit or this is an invalid EK. Thus, the Privacy CA does not provide an AC to the unit.

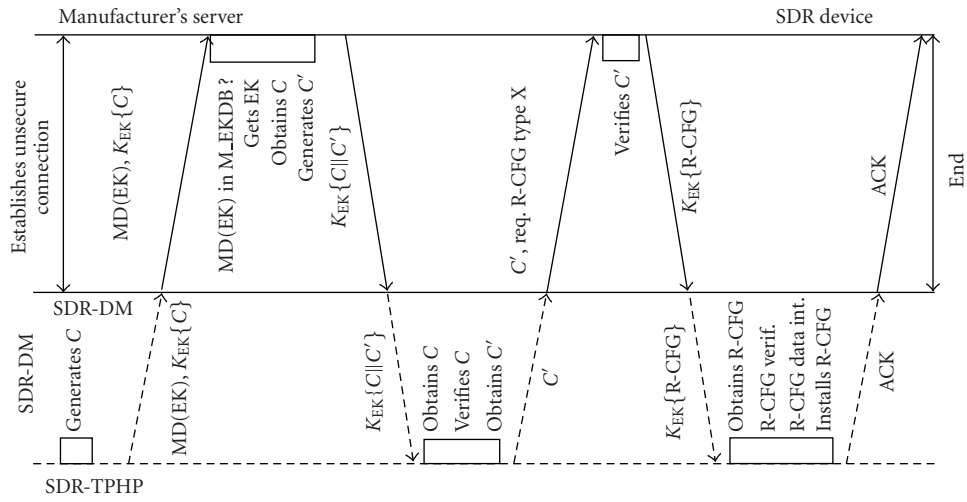

Figure 5: The secure SDR R-CFG download protocol.

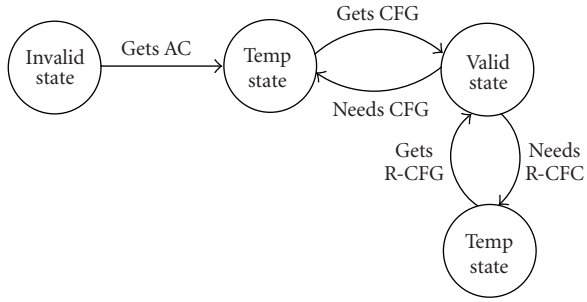

Figure 6: Transition states of an SDR-MD.

Second, the Privacy CA generates an AK pair and the unit authenticates the Privacy CA. The unit generates a nonce C and sends it to the Privacy CA encrypted by the EK. The Privacy CA obtains C and sends it back along with an encrypted message containing the AK pair. Upon receiving the message, the unit verifies C, authenticating the Privacy CA.

Third, after authenticating the Privacy CA, the unit obtains the AK pair and acknowledges the Privacy CA. The Privacy CA then generates the AC = $[AK_{pub}]_{Privacy\,CA}$ and sends it, encrypted by the AK_{pub} to the unit. The unit receives the AC, decrypts it, and stores it in its TPHP. After that, the connection is finally released.

After obtaining the AC, the final step to enter the valid state is to have the SDR-MD executing the CFG update protocol (CUP) to obtain a valid CFG. This protocol is executed whenever the unit needs a new phone number. Figure 8 depicts the CUP step by step.

After connecting to the WO's server, the unit sends its AC and the value of $R2$ = Att(CFG) along with a nonce C encrypted by the WO's public key. The WO's public key is obtained a priori through a secure protocol. If this is a new unit, the value of $R2$ is null.

Upon receiving the AC, the WO verifies if the AC is null. If the comparison is positive, the unit is a clone and the WO terminates the connection. Otherwise, the CUP continues its normal flow.

The WO uses the Privacy CA's public key and decrypts the AC, obtaining the AK_{pub}. The WO has a database (DB), indexed by the AK_{pub}, that contains information about each SDR-MD in a valid state, such as phone number and user name. Next, the WO looks for a matching AK_{pub} in the DB. If it finds a match, it verifies MD(CFG) = $R2$. If the comparison is negative, this is an invalid unit; either this is a cloned unit or a masquerade attack is occurring, and countermeasures are taken.

On the other hand, if the comparison is positive, this is a valid unit. The WO then obtains C and generates a nonce C' to authenticate the unit. C is concatenated with C' and sent encrypted by the AK_{pub} to the SDR-MD. If the AK_{pub} is not in the DB, this is a unit in the temporary state.

Upon receiving $K_{AK_{pub}}\{C\|C'\}$ from the WO, the unit authenticates the WO if the received C is equal to the one previously generated. If authentication fails, the SDR-MD terminates the connection. Otherwise, it sends C' back to the WO.

Next, the WO authenticates the unit by verifying C'. If authentication fails, the WO terminates the connection. Otherwise, the WO generates a new CFG and stores the MD(CFG) value in the DB. The unit receives the CFG encrypted by its AK_{pub} and decrypts it. The unit then stores the CFG in the protected storage of TRC2 and installs the new phone number.

Next, the AM Eng measures Att(CFG) and writes the value in $R2$. The unit then sends this value encrypted by the WO's public key to the WO. The WO verifies the value and acknowledges the unit if the comparison is positive. Otherwise, it informs the unit that an error occurred during the CFG installation step. This step is repeated in the case of

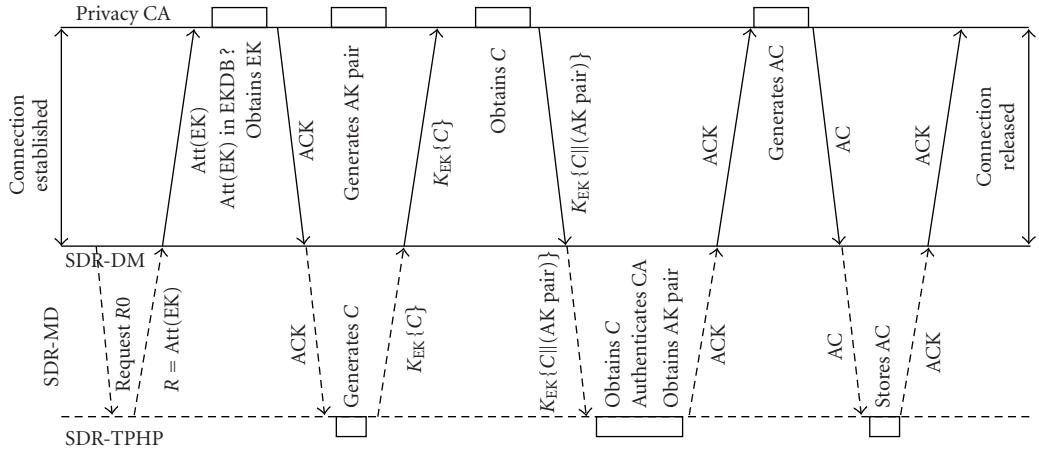

Figure 7: Attestation credential protocol.

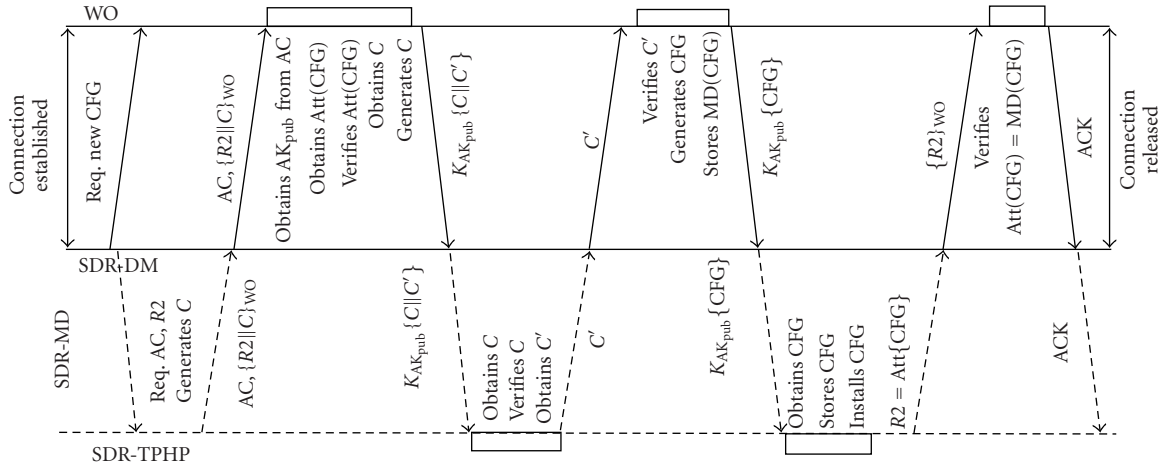

Figure 8: The CFG update protocol.

errors. After receiving an acknowledgment, the unit releases the connection.

After obtaining the AC from the Privacy CA and the CFG file from the WO, the SDR-MD finally reaches a valid state. Therefore, the unit is ready to use all the services offered by the WO. Figure 9 depicts the tamper-protected hardware package when the SDR-MD is in the valid state.

Note that the clone signal, sent by the AC Eng, propagates outside the TPHP to the CPU and inside the TPHP to the TRC2, where it sets the AC to null if the SDR-MD is a clone unit.

3.4.2. Cloning-aware procedure

The cloning-aware procedure is implemented in both sides, the WO and the SDR-MD, and is responsible for detecting whether the SDR-MD is a valid unit or a cloned unit.

After the unit has connected to the WO and requested a service, the cloning-aware procedure starts in the SDR-MD side. New Att(EK) and Att(CFG) values are measured by the AM Eng and sent to the AC Eng, which also receives the current value of $R0$ and $R2$ from the AR Eng. The AC Eng compares the values and signalizes 1 for a valid unit, if Att(EK) = $R0$ and Att(CFG) = $R2$, or 0 for a cloned unit, if Att(EK) $\neq$ $R0$ or Att(CFG) $\neq$ $R2$. In this fashion the SDR-MD is aware of cloning. Figure 10 illustrates the procedure. If the SDR-MD is a valid unit, the AC is sent and the WO cloning-aware procedure begins.

In the WO side, the procedure works basically as an authentication module. The WO obtains the AC and verifies if it is valid or null. If the AC is null, the WO terminates the connection, since the unit is a clone. Otherwise, the WO obtains the AK_{pub} from the AC and looks for a match in the DB. If there is no match, the service is denied. If there is a match, the WO prepares to authenticate the unit. If the unit is correctly authenticated, the WO allows the use of the service. On the other hand, if the unit is not authenticated, the WO concludes that this unit is trying to use other unit's AC (masquerade attack) and denies the service. Figure 11 illustrates the procedure.

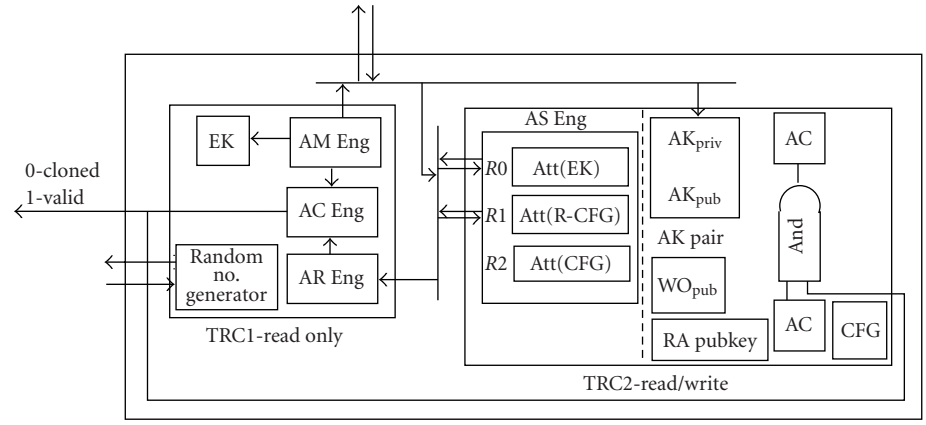

Figure 9: The tamper-protected hardware package in a valid state.

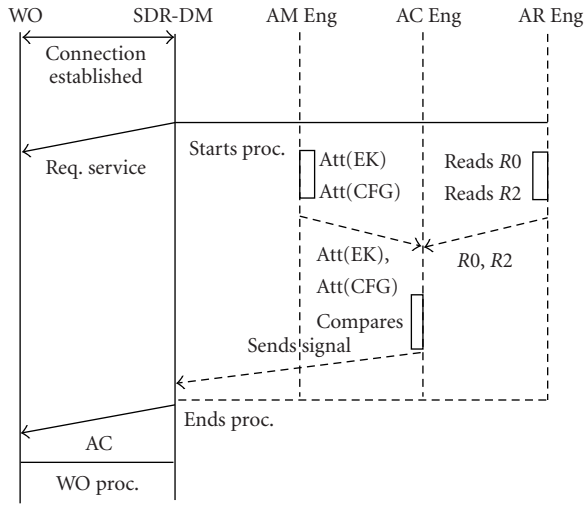

Figure 10: Cloning-aware procedure: SDR-MD side.

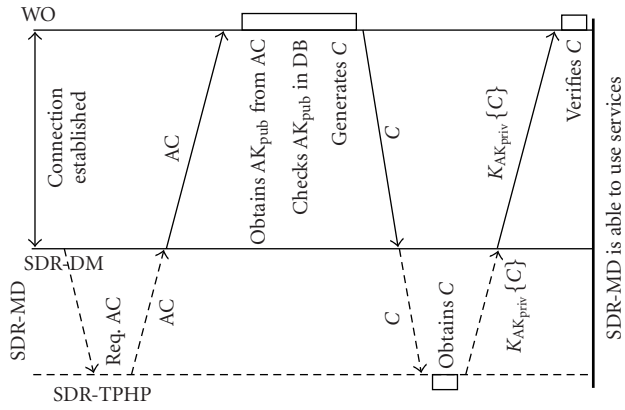

Figure 11: Cloning-aware procedure: WO side.

4. PRELIMINARY EXPERIMENTS

The experiments were executed using J2ME, which is a lightweight java version, specifically designed to be used with constrained devices. The experiments set-up is depicted in

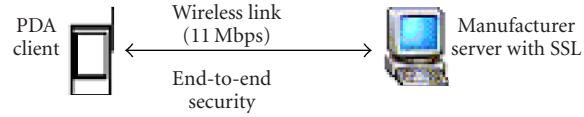

Figure 12: The experiment set-up.

Figure 12. An SDR-MD, in this case a Sharp Zaurus PDA SL-5600 with CPU speed of 400 MHz, 32 MB SDRAM, Linux OS, and J2ME support, connects through an 11 Mbps wireless link to a Pentium 4 2.6 GHz server with 256 MB RAM.

4.1. The secure SDR R-CFG download protocol

Two preliminary experiments involving the secure SDR R-CFG download protocol and the secure R-CFG/CFG module are described. In the first experiment, the time the R-CFG/CFG security module takes to identify invalid R-CFGs and delete them is measured. The second experiment compares the secure protocol execution when using an unsecured connection : HTTP, a lightweight secure connection, LSSL [19], and the SSL protocol [21].

The graph in Figure 13 shows the results of the first experiment. The MD5 algorithm is used to calculate the fingerprint and to perform the data integrity check. As expected, the larger the R-CFG is, the longer it takes to perform the security checks.

Figure 14 depicts the results of the second experiment. Note that the secure protocol with unsecured connection presents best performance, since it does not need to spend time with the cipher suite handshake and other extra steps needed by secure connections. In case secure connections are necessary, the use of the LSSL is suggested since it presents better performance than the SSL, as can be noticed in this experiment.

4.2. Anticloning scheme

It is expected that the anticlone scheme will not add any further delay on the obtainment of network services when comparing with the GSM and UMTS techniques. Although SDR mobile devices are constrained by nature, encryption and decryption operations are only executed for small pieces

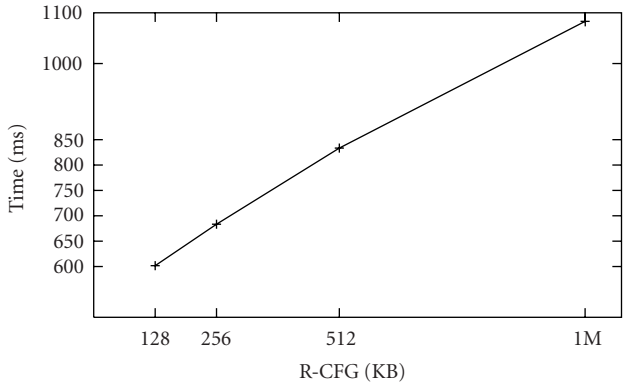

Figure 13: Time to identify invalid R-CFGs.

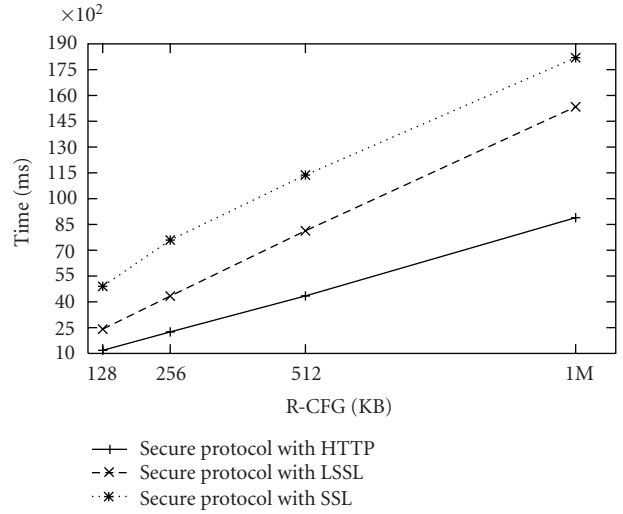

Figure 14: Comparing the secure protocol varying the connection type.

of information such as the 2048-bit EK and AK pair, and the 48-bit nonce C. Furthermore, the attestation engines and the random number generator in the TPHP are specialized pieces of hardware that can quickly execute data integrity measurements and generate a 48-bit random number.

5. CORRECTNESS PROOFS

This section presents a list of possible attacks involving the R-CFG files and how the secure SDR R-CFG protocol avoids those attacks. It then continues with correctness proofs that show that the fraud-prevention framework provides an environment free of cloned units.

Table 4 illustrates common methods of attacks that fail against the proposed protocol.

Next, the correctness proofs are presented. It begins with three lemmas. The first lemma shows that only an SDR-MD with a valid EK is provided an AC. The second lemma shows that an SDR-MD only obtains a new CFG when its identity is successfully proved. Finally, the third lemma shows that only valid CFGs, that is, CFGs that have been generated and signed by the WO, can be installed by an SDR-MD.

The proofs continue with two final theorems. The first theorem proves that there is no possibility to clone an SDR-MD over the air. The second theorem guarantees that only a valid SDR-MD can use the network services.

Lemma 1. *The Privacy CA only attests the identity of SDR-MDs that have valid EKs.*

Proof. Since the Privacy CA has a database of valid EKs and this database is assumed to be secured stored, any SDR-MD that requests an AC and sends an invalid MD(EK) value, that is, hash of an EK that is not generated by the manufacturer, has the AC denied.

A replay attack is not possible since the ACP is executed only once per each EK. Impersonation of the SDR-MD, that is, masquerade attack, is noticed by the authentication step. □

Lemma 2. *No SDR-MD obtains a CFG file unless its identity is successfully proved.*

Proof. According to the CUP definition, only after being authenticated by the WO, the SDR-MD is given a new CFG. This eliminates the possibility of masquerade attacks and replay attacks.

Only after responding correctly to the challenge generated by the WO, the SDR-MD is given a new CFG. Therefore, no SDR-MD obtains a new CFG file unless it has proved its identity. □

Lemma 3. *Only valid CFG files are installed in each SDR-MD.*

Proof. To install a new CFG, the SDR-MD must execute the CUP. According to the CUP definition, before receiving a new CFG the SDR-MD authenticates the WO by verifying $\{R2\}_{WO} = [\text{MD(CFG)}]$. If the comparison is positive, then the SDR-MD authenticates the WO. Thus, masquerade and replay attacks are eliminated.

After authentication, the SDR-MD receives a new CFG = $[\text{Phone no.}]_{WO}$. Since masquerade and replay attacks fail, only the WO could have sent this message, and the final step to validate the CFG occurs. The SDR-MD verifies the WO's signature in the CFG. When the signature is successfully verified, the CFG is considered valid and the TPHP stores and installs the new CFG. □

Theorem 1. *It is guaranteed that there is no possibility to clone an SDR-MD over the air.*

Proof. In order to clone an SDR-MD over the air, one attacker must obtain the EK of the victim or a combination of valid AK pair, valid AC, and valid CFG.

Since the EK and $AK_{private}$ are never disclosed by the TPHP, the attacker has no possibility to obtain the EK nor the AK pair of a victim. According to Lemma 2, the attacker must prove its identity to obtain a valid CFG, thus if the attacker uses an AC that is not his/hers, the WO will notice it and deny a new valid CFG.

Table 4: Possible attacks and how the secure protocol avoids them.

Attacks	Description	Protection
Access control	Clients using unauthorized services or trying to download data they should not	Protocol employs client authentication
Masquerade	An entity pretends to be the manufacturer server or a client	Protocol uses mutual authentication
Confidentiality	R-CFG might be confidential	By establishing secure connections or encrypting, the R-CFG proprietary information are kept secret
Replay	Messages are captured and retransmitted later	Mutual authentication avoids replay attacks
Invalid R-CFGs	Installing R-CFGs that are not approved by the RA	Every R-CFG is digitally signed by the RA and verified by the SDR-MD
R-CFG Integrity	R-CFG modified after it has been approved	Protocol employs one-way hash functions to guarantee data integrity

With no other way to clone an SDR-MD over the air, the only way to bill someone else's account is to capture his/her AC when transmitted over the air. However, the WO cloning-aware procedure will detect that the captured AC does not belong to that unit and it will deny any service. □

Theorem 2. *It is guaranteed that only a valid SDR-MD can use the wireless operator services.*

Proof. According to the WO cloning-aware procedure, in order to use the network services the SDR-MD must present a valid AC. By Lemma 1, only SDR-MDs with valid EKs are able to obtain a valid AC. Therefore, unit with an invalid EK does not have a valid AC and cannot use the WO's services.

According to Theorem 1, there is no way to clone an SDR-MD over the air, and impersonation of other SDR-MDs by capturing their AC is noticed by the WO cloning-aware procedure. Thus, the only other way to clone an SDR-MD is to have physical access to its TPHP.

However, if an attacker successfully disassembles the TPHP without damaging it and is able to copy the TPHP to another SDR-MD's TPHP, Lemma 3 and the SDR-MD cloning-aware procedure guarantee that the SDR-MD that received the cloned TPHP denies the use of the network services. The value of $R2$ on the cloned TPHP and the value of the current MD(CFG) in the device are different. Thus, the SDR-MD blocks the use of any services.

Since the SDR-MD cloning-aware procedure blocks the use of any service by cloned units and the WO cloning-aware procedure notices masquerade attacks, it is guaranteed that only a valid SDR-MD can use the wireless operator services. □

In summary, the fraud-prevention framework elevates the level of difficulty to clone an SDR-MD. The only way to clone one SDR-MD that employs the framework would be disassembling the TPHP from the SDR-MD and reading its contents. Since the TPHP is physically bound to other parts of the SDR-MD, attempts to disassemble it would probably damage the TPHP. Even if an attacker successfully disassembles the TPHP without damaging it, the equipment to read and copy the TPHP is so expensive that the attacker would practically have no gain, if any, in doing so.

6. CONCLUSION

To greatly enhance the overall security of SDR-MDs, a fraud-prevention framework is proposed. The fraud-prevention framework is composed of new pieces of hardware, modules, and protocols. The framework offers security monitoring against malicious attacks and viruses, protects sensitive information, creates and protects an identity for the system, employs a secure protocol for radio configuration download, and finally, establishes an anticloning scheme which guarantees that no units can be cloned over the air, and elevates the level of difficulty to clone units if the attacker has physical access to the mobile device. Even if cloned units exist, the anticloning scheme is able to identify and deny services to those units.

Preliminary experiments show that the framework is able to identify invalid R-CFGs with minimal delay. Proofs that analyze correctness of the framework show that the fraud-prevention framework provides an environment free of cloned units.

Future work includes the execution of several experiments that will measure performance of the fraud-prevention framework, and comparisons with other state-of-the-art related works.

REFERENCES

[1] B. Bing and N. Jayant, "A cellphone for all standards," *IEEE Spectr.*, vol. 39, no. 5, pp. 34–39, 2002.

[2] Software Defined Radio Forum website, http://www.sdrforum.org.

[3] Java 2 Micro Edition Technology website, http://wireless.java.sun.com/j2me.

[4] L. B. Michael, M. J. Mihaljevic, S. Haruyama, and R. Kohno, "A framework for secure download for software-defined radio," *IEEE Commun. Mag.*, vol. 40, no. 7, pp. 88–96, 2002.
[5] C. Castelluccia and F. Dupont, A Simple Privacy Extension for Mobile IPv6, *The Internet Engineering Task Force*, Internet Draft: Draft-Castellucia- MobileIP-Privacy, February 2001.
[6] A. Escudero, "Location privacy in IPv6—tracking binding updates," in *Proc. International Workshop on Interactive Distributed Multimedia Systems and Telecommunication (IDMS '01)*, Lancaster, UK, September 2001.
[7] CMS 88 Cellular Mobile TelephoneSystem, EN/LZT 101908, Ericsson.
[8] Global System for Mobile Communication, "The GSM security technical whitepaper for 2002," http://www.hackcanada.com/blackcrawl/.
[9] UC Berkeley. Internet Security, Applications, Authentication and Cryptography Group, "GSM cloning," http://www.isaac.cs.berkeley.edu/isaac/gsm.html.
[10] Overview of the Universal Mobile Telecommunication System. DRAFT, July 2002, http://www.umtsworld.com/technology/overview.htm.
[11] M. B. Frederick, "Cellular telephone fraud anit-fraud system," US Patent 5,448,760, September 1995.
[12] M. S. M. Annoni Notare, F. A. da Silva Cruz, B. Goncalves Riso, and C. B. Westphall, "Wireless communications: security management against clonedcellular phones," in *Proc. IEEE Wireless Communications and Networking Conference (WCNC '99)*, vol. 3, pp. 1412–1416 , New Orleans, La, USA, September 1999.
[13] The Trusted Computing Group, http://www.trustedcomputinggroup.org.
[14] The TCG PC Specific Implementation Specification, http://www.trustedcomputinggroup.org/downloads/.
[15] The RC5 encryption algorithm, http://www.secinf.net/cryptography/The_RC5_Encryption_Algorithm.html.
[16] RSA encryption, http://mathworld.wolfram.com/RSA-Encryption.html.
[17] Federal Communications Commission. Authorization and use of software defined radio: first report and order, September 2001, http://www.fcc.gov/Bureaus/Engineering_Technology/Notices/2000/fcc00430.txt.
[18] Intel, "Intel wireless trusted platform: security for mobile devices," http://www.intel.com/design/pca/applicationsprocessors/whitepapers/300868.htm.
[19] A. Brawerman, D. Blough, and B. Bing, "Securing the download of radio configuration files for software defined radio devices," in *Proc. ACM International Workshop on Mobility Management and Wireless Access (MobiWac '04)*, pp. 98–105, Philadelphia, Pa, USA, September–October 2004.
[20] US Secret Service Financial Crimes Division, http://www.secretservice.gov/financial_crimes.shtml#Telecommunications.
[21] A. O. Freier, P. Karlton, and P. C. Kocher, The SSL protocol Version 3.0, http://home.netscape.com/eng/ssl3.

Alessandro Brawerman is a graduate student pursuing a Doctoral degree in the School of Electrical and Computer Engineering, Georgia Institute of Technology. He received his B.S. and M.S. degrees in computer science from the Federal University of Parana, Brazil, during December 1997 and May 2000, respectively. His research interests are in security for software defined radio devices, such as cellphones and PDAs, and security and management of wireless networks. He currently holds a scholarship from Brazil and his research is funded by the Brazilian National Council of Research (CNPq). He has published and presented over 10 technical papers. He is an IEEE Member and was a Fellow of the Panasonic Information & Networking Technologies Laboratory (2003–2004). He was also the Instructor of several computer science courses at the Federal University of Parana (1998–2000).

John A. Copeland has been the John H. Weitnauer, Jr. Chaired Professor at the School of Electrical and Computer Engineering, Georgia Institute of Technology, from 1983 up to date. He teaches graduate and undergraduate courses on communications networks and network security, and does research in those areas (http://www.csc.gatech.edu). He was Director of the Georgia Center for Advanced Telecommunications Technology from 1993 to 1996, Vice President of Technology at Hayes Microcomputer Products from 1985 to 1993, and Vice President of Engineering Technology at Sangamo Weston, Inc. from 1982 to 1985, and did research at Bell Labs on semiconductor circuits and optical fiber networks from 1965 to 1982. He received the B.S., M.S., and Ph.D. degrees in physics from the Georgia Institute of Technology. He has been awarded 38 patents and has published over 60 technical papers. In 1970, he received the IEEE's Morris N. Liebmann Award. He is a Fellow of the IEEE and has served as the Editor of the IEEE Transactions on Electron Devices. He served on the Board of Trustees for the Georgia Tech Research Corporation from 1983 to 1993. He is a Member of Infragard, the ISSA, and the ACM SIGSAC. In 2000, he invented the StealthWatch network behavior anomaly detection system, and founded Lancope (http://www.lancope.com) which has deployed the StealthWatch system in over 100 corporate and government networks.