**1.**

Hello, thank you all for joining to my jury. Today I will present my thesis, about a secure and seamless prepayment system for wireless mesh networks. The prepayment scheme is called SSPayWMN.

**2.**

On this presentation, firstly I will give a brief introduction to my thesis. I will define the problem and propose a solution.
Then I will describe the building block of the system.
After that I will briefly explain our protocols and inform you about the simulation environment and client models.
Simulation results and performance evaluation will follow.
Finally we will discuss the success of SSPayWMN and I will give you the conclusions.

**3.**

In e-payment systems clients generally put full or partial trust to service providers. However service providers may unintentionally overcharge their clients. We intended to provide a fair system for both clients and to the service providers.

There has been some research on e-payment systems. Rivest and Shamir, proposed the first prepayment for e-payment systems. It is called Pay-Word. Their system used elliptic curve cryptography for public key operations and hash chains for e-cash, which is the e-currency of the system. There have been extensive improvements on their work, like using RSA based signature. The improvements speeded up the system but they did not provide seamless service providing.

Some other related works provide seamless service providing; however they did not provide anonymity and untraceability.

SSPayWMN is proposed to tackle these problems. Providing a secure and seamless prepayment for wireless mesh networks, while providing privacy and untraceability to some extent. Moreover, support steady state performance with reasonable delays.

**4.**

Our objectives are listed on this slide.
We intended to provide wide coverage since SSPayWMN is designed to provide service in metropolitan area.
Our system should be able to provide mobility support for clients and provide seamless connection without causing any interruption.

The system should provide anonymity and untraceability for the clients.
The system will provide mutual authentication. The client will be ensured that she is connected to the correct network.
Usage of hash tokens will provide two-way honesty. Overcharging will not be possible.
The system performance is evaluated using simulations conducted with network simulator 3 (a.k.a. ns-3).

**5.**

**Differences**

**6.**

Using the listed entities we have formed up the system.
There are mobile clients, access points that forms up the mesh backbone, gateways, several operator servers and a trusted third party server.

**7.**

The clients will receive their services with the connection cards. The clients buy their connection cards from the trusted third party.
Trusted third party selects a random value and forms up a hash chain. Hash chains are formed by using the output of a hash algorithm as an input to another hash algorithm.
The count of hash operations determines the length of the hash chain.
We use the irreversibility property of hash chains. It is easy to find a hash output using the input but it is infeasible for otherwise.
Aliases will be used and changed frequently to provide anonymity and untraceability to some extent.

**8.**

From now on I will explain the protocols of the system.
In this slide two protocols are depicted.
Initial Authorization and Reuse of a Connection Card.
Client encrypts the connection request using the trusted third party's public key. Sends the request through the mesh backbone.
TTP will decrypt the packet and calculate SN.
TTP will check the relevancy with SN and $H_0$
TTP will calculate the alias by performing on SN XOR Nonce and sends it to the serving access point via the medium. Every party between sends the response right away and verifies the packet afterwards.

**9.**

Access Point Authentication protocol is a simple handshake protocol
Access point starts the protocol by sending a command to the client.

Client sends a challenge to the access point.
Access point takes the HMAC of the challenge by using the hash token as a key, and sends this value to the client.
Client performs the same operation and compares the values.
If they match, client verifies the access point.

**10.**

After mutual authentication client starts to send data packets to access point.
This protocol is not so different from Can Yücel's packet transfer protocol.
The only difference is, Can was deciding the new hash token time by calculating the remaining service, I am calculating by remaining time.
The client sends the new hash token, whenever her service time is up.

**11.**

Changing Alias protocol is performed by every active client in the system periodically.
The protocol is executed in following of a change alias command broadcast by every access point.
The client encrypts the nonce and the SN and sends them by encrypting them with the public key of the TTP.
TTP decrypts the packet and calculates the new alias. TTP checks the new alias' uniqueness.
TTP signs the new alias and the new hash token and sends it to the access point.
This protocol provides untraceability against an adversary, in case of an alias compromise.

**12.**

Disconnection protocol is necessary for determining the ending point of the service.
Client sends an encrypted disconnection request to the TTP.
TTP decrypts the request and compute the SN.
TTP marks the client as disconnected.
TTP sends the signed alias and hash token to access point, and every party on the way marks client as disconnected.

**13.**

Seamless mobility protocol is performed in situation in which the client tries to handover between two access points of the same operator.
The client sends a mobility request to the old access point.

The old access point performs an encryption over a signature and sends it to the client.
The new access point performs the decryption and verification and understands that the client is already authenticated by the network.
Then the client and the new access point run an access point authentication protocol.

**14.**

In seamless roaming case we perform the exactly same operation but there is a disconnection protocol running in parallel this time. This decides the ending point of the service in old operator.

**15.**

All right enough of protocols.
We have tested our system in ns-3 network simulator.

**16.**

Our system has 300 clients, 100 access points covering a 1 $km^2$ area. 32 gateways, 2 operators and a TTP.
A depiction of the system is present on this slide.

**17.**

We have performed two types of simulations. First type is unit tests, where we tested the standalone performances of the protocols.
The second type is real-life scenario simulations where we considered client types and real life situations.

**18.**

We have grouped Initial Authorization, Reuse of a Connection Card, Change Alias and Disconnection protocols together since their transmitted packet lengths and computational delays are the same.
These protocols achieve 1.04 second of delay in unit tests.

**19.**

Access point authentication protocol has s 0.05 second of latency.

**20.**

Seamless mobility and roaming protocols achieve steady state performance around 3.2 second of delay.

**21.**

Packet transfer protocol achieves 0.0002 second of steady state performance in unit tests.

**22.**

For real life scenarios we have divided our clients into three groups according to their socio-economical status.
Students, employees and domestics.
The frequency and mobility of the clients partially depends on the client type.

**23.**

We have divided a day into 3. Morning, daytime and evening. The clients has different probabilistic values for different day parts.
The upper table shows the probabilistic values for becoming active when a client is inactive.
Every client decides what to do at the beginning of a minute.
For example if an employee is not connected in daytime then the next minute she has a 0,99 chance to become active since it is working time. However in the evening she has a 0,20 chance to become active.

**24.**

And these probabilistic values trigger some protocols in our system.
As you can see if a client is not connected and the random action is in become active probability then she performs initial authorization or reuse of a connection card protocol depending on the status of the connection card.

**25.**

Our clients move on a grid by selecting a random destination and a random speed between 2 and 6 km/h.
However the random values are also multiplied by the client type constants. These constants are 6 for students 2-3 for employees. 1 for domestics. These constants are also affective in the selection of random locations and speeds.

**26.**

Initial authorization reaches a steady state performance around 2.6 seconds.

**27.**

Reuse of a connection card reaches a steady state around 2 seconds.

**28.**

Change alias protocol has a steady state delay value around 1.9 seconds.

**29.**

Disconnection protocol takes approximately 2 seconds as well.

**30.**

Seamless mobility protocol has a 3.3 seconds of delay for hash token transfer but the client does not stop getting service from an access point unless this protocol is finished; therefore the protocol is seamless to the client.

**31.**

Seamless roaming protocol shows more variance but at the end it achieves a steady state performance new 3.3 seconds.

**32.**

Packet transfer protocol has a 0.02 second of steady state performance.

**33.**

Overall burden on the system means the time that a client waits for protocols to get service. Please note that these charts are drawn using logarithmic scale.

| | Total Internet Usage Time | Total Internet Usage Delay | Average Internet Usage Time for a Client | Average Internet Usage Delay for a Client |
|---|---|---|---|---|
| Student | 95899 Minutes | 2078 Minutes | 958 Minutes | 20 Minutes |
| Worker | 101681 Minutes | 1756 Minutes | 1016 Minutes | 17 Minutes |
| Non-Worker | 105335 Minutes | 1832 Minutes | 1053 Minutes | 18 Minutes |

For example total internet usage of students is 95899 minutes in total which corresponds to 958 minutes of internet usage by a student in average. In average a student waits for 20 minutes, whereas these values are lower for employees and domestics.

Our results showed us a 1-2% of burden of total service time.

**34.**

Wide Coverage: Every access point could serve within a 100 meters radius. It is proven that with 100 access points, 1 $km^2$ area is covered for Internet service.

Seamless Mobility and Roaming: Users are able to switch between access points no matter what operator they belong to. The delays are low enough to maintain current connection without any interruption.
Anonymity: For law-enforcement reasons, users must give their identities to Trusted Third Party (*TTP*) for getting connection cards. Therefore, as far as *TTP* keeps clients' identities secret, users can remain anonymous. Identity verifications are performed using aliases, which change periodically.
Mutual authentication: Initial Authorization and Reuse-CC protocols ensure the authentication of the user. TTP signs the acknowledgement values and a handshake protocol is run between the serving access point and the client. These processes ensure mutual authentication.
Two-way honesty: Since the tokens are issued by *TTP*, only the *TTP* and connection cardholder knows all the tokens that are related with a specific connection card. Hence whenever a Client sends a new token, it is not possible for him to say, "I did not use it". Since *TTP* has verified the token, in the authentication phase, operators cannot say that they provided service for non-used tokens.
Untraceability: Our protocols provide untraceability by changing aliases periodically. Clients are traceable between the times they change their aliases nonetheless they could not be related to future actions after the alias change. The period of time to change the aliases is a choice of the designer.
Performance: The performance of SSPayWMN has been evaluated with simulations using ns-3. Two groups of simulations are performed: unit tests and a real-life scenario. In both groups, our protocols achieved steady state with reasonable delays. Moreover, in overall, the latency cost of SSPayWMN is 1% of the actual usage.

**35.**

Our payment scheme is based on hash chains and provides undeniable payment. This system provides fairness to both operators and to clients. Moreover, SSPayWMN provides privacy and untraceability to some extent. Furthermore; SSPayWMN can successfully handle seamless handover between operators by eliminating the need for re-authentication from the scratch.

Our results for unit tests show that SSPayWMN protocols achieve steady state performances in independent executions of system protocols. Real-life scenario simulations show that SSPayWMN protocols could achieve steady state performance in a daily basis. Both simulations ensure the stable performance of SSPayWMN and shows that our system is a flexible and robust system.

**36.**

Thank you
Very much