Comparison of Online Charging Mechanisms for SIP Services

Abdelrhman Ahmed¹, Sherif Khattab², Khaled Mostafa², and Salwa El-Gamal²

¹Faculty of Computers & Information, Fayoum University, Fayoum, Egypt
²Faculty of Computers & Information, Cairo University, Cairo, Egypt
¹aas03@fayoum.edu.eg
²{s.khattab, k.mostafa, s.elgamal}@fci-cu.edu.eg

Abstract—Session Initiation Protocol (SIP) has been adopted by telecom Next Generation Networks (NGN) and 3rd Generation Partnership Project’s IP Multimedia Subsystem (IMS). Charging is one of the most fundamental components of any commercial SIP service.

Service providers have to employ an online charging mechanism to support the prepaid payment model and to create flexible pricing policies that interwork with customers in real-time, to achieve marketing targets while improving customer experience.

In this paper, we discuss the requirements of online charging mechanisms for SIP services, investigate the existing online charging mechanisms for SIP services namely AAA-based and SIP-based, and compare between them according to their security, performance, supported charging services, and supported service access types.

Our contributions are: (1) comparing between AAA-based and SIP-based online charging mechanisms, (2) checking the vulnerability of the AAA-based online charging mechanism to hacking attacks implemented by R. Zhang et al. on Vonage and AT&T VoIP service providers, (3) identifying new security attacks against the latest SIP-based online charging mechanism—SIPCoin.

Index Term—AAA, charging, micro-payment, SIP

1. INTRODUCTION

By 2012, ABI Research expects that almost half of all telecom users will be using at least one SIP service. This will generate over $150 billion in service revenue annually [1]. Charging is one of the most fundamental components of any commercial SIP service. Indeed, accurate charging will boost the trustworthiness and popularity of SIP services. Moreover, charging plays an important role in merchandising and marketing by enabling a wide variety of services such as marketing promotions.

In some places, especially the developing countries, the prepaid model is the dominant payment model [2]. According to Informa Telecoms & Media, during 2007 and into 2008, the market for prepaid mobile services has continued to grow more than twice as fast as the postpaid market. At the end of 2007, there were 2.33 billion prepaid subscriptions in the world, generated revenues of $241.9 billion. Informa Telecoms & Media predicts that by 2013 there will be 3.93 billion prepaid subscriptions, generating revenues of $382.2 billion [3].

An online charging mechanism charges customers in real-time where charging information can affect, in real-time, the service offered and therefore a direct interaction of the charging mechanism with the service control is required [4]. The telecom industry has achieved overwhelming success through commercial applications of the online charging mechanism [5].

The benefits of online charging are twofold. Firstly, service providers have to employ an online charging mechanism to support the prepaid model requirements such as: the service provider needs to check the user account balance before accepting a service request. Also the service provider needs to monitor the customer’s transactions to take the proper action when the customer account balance is about to reach zero. Secondly, online charging mechanism allows service providers to create flexible pricing policies that interwork with customers in real-time, to achieve marketing targets while improving customer experience. For example, the service provider can provide an extra option to the user for free while the session is running if the session charge reaches a certain threshold.

SIP background. SIP [6] is an application layer signaling protocol specified by the Internet Engineering Task Force (IETF) [7]. SIP is mainly designed for initiation, modification, and termination of interactive sessions.

SIP can operate with other application layer protocols including Real-Time Transport Protocol (RTP) [8] and Session Description Protocol (SDP) [9] to provide various real-time multimedia services [10], such as Voice over Internet Protocol (VoIP), multi-party conferences, Internet Protocol Television (IPTV), and instant messaging (IM).

Currently, due to its many favorable features including simplicity and extensibility, SIP has been adopted by telecom Next Generation Networks (NGN). NGN is a packet-based network able to provide services including telecommunications services. Also 3rd Generation Partnership Project (3GP) [11] adopts SIP at IP Multimedia Subsystem (IMS). IMS is an architectural framework for delivering internet protocol multimedia services.

In this paper, we discuss the requirements of online charging mechanisms for SIP services, investigate the existing online charging mechanisms namely AAA-based and SIPCoin—the latest SIP-based mechanism—and compare between them according to their security: performance; supported charging services; and supported service access types.

The AAA-based online charging mechanism is based on the Authentication, Authorization and Accounting (AAA) architecture and consists of the Diameter SIP application and the Diameter credit-control application. While SIPCoin is a peer-to-peer online charging mechanism based on SIP and the hash-chain micro-payment technique.

We can summarize the comparison of the two charging mechanisms in the following points. (1) Both AAA-based and SIPCoin mechanisms are vulnerable to attacks by untrusted intermediate parties in absence of end-to-end security, but they have their end-to-end security techniques to mitigate such attacks. (2) SIPCoin has higher efficiency.
and scalability than AAA-based mechanisms. (3) AAA-based mechanisms supports more charging services than SIPCoin. (4) AAA-based mechanism supports only subscription-based service access type, while SIPCoin supports both subscription-based and pay-per-use service access types.

The rest of this paper is organized as the following. Section 2 discusses the requirements of online charging mechanisms for SIP services. In Section 3, we investigate the AAA-based online charging mechanism. Section 4 investigates SIPCoin, the latest SIP-based online charging mechanism. A comparison between the two online charging mechanisms, according to their security, performance, supported charging services, and supported service access types can be found in Section 5. Finally, Section 6 concludes the paper.

2. REQUIREMENTS OF ONLINE CHARGING MECHANISMS FOR SIP SERVICES

In this section, we will discuss the requirements of online charging mechanisms for SIP services.

Real-time. First, Online charging mechanism must be real-time with negligible delay as SIP services are based on real-time multimedia communication [12].

Secure. Second, R. Zhang et al. presented four attacks on SIP-based VoIP systems that target either charge users for sessions established by attackers, or overcharge users for their sessions [13].

Flexible charging mechanism. Third, service providers should support flat rate, time, volume, and session-based charging [14]. For example, most VoIP providers use simple pricing models where calls over Internet are charged using flat rate and calls that terminate in Public Switched Telephone Network (PSTN) are charged in a time-based fashion, while other services such as E-Learning services may be charged in a session-based fashion.

Scalable. Fourth, scalability can be improved by eliminating the need for establishing a direct trust relationship between the customer and the service provider [14]. As the customer may use different services provided by different providers, then for more efficiency and scalability, it is recommended to use a trusted third party for the customer and the service provider rather than forcing the service provider to establish a direct trust relationship with each one of its customers.

Optimized for low-valued transactions [14]. Fifth, service providers provide access to SIP services through subscription and pay-per-use. Customers usually prefer pay-per-use to pay only for their usage and to have freedom to move from one service provider to another [15].

Pay-per-use systems require support of low-valued transactions. Credit cards or e-checks cannot be used for low-valued transactions; they add a time delay and increase the processing costs. Therefore, Micro-payment techniques are the proper choice; they are designed especially for payments of small amounts ranging from one cent to several dollars [16].

Get offer prior to session establishment. Sixth, SIP is able to provide different real-time multimedia services with different options. A customer needs to get an offer from the service provider prior to session establishment [14], and based on his account balance or a service-specific criterion approves the offer with certain options or rejects it completely.

Ability to debit as well as credit the user account. Seventh, there are services such as gaming and advertising that may credit as well as debit the user account [17].

3. AAA-BASED ONLINE CHARGING MECHANISM FOR SIP SERVICES

In this section, we summarize previous work towards building online charging mechanism for SIP services based on the Authentication, Authorization and Accounting (AAA) architecture. The AAA-based online charging mechanism consists of the Diameter SIP application and the Diameter credit-control application, in what follows we present both of them.

IETF AAA working group puts standards for authentication, authorization and accounting solutions for Internet [18]. They presented Diameter Base protocol [19] – also referred to as Diameter –, the IETF standard for AAA in IP networks and Internet [20], which is defined in RFC 3588. The 3GPP has selected Diameter to provide AAA functionalities.

Diameter is defined as a base protocol which is used in conjunction with a set of applications. It is designed as a peer-to-peer protocol and uses the client/server architecture for message exchange between Diameter nodes [21]. Diameter addresses the issues of the previously used AAA protocol, Remote Authentication Dial In User Service (RADIUS) [22].

The AAA working group presented the Authentication, Authorization and Accounting requirements for SIP in RFC 3702 [23]. Then the Diameter SIP application was presented in RFC 4740 [24]. The Diameter SIP application discusses how a SIP implementation can use the AAA architecture to support authentication, authorization and accounting. Although the Diameter SIP application supports only SIP user agents and proxies that use HTTP digest authentication [25], it can be extended to adopt other authentication mechanisms supported by SIP [1] such as IP Security (IPSec) [26], Secure Multipurpose Internet Mail Extensions (S/MIME) [27], Transport Layer Security (TLS) [28] and Datagram Transport Layer Security (DTLS) [29].

Diameter SIP application can collect charging information while the SIP session is running but cannot provide online charging. To address this issue, the Diameter Credit-Control application implements online charging for a variety of end user services [17].

3.1 Architecture model

In this subsection, we present the architecture model of the AAA-based online charging mechanism for SIP services.

As depicted in Figure 1, the startup is when the end user requests a service from the service provider. The service provider authenticates and authorizes the user through the AAA server using the Diameter SIP application. When online charging is required, the credit-control client interacts with the credit-control server using the Diameter credit-control application. Details of this interaction will be presented in the next subsection.

The credit-control server keeps track of users’ accounts. Upon request from the credit-control client, the credit-control server reserves a quota of service units from the user’s account and reports the granted service units to the user.
credit-control client. The credit-control client monitors the usage of granted service units, requests more units when the granted units are about to be exhausted and reports to the credit-control server the actual number of used units to debit from the user’s account.

End User End User

Service Provider
Credit-Control Client

Diameter SIP & Credit-Control applications

AAA Server

Diameter Credit-Control application Billing System

Credit-Control Client

Fig. 1. Architecture model for AAA-based mechanism (This figure is a modified version of Figure1 in [17], to be adapted to SIP services.)

The service fee may be calculated by the credit-control client and included in the credit request or left to the credit-control server to calculate it.

3.2 Message flow

Figure2 presents the message flow for the Diameter SIP application and Diameter credit-control application to provide authentication and authorization; advice of charge; session credit-control; and graceful service termination [17].

Authentication and Authorization. User sends a Register request to the configured SIP registrar (1). The SIP proxy sends a request using the Diameter SIP application to the AAA server to authenticate and authorize the user (2).

Advice of Charge (AoC). Caller sends a service request (3). SIP proxy detects that the caller is subscribed to AoC service, then sends a diameter credit-control request including session information to the credit-control server to rate the requested service (4). The credit-control server sends the service cost to the SIP proxy (5). The SIP proxy sends the caller an AoC web page contains the session information collected from SIP signaling and Session Description Protocol (SDP) attributes, the cost information, and a button to accept or reject the service.

Session Credit-Control. When the SIP proxy receives a service request—and after the caller accepts the service cost if he is subscribed to AoC service— and the online charging is required, the credit-control client sends a request to the credit-control server to initiate a credit-control session (6).

The session-initiation request contains session information collected from SIP signaling and SDP attributes. The credit-control server checks the user’s account balance, rates the service—if service fee has not been reported at the credit-control client request—, and reserves quota of service units from the user’s account. The granted quota is reported back to the SIP proxy (7), and then the SIP proxy forwards the service request to callee (8).

At expiry of the granted quota, the credit-control client sends an update request to the credit-control server to report the used units and request new quota (9). The credit-control server debits the used units from the user’s account, reserves new quota and reports back to the credit-control client (10).

At session termination (24 and 25), the credit-control client sends a termination request containing the used units to the credit-control server (26). The credit-control server debits the used units from the user’s account, terminates the credit-control session and acknowledges the credit-control client (27).

Graceful Service Termination. While session is running, the credit-control server debits the used units from the user’s account, and reserves a new quota, if the user’s account balance reaches zero, then the new quota is reported back to the credit-control client combined with an indicator that this is the final quota and the action to be taken after exhaust of this final quota, in this example, the action is to redirect the user to the Top-Up server to replenish his account (12).

At expiry of the final allocated quota, the credit-control client sends an update request to the credit-control server to report the used units (13), and puts the session on hold (14). The credit-control server debits the used units from the user’s account and sends to the credit-control client the validity time that represents how long does the user have to replenish his account (15).

SIP proxy starts a session between the user and the Top-Up server where user is to be asked for a credit card number and an amount of money to replenish his account (16, 17, and 18). After account replenish or at expiry of validity time, the credit-control client sends an update request to check whether the account has been replenished or not (21). If account has been replenished, the credit-control server reports back the granted units and the SIP proxy puts the session off hold (22 and 23); otherwise SIP proxy terminates the session.
Fig. 2. Simplified message flow for Diameter SIP application and Diameter credit-control application to provide online charging.

Fig. 3. SIPCoin message flow. (This figure is a simplified version of Fig. 2 in [7], simplification is only for presentation purpose)
4. SIPCoin Mechanism

In this section, we present the latest SIP-based online charging mechanism for SIP services that called SIPCoin.

The SIP-based online charging mechanism was first introduced in 2004 through an Internet draft [14] by IETF SIP Protocol Investigation (SIPPING) working group [30] to overcome some of the limitations of the AAA-based online charging mechanism, it is a peer-to-peer (P2P) micro-payment mechanism based on the Security Assertion Markup Language (SAML) [31].

SIPPING online charging mechanism employs a payment provider as a Trusted-Third-Party (TTP) for the customer and the service provider. Therefore, customer and service provider do not need to have any direct trust-relationship with each other. Also the mechanism is P2P which improves the scalability and efficiency of the mechanism.

J. Hao et al. [12] found two drawbacks in the SIPPING mechanism. First, SAML introduces complexity to payment messages and payment flow, thus increases overhead on service and payment providers and makes it difficult to meet the real-time requirement. Second, payment provider has to participate in each payment, which affects scalability as the payment provider may become a bottle-neck.

SIPCoin is a P2P online charging mechanism based on the hash-chain micro-payment technique [16]. SIPCoin [12] is introduced to address the two drawbacks of the SIPPING mechanism. SIPCoin does not use SAML and the payment provider does not have to participate in each payment, improving both efficiency and scalability and meeting the real-time requirement. SIPCoin creates P2P market for real-time communication services.

Figure 3 presents SIPCoin message flow, which can be divided into four stages: (1. Offer) the service provider presents an offer to the customer; (2. Withdrawal) the customer withdraws SIP coins from the payment provider; (3. Payment) the customer pays the service provider; and (4. Redemption) the service provider redeems the coins from the payment provider.

Stage 1: Offer. In this stage, the service provider presents an offer to the customer.

The customer sends a request to the service provider (1). The service provider sends an offer to the service provider including order identity, supported currencies, supported payment providers, service initial cost, and service-unit cost (2). For free sessions, like calls over Internet that are charged by monthly flat rate, there is no a payment message nor interaction with the payment provider.

Stage 2: Withdrawal. In this stage, the customer withdraws SIP coins from the payment provider.

The customer chooses both currency and payment provider that are supported by the service provider and sends a request to withdraw an amount of SIP coins including the order identity including the service provider's identity (3).

The payment provider verifies the service provider identity, and then generates the required SIP coins as a hash chain –using “(1)” – (4). Then the payment provider sends the customer the hash function (H_C), and both the root (W_N) and the anchor (W_0) of the hash chain. The customer regenerates the hash chain by hashing the received root (W_N) N times, and verifies the SIP coins by comparing the calculated anchor to the received one (W_0).

\[ W_i = H_C(W_{i+1}) \text{ where } i = N - 1, N - 2...0 \] (1) in [12]

Stage 3: Payment. In this stage, the customer pays the service fee to the service provider.

In order to enable the service provider to verify the customer's payment, the payment provider sends to the service provider the hash function (H_C), the anchor of the hash chain (W_0), and the customer identity (5).

The customer sends a payment to the service provider to pay the service's initial cost (6). The message includes the payment provider identity, a hash value (W_1), and its index (I) in the hash chain. For example, if initial payment is ten cents, then (W_1) is the tenth value in the hash chain.

The service provider verifies the payment by hashing (W_1), (I) times using (H_C) hash function, and comparing the calculated anchor with the one received before (W_0) from the payment provider. This verification ensures that the SIP coins can be only paid to the service provider identified at the withdrawal request.

When the paid service is about to be exhausted, the service provider requests an increment payment from the customer (7).

The customer sends a payment to the service provider to pay the service-unit cost (8). The message includes the payment provider identity, a hash value (W_X) and its index (X) in the hash chain, where 1 < X ≤ N. The service provider verifies the payment by hashing (W_X), (X-I) times using (H_C) hash function, and comparing the result with the previous received hash value from the customer (W_0).

Messages 7 and 8 can be repeated until session end. If the coins withdrawn using the message 3 have been exhausted, then the customer sends a withdrawal request to the payment provider; otherwise the service provider terminates the session.

Stage 4: Redemption. In this stage, the service provider redeems the coins from the payment provider. Periodically,
the service provider sends a redemption request to the payment provider including customer identity, order identity, hash value \( W_4 \), and its index \( X \) in the hash chain, where \( X \leq N \). The payment provider verifies the payment and transfers money into service provider's account.

5. COMPARISON OF AAA-BASED AND SIPCOIN

In this section, we compare the two online charging mechanisms namely AAA-based and SIPCoin, according to their security, performance, supported charging services, and supported service access types.

5.1 Security

In this subsection, we compare the security of the two mechanisms as summarized in Table I.

<table>
<thead>
<tr>
<th>Attacks</th>
<th>AAA-based</th>
<th>SIPCoin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Availability</td>
<td>x</td>
<td>✓</td>
</tr>
<tr>
<td>Eavesdropping</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Man-In-The-Middle (MITM)</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Replay</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>FakeBusy</td>
<td>✓</td>
<td>x</td>
</tr>
<tr>
<td>ByeDelay</td>
<td>✓</td>
<td>x</td>
</tr>
</tbody>
</table>

Security attacks. In this subsection, we describe the FakeBusy and ByeDelay attacks presented by R. Zhang et al. [13] that charge users for sessions established by attackers and overcharge users for their sessions, respectively. R. Zhang et al. presented two more attacks, InviteReplay and ByeDrop. InviteReplay attack can be applied only on SIP servers that implement the anti-replay functionality in the SIP authentication wrongly. ByeDrop attack can be applied only on SIP servers that don’t terminate sessions of no RTP activity.

FakeBusy. The attack rejects a user's service-request, and then an attacker uses this request to establish a session that will be charged to the user. The attack is launched by two MITMs, one stands between the caller and the SIP proxy, and the other stands between the callee and the SIP proxy.

The attack message flow is depicted in Figure4. SIP proxy authenticates the user’s service-request (1 – 3). MITM1 intercepts the authenticated request sent to the SIP proxy and modifies the IP address and port number of the RTP stream to its own rather than the caller ones (4). MITM1 sends a fake busy message to the caller to inform that the callee is on the phone (5). MITM2 intercepts the request sent to the callee (6) and replies with Trying (7), Ringing (9), and OK (11) messages. MITM2 includes its IP address and a chosen port number for the RTP stream in the OK message. Now the RTP stream is established between MITM1 and MITM2 while SIP signaling refers to a session between the caller and the callee. Then the caller will be wrongly charged for the session.

ByeDelay. The attack prolongs the session duration by delaying the service's termination request –Bye message–. The attack is launched by two MITMs, one stands between the caller and the SIP proxy, and the other stands between the callee and the SIP proxy. After delaying the Bye message, the two MITMs can exchange the RTP packets rather than the session intended participants.

The attack message flow is depicted in Figure5. Call is established properly (1 - 11). When any of the two participants terminates the session, for example, callee in Figure5, MITM2 intercepts the Bye message (12), and replies with an OK message to inform the callee that the session has been successfully terminated (13). MITM2 reports MITM1 with callee request, and then MITM1 sends a Bye message to the caller to inform that callee terminated the session (14). Now the two MITMs can exchange RTP packets until one of them terminates the session (16 - 19). If the SIP proxy authenticates the Bye message, MITM2 sends the intercepted Bye message sent by the callee, otherwise, MITM2 can generate the Bye message. The callee is overcharged for the session based on the time spent by the two MITMs.

Security of AAA-based mechanism. The Diameter Base protocol [19] requires that messages are secured by using IPSec or TLS. TLS is recommended for protection of inter-domain exchanges, while IPSec for protection of intra-domain exchanges where IPSec tunnels are pre-established [24], in what follows we justify these recommendations.

TLS and IPSec cannot support end-to-end security between two users that are connected through intermediate SIP proxies as intermediate SIP proxies need to parse the SIP headers to route the message properly [32]. Therefore TLS and IPSec support only hop-by-hop security by establishing a secure connection between each two adjacent entities on the path between the two end users. At call establish, time required to establish IPSec tunnels between each hop is much higher than the industry-acceptable call setup time which is 250 milliseconds [32]. Therefore IPSec is feasible where IPSec tunnels are pre-established and is impractical for dynamically allocated sessions.

When untrusted third-party relay or proxy agents are involved –relay agents route messages while proxy agents may modify messages to enforce policies– the TLS or IPSec hop-by-hop mechanism does not provide sufficient security for credit-control sessions [17].

![Fig. 5. Message flow of ByeDelay attack.](image-url)
Therefore the credit-control sessions will be vulnerable to attacks such as the following. First, eavesdropper can collect the volume of customers' transactions, and pricing plans and business volume of the credit-control servers [17]. Second, Malicious modification, injection, or deletion of service specific Attribute-Value pairs (AVPs) —service specific AVPs contain service attributes and are used to rate the service— or complete credit-control messages lead to financial consequences [17]. Third, FakeBusy attack charges customers for sessions established by attackers. Fourth, ByeDelay attack overcharges customers for their sessions.

To mitigate the previous attacks, provide end-to-end security by employing the Diameter Cryptographic Message Syntax (CMS) Security Application [33].

The diameter credit-control application mitigates application-layer replay attack by employing Session-Id defined in the Diameter Base protocol, and the credit-control request-number defined in the Diameter credit-control application [17].

Security of SIPCoin mechanism. SIPCoin employs S/MIME to provide end-to-end confidentiality, integrity, authentication, and replay protection for the communication between the customer and the payment provider. It also employs TLS to provide hop-by-hop protection for the communication between the customer and the service provider [12].

J. Hao et al. suggested the use of S/MIME to provide the confidentiality and integrity protection for the important messages of the payment message flow according to the demands [12]. But, they did not clarify which payment messages are important and which are not, in what follows we discuss the protection of all payment messages.

End-to-end protection for the communication between the customer and the payment provider protects withdrawal requests and responses so that attackers can neither steal SIP coins nor gain services from the service provider on behalf of legitimate customers [12].

Hop-by-hop protection for the communication between the customer and the service provider does not provide sufficient protection leading to that untrusted intermediate proxies can run attacks such as the following. First, Collection of the volume of customers' transactions, and pricing plans and business volume of the payment providers by eavesdropping offer, initial payment, and increment payment messages. Second, Reduction of the service provider's revenue by redirecting its customers to another colluding service provider rather than the intended one. This can be done by replying a service request with a fake response "301 moved permanently" or "302 moved temporarily" combined with the colluding service provider address, and then the customer sends a new request to the colluding service provider. Third, dropping initial or increment payment messages leading to that the service provider terminates the service. Fourth, modification of SIP coins in initial or increment payment messages, but when the service provider verifies the payment, it will reject the payment and terminate the service. Fifth, Increasing initial or unit costs in the service provider's offer overcharges the customer. The service provider can resolve the attack by refunding extra payment to the customer but the customer will not trust the service provider any more.

SIPCoin is neither vulnerable to FakeBusy nor ByeDelay as payment is handled directly with the caller. For FakeBusy attack, caller pays only an initial payment and service provider will refund it as session has not been established because callee is on the phone. For ByeDelay attack, caller does not pay any more increment payment once session has been terminated by caller or callee.

End-to-end security for all payment messages exchanged between the customer and the service provider can mitigate the previous attacks. Also the communication between the service provider and the payment provider must employ end-to-end security to protect withdrawal response and redemption request messages.

5.2 Performance

In this subsection, we compare the performance of the two mechanisms.

Efficiency. SIPCoin is more efficient than AAA-based mechanism in terms of processing and network traffic. In AAA-based mechanism, the credit-control server must participate in each payment. While in SIPCoin, the payment provider does not have to participate at each payment. This not only reduces the number of payment messages between the customer and the payment provider, but also reduces the SIPCoin's usage frequency of S/MIME.

Scalability. SIPCoin is more scalable than AAA-based mechanism in terms of the architecture model and the management of trust relationship between involved parties. Since the customer may use different services provided by different providers, in AAA-based mechanism, each service provider must authenticate and authorize its customers against the AAA server. While in SIPCoin, no need to have a direct trust-relationship between the service provider and each one of its customers; SIPCoin uses a trusted third party for the customer and the service provider.

In AAA-based mechanism, the service fee is calculated by the credit-control client, or service- specific parameters are passed to the credit-control server to calculate the fee. While in SIPCoin, the service fee is determined by the service provider without the need to contact another entity. Finally, SIPCoin is based on the scalable peer-to-peer architecture, while AAA-based mechanism is based on the client-server architecture.

5.3 Supported charging services

In this subsection, we compare the two mechanisms according to their support of charging services as summarized in Table 2.

Before comparison, we define two charging services. Firstly, direct debit. There are certain services that always succeed but may encounter long delay between the service invocation and the actual service delivery. For more efficiency, such services can be charged by direct debiting at service delivery rather than establishing a credit-control session from the service invocation until the service delivery [17]. Instant messaging is an example for such services, there may be long delay between sending a message and its retrieval by the recipient. Secondly, refund. Service provider may refund service-units or monetary amount to the customer's account; for example, the customer pays for the service but the service provider fails to deliver the service.
TABLE II
SUPPORTED CHARGING SERVICES BY AAA-BASED AND SIPCOIN
MECHANISMS

<table>
<thead>
<tr>
<th>Charging Services</th>
<th>AAA-based</th>
<th>SIPCoin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit Reservation</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Initial &amp; Increment Payments</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Balance Check</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Direct Debit</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Refund</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Graceful Service Termination</td>
<td>√</td>
<td>√</td>
</tr>
</tbody>
</table>

5.4 Supported service access types

AAA-based mechanism supports only subscription-based service access type, while SIPCoin supports both subscription-based and pay-per-use service access types then it appeals to a wider range of customers.

6. CONCLUSION

In this paper, we discuss the requirements of online charging mechanisms for SIP services, investigate the existing online charging mechanisms for SIP services namely AAA-based and SIPCoin, and compare between them according to their security; performance; supported charging services; and supported service access types.

We can summarize the comparison of the two charging mechanisms in the following points. (1) Both AAA-based and SIPCoin mechanisms are vulnerable to attacks by untrusted intermediate parties in absence of end-to-end security, but they have their end-to-end security techniques to mitigate such attacks. (2) SIPCoin has higher efficiency and scalability than AAA-based mechanism. (3) AAA-based mechanism supports more charging services than SIPCoin. SIPCoin needs to incorporate services such as, balance check, direct debit, refund, and graceful service termination. (4) AAA-based mechanism supports only subscription-based service access type, while SIPCoin supports both subscription-based and pay-per-use service access types.

Future research will evaluate SIPCoin’s effect on SIP performance, and incorporate missing charging services into SIPCoin.

REFERENCES


