Secure and Seamless Payment for Wireless Mesh Networks

Albert Levi, Serhat Can Leloglu

Sabanci University, Turkey

levi@sabanciuniv.edu

canleloglu@sabanciuniv.edu

Abstract— **Wireless Mesh Network (WMN) is a multi-hop high-speed networking technology for broadband network access. Compared to conventional service providing systems it is fairly cost-effective and easy to deploy. WMNs are often used for service providing since they provide service to mobile and immobile clients. In this paper a secure and seamless pre-payment system is proposed moreover network simulations for this system are presented.**

Keywords—**Wireless** Mesh Networks, Cryptography, Payment Systems, Security, Network Simulation

1. Introduction

Wireless Mesh Networks [1] are often used for service providing; moreover a secure system built using WMNs should support user identification, authentication as well as authorization and accounting.

Commonly payment systems service providers do not fully trust clients, but in reality service providers –intentionally or not- may over charge the clients or charge for services that they did not provide. It is proven that using native cryptographic algorithms, every action could have an undeniable cryptographic proof so that the client could not get service without payment and service providers could not charge without serving.

The secure and seamless pre-payment system presented in this paper, has the properties such as wide-coverage, seamless mobility and roaming, anonymity, mutual authentication, two-way honesty, preventing double spending and unlinkability. Ten protocols are designed for actions of the system entities, and these protocols are tested using network simulator 3 [2]. The designed system had formidable results in unit tests and the results are explained in this paper too.

1. Cryptographic Notes

The designed protocols are formed by the usage of some cryptographic primitives such as public key cryptosystems and hash functions forms up the designed protocol. 2048-bit RSA [3] is employed for public key encryption-decryption and signature purposes. AES-128 [4] is utilized for symmetric key cryptography and SHA-256 [4, 5] is used as a hash algorithm in the system. HMAC [5, 6] algorithm is used for challenge-response protocols.

1. General Overview of Proposed Scheme and System Entities

The proposed system is a secure pre-payment infrastructure for WMNs that also considers users' privacy and fairness. In this infrastructure there are mobile phones or laptops as clients, as well as tools that are used for service providing. Table 1 gives a list of system entities that function in the proposed system.

TABLE I  
System Entities

|  |  |
| --- | --- |
| Description: D:\My Documents\albert\tt proje\phone.png | Mobile user (client) |
| Description: D:\My Documents\albert\tt proje\accessPoint.png | Access Point (AP). From now on in this document, it is called as AP, but please note that it also has routing capability. |
| Description: D:\My Documents\albert\tt proje\cloudWithoutDots.png | Mesh backbone |
| Description: D:\My Documents\albert\tt proje\gateway.png | Gateway (GW) that connects the mesh backbone to outer world and also to the operator's server |
| Description: D:\My Documents\albert\tt proje\operator.png | Operator's server (OP). Keeps necessary logs and user info. |
| Description: D:\My Documents\albert\tt proje\trustedThirdParty.png | Trusted Third Party (TTP). Payment related logs are mostly to be generated by the TTP. |

Figure 1 shows the topology of the network and connections between entities.



Figure 1. Network Topology

Figure 1 shows the topology of the network and connections between entities.

Connection between serving access points is wireless, and they use 802.11s protocol [6]. The mesh backbone emulates a cloud from the mobile user’s perspective. It is a black box; which receives packets from mobile user and delivers them to the gateway in a multi-hop manner. Mesh backbone uses Hybrid Wireless Mesh Protocol (HWMP) [7], which is a hybrid routing protocol, which has routing tables.

Connection medium between mesh backbone and gateway (GW) is wireless. GWs and operators communicate through wired connection. The connection between an operator and TTP is also wired. These connections use 802.3(Ethernet protocol).

1. Connection Card Structure

A Connection Card (CC) is the main deed that clients buy from operators and use to get Internet service. CCs include credits as tokens. Hash tokens are generated using hash chains as discussed below. CCs also have unique Serial Numbers (), which are to be used for alias computation later.

Tokens for getting Internet service are basically links in a hash chain. For each set of tokens, the operator picks on a random Initialization Vector (IV) and takes hashes of it several times. The number of hash operations is actually the number of tokens in a set.

…

The first link in the chain, , is the first token to be used, and then tokens are used in increasing order by token index. In this manner, one-way property of hash algorithms is exploited such that an attacker cannot learn the next token even if she knows the previous tokens.

1. Alias Computation

Aliases are temporary identifiers for clients. They change frequently using a secure protocol. Anonymity is achieved by changing aliases as previously stated way however it is durable to some extent.

The serial number (SN) of the CC, which is bought from an operator, will be used as a base for client’s aliases. An alias will be computed by performing the following operations:

1. Client will pick a random 128-bit unsigned number and call it his nonce .
2. Perform XOR operation with and his nonce,
3. Client will use this alias whenever his identity is required.

One may argue that this kind of alias computation would run a risk of producing same alias for several users. However making TTP to check the proposed alias to be a unique one solves this problem.

1. Protocols

There exist ten protocols to make the system work. These protocols define packet transfers and routes. Cryptographic primitives and the way they are used are also explained in the protocol designs.

Some protocols show similarity e.g. *Initial Authorization* and *Reuse of a Connection Card*. The only difference between these two protocols is their hash token index. *Initial Authorization* uses the very first hash token while *Reuse of a Connection* Card using the other hash tokens on the hash chain. This kind of similar protocols will be explained simultaneously.

* 1. End-to-End Two-Way Protocols

The main protocol in the system is the End-to-End Two-way protocols, which are also the most common ones in the system.

The protocols classified as End-to-End Two-way are *Initial Authorization, Reuse of a Connection Card, Disconnection, Change Alias* protocols. These protocols transmit equally sized packets from client to TTP. TTP executes the same cryptographic operations on the packet and forwards the packet to the client. In these protocols client performs an encryption over a 384-bit packet using RSA-2048 and sends it to the TTP. TTP decrypts this cipher using RSA-2048 private key then signs 256-bit data using RSA-2048 private key. TTP sends this signed data to client through mesh backbone. Every station receiving the encrypted and signed packet; verifies the signature and forwards the original packet until it reaches the destination.



Figure 2. End-to-End Two Way Protocol Flow

*Initial Authorization* is the first protocol that a client uses in the system in order to get authorized. It is used only once by a particular user.

*Reuse of a Connection Card* protocol is used when a user does not finish the tokens in a connection card and would like to use the remaining tokens at a later time. *Initial Authorization* and *Reuse of a Connection Card* protocols only differ in their hash token index.

The initial time of the session for a user is stored when a user performs one of the two previously mentioned two protocols. Disconnection protocol yields the ending time of the session. In this way, the TTP learns the amount of time that the user got served. This information is used for settlement purposes.

One of the privacy preserving features of the proposed system is that access points ask every user to change their aliases from time to time. When received such a command from the access point, clients compute aliases and send it to the TTP for signature. The overall process is called *Change Alias* protocol.

* 1. Access Point Authentication



Figure 3. Access Point Authentication

*Access Point Authentication*, which is shown in Figure 3, takes place between a mobile client and an access point. It is a challenge-response type of protocol to authenticate the access point to the client.

*Access Point Authentication* starts with the serving access point by sending a request to the client. Client sends a 128-bit challenge to the access point. The access point performs an HMAC operation on this challenge using the last hash token as a key. The client performs the same operation and compares two results. If they match, the access point is verified as authenticated.

* 1. Packet Transfer



Figure 4. Packet Transfer

*Packet Transfer* protocol, shown in Figure 4, protocol is the simplest and the most commonly used protocol among others. It is the main service access protocol that uses tokens one by one. One token of the hash chain is sent from client to AP and the client starts to use the broadband access service. Usage is charged in time basis. Every five minutes client sends a new hash token to continue to get Internet service. When a user sends a hash token it means that she already has paid for the service and in case of disconnection, the protocol is called after e.g. 2 minutes, user could not get a refund for the remaining 3 minutes.

The time measurement happens between access point and client. The access point does decrementing from 5 minutes. If client tries to get service after 5 minutes, access point sends a request to client to make her to send a new hash token.

* 1. Update Packets



Figure 5. Update Packets

*Update Packets* protocol, shown in Figure 5, is used in case of an unexpected behaviour in network. If a client drops out of the network, operators and TTP needs to be informed that this client is not active anymore. In order to handle this unexpected behaviour, the access points periodically update operators using *Update Packets* protocol.

In this protocol, client sends concatenation of 128-bit alias and 128-bit hash token to the operator. Operators update TTP in case of a drop. This protocol is a one way end-to-end protocol.

* 1. Seamless Mobility and Roaming (Payment Related)



Figure 6. Seamless Mobility and Roaming

*Seamless Mobility* and *Roaming* protocols, shown in Figure 6, are run whenever the client changes the serving access point. The running protocol is called *Seamless Mobility* if the new access point belongs to the same operator as the previous access point. If the operators differ, then the protocol is called *Seamless Roaming.*

In these two protocols client sends a 384-bit request packet to the old access point. The old access point receives this packet and performs an encryption on it using RSA-2048, than signs this cipher text using RSA-2048 private key. The old access point sends this packet to client and the client relays it to the new access point. New access point decrypts the packet using RSA-2048 private key and verifies the signature using RSA-2048 public key.

Finally the new access point and the client run a *Challenge-Response Protocol* to authenticate the new access point.

If the running protocol is *Seamless Roaming*, then receiving break-off request from the client triggers the old access point to send a disconnection request to the TTP. This part of the protocol is not implemented in the unit test because it runs in background.

* 1. Distributing Access Point Public Keys



Figure 7. Distributing Access Point Public Keys

A public key distribution mechanism is placed within the system in order to achieve *Seamless Mobility* in home operator and also to support *Seamless Roaming*.

In Figure 7, a generic model for public key distribution is shown. This protocol has two parts; one is certificate generation for the access point public keys, second is distribution of the public keys. The part between operator and the TTP is offline; it runs during the set-up, before the deployment of the access points in the field.

1. Unit Test Results

Unit tests cover protocol behaviours under low pressure. In these tests there is only one user, and this user performs the same protocol every minute. These tests are done to ensure that modules of the system are fit for use.

As discussed earlier some protocols show similarity considering packet sizes, cryptographic operations and packet routes. Since there would be no difference between unit tests of protocols that are in the same group, there is one result chart for a particular group of protocols.

* 1. Unit Test Result for End-to-End Two-Way Protocols

Unit tests for end-to-end two-way protocols consist of a user, running the same protocol every minute. Charts present the average delay of packet delivery over time. In this simulation the user sends the packet to a serving access point and the packet hops 2 times in the mesh backbone until it reaches the gateway. Gateway forwards the packet to operator and operator transmits the packet to TTP. TTP processes this packet and sends it back to the client through the same route.

Figure 8 gives the result for unit test of end-to-end two-way protocols.



Figure 8. End-to-End Two-Way protocols Unit Test Result

As shown in Figure 8, there is a delay that shows variation around 0.04 second. This unstable behaviour is caused by different initial packet delays. System needs some packets to set up paths between mesh nodes. The performance stabilizes in time. Average delay shows a peak by the end however the difference between highest and lowest values of the results is inconsiderable.

* 1. Unit Test Result for Access Point Authentication

*Access Point Authentication* protocol consists of a challenge-response protocol. It contains two HMAC operations.

Unit test for this protocol contains a user, trying to run access point authentication protocol with a serving access point every minute. The resulting chart, presented on Figure 9, shows the average delay of the protocol versus time.



Figure 9. Access Point Authentication protocol Unit Test Result

As shown in Figure 9, average delay of access point authentication converges to 0.05 second in the steady state. The initial delay values are higher than the later ones, because nodes need some time to establish and see who is around. At the time of initial deployment, wireless nodes send and receive beacons and perform operations using them.

* 1. Unit Test Result for Seamless Mobility and Roaming

*Seamless Mobility* and *Seamless Roaming* protocols have the same behaviour since client sends and receives same length of packets. Thus, they are grouped together for unit tests.

Unit test for *Seamless Mobility* and *Seamless Roaming* protocols consists of a client changes serving access point every minute. Client is located in between two access points and these access points are both eligible for service. Since these protocols must be seamless to the user it is important to get reasonable delays for these protocols.

Figure 10 presents the unit test result for *Seamless Mobility* and Roaming protocols.



Figure 10. Seamless Mobility and Roaming protocols Unit Test Result

In unit test for these protocols, a 0.15 second of network delay for access point change is observed. Similar to other protocols, there is a transitive period at the beginning of the simulations, however it reaches steady state in time and gains balance.

* 1. Unit Test Result for Packet Transfer

*Packet Transfer* is the mostly used protocol in the system. It is crucial to have small amount of network delay for this protocol because of it’s often use. Unit test scenario of *Packet Transfer* protocol is that a client sends a 512-byte packet every minute.

Figure 11 shows the unit test result for Packet Transfer protocol.



Figure 11. Packet Transfer protocol Unit Test Result

Unit test gave a higher average delay value at the early parts of the simulation but expectedly it reaches a balance through time. As seen on Figure 11, at steady state, packets are received in a very short amount of time, which is around 0.0002 second.

* 1. Unit Test Result for Update Packets

*Update Packets* protocol takes place between AP and TTP. In this simulation access point updates the user info stored at operator. Figure 12 shows the average delay of *Update Packets* protocol over time.



Figure 12. Update Packets protocol Unit Test Result

In the simulation scenario, APs update operator once in every second. Our simulation showed that there is a 0.02 second maximum network delay for updating operator for the client usage.

1. User Modelling And Mobility

The proposed system intends to serve a variety of users (a.k.a. network clients). Network clients differ in their network usage frequency with respect to time of day, their mobility patterns and frequency of usage.

Certain kinds of actions are defined, such as authorization (initial or reuse of a connection card), disconnection, packet transfer (network usage), payment related roaming and payment related AP handover. All of these actions are triggered as a result of a random event. Connection and network usage related actions are triggered according to a two-state Markov Chain model [8]. Roaming and handoff related actions are triggered by user mobility.

* 1. User Actions

In real-life scenario simulations, network usage related actions are modelled using two-state Markov Chain as shown in Figure 13. There are two states that a user could be in: *Connected* and *Not Connected*. State transitions or staying in the same state triggers some actions as described below.



Figure 13. State Diagram of Clients

The initial state is *Not Connected*. In this state, the user switches to *Connected* state with the probability value of . This state transition triggers *Initial Authorization* (if the CC is used for the first time) or *Reuse of a Connection Card* protocol (if the connection has been used before). In this way, the user starts consuming the network and getting the service. While in *Not Connected* state, the user stays in the same state with probability value of .

While in *Connected* state, the user remains connected (i.e. stay in the same state) with the probability of . Staying connected triggers *Packet Transfer* protocol. In other words, the user continues to get service via the currently connected AP. In *Connected* state, transition to *Not Connected* state occurs with probability of. This transition disconnects the user via *Disconnection* protocol.

In this 2-state Markov chain model, the average connection duration, , is calculated as the expected value of staying in *Connected* state, as given below.

(1)

Where, denotes .

The expected value of staying in *Not Connected* state is the average idle time for a user between two connections. This value, , is calculated as follows.

(2)

Where, denotes .

* 1. Client Types

Three different user types are outlined with different networking and mobility requirements. Considering whether they are working, studying or domestic provides the differentiation among user types.

The network usage within one day has been modelled in three time slots: (i) night (00:00 – 07:59), (ii) daytime (08:00 – 15:59), and (iii) evening (16:00 – 23:59).

User types are described as follows:

* **Students:** This kind of clients uses network services mostly in the evening when they return back from school. Their possibility to use network services during morning and night is relatively small comparing to mid-day time. Thus, the probabilities for being active are higher for evening. Students are assumed to be mobile at the beginning and end of the *daytime* slot since they go to their school. Until the end of the *night* slot, students would more likely to get service in their homes in an immobile way.
* **Employees:** This kind of clients has routine lives. They are immobile and not so active during nights. However, during the daytime, they are very active and use network services at their work places. Moreover, they are mobile as they commute to/from work from/to home at the beginning and end of the working times.
* **Domestics:** This type of users does not work outside and spend their time at home. Usually the domestics get Internet service in an immobile way. These users are highly active at all times.

The parameters of and are determined based on the abovementioned discussion about the client type characteristics and the time slots. These values are given below. The triplet specify the probability values for night, daytime and evening, respectively.

These values also determine the average connection duration and idle time by using Eq. 1 and 2. For example, a domestic client remains idle during daytime for minutes between connections. Once connected, average connection time for this category is minutes.

* 1. User Mobility and Timing

Real-time scenario covers Internet usage of 300 users in a 1-km2 metropolitan area. The simulations time begins at 00:00 a.m. and lasts for 24 hours. Simulation time is divided into 3 parts considering night, daytime and evening. Every part of the day has different statistical values for client behaviours.

Simulations are run for 1440 seconds, however every second in the simulation stands for 1 minute in real life.

In real-life scenario simulations clients are able to move from one location to another. The time and direction of their movement is selected at random but probabilities are affected by user roles. For example, when school is over, a student is most likely to move towards her target destination (e.g. her home).

Clients are assigned a random target access point. Every one of 100 access points has 3 initial clients. The client moves from its current access point to the target access point on the grid. An example movement pattern is shown in Figure 14. As a client moves from access point A to the access points B, if she needs to connect to the Internet, she forms up a new connection with the access point, which is closest to client’s current location.



Figure 14. User Mobility

In real-life scenario simulations, there are two operators and they have same amount of access points. In current simulations, each operator has 50 access points. The client handovers or roams if there is an active connection during movement between access points. In such a case, depending on the new access point’s affiliated operator, user’s movement triggers either *Seamless Mobility* or *Roaming* protocols. If new access point’s affiliated operator is same as the one that client currently uses, then it means the client would perform *Seamless Mobility* protocol for handover. Otherwise, the client would run *Seamless Roaming* protocol.

Clients are assigned uniformly distributed random speeds between 2 km/h to 6 km/h. The clients are assumed to move without a motor vehicle.

1. Results for Real-Life Scenario Simulation

Results for unit test simulations are described before, however the most significant results are real-life scenario simulation results. Despite the randomness of the system, users’ actions are highly related to their group and current simulation time.

Charts for the results display the average delay for a particular protocol.

* 1. Overview

Final simulations provided the results in Table 2. Charts on Figure 15 and Figure 16 are drawn exploiting the results in Table 2. Considering the results it could be calculated that over 100 minutes of Internet service, workers have only waited for 1 minute for system delays. In average, over 1000 minutes of Internet service needs a delay of 13 to 16 minutes of waiting.

TABLE II  
Simulation Results for Client Types

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Total Internet Usage Time | Total Internet Usage Delay | Average Internet Usage Time for a Client | Average Internet Usage Delay for a Client |
| Student | 95899,26 Minutes | 1698,95 Minutes | 958,99 Minutes | 16,98 Minutes |
| Worker | 101681,64  Minutes | 1316,35 Minutes | 1016,81  Minutes | 13,16 Minutes |
| Non-Worker | 105335,08 Minutes | 1456,12 Minutes | 1053,35 Minutes | 14,56 Minutes |

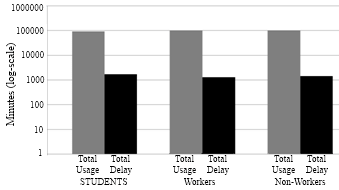


Figure 15. Total Amount of Internet Usage Times for Client Types vs. Total Delays

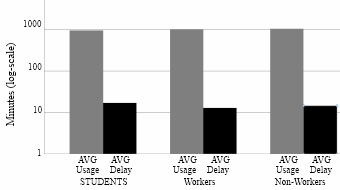


Figure 16. Average Usage Times for Client Types vs. Average Delays

As described before the clients are grouped into 3 groups. The client roles and probabilistic values affect their behaviour in the system, which results difference between overall values of the simulations.

Figure 15 and Figure 16 shows the overall results for real-life scenario simulation. Figure 15 shows comparison of minutes clients used as idle or active. Figure 16. Shows the average value for the clients of the same group.

* 1. Real-Life Scenario Simulation Result for Initial Authorization



Figure 17. Result for Initial Authorization Protocol

*Initial Authorization* protocol is used at the beginning of the service for each user. As it is seen on the chart every one of the 300 users are authenticated at the end of 40th minute.

Simulation starts around the 10th minute in the morning. At the beginning there is a huge amount of users, trying to authenticate. Figure 17 indicates that, this process varies between 0.6 and 2.5 seconds. After 10 minutes it attains a balance and *Initial Authorization* protocol meets a delay of 1 second, which means when users open up their mobile device they would have Internet service after 1 second.

* 1. Real-Life Scenario Simulation Result for Reuse of a Connection Card Protocol



Figure 18. Result for Reuse of a Connection Card

*Reuse of a Connection Card* protocol is used after disconnecting from the system. As it is seen it is a highly used protocol in the system. It starts around the 50th minute and used for the entire time of the simulation.

As seen on Figure 18, at the beginning of the protocol the delay changes between 0.1 and 0.6 second. After some time protocol achieves a balance and a 0.4 second of network delay is observed.

* 1. Real-Life Scenario Simulation Result for Changing Alias



Figure 19. Result for Changing Alias Protocol

Every active client uses *Changing Alias* protocol in the system in every 50 minutes. The protocol is first used at 50th minute and it is used entire time of the simulation.

As one can see on Figure 19, at the beginning of the protocol the delay for the protocol varies between 0.1 and 0.4 seconds. After some time the average delay for the protocol converges to 0.4 seconds.

* 1. Real-Life Scenario Simulation Result for Disconnection



Figure 20. Result for Disconnection Protocol

*Disconnection* protocol first appears around 30th minute and it is used through the entire time of the simulation. Figure 20 shows that, at the beginning of the system Disconnection protocol average delay vary between 0.1 and 0.5 second but through time the average delay meets 0.4 second.

* 1. Real-Life Scenario Simulation Result for Update Packets



Figure 21. Result for Update Packets Protocol

*Update Packets* protocol is an end-to-end one-way protocol. It is expected to get lower delay values for this one. Only access points use *Update Packets* protocol and they send packets to TTP. The packets are sent every 10 minutes.

As it is seen on Figure 21, at the early stages of the protocol, the average delay value varies between 0.6 and 1.4 second but then after some time the protocol stabilized around 0.4 second.

* 1. Real-Life Scenario Simulation Result for Seamless Mobility in Home Operator Protocol



Figure 22. Result for Seamless Mobility in Home Operator Protocol

*Seamless Mobility* protocol is used when a handover happens between access points. If these access points are belonging to the same operator then it means the client is using *Seamless Mobility* protocol.

By looking at Figure 22, it could be said that*, Seamless Mobility* protocol has an initial average delay that shows difference between 0.2 and 1.2 seconds. A user loses around 0.1 second to make a handover to the new access point.

* 1. Real-Life Scenario Simulation Result for Roaming Protocol



Figure 23. Result for Roaming Protocol

*Roaming* protocol is used when a handover happens between access points. If these access points are belongings of different operators then it means the client is using *Roaming* protocol.

*Roaming* protocol has an average delay that varies between 0.05 and 0.2 seconds. There are 2 operators so a client has a %50 chances to make a *Seamless Mobility* or *Roaming* protocols. After some time protocol reaches a balance around 0.2 second of delay.

As one can see on Figure 23, the results for *Roaming* protocol shows a boost until the middle of the simulation time but it decreases and achieves a balance

* 1. Real-Life Scenario Simulation Result for Packet Transfer



Figure 24. Result for Packet Transfer Protocol

*Packet Transfer* protocol is the mostly used protocol in the system.

Figure 24 states that, at the beginning of the protocol the average delay value varies between 0.005 and 0.025 but then the protocol achieves a balance around 0.02 second.

1. Conclusion

In unit tests, standalone performances of the protocols under trivial usage scenarios are analysed. Unit tests set an example for how the system will behave in empty hours. In this way, the first proof-of-concept implementation of the system is provided and showed that the designed protocols reach steady state and reasonable performance in time.

Uniform probability distribution model enables us to simulate real time scenarios in simulation environment, and gets results closer to real time situations.

There are different user types, as there are different types of clients in real life. There is also randomness at the system, so there could occur different outcomes for the same simulations, as there is change in real-life network traffic everyday. The average of those simulations would cover even the most unexpected situations

Results are very important since the actual usage of the system is a combination of these protocols. Unit tests and real-life scenario simulation results show that the proposed system is a considerable and an effective pre-payment system.

References

1. Akyildiz, I. F., Wang, X., and Wang, W. (2005) Wireless mesh networks: a survey, *Computer Networks and ISDN Systems*, 47(4): 445-487.
2. Network Simulator 3 Official Web Site: <http://www.nsnam.org/>
3. Rivest, R., Shamir, A., and Adleman, L. (1978) A method for obtaining digital signatures and public-key cryptosystems, *Communications of the ACM*, 21(2): 120–126
4. FIPS PUB 197 (2001) Announcing the Advanced Encryption Standard (AES), <http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>
5. Trappe, W., and Washington, L. (2006) Introduction to cryptography with coding theory, *Person Education, Inc.*
6. Joseph D. Camp and Edward W. Knightly The IEEE 802.11s Extended Service Set Mesh Networking Standard
7. Kai Yang, Jian-feng Ma, Zi-hui Miao (2009) Hybrid routing protocol for wireless mesh network, Computational Intelligence and Security – CIS ’09
8. Everitt, B.S. (2002) *The Cambridge Dictionary of Statistics*. CUP. [ISBN 0-521-81099-X](http://en.wikipedia.org/wiki/Special:BookSources/052181099X)