高度交通システム 7−5
（２００１．１１．１６）

Proposal and Design of Secure Protocols for ITS Reservation and Downloading Services

清本晋作[*]　田中俊昭[*]　中尾康二[*]　児島史秀[†]　佐藤勝善[†]　藤瀬雅行[†]

[*]KDDI 研究所
〒356-8502 埼玉県上福岡市大原 2-1-15
[†]通信総合研究所
〒239-0847 神奈川県横須賀市光の丘 3-4

Tel: 049-278-7885
E-mail: sh-kiyomoto@kddi.com

あらまし　近年、ITS における主要な通信技術であるＤＳＲＣについて、その特徴を生かしたアプリケーション構築に関する研究が進められてきている。今回、我々は、そのようなアプリケーションの１つである「マルチメディアコンテンツの予約・配信サービス（ＲＤＳ）」におけるセキュリティアーキテクチャについて検討を行ったので報告する。特に、利用者とＲＤＳとの認証処理に着目し、簡易でかつ安全性の保証された認証プロトコルの設計を行なった。

Proposal and Design of Secure Protocols for ITS Reservation and Downloading Services

Shinsaku KIYOMOTO*,　Toshiaki TANAKA*,　Koji NAKAO*,

Fumihide KOJIMA[†],　Katsuyoshi SATO[†],　Masayuki FUJISE[†]

*KDDI R&D Laboratories Inc.
2-1-15 Ohara Kamihukuoka-shi Saitama 356-8502, Japan
[†]Communication Research Laboratory
3-4 Hikari-no-oka Yokosuka-shi Kanagawa 239-0847, Japan

Tel：+81 492 78 7885
E-mail：sh-kiyomoto@kddi.com

Abstract　Since DSRC is one of the promising mobile platforms in ITS, several applications have been recently studied and discussed on top of the DSRC. In this paper, we propose and design security protocols for ITS Reservation and Downloading services in DSRC. In our proposed concept, a strict authentication is carried out in the reservation protocols, RDS issues a reservation ticket as an evidence of authentication, and a simple authentication based on the shared ticket is performed in the downloading protocol. These authentication protocols are lightweight, because authentication mechanism is mainly composed of hash functions. Our mechanism would also be scalable with respect to the number of users, because reservation server is not necessary to manage users' information such as user IDs or user keys.

1. Introduction

ITS (Intelligent Transportation Systems) is an innovating infrastructure towards the realization of automatic driving such as AHS service [1]. Furthermore, it provides users with a great benefit by supporting various kinds of fruitful and attractive services such as ETC (Electronic Toll Collection), Information services, E-shopping services or Reservation services.

Since DSRC [2][3][4] (Dedicated Short Range Communications) is one of the promising mobile platforms in ITS, several applications have been recently studied and discussed on top of the DSRC[5][6][7]. In this paper, we are focusing on a promising application, which provides content Reservation and its Downloading System, called RDS in short, on the top of DSRC. More specifically, a reservation to request downloading in bulk from user will take place at his home or office, or at the first stage of car drive. While driving his vehicle, the requested content will be distributed to a nearest spool server, which is requested to be a distributor of the content to the user. When he arrives his point for getting his requested content, the spool server will download the content to his vehicle terminal through DSRC with high performance.

Under the above circumstance, some security issues, such as peer entities authentication between a user and the RDS system, confidentiality of user's sensitive information, key managements on RDS and non-repudiation of a reservation request could be identified as important study issues to work out solutions for RDS realization. In this paper, we specifically concentrate on the authentication and the non-repudiation security functions, and propose a new secure protocol using ticket to realize secure RDS under DSRC. The ticket is to guarantee evidences of the actions for reservation and to provide efficient authentication.

2. Introduction of RDS

The RDS is basically composed of the following elements. 1) User Terminal, 2)Media Server, 3)Reservation Server, 4)Spool Server, 5)Base Stations and 6)Vehicle Terminal as shown in Figure 1. Among the above elements, the RDS is realized by the 2) 3) 4) and 5).

Application scenario of the RDS is designed as follows:

a) A user accesses to the Reservation Server by his home computer (User Terminal) to reserve downloading a content, such as movies, music, electronic books, to his Vehicle Terminal. At this reservation, the user registers location, time, and content to the Reservation Server for his downloading;

b) The reservation server will ask a Spool Server to prepare the requested content for the downloading to the user vehicle. The Spool Server is selected to locate at the nearest place to the registered location from the user;

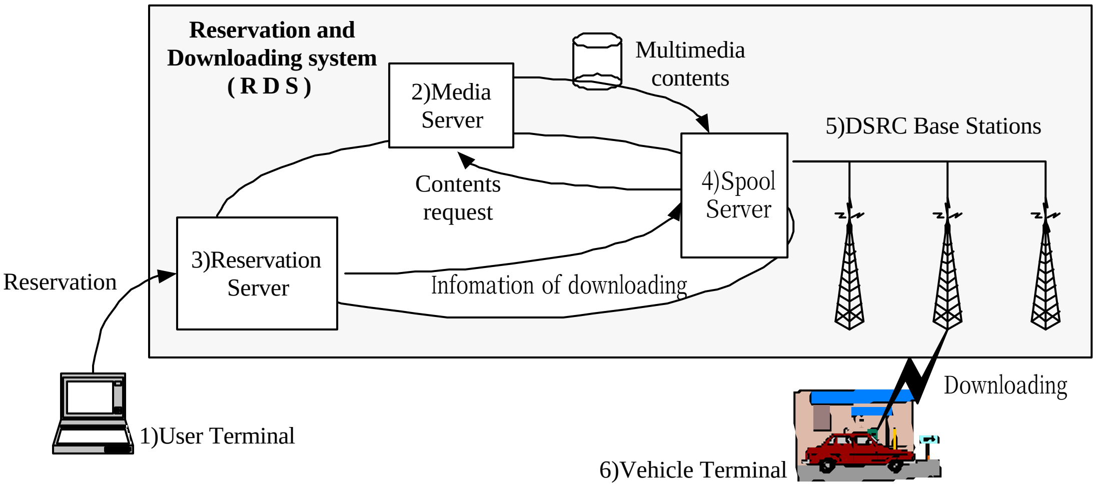

Fig. 1. Reservation and Downloading Services (RDS)

c) If the Spool Server has already stored the requested content, it is not necessary for the Spool to get the content via a Media Server (contents server). However, if the requested content does not exist in the spool, the content should be transferred from the appropriate Media Server when requested;
d) When the user's Vehicle arrives at the location registered, the content is downloaded to the user's Vehicle Terminal with high performance from the Spool.

The above scenario is the basic procedures, which contains the following characteristics recognized in the RDS:
1) Since RDS is operated in the huge network, it be constructed by several Spool Server, Media Server and Reservation Server;
2) Depending on traffic, the user vehicle does not always reach to the registered location at the registered time. Therefore, the RDS has to take into account adaptation of the downloading point dependent of the user location in an flexible manner;
3) Downloading the content should be basically charged to the user who makes the reservation in this RDS. The charge takes place after an acknowledgement of the successful completion of download;
4) In the case of reservation, it is not necessary to do it from his vehicle, but from his personal handy-phone or personal computer;
5) In the normal case, the user, who made the reservation, will get the content at the registered location and time. However, it is also considered in the RDS that the user can pass the right to get the content to another person. So that the person, who was given the downloading right by the initial user, will get the content on behalf of the initial user.

3. Security requirements for ITS reservation and downloading system

Based on the above characteristics of the RDS, the following security requirements can be identified:

1) Entity Authentication:
Reservation by unauthorized users should be protected. Furthermore, content download by unauthorized users should also be protected.
2) Non-repudiation:
Evidences of downloading should be securely proved and managed by the RDS.
3) Inner Security of RDS:
User information related to the reservation, user preference and access logs should be securely managed and handled by the RDS.
4) Network Confidentiality:
Information exchanged between users and RDS in both reservation and downloading phases over the network (Mobile and DSRC) should be protected against eavesdrop.

In this paper hereafter, we will concentrate on the above three security requirements 1), 2) and 4). As for 3), at this point, the security issues within RDS are left for the later considerations when we implement the proto-type system of RDS. In the following sections, we propose secure protocols to satisfy the above three requirements.

4. Details of our proposal

4.1 Basic Concept

On an assumption that the RDS is trusted, we propose a ticket-based reservation and downloading protocols to satisfy the above requirements as shown in Figure 2. As the basic concept, a ticket proposed in this paper is used to guarantee the facts that the reservation is successfully performed and the downloading is perfectly completed for non-repudiation. The ticket is also used for peer entity authentication between the user and the spool just before downloading, and is further used to share a secret key for confidentiality. Outline of the proposed secure protocols is as follows.

(1) Authentication is firstly performed a user terminal and the RDS in conjunction with generation of a shared communication key. After that, the user terminal sends his privilege to the RDS.
(2) Information, exchanged between the user terminal and the RDS, is encrypted by the shared communication key. A list of permitted contents is arranged by the RDS according to the user's privilege, and it is sent back to the user terminal. The user selects one of the contents from the list and reserves contents by specifying the downloading names of content (e.g. movies titles), time and

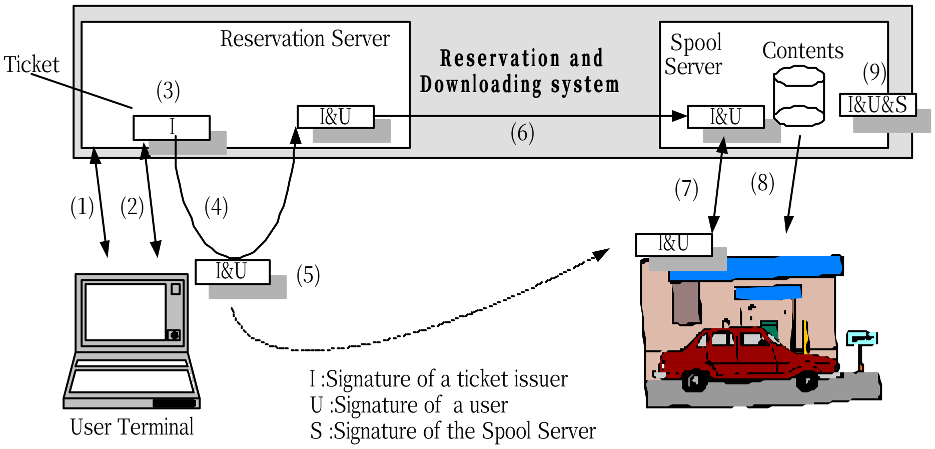

Fig. 2. Ticket-based reservation and downloading protocols

location.

(3) The RDS issues a reservation ticket that contains the above information and is digitally signed by the secret key of the RDS.

(4) The RDS sends it to the user terminal.

(5) If the ticket is correctly verified by the user terminal, then it is digitally signed by the user, and he replies its copy to the RDS.

(6) At the downloading phase, the ticket is sent to a Spool Server specified by the user.

(7) On the other hand, the user terminal stores the ticket in a tamper-resistant module such as a smart card, and the user sets the temper-resistant module into vehicle terminal. Consequently, the ticket is shared by both the Spool Server and vehicle terminal.

(8) Before downloading, the ticket stored at the vehicle terminal is compared with that stored at the RDS for the purpose of the peer authentication. If the authentication is successfully done, the downloading is started.

(9) At the downloading, the content is encrypted by the communication key derived from the shared ticket. When the downloading is finished, the Spool Server signs on the ticket digitally to show that the ticket is used. The used ticket is stored by the RDS as an evidence of downloading.

4.2 Detail protocol of reservation

In advance, a user executes a registration for the Reservation Server, and gets a pre-shared key Pa. The pre-shared key Pa is derived from a master key P and the user's identification number IDa. The master key P is securely stored in the server.

$$Pa = g\ (P \mid IDa)$$

Where g is a one-way hash function, and g (P | IDa) indicates a hash result of P and IDa.

Detail protocol of reservation is the followings

STEP1: The user terminal generates a random number R.

STEP2: The user terminal sends R and IDa to the Reservation Server.

STEP3: The user terminal computes a pre-communication key Kp.

$$Kp = f_{Pa}\,(IDa \mid R)$$

Where f is keyed hash function, and f_{Pa} (IDa | R) indicates a keyed hash IDa and R by key Pa.

STEP4: The server computes the pre-shared key Pa from P and IDa.

STEP5: The server also computes the pre-communication key Kp in the same manner.

STEP6: The server generates a random number R'.

STEP7: The server encrypts R and R' by Kp, and send it to the user terminal.

STEP8: The user terminal decrypts R and R' by Kp, and verify that R is correct.
STEP9: The user terminal computes a communication key Ka.

$$Ka = f_{Kp}(IDa \mid R')$$

STEP10: The user terminal encrypts R' and an attribute certificate by Ka, and sends it to the server.

Where, an attribute certificate is an extended X.509 certificate, which contains a user's privilege.

STEP11: The server computes the communication key Ka in the same manner.
STEP12: The server decrypts R' and the attribute certificate by Ka, and verifies R' is correct. The server also verifies the attribute certificate and extracts the user's privilege from it.

The following all communication are encrypted by Ka.

STEP13: The server creates a list of permitted contents according to the user's privilege, and sends it to the terminal.
STEP14: The user terminal issues a reservation request.
STEP15: The server creates a reservation ticket based on the reservation request.
STEP16: The ticket is digitally signed by the server's secret key, and sent to the terminal.
STEP17: The user terminal verifies the ticket, and the ticket is signed by the user's secret key, only if the verification is successfully done.
STEP18: The user terminal replies the signed ticket to the server, and stored its copy.
STEP19: The server verifies the ticket.

4.3 Detail protocol of downloading

We design such that downloading protocol is simple from the following viewpoints.
-There is no key sharing protocol at downloading phase, because Vehicle terminal and Spool server authenticate each other explicitly by using reservation ticket, as a shared key, that is securely generated and stored on both sides in advance. This security holds under the assumption that RDS is within one security domain and Reservation Server securely transfers reservation ticket to Spool Server.
-Confidentiality of contents is assured implicitly without key agreement protocol at the downloading phase, because encryption key derived from reservation ticket that is securely generated and stored on both sides.

Detail protocol of reservation is the followings.

STEP1: The Spool Server tries to connect with user's Vehicle Terminal when the reservation time has come.
STEP2: The vehicle terminal generates a random number R''.
STEP3: The vehicle terminal sends the ticket number n and R'' to the spool server.
STEP4: The spool server searches the ticket number n in his stored tickets. If he cannot find it, he sends an error message to the vehicle terminal and stops the protocol
STEP5: Otherwise, the server computes a hash result of the ticket Tn.

$$Ht = g(Tn)$$

STEP6: The server computes a communication key Kt from n and R''.

$$Kt = f_{Ht}(n \mid R'')$$

STEP7: The server encrypts R'' by Kt, and sends it to the terminal.
STEP8: The vehicle terminal computes the communication key Kt in the same manner.
STEP9: The vehicle terminal decrypt R'', and verifies R'' is correct.
STEP10: The vehicle terminal encrypts ticket Tn by Kt, and sends it to the server.
STEP11: The server decrypts the ticket by Kt, and conforms it is same as the ticket storing in the server.

The following all communication are encrypted by Kt.

STEP12: The server starts sending Multimedia Contents.
STEP13: When the downloading is finished, the vehicle terminal deletes used ticket.
STEP14: To show that the ticket is used, it is digitally signed by server's secret key.

4.4 Ticket format

We show a reservation ticket format in Figure 3.

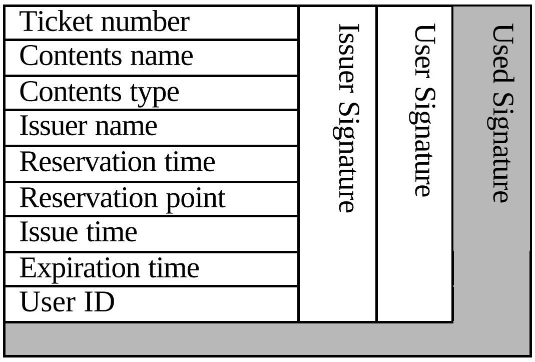

Fig. 3. Reservation ticket format

Where, Ticket number is initial number uniquely assigned, Issuer name is the name of the Reservation server which plays the role of ticket issuer, Issue time is the time when the ticket is issued, and Expiration time is a limited time that ticket is valid. Contents name, Contents type are information with respect to reserved content, Reservation time is the time when the content is downloaded, and Reservation point is the location where the content is downloaded.

Issuer Signature and User Signature is information digitally signed by Issuer's secret key and User's secret key respectively, where the message to be singed contains text from Ticket number to User ID as in Figure 3. On the other hand, Used Signature is the information digitally signed by the Spool Server, where the message to be singed contains all part of the format except for the Used signature..

In our protocol, the used ticket is created only if both the reservation process and the downloading process are successfully performed. Therefore, downloaded request can be cancelled, under the situation that the reservation process is completed and the downloaded process in not finished. This advantage is very useful for ITS application that the downloading is canceled or failed according to traffic conditions.

5. Conclusion

In this paper, we propose and design security considerations for RDS service, especially focusing on practical and efficient reservation and downloading protocols. In our proposed concept, a strict authentication is carried out in the reservation protocols, RDS issues a reservation ticket as an evidence of authentication, and a simple authentication based on the shared ticket is performed in the downloading protocol. These authentication protocols are lightweight, because authentication mechanism is mainly composed of hash functions. We also proposed canceling mechanism of reservation transaction. With the result, our protocol is tolerated against failure of contents downloading because of traffic conditions. Our mechanism would also be scalable with respect to the number of users, because spool server is not necessary to manage users' information such as user IDs or user keys.

As for our future research, we will study an encryption method in the contents downloading. We will have to study faster encryption method customized to TCP/IP translations on DSRC environment.

References

[1]Y.Kasuga et. al., "Development of AHS Key Technologies in Japan", 6th ITS World Congress 1999.

[2]prTR2/204/15, DSRC Application Layer, ISO TC204 WG15 Document, 1999.

[3]ITS Team, "A Study on the Development of Dedicated Short Range Communication System for ITS", ETRI report, 1999.

[4]Iwata et. al., "DSRC Communication System", 5th ITS World Congress, 1998.

[5]H.Cho et.al., "Bus Information Service and Internet Service Plan using Active DSRC System", 7th ITS World Congress, 2000.

[6]Y.Hasegawa et. al., "An Application Study of The ETC/DSRC Technologies", 7th ITS World Congress, 2000.

[7]K.Tokuda et. al., "Activity on ROF-RVC Working Group of ITS Joint Research Group at YRP", The 1st workshop on ITS Telecommunications, 2000.