**Mini Project Report on**

https://lh4.googleusercontent.com/Tf6D4vPBelPcf7k-e21Y34NwNtD0_cCOX1C4xo5aGNST5KCKv_t-XuA2LMxvFz6PH-tr6Swp6tJES_bRcbFiGgGVTwQAH77UxM-u5m5btf7fdDxKTyYFuE447nV-Ir4GzekrBCWlfl2M1fR6HBo-fQ

**Key management scheme for secure communication**

https://lh4.googleusercontent.com/Tf6D4vPBelPcf7k-e21Y34NwNtD0_cCOX1C4xo5aGNST5KCKv_t-XuA2LMxvFz6PH-tr6Swp6tJES_bRcbFiGgGVTwQAH77UxM-u5m5btf7fdDxKTyYFuE447nV-Ir4GzekrBCWlfl2M1fR6HBo-fQ

**Submitted in partial fulfillment of the requirement for the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE & ENGINEERING**

**Submitted by:**

**Student Name: Shreya Singh**  **University Roll No.: 2019118**

***Under the Mentorship of***

**- Dr. Mohammad Wazid**

**Professor**



**Department of Computer Science and Engineering**

**Graphic Era (Deemed to be University)**

**Dehradun, Uttarakhand**

**July-2023**



**CANDIDATE’S DECLARATION**

I hereby certify that the work which is being presented in the project report entitled **“** **Key management scheme for secure communication”** in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineeringof theGraphic Era (Deemed to be University), Dehradun shall be carried out by the under the mentorship of  **Dr. Mohammad Wazid, Professor**, Department of Computer Science and Engineering, Graphic Era (Deemed to be University), Dehradun.

Name: Shreya Singh     University Roll no.: 2019118

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Description** | **Page No.** |
| Chapter 1 | Introduction | 1 |
| Chapter 2 | Literature Survey | 6 |
| Chapter 3 | Methodology | 8 |
| Chapter 4 | Result and Discussion | 11 |
| Chapter 5 | Conclusion and Future Work | 13 |
|  | References | 14 |

**Chapter 1**

**Introduction**

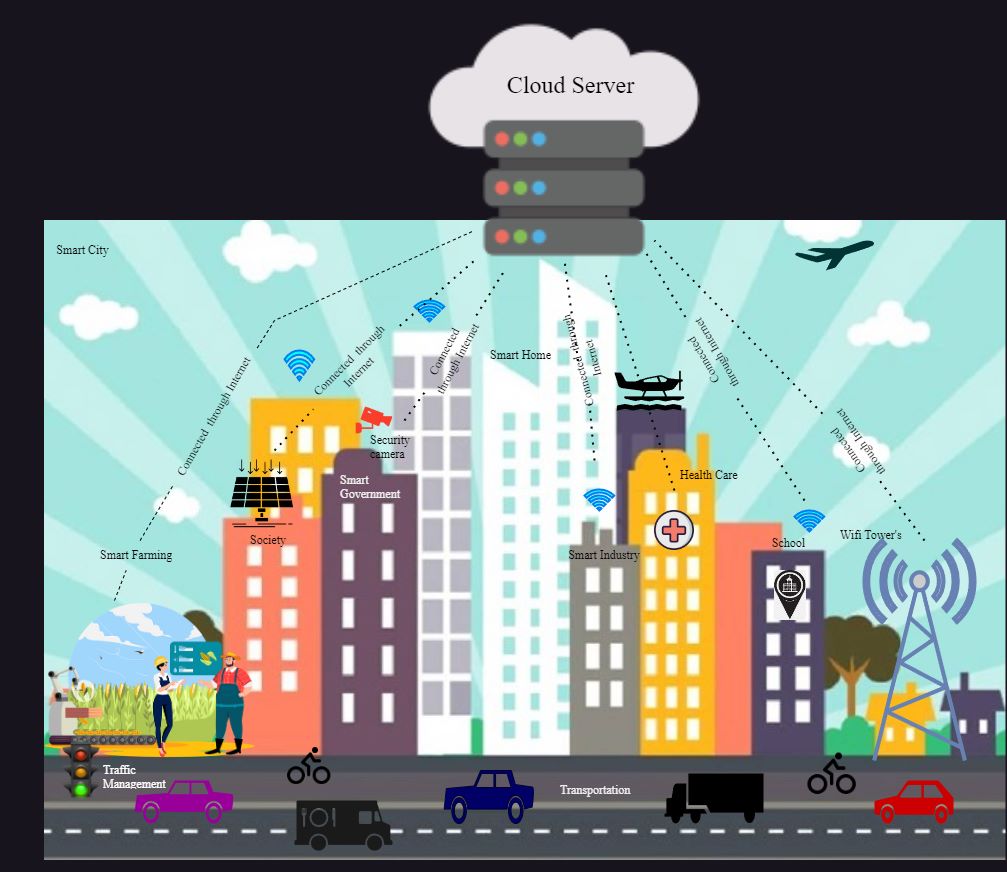
Smart City is basically an mixture of various existing fields of IoT deployed in a city to ensure better and smoother functioning of a society. It utilizes the concept in which various physical entities equipped with embedded systems can communicate with each other in order to exchange data and information of the users over a cloud server reduce labour costs, make decisions on behalf of humans through previously available data, and provide better living opportunities to people. The proper implementation of a Smart city is a long tedious process which requires equal resources distribution to all sectors of society. With the deployment of such a vast network of devices that continuously exchange data between each other and the server and the idea of connecting an entire city through this information exchange, it becomes obvious that a lot of this data is private and sensitive and thus needs to be protected from people with malicious intent. So, it becomes extremely important that the three pillars of security namely Confidentiality, Integrity and Accessibility (CIA) remain unharmed. Hence we have proposed authentication and key agreement (AKA) protocol model.

* 1. **Problem Statement**

Key Management Scheme using Secure Communication means management of keys is required for secure communication between the sender and receiver entity . Effective key management is essential for ensuring the security of cryptographic systems and establishing secure communication channels. This report explores a key management scheme that gives secure communication techniques to enhance the security of cryptographic systems.

**1.2 System Model**

The network model of our “Key Management Scheme For Secure Communication” given in Fig. 1.2.1 consists of smart devices, which are the end devices. Furthermore, the cloud servers are located at the cloud layer. The devices and cloud communicate by transferring data through internet. But it is seen that this kind of communication environment is not secure as it is vulnerable to different kinds of attacks. Therefore, it is then essential to provide a secure communication. By using AKA scheme, a smart devices can communicate in a secure way (or visa-versa) and can maintain a secure session with the cloud server for the secure data transfer. After mutual authentication, a smart device and the cloud server establish a “session key” for their secure communication in future.



**Fig. 12.1 Authentication and key agreement protocol**

**1.3 Requirement**

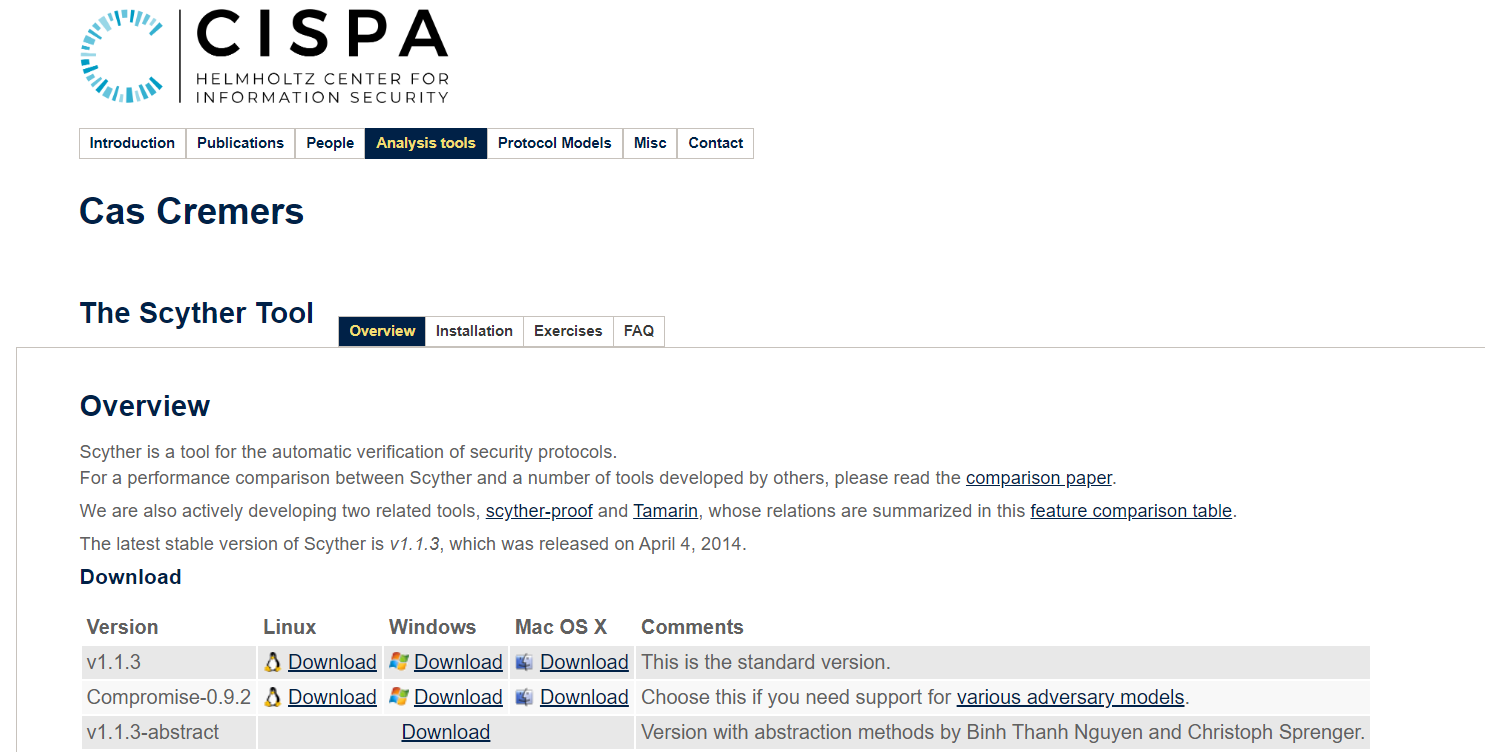
In this project we’ve developed a protocol in SPDL (Security Protocol Description Language) for management of keys using secure communication between sender and receiver entity. After designing the protocol we’ve verified it using Scyther tool.

Scyther, an automated security tool that is used for analyzing the security of cryptographic protocols. It can be used to detect vulnerabilities and weaknesses in protocols and verify their security properties. It was developed by Cas Cremers and his colleagues as a tool for formal analysis of cryptographic protocols. It is an open-source tool that is available for free and can be downloaded from the official Scyther website. The main purpose of Scyther is to analyze the security of cryptographic protocols by modeling them and then automatically searching for attacks or vulnerabilities. It uses formal verification techniques and symbolic analysis to explore the possible execution paths of a protocol and check for security properties such as confidentiality , integrity and authentication.

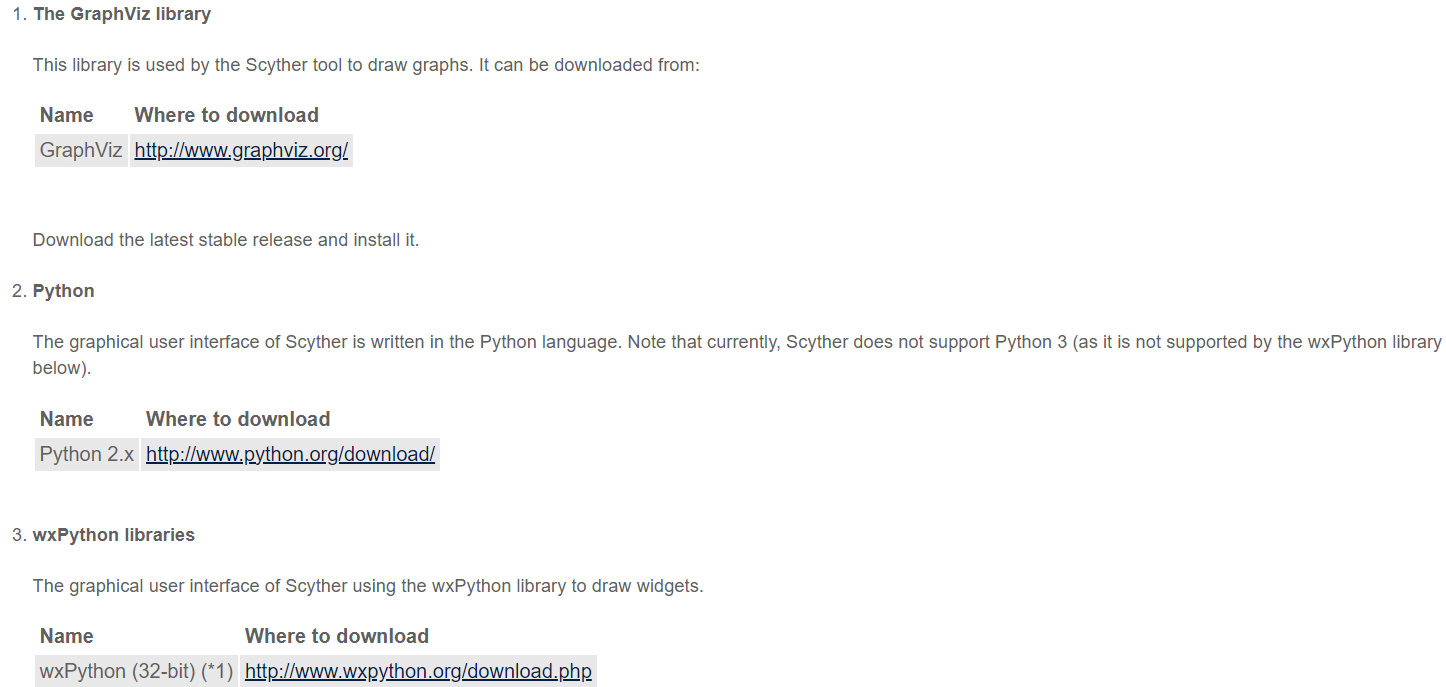
Scyther supports various security models and cryptographic primitives, including symmetric and asymmetric encryption, digital signatures, key exchange protocols, and more. It provides a high-level language called the Scyther Security Protocol Description Language (SPDL), which allows users to describe their protocols in a concise and readable manner. One of the key features of Scyther is its ability to automatically generate attack scenarios and identify potential vulnerabilities in protocols. It can generate an attack tree that shows the different attack paths and the conditions required for each attack to succeed. This helps protocol designers and security analysts to understand the weaknesses in their protocols and take appropriate measures to address them.

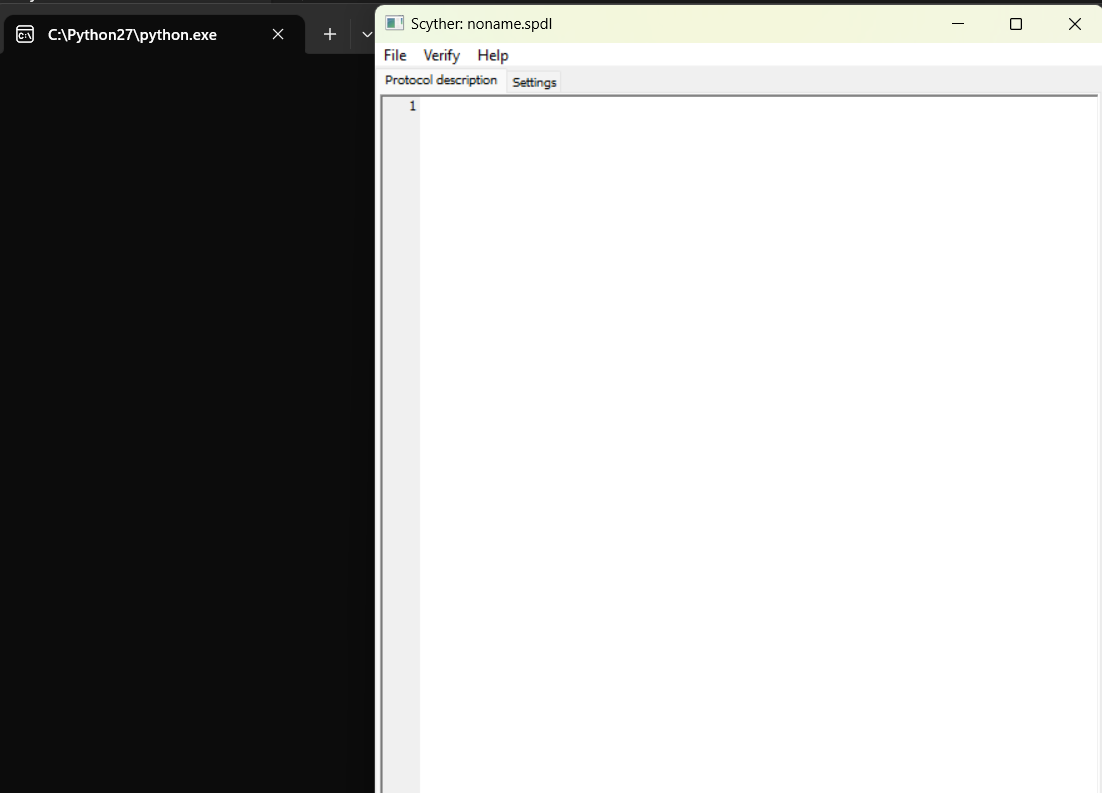
Tools required to run/verify any protocol are given below [1][2][3][4]:

* Sycther
* GraphViz library3
* Python
* wxPython libraries



**Fig 1.3.1 Scyther Tool Installation**



**Fig 1.3.2 Installation of tools required for verification**  
  
  
**Fig 1.3.3 (a) About Scyther**   
  
**Fig 1.3.3 (b) Window for writing protocol**

After the installation is completed, we write the designed protocol in the above window.**Chapter 2**

**Literature Survey**

Key management scheme are very important in providing security to several smart devices. Numerous work in this field has already taken place. Some of them are listed below :

Das [5] presented a new “user authentication scheme” for Industrial Iot deployment in which a registered user can access the data of a smart industrial device remotely through the established secure session key.

Challa [6] designed a “signature based user authentication scheme” for IoT applications.

Ntshabele, K., Isong, B., Gasela [7] developed a security key management server using LoRaWAN(Low Range Wide Area Network) using a cryptographic system similar to AES(Advanced Encryption System)- 128 bits to secure data in transmission. The security verification tool Scyther has been used to verify the strength of the security protocol against various attacks. Two algorithms A and B are designed for a trusted Key Management Server (TKMS). It is important to note that the algorithm B is just a modification or an enhanced version of A as when algorithm A goes under verification in Scyther if the model is unable to protect against attacks it goes under an improvement phase where it is again tested and once it runs successfully it is named Algorithm B. It is also important to note that the authors use an XOR function algorithm to compute the Keys.

Result: After 7 rounds of improvement algorithm A finally becomes algorithm B and is finally secure.

M. Wazid, M. S. Obaidat, A. K. Das and P. Vijayakumar[8] developed a SAC-FIIoT which is a secure access control scheme for Fog-Based Industrial Internet of Things. IIoT refers to an industry-level connection of various devices over the internet that communicate with cloud servers as well as each other to exchange information. To design a security model for IIoT they have used Fog-based architecture where devices such as temperature-sensing devices, machines, and gas-measuring devices act as the end devices. The device layers is connected to an intermediate fog layer that is present between the end devices and the main cloud servers. The main concept behind having a fog layer is that the data that is more frequently used can be stored in the fog layers whereas the data less frequently used is stored in the cloud layers. The data stored in these fog layers is also vulnerable to attacks and hence a security protocol is needed. The authors use the “Dolev Yao” threat model is used to design this protocol.

Result: Comparative analysis of this new model proves to be more efficient than the existing models.

Saha [9] presented a blockchain based access control protocol for IoT-enabled healthcare applications. Some of the features of Saha protocol are – the technology was an access control mechanism with inclusion of private blockchain technology. Their protocol could be used by the trusted group of hospitals for the sharing of sensitive healthcare data securely. ECC-based signature technique was used in their protocol. The security of the presented protocol was dependent on solving the ECDLP and “collision-resistant one-way hash function”. Their protocol contained important phases, like registration/enrolment phase, login and access control phase and blockchain formation phase. Their protocol could resist various types of known attacks “replay attack”, “MiTM”, “impersonation attacks” and “Ephemeral Secret Leakage (ESL) attack”. Their protocol also preserved the “anonymity” properties.

**Chapter 3**

**Methodology**

The proposed authentication and key agreement (AKA) scheme is elaborated on in this section. After performing the steps of the proposed scheme, secure authentication of the entities can be achieved. The Cloud Server (CS) and its associated smart device (DV) establish the session keys for their secure communication.

The proposed scheme contains the following essential phases:

1. The registration phase, and
2. Authentication and key establishment phase.

**A. Registration Phase**

The trusted authority TA performs the registration of various entities. The TA is presumed to be an unbiased authority and provides the secret key.

a) Registration of Smart Device :

The TA does registration of each smart device DV using the following steps:

• The TA has its own secret key and identity as also the smart device has its own secret key and identity . After this, the TA computes the pseudo-identity of DV by concatenating the identity and secret key of smart device and applying a hash function on it and its own pseudo-identity by concatenating its identity and secret key and then applying a hash function on it.

• The TA computes the temporal credentials of smart devices DV by concatenating the identity ,secret key of smart device (DV) and pseudo-identity of TA and a registration timestamp of DV and then applying a hash function on it. It also generates a random temporary identity of DV.

• The TA finally stores these values in the memory of smart device (DV) before its deployment in the network.

b) Registration of Cloud Server :

The TA does registration of each cloud server (CS) using the following steps:

• The TA chooses the identity of the cloud server and then computes the pseudo-identity of CS by concatenating the identity and secret key of cloud server and applying a hash function on it

• The TA computes the temporal credentials of CS by concatenating the identity ,secret key of cloud server (CS) and pseudo-identity of TA and a registration timestamp of CS and then applying a hash function on it .

• The TA finally stores these values in the memory of cloud server (CS) before its deployment in the network.

**B.**  **Authentication and key establishment phase**

.This phase is required for secure authentication of the deployed smart devices (DV) with the cloud server (CS). It is done using following steps:

• DV is the initiator therefore it generates a random nonce rs1 and T1 where T1 is timestamp. Concatenating rs1 and temporal credentials of smart devices and applying a hash function on it and concatenating T1 and pseudo-identity of smart devices and applying a hash function on it. By Ex-oring these values we compute message M1 and message M2 is computed by concatenating rs1 and temporal credentials of smart devices and applying a hash function on it, then concatenating this value with T1 and pseudo-identity of smart devices and applying a hash function on it. After computing these values, DV sends an authentication message Msg1 containing M1 ,M2 and T1 to CS .

• After receiving Msg1 from DV, CS first verifies the timeliness of T1 by using the condition: T1 minus T′1 is less than equal to △T, where △ T is “maximum transmission delay” and T′1 is the reception time of Msg1. If it matches, M’1 is computed as M1 Ex-or hash function of concatenation of pseudo-identity of DV and T1. M′2 is computed as hash of concatenation of rs1 and temporal credentials of smart devices then concatenation with pseudo-identity of DV and T1 and apply a hash function on it. If M2 and M′2 are equal then CS generates rs2 and T2 and computes message M3 = concatenating rs2 , temporal credentials of CS , pseudo-identity of CS and applying a hash function on it then Ex-oring it with the hash result obtained after concatenating pseudo-identity of DV and T2..

• The session key on the cloud end (SK CS-DV) is now generated. Using the session key, message M4 is generated as M2 by concatenating session key, T1, and T2 then applying a hash function on it. A new temporary ID of smart device is generated and a message M5 is computed using this new ID. Now, cloud server (CS) sends the authentication message Msg2 consisting M3 , M4 , M5 and T2 to smart device (DV).

.

• After receiving Msg2 from CS , DV now verifies the timeliness of T2 by using the condition: T2 minus T′2 less than equal to △T, where △ T is “maximum transmission delay” and T′2 is the reception time of Msg2.

• The session key (SK DV-CS ) is now generated by DV. M′4 is generated by concatenating the session key, pseudo-identity of DV and T2 then applying a hash function on it . If M4 and M′4 are equal then CS is authenticated with DV and the session key computed by DV is correct.

• DV generates timestamp T3 and computes a session key verifier message as SKV by concatenating secret key and T3 and applying a hash function on it. DV sends authentication message Msg3 containing session key and T3 to CS .

• After receiving Msg3 from CS , DV now verifies the timeliness of T3 by using the condition: T3 minus T′3 less than equal to △T, where △ T is “maximum transmission delay” and T′3 is the reception time of Msg3. If it matches, CS computes SKV′. Now check if SKV is equal to SKV′ , if so assume that the session key computed by DV is correct.

Now DV and CS establish session key as SK DV-CS (= SK CS-DV ) for their secure data transmission. Using this session key the smart device and the cloud server can communicate efficiently and securely.

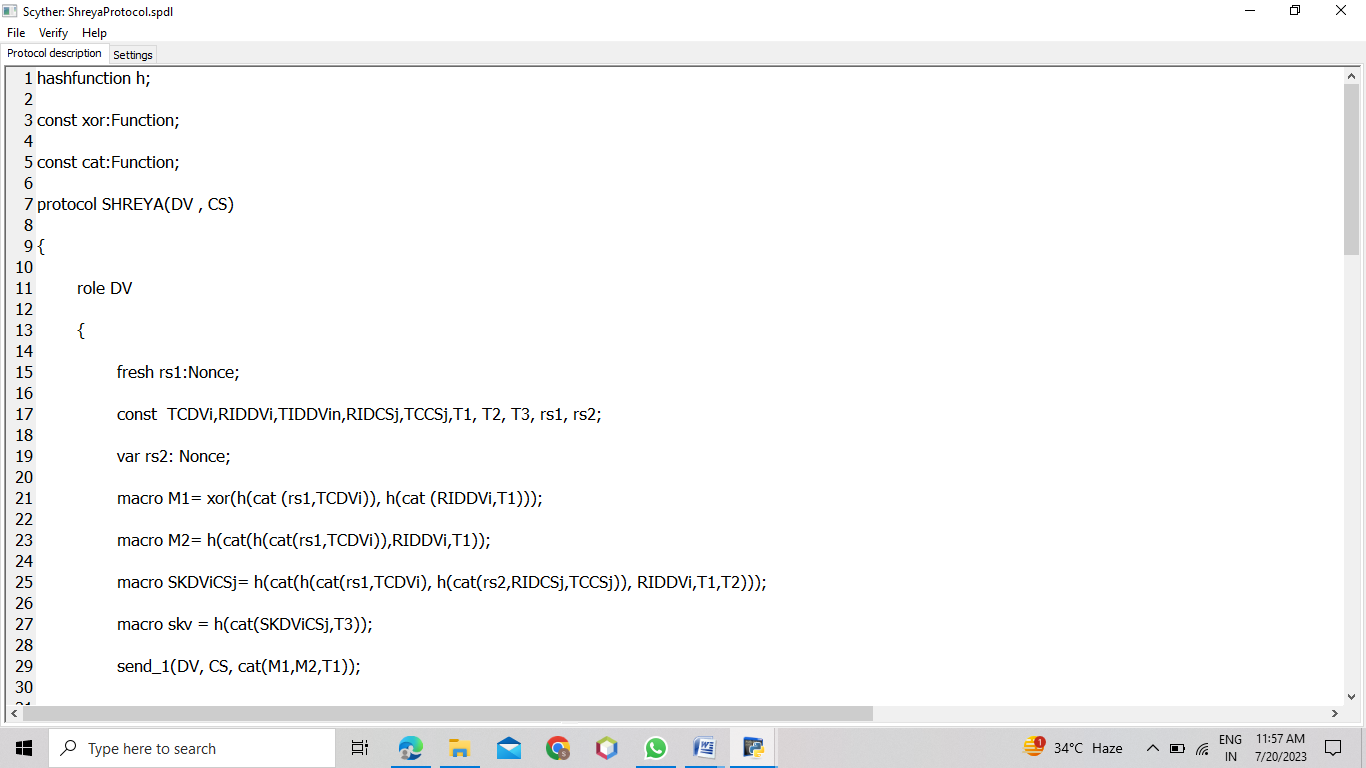
**Chapter 4**

**Result and Discussion**

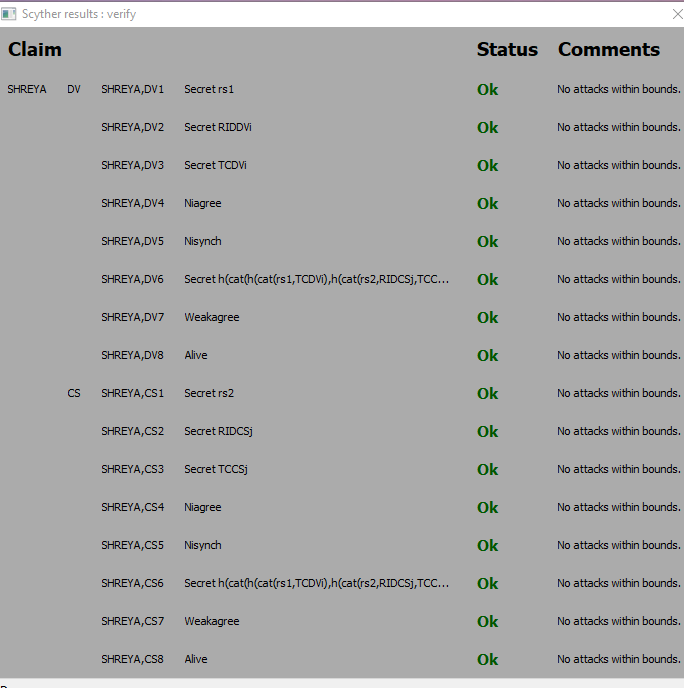
The figure below is snapshot of the proposed authentication and key agreement (AKA) protocol.

Here DV stands for smart device and CS stands for cloud server.

Now, we verify the written protocol by pressing F1 button on the keyboard or by clicking on ‘Verify’ then on ‘Verify protocol’ .



**Fig 4.1 Key Management Protocol**



**Fig 4.2 Protocol verification**

In the above figure, we can observe that the proposed protocol is secured & completely verified. Since the protocol is secured, hence it can be used for communication between the sender and receiver entity.

**Chapter 5**

**Conclusion and Future Work**

**5.1** **Conclusion**

A key management scheme (AKA protocol) for secure communicationenvironment is presented. The verification of proposed protocol proved that it is secured against the differenttypes of possible attacks.

A robust key management scheme utilizing secure communication techniques is crucial for ensuring the security of cryptographic systems. By employing secure communication protocols, such as symmetric and asymmetric encryption, key exchange protocols, and digital signatures, organizations can establish and maintain secure communication channels. Proper key generation, distribution, storage, and revocation mechanisms further enhance the security of cryptographic systems, protecting sensitive information from unauthorized access and ensuring the confidentiality, integrity, and authenticity of communication.

**5.2 Future Work :**

There are several possible future extensions to this work. For example, we can explore the integration of blockchain in the smart device environment, given the lack of trust between participating entities and the need to transmit significant information. The proposed authentication and key agreement (AKA) protocol can be compared with the other related existing competing schemes. We can add more functionality features to the proposed AKA and also provide simulation study to measure its impact on the performance of various network parameters. Some future research directions of this particular domain can be done:

* Security Enhancement
* Handling of Privacy Issues
* Improvement of Accuracy of the System

**References**

[1] https://people.cispa.io/cas.cremers/scyther/

[2] <http://www.graphviz.org/Download.php>

[3] <http://www.python.org/download/>

[4] <http://www.wxpython.org/download.php>

[5] A. K. Das, M. Wazid, N. Kumar, A. V. Vasilakos, and J. J. P. C. Rodrigues, “Biometrics Based Privacy-Preserving User Authentication Scheme for Cloud-Based Industrial Internet of Things Deployment,” IEEE Internet of Things Journal, vol. 5, no. 6, pp. 4900–4913, 2018.

[6] S. Challa, M. Wazid, A. K. Das, N. Kumar, A. G. Reddy, E. Yoon, and K. Yoo, “Secure Signature-Based Authenticated Key Establishment Scheme for Future IoT Applications,” IEEE Access, vol. 5, pp. 3028–3043, 2017.

[7] Ntshabele, K., Isong, B., Gasela developed a security key management server using LoRaWAN(Low Range Wide Area Network), IEEE, 2019.

[8] Mohammad Wazid, Mohammad S. Obaidat, Ashok Kumar Das and Pandi Vijayakumar proposed “SAC-FIIoT: Secure Access Control Scheme for Fog-Based Industrial Internet of Things” IEEE Access,

[9] K. Sutrala, A. K. Das, N. Kumar, and J. J. P. C. Rodrigues. On the Design of BlockchainBased Access Control Protocol for IoT-Enabled Healthcare Applications. In ICC 2020 - 2020 IEEE International Conference on Communications (ICC), pages 1–6, Dublin, Ireland, Ireland, 2020