A wireless network is more vulnerable to denial of service (DoS) attacks than a wired one. In this paper we propose a new DoS defense scheme toward actively resisting DoS attacks. A mobile terminal generates an authorized anonymous ID (AAI) using its true ID, and assigns its true ID with the produced AAI. Using an AAI, a legitimate mobile terminal will be authenticated by the wireless network, however, its true ID is concealed, and it ‘disappears’ to potential attackers. This method can be used to defend several kinds of DoS attacks at the same time. Additionally it can also be used to alleviate other kinds of security threats in wireless networks, such as eavesdropping. We demonstrate our proposed method in detail in a new application network: UMTS-WLAN network, and provide some simulation results in OPNET 10.0 A environment.

Keywords: DoS attack, Authorized Anonymous ID, Mobile IP, UMTS-WLAN, OPNET.

1 INTRODUCTION

Wireless networks use an open medium to transmit data, so all transmissions are subject to interception and eavesdropping. For example, malicious users may spoof the identities of legitimate mobile terminals through wireless channels, and launch denial of service (DoS) attacks which will congest the whole wireless network. However, any kind of congestion is intolerable when mobile terminals are used to transmit continuous and real-time data. Moreover, wireless systems usually have a much narrower bandwidth than wire-line ones. So protection of mobile terminals from DoS attacks is crucial for wireless networks.

There are many kinds of DoS attacks in wireless networks, and resisting all of them becomes a real challenge. Unlike existing DoS defense methods, we propose a new method to defend against DoS attacks. In our scheme, we generate an authorized anonymous ID (AAI) using the user’s true ID, and then replace the true ID with the produced AAI. In this way, a legitimate mobile terminal will conceal its personal information which may be used by DoS attackers, while still obtaining the wireless service.

We show the efficiency of our method in a new application network: UMTS-WLAN network. The rest of the paper is arranged as follows: Section 2 introduces the DoS attacks in wireless networks; Section 3 explains one specific wireless system, UMTS-WLAN hybrid network, which is vulnerable to DoS attacks; Section 4 presents a security protocol to resist DoS attacks; Section 5 is a discussion of our scheme; Section 6 provides simulation results in the OPNET environment; Session 7 concludes the paper.

2 DOS ATTACKS IN WIRELESS SYSTEMS

2.1 Types of Attacks for Wireless Communications

Compared to a wired line, a wireless channel is more susceptible to attacks from both passive eavesdropping and active interfering. There are several main common security threats in wireless network.

2.1.1 Eavesdropping

An attacker steals private keys, decryption keys, session keys, etc, from the mobile terminals. Using
corresponding keys, the attacker can eavesdrop on
the communication through wireless channels, and
extract useful information.

2.1.2 Denial of Service

An attacker can cause congestion in a wireless
network either by generating an excessive amount of
traffic itself, or by making other nodes generate
excessive amounts of traffic [1]. In general, attackers
try to keep the legitimate users away from expected
services using DoS attacks.

2.1.3 Theft of Service

A malicious user may spoof the IP (Internet
Protocol) address and/or MAC (Medium Access
Control) address of legitimate users to take over the
wireless communication service [2]. Note that the
MAC address of a wireless device is a kind of
hardware address that is a unique identification
number assigned by manufacturers. Actually, the
theft of service attack can be considered a special
kind of denial of service attack, as it also keeps the
legitimate users away from its services.

Much work has been done on eavesdropping
attack resistance to obtain enhanced security. For
element, Burton Group offers an immediate, strong
solution for WLAN, i.e. Wi-Fi Protected Access
(WPA) [3]. Matsunaga et al. [2] designed a secure
authentication system to enhance the security of
wireless channels. However, defense solutions are
hard to produce for DoS attacks, because some holes
are inherent in the wireless MAC protocol. For
example, in general every user is given link-layer
access in 802.11 protocol, but a malicious user can
disturb a legitimate user’s communications by
spoofing the MAC address or flooding frames in
layer 2 network [2]. Safeguarding a legitimate user
from DoS attacks is a challenging task.

2.2 Denial of Service Attack in Wireless
Network

DoS attack is one of the active interfering
attacks and it is difficult to protect against. Besides
the common DoS attacks in the wired network, such
as transmitting falsified route updates, and reducing
the TTL (time-to-live) field in the IP header [1], the
wireless network has its own DoS attacks. For
example, an attacker can send a message to keep the
wireless channel busy, so no other legitimate devices
can utilize the channel. Another example is that an
attacker may use up the battery of a particular node
by making that node continually dump data [1]. In
general, DoS attacks in wireless networks can be
classified into two categories; one is pure resource
consumption DoS attacks, the other is protocol
related DoS attacks. Following, we will briefly
summarize these two.

2.2.1 Resource Consumption DoS Attacks

Attackers try to exhaust either the resources
allocated for public usage or the resources allocated
for a particular user. Typical resource consumption
DoS attacks include congestion-based MAC layer
attack, mass-produced junk message attack, virtual
carrier-sense attack [3], battery draining attack by
relaying spurious data, etc.

2.2.2 Protocol Related DoS Attacks

Attackers modify protocols or use existing
protocols to generate spurious messages. Typical
attacks include de-authentication attack/de-
associating attack [3], route updates falsification/overdue route date replaying attack [1],
TTL field of IP header modification attack, spoofing
time-mean DoS attack, etc.

2.3 Existing Defense Methods for DOS Attacks

Some studies have been done in DoS attacks.
Gupta, et al. analyzed congestion-based attacks that
deny channel access by causing packet congestion in
mobile ad hoc networks, and proposed a method of
using MAC layer fairness to alleviate the effects of
such attacks [1]. Faria and Cheriton considered DoS
attacks coming from authentications, and proposed a
new authentication structure to address the problem
[4]. Kyasanur and Vaidya studied and simulated
some misbehaviors in wireless networks, where
selfish hosts fail to follow the MAC protocol and try
to obtain an unfair share of the channel bandwidth.
They presented a scheme to detect and penalize such
selfish behaviors [5]. Bellardo and Savage focused
on DoS attacks on the MAC protocol itself. They
described software infrastructure for generating
arbitrary 802.11 frames using commodity hardware
and used this platform to implement de-
authentication DoS attack and virtual carrier-sense
DoS attack. They then proposed potential low-
overhead implementation changes to mitigate the
underlying DoS attacks [3]. Karlof and Wagner
worked with DoS attacks on wireless sensor
networks [6]. They identified several DoS attacks
including black holes, resource exhaustion, sinkholes,
induced routing loops, wormholes, hello flooding,
etc, which are directed against the routing protocol
employed by wireless sensor networks [7]. They then
designed several countermeasures [6] for
responding DoS attacks, such as: using a globally
shared key do link-layer encryption and
authentication; verifying the bi-directionality of a
link before taking meaningful actions based on a
message received over that link; carefully designing
routing protocols, such as geographic protocols, in
which wormholes and sinkholes are meaningless;
using multi-path whose nodes are completely
disjointed; and exploiting authenticated broadcast
and flooding. Houle [8] focused on DoS attacks with
name-servers to execute packet flooding, and
introduced a solution using packet filtering to
prevent DoS attacks based on IP source spoofing [9].
A common feature of previous DoS attack
defense methods is that they can be used to resist only one type of DoS attack. Unlike these DoS attack resistance methods, we designed a new DoS defense method which can be used to resist multiple DoS attacks at the same time as long as the DoS attacks are launched on a particular victim [28], such as a mass-produced junk message attack, battery draining attack, etc. Moreover our method can be used when mobile terminal is roaming away from its home network. In the following sections, we illustrate our new method using a UMTS-WLAN hybrid network for the following reasons: First of all, UMTS networks have a much slower transmission rate than WLAN networks (we will discuss in more detail in the next section), so there is an inherent bottle-neck in the UMTS-WLAN hybrid network, and DoS attacks will congest the UMTS-WLAN network more easily than the pure WLAN network. Secondly, UMTS-WLAN is a hybrid wireless network, and there is no works related to the DoS attacks defense in UMTS-WLAN before. Additionally, we have already built a UMTS-WLAN network in OPNET 10.0A environment [10]. Using this model we can clearly show the effectiveness of our method in terms of DoS attack resistance. It is to be noted that our method is not restricted to this model, and it can be used in other wireless communication systems, such as ad hoc network, sensor network, etc.

3 INTRODUCTION OF UMTS-WLAN NETWORK

In section, we introduce the UMTS-WLAN network.

3.1 UMTS-WLAN Technology

Universal Mobile Telecommunication Systems (UMTS) and Wireless Local Area Networks (WLAN) are two complementary technologies [11]. The UMTS third generation (3G) network provides a wide area of coverage, high mobility, and relatively low speed, whereas WLAN provides local coverage, low mobility, and relatively high speed.

The UMTS 3G network is an evolutionary system based on the current time division multiple access (TDMA) system. It works in a frequency division duplex (FDD) mode, and uplink and downlink transmissions use different frequency bands with a transmission rate of 384 kbps [12] in a wide coverage area. Figure 1 shows a typical UMTS 3G network. Basically, it is comprised of three parts: User Equipment (UE), the UMTS Terrestrial Radio Access Network (UTRAN), and the Core Network (CN). UTRAN has two nodes: Node B and the Radio Network Controller (RNC). CN also has two nodes: Serving General packet radio service (GPRS) Support Node (SGSN) and Gateway GPRS Support Node (GGSN). The SGSN provides authentication, access control, and location management [13]. The GGSN connects the UMTS network to the Internet.

WLAN has several types of standards: IEEE 802.11a, IEEE802.11b, IEEE802.11g, and so on. For example, IEEE 802.11b supports a transmission rate up of to 11 mbps and covers no more than 100 meters in an urban area [14]. There are two basic architectures for constructing a WLAN: ad-hoc and infrastructure. For an ad-hoc architecture, every mobile station (STA) can communicate with every other station (STA) in IEEE 801.11b piconet (The smallest network unit in WLAN). For an infrastructure architecture, every STA must pass through an Access Point (AP) to communicate with other STAs.

UMTS-WLAN technology couples a UMTS 3G wireless network with Wireless LANs.

![Figure 1: UMTS 3G Network](image)

3.2 Handover in UMTS-WLAN network

In the UMTS-WLAN hybrid network, mobile terminals communicate with a WLAN-enabled UMTS user equipment point, i.e. UMTS cellular phone, or a WLAN access point connected to UMTS SGSN/GGSN node [10]. Through UMTS-WLAN network, mobile terminals can connect to Internet almost without any location restrictions.

As an example, Figure 2 shows an unconfined e-healthcare system we built in OPNET 10.0A simulation environment [10], which connects WLAN with UMTS at both UE and SGSN points (Note: in our model, wireless sensor 1&2 transmit data through UE point, wireless sensor 3&4 transmit data through SGSN point), and uses mobile IP approach to interconnect UMTS and WLAN. The following analyses and conclusions are based on but not restricted to this model.

UMTS and WLAN are two different protocols, and the procedure for interworking between UMTS and WLAN is through the handover. In UMTS-WLAN hybrid mobile network, handover is important for both UMTS and WLAN. Good
handover technologies enable mobile terminals to roam between UMTS and WLAN without losing connection. Basically, there are three kinds of UMTS-WLAN interworking strategies: mobile IP, gateway, and emulator. Tsao and Lin [15] gave detailed descriptions of all three approaches. If a mobile station (STA) or mobile user equipment (UE) wants to keep the IP address unchanged when roaming between UMTS and WLAN, a mobile IP approach should be involved. Also, the mobile IP can keep the connection when STAs or UEs roam in UMTS-WLAN network. So our model is equipped with mobile IP.

A mobile node may change its location without changing its IP address. (2) Home Agent (HA). It is an entity that tunnels datagrams for delivery to mobile node when it is away from the home network, and that maintains location information for the mobile node. (3) Foreign Agent (FA). Is an entity that gives local access when a mobile terminal is away from its home agent, de-tunnels and delivers datagrams to the mobile node that was tunneled by the home agent, and tells the HA where the mobile terminal is.

In our UMTS-WLAN model shown in Figure 2, Mobile nodes are wireless sensors, HA is installed in SGSN point, and FAs are installed in corresponding wireless LAN access points (APs): UWLAN_AP or UWLAN_UE. When a wireless sensor is in the areas covered by APs, the FAs will inform HA of the sensor where it is. Afterwards, HA will encapsulate and tunnel the datagrams to FAs, and the FAs will de-tunnel and deliver the packets to the wireless sensor.

Figure 2: An Example of UMTS-WLAN in OPNET

3.3 Mobile IP in UMTS-WLAN

Mobile Internet Protocol (MIP) is a specific base protocol for mobility handling in wireless communication systems. As MIP is independent of the underlying transmission technology and has unconstrained mobility based on internet protocols, it can be used in internet service over heterogeneous networks such as UMTS-WLAN, and provides seamless mobility across networks and technologies.

MIP protocol has two versions: mobile IPv4 (a base MIP standard from 1996[16]) and mobile IPv6 (MIP standard being standardized). Both are comprised of three components [16]: (1) Mobile Node. It is a host that can change its point of attachment from one network/sub-network to another.

Figure 3 shows the protocol stack of the UMTS-WLAN hybrid network using the MIP approach [15]. In the UMTS network, a UE uses standard UMTS protocol, i.e. session management (SM), GPRS mobility management (GMM), GPRS tunneling protocol (GTP), medium access control (MAC), etc., to handle data packets transmission and roaming
between UMTS cells. In WLAN network, a mobile STA uses IP protocol directly to transmit data packets, and uses MIP to handle roaming between different APs. In order to handover smoothly in UMTS-WLAN, it is necessary to install HA and FAs in UMTS GGSN and WLAN access routers. HA or FAs tunnel and forward the data packets using the MIP protocol when mobile nodes roam between UMTS and WLAN.

4 RESISTING DOS ATTACK USING AAI

4.1 Authorized Anonymous ID (AAI)

An AAI is a pseudo ID that only tells the wireless system whether the provider of the ID is a legitimate user or not. There are several AAI-related techniques. For example AAI has been applied in location privacy area in [17], where by using an authorized anonymous ID, a mobile user can get personal control over his/her location privacy. Another important AAI-related technique is blind signature. Blind signature schemes, first introduced by Chaum [25][26], allow a person to get a message signed by another party without revealing any information about the message to the other party. Blind signatures have numerous uses including anonymous access control, and digital cash [27].

In this paper we propose a new AAI generation method, and use the AAI to resist DoS attacks in UMTS-WLAN network. With our scheme a legitimate mobile terminal can successfully register the wireless network with its AAI and transmit packets, however the mobile user ‘disappears’ to any potential DoS attackers.

4.2 AAI Generation for MTs in a Foreign AP

We shall design a protocol to generate an AAI when mobile terminals (MTs) are roaming into the coverage of a foreign access point. For the convenience in describing our AAI generation procedure, we list the notations as follows:

MT: mobile terminal
HA: home agent
FA: Foreign agent
AP: access point
Eh: Public key of HA
Dh: Private key of HA
Ep: Private key shared by legitimate MT and HA
Eh(I): Encrypt information I using public key of HA
Dh(C): Decrypt the encrypted information C using private key of HA

\( k_{FH}^{(I)} \): Encrypt information I using symmetric key shared by FA and HA

RC: A random number, different random numbers are used in different AAI generations

Timestamp: The current time of day. It is used as replay protection, the node generating a message inserts the current time of day, and the node receiving the message checks that this timestamp is sufficiently close to its own time of day [16]

IDmt: The true identity of a mobile terminal
IDap: The identity of an access point equipped with FA

Crm: The encrypted message, \( n=0,1,2,3 \)

H(x): A secure one-way, nonreversible hash function (e.g. MD5) with input x

ID_aym: The generated AAI

g(x): A monotonic function

\( P(I) \): Processed operations in information I

Figure 4 shows the AAI generation architecture when a MT (such as the wireless_sensor_3&4 in our UMTS-WLAN model in Figure 2) is roaming to the coverage of a foreign AP (such as UWLAN_AP node in our UMTS-WLAN model in Figure 2). Seven steps are needed to generate an AAI. Here we assume that HA are trustable (If HA are not trustable, HA and MT need authenticate each other before AAI generation).

In the first step, the MT encrypts its true identity (IDmt), a random number RC, and the timestamp using the public key of the HA (SGNG in our model as shown in Figure 2). It gets \( \text{Cr}_0 = \text{Eh}(\text{IDmt}, \text{RC}, \text{timestamp}) \), and sends \( \text{Cr}_0 \) to FA (UWLAN_AP in our model) via the wireless LAN channel.

In the second step, the FA encrypts received \( \text{Cr}_0 \) and its identity IDap using symmetric key shared by FA and HA to generate \( \text{Cr}_1 \), and forwards \( \text{Cr}_1 \) to the HA via wired a line between FA and HA.

In the third step, HA decrypts \( \text{Cr}_1 \) using a symmetric key shared with FA and obtains (Cr0, IDap). HA then searches the database to check whether the identity of FA i.e. IDap exists or not. If it does not exist, then the FA is considered as illegal and HA terminates the process; otherwise, HA further decrypts \( \text{Cr}_0 \) using its private key and obtains (IDmt, RC, timestamp). Also HA checks whether the identity of MT i.e. IDmt is legal or not. If IDmt is legitimate, HA authenticates the RC and timestamp. Furthermore, it compares the RC received with the RC pre-stored in memory to see whether the two RCs are identical, and it compares the timestamp received with its own time of day to determine whether they are sufficiently close. If these two comparisons are correct, the MT is accepted as legitimate, otherwise, HA terminates the
authorization procedure.

In the fourth step, HA encrypts hashing function $H(RC)$ using its private key and obtains $Dh(H(RC))$, and selects a new random number $RC_n$ to compute $XR = RC ⊕ RC_n$ (‘$\oplus$’ is exclusive-OR). It then gets its current time of day i.e. timestamp_n, and encrypts $(XR, timestamp_n, Dh(H(RC)))$ using symmetric key: $K_o$ to compute $Cr_2$, and forward $Cr_2$ to FA via a wired line.

In the fifth step, FA simply decrypts $Cr_2$ using symmetric key: $K_o$ and sends the results (i.e. $XR$, $timestamp_n$, $Dh(H(RC))$) as well as a temporary symmetric key: $Key_FA$ which will be used in situation of handoff to the MT via a wireless channel.

In the sixth step, after receiving $(XR, timestamp_n, Dh(H(RC)))$ and $Key_FA$ the MT authenticates $timestamp_n$ and hashing function $H(RC)$. It first compares $timestamp_n$ received with its own time of day to see whether they are sufficiently close. Then it compares $Eh(Dh(H(RC)))$ with $H(RC)$ to check whether they are the same. If these two verifications are correct, the MT will keep $Dh(H(RC))$, and further generate the AAI using $Dh(H(RC))$ and current timestamp, namely, $AAI = ID_{aym} = g(Dh(H(RC)), timestamp)$. Afterwards MT updates random number RC with $(RC \oplus XR)$ for the next AAI generation procedure, and saves $Key_FA$ for the situation of handoff.

Finally, the MT informs HA of the successful AAI generation, and HA updates the memory with the new random number $RC_n$.

Proposition: $Eh(Dh(H(RC))) = H(RC)$

Proof: Hashing function $H(x)$ is shared by the legitimate MT and HA, and for a specific authentication procedure the RC are the same for the legitimate MT and HA. So if $Dh(H(RC))$, which is used as AAI, is from a legitimate agent, the MT should hold $Eh(Dh(H(RC))) = H(RC)$.

Through the above steps, an MT generates an AAI when it is in the coverage of a foreign AP. If the MT is in its home personal network, the AAI generation procedures are even simpler. It can generate AAIs only through HA (such as such as SGSN node in our UMTS-WLAN model in Figure 2). For succinctness, we will not show the detailed procedure here.

### 4.3 AAI Generation in the Situation of Handoff

When an MT roams from one WLAN to another WLAN, it will switch from the old foreign agent, $FA_o$, to the new foreign agent, $FA_n$. This is handled by the handoff procedure.

Figure 5 shows the AAI generation architecture when an MT roams from one FA to another FA. A new AAI is generated from the old AAI in the situation of a handoff. This protocol makes it extremely difficult for an attacker to guess the new AAI without knowing the old AAI. Six steps are needed to generate the new AAI.

In the first step, the old FA ($FA_o$) generates a random number NR, encrypts it using symmetric key shard by HA ($K_{n.o}$) and temporary symmetric key shard by MT ($Key_FA$), and sends them to MT and HA respectively.

In the second step, MT decrypts the message using the temporary symmetric key receiving from Fig 4. to get NR, updates its random number RC with $RC' = RC \oplus NR$ (‘$\oplus$’ is exclusive or), and
computes \( E_1 = \text{Ep}(RC') \) using the key shared by legitimate MT and HA. Then MT sends \( E_1 \) to the new foreign agent (\( \text{FA}_n \)) via a wireless channel.

In the third step, HA updates its random number \( RC \) with \( RC' = RC \oplus NR \) after getting \( NR \) by decrypting the received message, computes \( E_2 = \text{Ep}(RC') \) using the key shared by legitimate MT and HA, and encrypts hashing function \( H(RC') \) using its private key and obtains \( \text{Dh}(H(RC')) \), HA then generates a new random number \( RC_n \) to compute \( XR=RC' \oplus RC_n \), and encrypts \( (\text{AAI}_t, XR) \) using symmetric key \( K_{fh_n} \), then forwards \( E_2 \) and the encrypted result \( \text{Cr}_4 \) to \( \text{FA}_n \) via a wired line.

In the fourth step, \( \text{FA}_n \) receives \( E_1 \) from MT through a wireless channel, and receives \( E_2, \text{Cr}_4 \) from HA through a wired line. \( \text{FA}_n \) compares \( E_1' \) with \( E_2 \). If they are unequal, MT is considered an illegitimate terminal, and \( \text{FA}_n \) terminates the authorization procedure. Otherwise, \( \text{FA}_n \) decrypts \( \text{Cr}_4 \) using symmetric key \( K_{fh_n} \) and sends result \( (\text{AAI}_t, XR) \) as well as a new temporary symmetric key: \( \text{Key}_{FA_n} \) which will be used in the next handoff to the MT via a wireless channel.

In the fifth step, after receiving \( (\text{AAI}_t, XR) \), the MT compares \( \text{Eh}(\text{AAI}_t) \) with \( \text{Eh}(\text{AAI}) \cdot H(RC') \) to check if they are the same. If they match, the MT will generate a new authorized anonymous ID using \( \text{AAI}_t \) and current timestamp, namely, \( \text{AAI}' = g(\text{AAI}_t, \text{timestamp}) \). Afterwards, MT updates the random number \( RC' \) with \( (RC' \oplus XR) \) for the next AAI generation procedure, and saves \( \text{Key}_{FA_n} \) for the next handoff.

Finally, the MT informs HA of the successful AAI' generation, and HA updates the memory with the new random number \( RC_n \).

**Proposition:** The AAI\(_t\) has the property of \( \text{Eh}(\text{AAI}_t) = \text{Eh}(\text{AAI}) \cdot \text{H}(RC') \).

**Proof:**

In the fifth step we use the concept of privacy homomorphism, which was introduced by Rivest\[23\], to authenticate the AAI\(_t\). Privacy homomorphism can be described as follows:

\[
\text{Dh}\{ \text{P}\{ \text{Eh}(I) \} \} = \text{Eh}\{ \text{P}\{ \text{Dh}(I) \} \} = \text{P}(I) \quad (1)
\]

Equation (1) shows that the result of decryption, after processing the operations of the encrypted information, is the same as the processed operations in the plain information \[24\]. With privacy homomorphism, the secret information kept in the old foreign agent will be safely forwarded to the new foreign agent in the situation of a handoff.

AAI\(_t\) is the result of the multiplication of two messages, i.e., AAI and Dh(RC\'). By the property of privacy homomorphism, AAI and Dh(RC\') do not need to be decrypted respectively at the mobile terminal when hand off occurs. So we have the following equations:

\[
\begin{align*}
\text{Eh}(\text{AAI}_t) &= \text{Eh}(\text{AAI}) \cdot \text{Dh}(H(RC')) \\
&= \text{Eh}(\text{AAI}) \cdot \text{Eh}(\text{Dh}(H(RC'))) \\
&= \text{Eh}(\text{AAI}) \cdot \text{Eh}(\text{Dh}(H(RC'))) = \text{Eh}(\text{AAI}) \cdot \text{H}(RC').
\end{align*}
\]

QED.

### 4.4 Resisting DoS Attack with AAI

Normally an individual MT is identified initially
by its MAC address, but when it generates traffic, a slightly modified version ofdsniff [18] can be used as a better identifier such as a user ID, a custom DNS (domain name server), and others. These identifiers can be used by malicious users to select an individual host for DoS attacks[3].

Before a malicious attacker can successfully launch a DoS attack to a specific device in UMTS-WLAN network, he/she must get the sufficient identity of that device, including the MAC address, user ID, or DNS address. Actually, snooping a MAC address or user ID of a legal device is not a challenge for the attackers. Using iPAQ H3600 COMPAQ packet PC with Dlink DWL-650 card running the Swat attack testing tool, Bellardo and Savage [3] showed how to get the identities of individual clients and APs by passively monitoring the wireless channels. Our proposed authorized AAI provides an approach toward protecting wireless devices from DoS attacks by preventing the critical personal information from being snooped.

If an MT is to start a communication session, it first uses its true identity (e.g. MAC address or user ID) to achieve authorization and generate an AAI according to the procedures described in 4.2 or in 4.3, then it replaces its true ID with the AAI (MAC addresses are software updateable on most wireless interface cards [19]) and registers to the UMTS-WLAN network. Furthermore, this AAI can be used as the key for packet authentication [17], i.e. generates a message authentication code by the AAI, and controls the access with the authentication code [20]. In this way, the HA and FA can grant authorized MTs access to the UMTS-WLAN network and start a communication session. It need not disclose its true ID, which may be used by an attacker to launch DoS attacks. To enhance the security, the MT must generate a new AAI if one of the following conditions happens: 1) Lifetime of the AAI expires; 2) The MT startups a new communication session.

5 DISCUSSIONS

In our protocol, the true ID of a wireless device is replaced by an AAI. A periodically changed AAI makes it hard for a malicious user to find the correspondence between the AAI and the wireless device. As an additional benefit, our proposed scheme can also be used to resist other attacks, such as eavesdropping, because it will be hard for an attacker to launch an intended eavesdropping without the true ID of the victim. When using our method, we need to consider the following situations.

5.1 All Hosts are Attacked in Burst

When attackers launch random DoS attacks, or attack all the legal devices in burst, our proposed method will not be much help. In these situations, the attacker need not know the relationship of the AAI and the real device since IDs, no matter if they are true IDs or AAIs, are randomly chosen from wireless channels. For this kind of attacker, we may use the covert channel method [21] to trace back and find the malicious attackers. First, covert channels are designed in the mobile IP packet headers. Then some information of the intermediate nodes (SGSN, GGSN, APs, etc) is inserted into the covert channels. The inserted information is resumed on the victims’ side. Finally, the paths from attackers to the victims can be identified with the help of the inserted information, and victims may isolate the attackers after achieving the paths.

5.2 Identity Collision

In our protocol, we replace the true identity with AAI. It may seem there might be an ID collision. However, we use two steps to avoid ID collision, first using a hashing function mechanism to generate a data: Dh(H(RC)) which has little chance of collision. Second, using a monotonous function of timestamp and Dh(H(RC)) to ensure the unique of AAI.

5.3 Needed Computation

Here most of the calculations and authentications are done at HA, to which computation time is not a large concern as HAs are always be equipped with a powerful computer and supplied with continuous power.

However, mobile terminals, which have only limited computing capability, need to compute 2 times encryptions (i.e. compute C r0, Eh(Dh(H(RC))) ), 1 time exclusive OR, 2 times comparisons, 2 times data updates (i.e. update RC and ID) in the AAI generation protocol. Most of computation time is used in the encryption procedure. For example, in some chip-designed technology [22], a number of milliseconds are needed if using an 8-bit micro-controller to perform a 1024 bits RSA encryption [24].

5.3 Power Consumption

To implement our scheme, the MTs need do encryption, decryption, authentication, and true ID replacement as shown in Figure 4 and 5. All of these procedures consume much energy. Battery power is a precious resource for a MT, especially for small hosts, such as wireless medical sensors in our proposed model (see Figure 2.). In order to save the energy, MTs should have the option to extend the
lifetime of an AAI. Another way to mitigate the power consumption issue is to use pre-generated AAs that are stored in its memory in advance.

6 SIMULATIONS

We used the UMTS-WLAN model we created earlier (see Figure 2) and modified it to show the results of our proposed method using OPNET simulation. For concision, we chose wireless_sensor_1 as an example to show the performance (wireless_sensor_2 & 3 & 4 have almost the same performance).

Experiment 1: Effect of DoS Attack without AAI

We simulate the effect of DoS attacks in OPNET 10.0 A environment. The simulation lasts 3 minutes, meanwhile an attacker launches a mass-produced junk message attack, one kind of resource consumption DoS attacks, to wireless_sensor_1 between minute 1 and minute 2. Figure 6 shows the media access delay of wireless_sensor_1, Figure 7 shows the packet delivery delay of wireless_sensor_1, and Figure 8 shows the throughput of wireless_sensor_1. We can see that during the period of DoS attacks, both media access and packet delivery delays are greatly increased, and most of the packets transmitted in the wireless channel are junk packets.

Experiment 2: Effect of DoS Attack with AAI

To show the efficiency of our AAI method, we setup three scenarios. In the first scenario wireless_sensor_1 transmits normal traffic to the server, and no malicious user launches a DoS attack. In the second scenario an attacker launches a mass-produced junk message DoS attack directly to wireless_sensor_1. In the third scenario wireless_sensor_1 uses AAI method to conceal its true ID, so the attacker can only randomly launch a mass-produced junk message DoS attack to wireless_sensor_1.

Figure 9 is the comparison of packet delivery delay in the three scenarios. It shows that the packet delivery delay of wireless_sensor_1 will decrease to less than 1 second in the situation of DoS attack if
wireless_sensor_1 use the AAI method we proposed. Figure 10 shows the comparison of throughput in the three scenarios. We can see that the throughput of wireless_sensor_1 in scenario 1 is almost the same as in scenario 3. This means that most of the packets transmitted in the wireless channel are normal, useful information transmitted by wireless_sensor_1. Using the AAI method will greatly diminish the impact of a mass-produced junk message DoS attack.

Experiment 3: Efficiency of AAI with Different Number of MTs

To evaluate the effects of the number of MTs in the same infrastructure network using our AAI method, we set up three scenarios. In the first scenario there are 4 MTs in the infrastructure network, in the second there are 8 MTs, and in the third, 12 MTs. A malicious user launches the same DoS attack in all three scenarios.

Figure 11 shows the comparison of average packets delivery delay. The number of MTs will affect the efficiency of our AAI method. We can observe that though the average packet delivery delay in all three scenarios is less than 0.7 second, the more MTs in the same infrastructure, the less significant the impact of DoS attacks to the wireless_sensor_1. However, when the number of MTs increases, the average backoff slots [14] of wireless_sensor_1 also increase, which may affect the performance of wireless_sensor_1. Figure 12 shows the comparison of average backoff slots.

Figure 9: Comparison of Packet Delivery Delay

Figure 10: Comparison of Throughput

Figure 11: Packet Delivery Delay with Number of MTs

Figure 12: Backoff Slots with Different Number of MTs
7 CONCLUSION

We propose a new DoS attack resistance method in this paper. Instead of using the true ID, the MT uses its AAI to communicate with others. The AAI reveals no information about the MT because it is disassociate with the true ID. The AAI also changes frequently from one communication session to another. All these make it difficult for a malicious user to launch a DoS attack on a specific legitimate user. Simulation results show that our AAI method greatly alleviates the effect of a DoS attack. Furthermore, the AAI method can be combined with the covert channel method to trace back to and segregate the malicious user [21].

There are many kinds of DoS attacks in wireless networks, and it is hard to design a general-for-all method. Our scheme is a step closer toward defending against DoS attacks.

8 REFERENCES


