Open Services –Need To Restrict Access in Bluetooth

Pushpa Suri [†]***, Sona Rani*** [††]

[†] ***Department of Computer Science and Applications, Kurukshetra University, Kurukshetra, India.***
[††] ***Department of Computer Science and Applications, Kurukshetra University, Kurukshetra, India.***

Summary

Security is essential to many applications, which will use Bluetooth links, but hiding the complexity of Bluetooth security from the user is essential if Bluetooth devices are to be easy to use. Through the security architecture, it is possible to implement security at a variety of levels with minimal intervention from the user.

The Security manager is the central entity that manages and enforces security policy in the Bluetooth Security Architecture. In this research paper we have proposed the solution for the escaping from the attack from the unknown devices by the use of security manager information. Our solution deals with the security-related information stored in the security manager about the unsuccessful connection request by the unknown device.

Keywords
Bluetooth, security manager, trusted device, untrusted device

1 Introduction

The security manager component is the entity that decides what policies are to be enforced when a connection request is there. Based on the service, device type and whether the device is trusted or untrusted the security manager can enforce application level authentication, encryption of the session and any other specific access policies.

The Security manager needs information regarding devices as well as services before it can take a decision whether or not to allow access and if so, to what services. This information is stored in two databases namely, the Device Database and the Service Database.

The process followed by the security manager in granting access to a remote device to connect to a particular service is as follows:

1. Request by remote device.
2. L2CAP receives the Connection request
3. Security manager receives the request from L2CAP to grant access
4. Security Manager searches for query both device and service databases
5. If device is trusted, then security manager may ask for authentication or authorization (depending on the implementation).
6. If the device is untrusted, the security manager enforces authorization.

7. The Security manager will then decide if the service access requires link encryption. If so, keys will be negotiated and exchanged at the L2CAP protocol level and the connection will continue to be setup. Alternatively, if the device is in security mode 3, the security manager instructs the LMP to authenticate and encrypt (if desired) the communication before the connection to the service is set up.

2 Databases

Since there exist trusted devices and different levels of authorization, databases are needed to hold device and service information Different protocols will access the information on in these databases according to the implemented profile To allow uniform access to the databases by all protocols, a security manager handles security transactions with the different protocols

The Device database stores information about the device type, the trust level (whether trusted or untrusted) and about the link key (used for encryption) length. It holds information on whether devices are authenticated and authorized.

The Service database stores information regarding the authentication, authorization and encryption requirements for access to a service. It also stores other routing information for the services.

2.1 Exchange of information with the security databases

To allow uniform access to the databases by all layers, a security manager handles security transactions with the various layers. All exchange of information with the security databases goes through the security manager as illustrated in Figure 1.

Applications and protocols wishing to use security features register with the security manager. The security manager stores security information in the security databases on behalf of the rest of the system. Security policies are enforced by exchanging queries with the security manager:

- Applications query to find out whether a particular device is allowed to access a service.

Manuscript received May 5, 2007.
Manuscript revised May 20, 2007.

• HCI queries to find out whether to apply authentication and/or encryption to a connection.
• The user interface is queried by the security manager to get PINs.
• The user interface is queried by the security manager to authorize new devices.
• Protocol layers query the security manager with access requests.

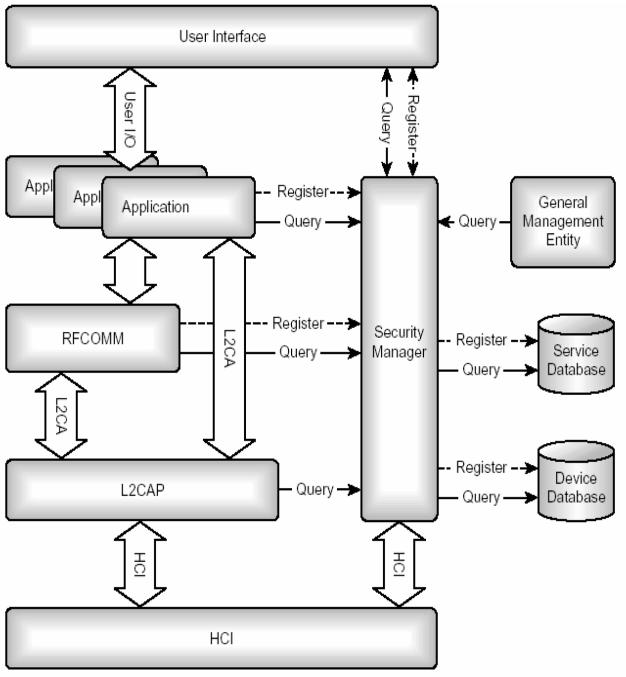

Figure 1 Information Exchange

3 Problem in the existing system

In the existing system the open services are accessed by all types of devices. And if the database about the open services can be assessed by attacker device. The connection will be made between the devices, because there is no need for the authentication. So the attacker will misuse this pairing process.
The table 1 shows the types of devices and services and their respective relationship accordingly.

But in the security database, there is no provision for storage of information about the attacker device attempts. And if we store the information about the unsuccessful connection attempts in the database and the same devices makes the connection through the accessing the information of open services. we can detect the attack . There should be provision for storage of attacker information. That is if more than one attempt is made to make the connection. Then the address of that device should be stored in the device database.

4 Proposed method

In our proposed method following terms are used:

Device Level
Devices are distinguished in five levels:
(I) Trusted Device: A device that has been authorized as the trusted fixed relationship (paired) and has unrestricted access to all services.
(2) Untrusted Device: A device that has been authenticated successfully but has no permanent fixed relationship (but possibly a temporary one), is not considered as trusted. The access to services is restricted.
(3) Authenticated Device: A device that has been authenticated successfully but still not process authorization. The access to services is restricted.
(4) UnAuthenticated Device: A device that has failed to authenticate. It has the access to services with the lowest privilege.
(5) Unknown Device: A device that has not passed any authentication and authorization process. It has the lowest privilege
Service Level
Based on the requirements of services for authorization, authentication and encryption, the access services request are defined as the following three security levels:
(1) Need Authorization and authentication both Service: Services that require authorization and authentication. Automatic access is only granted to trusted devices. Other devices need a manual authorization.
(2) Need Authenticate Service: Services that require authentication only. Authorization is not necessary.
(3) Open Service: Services open to all devices; authentication is not required, no access approval is required before service access is granted. Only need to encrypt the data. .

For the enhancing the security in bluetooth is that we make use of database of security manager efficiently. That is before making the connections, one should follow the Relationships among them. And in our proposed method we recommend that if in a particular time period one device make use of more than one attempt that it fails to make the connection in the first attempt. So we can estimate that the claimant device is not the right one, it is the attacker and only estimating the keys.
With the help of security manager, the information about the attacker device can also be saved. With this method

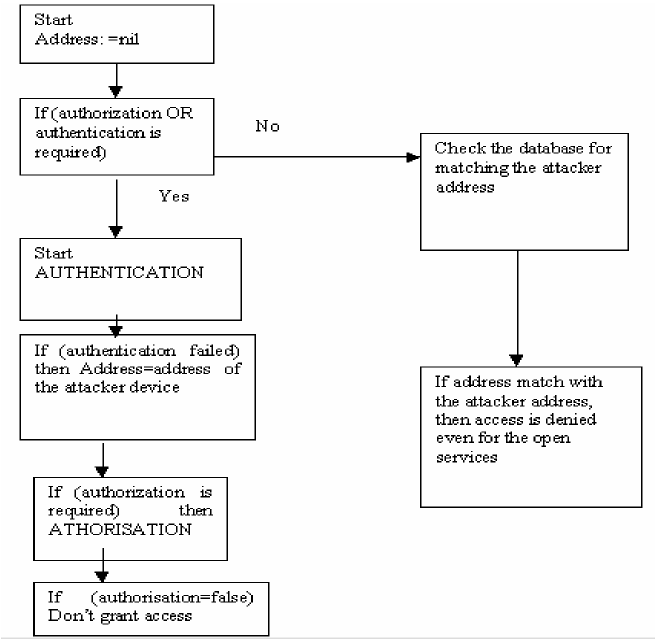
Start
Address: =nil
If (authorization OR authentication is required)
No
Yes
Check the database for matching the attacker address
Start AUTHENTICATION
If (authentication failed) then Address=address of the attacker device
If address match with the attacker address, then access is denied even for the open services
If (authorization is required) then ATHORISATION
If (authorisation=false) Don't grant access

Figure 2 Authentication process of proposed method

Whenever an attacker wants to make the connection the security manager will check the database and accordingly it can alert the device that the attacker is in the position to try for attacker and the request for the connection from the attacker will be denied as shown in the figure 2. Even for the services that are open to all the devices. There should be check for the open services, so that through the access to open services the attacker will get the connection and can know about the keys of the victim device.

Conclusion

Because it is the service that decides the level of security to be enforced, security cannot be enforced when an ACL (data) connection is first set up. Instead, security is enforced only when access is requested to a protocol or service that requires security. The protocol or service requests access from the security manager. The security manager looks up the service or protocol in the service database to see what level of security to impose. Then it looks up the connecting device in the device database to see whether it meets the requirements of the service. If necessary, the security manager enforces authentication and/or encryption, and ends any necessary queries for PINs or authorization to the user interface. Access is then granted or refused, and if access was granted, the service can be used.

So in this paper the methods for storage the attacker data about the number of attempts made by it can help in the detection in the attacks and the observation of the time delay can solve the problem.

References

[1] Bluetooth Special Interest Group. Specification of the bluetooth system: core package version 1.2, 2003.

[2] Bluetooth Special Interest Group. Specification of the bluetooth system: core package, 2004.

[3] Markus Jakobsson and Susanne Wetzel. Security weaknesses in bluetooth.
Lecture Notes in Computer Science, 2001.

Table 1 Relationships between Device Type and Service Type

Device type → Service type ↓	Trusted	Untrusted	Authenticated	Unauthenticated	Unknown	Attacker device
Needs authentication only	Access is Allowed	Does not allow access	Does not allow access	Does not allow access	Does not allow access	Does not allow access
Needs authentication and authorization both	Access is Allowed	Access is Allowed	Access is Allowed	Does not allow access	Does not allow access	Does not allow access
Open services	Access is Allowed	Access is Allowed	Access is Allowed	Does not allow access	Access is Allowed	**Does not allow access**