



NTNU – Trondheim
Norwegian University of
Science and Technology

Security Analysis of TETRA

Shuwen Duan

Master of Telematics - Communication Networks and Networked Services [2

Submission date: June 2013

Supervisor: Stig Frode Mjøl̄snes, ITEM

Co-supervisor: Joe-Kai Tsay, ITEM

Norwegian University of Science and Technology
Department of Telematics

Problem Description

TETRA (Terrestrial Trunked Radio) is an ETSI (European Telecommunications Standards Institute) standard (first published in 1995) for a mobile communication system designed to be used by law enforcement, emergency and rescue service organizations, in public transportation organizations, and as a general national safety communication network. TETRA systems have been built and are in operation in more than 100 nations. The main service is voice communications. Some special features are very short call setup time, push-to-talk group calling mode, and direct terminal-to-terminal radio transmission. TETRA provides authentication protocols and both radio channel and end-to-end encryption.

This report will describe the security system of TETRA, including the cryptographic primitives used, the authentication and encryption protocols, the key management, and mechanisms provided for cooperation across security domains. The candidate will perform a security analysis of the system, by setting up the security assumptions, the attacker model, and describe the result of her analysis. In particular, a formal security analysis of the authentication protocols could be done. Furthermore, the candidate will attempt to understand and describe publicly reported technical problems, if any, pertaining to the communication security of TETRA.

Abstract

TETRA is designed to be used in private mobile radio environment, and PMR users have a requirement for high level of security. Security takes a number of different forms, both in terms of availability, reliability of the system and confidentiality of transmitted information.

This thesis identified the key security features of TETRA system which includes authentication, encryption and key management. A formal security analysis of authentication protocol was made exploring possible attacks during authentication. The automatic security verification tool used in this thesis is Scyther. Inspired by the result given by Scyther, possible attacks were discussed with different scenarios.

It was concluded that some of the attacks found by Scyther might not be the most efficient ones due to their complexity.

Preface

This master thesis has been carried out at the Department of Telematics at the Norwegian University of Science and Technology (NTNU) during the period February to June 2013.

I take this opportunity to express my profound gratitude and deep regards to my supervisor Joe-Kai Tsay for his guidance, monitoring and encouragement throughout the project. I would also like to thank Professor Stig Frode Mjølunes for feedback and great support, and the Department of Telematics for giving me the opportunity to write this thesis.

Contents

I	Introduction	1
1	Introduction	3
1.1	Scope of thesis	3
1.2	Work methods used	3
1.3	Outline of the Thesis	4
II	Background	7
2	Services of TETRA	9
2.1	Services of the TETRA System	10
3	TETRA Network Architecture	13
3.1	Interfaces	14
3.2	Network Components	15
3.3	Operation of the TETRA System	16
3.3.1	Identities	16
3.3.2	Mobility Management	17
III	TETRA Network Security	19
4	Security Features	21

4.1	Security Requirements	21
4.2	Authentication in TETRA	22
4.2.1	Authentication Key	22
4.2.2	Authentication Procedures	24
4.2.3	Authentication PDUs	30
4.2.4	Authentication Algorithm	31
4.2.5	Analysis of Authentication Protocol	32
4.3	Key Management	33
4.3.1	Air Interface Key management mechanisms	33
4.3.2	Over The Air Re-keying	35
4.4	Encryption	36
4.4.1	Classes of Security	36
4.4.2	Air Interface Encryption	36
4.4.3	End-to-End Encryption	38
4.5	Replay Prevention	40
IV	Analysis of the Authentication Protocol	41
5	Analysis of Security Protocol	43
5.1	The Scyther Tool	43
5.2	Security Protocol Specification	44
5.3	Security Properties	46
5.4	Verification Algorithm	47
5.5	Adversary Models	48
6	Verification of TETRA Authentication Protocol	49
6.1	TETRA Authentication Protocol Specification	49
6.2	Verification of TETRA Authentication Protocol	53
7	Attack Scenarios	61
7.1	Denial of Service (DoS) Attacks	61

<i>CONTENTS</i>	ix
7.2 Man-in-the-Middle Attacks	63
7.3 Summary	66
V Conclusion	69
8 Conclusion	71
VI Appendix	75

List of Figures

3.0.1 TETRA network architecture[2, 6]	14
3.3.1 The TETRA Equipment Identity[22]	16
3.3.2 The TETRA Subscriber Identity[22]	17
4.2.1 Generation of authentication key to MS[8]	24
4.2.2 Authentication of a user[8]	25
4.2.3 Authentication of the infrastructure[8]	26
4.2.4 Mutual Authentication Initiated by Infrastructure[8]	27
4.2.5 Mutual Authentication Initiated by a User[8]	29
4.2.6 D-AUTHENTICATION DEMAND[8]	30
4.2.7 U-AUTHENTICATION RESPONSE[8]	30
4.2.8 D-AUTHENTICATION RESULT[8]	31
4.3.1 Distribution of a common cipher key[8]	34
4.4.1 Speech and control information encryption[8]	37
4.4.2 Generation of ECK[8]	38
4.4.3 Synchronization [9]	39
6.1.1 MSC of TETRA Authentication Protocol	50
6.1.2 Example input to the Scyther tool	52
6.1.3 Example input to the Scyther tool	53
6.2.1 Scyther Verification Result	54

6.2.2 Trace patterns	55
6.2.3 Attack 1	56
6.2.4 Attack 2	58
6.2.5 Attack 3	60
7.2.1 False base station attack	64

List of Tables

- 5.1 Basic Term Sets[29] 45
- 5.2 Basic Run Sets [29] 45

Nomenclature

AC	Authentication Code
CC	Colour Code
CCK	Common Cipher Key
CCK-id	CCK Identifier
CK	Cipher Key
CN	Carrier Number
CSMA	Carrier Sense Multiple Access
DCK	Derived Cipher Key
DMO	Direct Mode Operation
DoS	Denial of Service
DSMA	Data Sense Multiple Access
DSSS	Direct Sequence Spread Spectrum
ECK	Encryption Cipher Key
EKSG	End-to-end Key Stream Generator

EKSS	Key Stream Segment
ESN	Electronic Serial Number
ETSI	European Telecommunications Standards Institute
ETSI	European Telecommunications Standards Institute
FAC	Final Assembly Code
FHSS	Frequency Hopping Spread Spectrum
GCK	Group Cipher Key
GSM	Global System for Mobile Communications
GTSI	Group TETRA Subscriber Identity
HDB	Home Database
IMEI	International Mobile Equipment Identity
ISDN	Integrated Services Digital Network
ITSI	Individual TETRA Subscriber Identity
IV	Initial Value
KGS	Key Stream Generator
KSO	Session Key for OTAR
KSS	Key Stream Segment
LS	Line Station
MAF	Mutual Authentication Flag
MCC	Mobile Country Code
MF	Manipulation Flag

LIST OF TABLES

xvii

MGCK	Modified Group Cipher Key
MNC	Mobile Network Code
MNI	Mobile Network Identity
MSC	Message Sequence Chart
OTAR	Over The Air Re-keying
PDN	Public Data Network
PDO	Packet Data Optimized
PDU	Protocol Data Unit
PMR	Private Mobile Radio
PSTN	Public Switched Telephone Network
RS	Random Seed
SAGE	Security Algorithm Group of Experts
SCCK	Sealed Common Cipher Key
SCK	Static Cipher Key
SDR	Software Defined Radio
SIM	Subscriber Identity Module
SIM	Subscriber Identity Module
SMS	Short Messaging Service
SPR	Spare

SSI	Short Subscriber Identity
SwMI	Switching and Management Infrastructure
TAC	Type Approval Code
TEA	TETRA Encryption Algorithm
TEI	TETRA Equipment Identity
TEI	TETRA Equipment Identity
TETRA	Terrestrial Trunked Radio
UAK	User Authentication Key
USRP	Universal Software Radio Peripheral
VDB	Visitor Database

Part I

Introduction

Chapter 1

Introduction

1.1 Scope of thesis

TETRA is an ETSI standard for a mobile communication system designed to be used in private mobile radio environment as a general national safety communication network. This thesis is going to identify the key security features of TETRA system which include authentication, encryption and key management. In particular, the thesis focuses on analyzing the authentication protocol. A formal security analysis of authentication protocol will be done. The consequences of possible attacks will be discussed and attack scenarios will be provided.

1.2 Work methods used

Background knowledge was studied from relevant literature and books. The security protocols were studied from ETSI standards.

The Scyther tool is used to analysis the TETRA authentication protocol. It is a tool used for automatic verification of security protocols.

1.3 Outline of the Thesis

The thesis is written at a level that expects the reader to have general technical understanding of telecommunication networks. Some background knowledge of information security is required. This thesis content five parts, and each part includes one or more chapters. The depth and detail varies from section to section, and some parts are covered into more details than others.

- PartI Introduction
- PartII Background information of TETRA system
This part will cover services provided by TETRA system and architecture of TETRA networks.
- Part III Security features of TETRA
This part will describe security feature of TETRA system including authentication, key management, air interface encryption and end-to-end encryption.
- Part IV Analysis of authentication protocol
This part will cover information of the automatic security protocol verification tool, and present the result of verification. Possible attack scenarios will also be discussed.

- Part V Conclusion

Part II

Background

Chapter 2

Services of TETRA

Trunking, is the technique used in the radio systems to expand the availability of communication resources. i.e. all users have automatic access to all channels. Two assumptions of the system is that the average message is short and many stations need to communicate simultaneously is not likely to happen[1].

The TETRA standard supports three types of trunking methods: message trunking, transmission trunking and quasi-transmission trunking [2]. In message trunking, a radio channel is assigned for the entire duration of the conversation while in transmission trunking a radio channel is assigned only for the duration of a single half-duplex radio transmission. Since each transmission in a message must obtain a new voice channel, the subscriber may experience delays in “busy hours”. In quasi-transmission trunked system the requests from a recently terminated group have higher priority over other requests [1]. This mechanism guarantees message continuity for end users.

TETRA is a standard developed by ETSI for Private Mobile Radio (PMR) environment. It operates in the frequency range from 150 MHz to 900MHz and it is capable to offering a bit rate up to 28.8 kb/s [2].

2.1 Services of the TETRA System

ETSI specified three operation modes for TETRA[3]:

- Circuit mode (Voice plus Data) (ETS 300 392 series), that provide circuit switched speech and data transmission.
- Packet Data Optimized (PDO) (ETS 300 393 series), that provide data traffic based on packet switching.
- Direct Mode Operation (DMO) (ETS 300 396 series), where voice transmission between two terminals without using a network.

TETRA services can be divided into teleservices and bearer services [2]:

- Bearer services provide information transfer between network interfaces using only low layer functions.
- Teleservices provide complete capability for communication including terminal functions.

Some of the main Bearer services include [3]:

- **User Status Transmission** used to transfer short, predefined messages from user to the dispatching control or vice versa.

- **Short Data Service** transmits short text messages between users comparable to SMS in GSM.
- **Circuit Switched Data Services** in unprotected mode or encryption mode.
- **Packet Switched Data Services** based on TCP/IP or X.25 protocol.

Some of the main Teleservices include [4]:

- **Individual call** Point-to-point connection between two subscribers
- **Group call** Point-to-multipoint connection between calling subscriber and a group called through a common group number. It employs Half-duplex mode through the *push-to-talk* switch.
- **Broadcast call** Point-to-multipoint connection in which the subscriber group dialed through a broadcast number can only hear the calling subscriber.
- **Acknowledged group call** In a group call the presence of the group members is confirmed to the calling subscriber using an acknowledgment.
- **Direct Mode (DMO)** Point-to-point connection between two mobile terminals without using of the TETRA network.

Chapter 3

TETRA Network Architecture

This chapter introduce interfaces and components in a TETRA network. Architecture of a TETRA network is shown in figure 3.0.1.

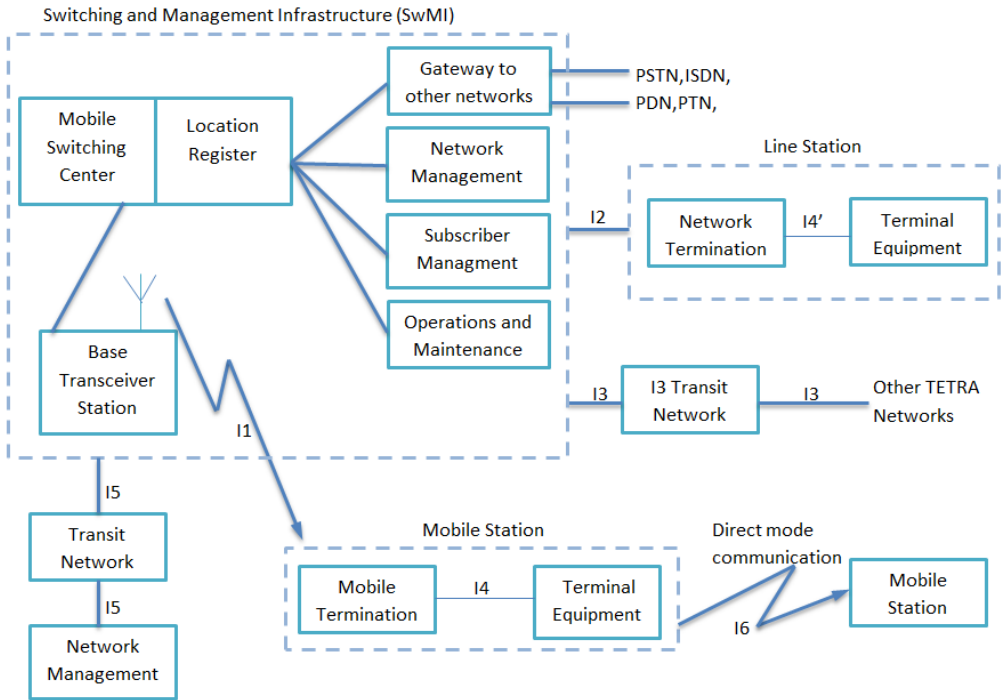


Figure 3.0.1: TETRA network architecture[2, 6]

3.1 Interfaces

The interfaces in a TETRA network are [2]:

- I1 = Radio air interface
- I2 = Line station interface
- I3 = Inter-system interface. This interface allows interconnection of TETRA networks from different manufacturers.

- I4 = Terminal equipment interface for a mobile station
I4' = Terminal equipment interface for a line station
- I5 = Network management interface
- I6 = Direct mode interface

3.2 Network Components

Functional structure of a TETRA network includes [4]:

- **Mobile Station (MS)** comprises subscribers physical equipment, a *Subscriber Identity Module* (SIM) and a *TETRA Equipment Identity* (TEI) specified to each device. TEI is input by the operator which means a stolen device can be disabled immediately.
- **Line Station (LS)** has similar structure as a mobile station but with the switching and management infrastructure connected over ISDN. It provides the same function and services as a mobile station.
- **Switching and Management Infrastructure (SwMI)** contains base stations that establish and maintain communication between mobile stations and line stations over ISDN. It allocates channels, switches calls and contains databases with subscriber's information.
- **Network Management Unit** provides local and remote management functionality [2].

- **Gateways** interconnect a TETRA network with a non-TETRA network such as PSTN, ISDN and PDN. Translation or conversion of information formats and communication protocols might be necessary [2].

3.3 Operation of the TETRA System

3.3.1 Identities

Identities used to distinguish communication parties in a TETRA network. Equipment is manufactured with the TETRA Equipment Identity (TEI) which is similar with the *International Mobile Equipment Identity* (IMEI) used in GSM. The contents of TEI presented in figure 3.3.1.

Type Approval Code (TAC)	Final Assembly Code (FAC)	Electronic Serial Number (ESN)	Spare (SPR)
24 bits	8 bits	24 bits	4 bits

Figure 3.3.1: The TETRA Equipment Identity[22]

The Mobile Network Identity (MNI) identifies different TETRA networks, and base station will broadcast its MNI. The MNI includes country code and operator information.

The TETRA Subscriber Identity (TSI) is used to relate subscribers with their services and billing. The contents of TSI is shown in figure 3.3.2. The subscriber identity module (SIM) is inserted in terminal equipment. A unique service of TETRA system is group call. In addition to individual TSI (ITSI), there is also group TSI (GTSI).

Mobile Country Code (MCC)	Mobile Network Code (MNC)	Short Subscriber Identity (SSI)
10 bits	14 bits	24 bits

Figure 3.3.2: The TETRA Subscriber Identity[22]

3.3.2 Mobility Management

Mobility management in TETRA is similar with that in the GSM. The home database (HDB) holds information of MSs such as user identities, cipher keys and subscribed services. In the visited network, authentication take place through HDB, and essential user information is downloaded to the visitor database (VDB).

Part III

TETRA Network Security

Chapter 4

Security Features

This chapter covers a description of security features in TETAR including authentication, key management and encryption mechanisms.

4.1 Security Requirements

A secure communication network should provide Confidentiality, Integrity, Authentication, Nonrepudiation and Reliability [5].

Confidentiality Only authorized users should have access to the information being exchanged.

Integrity Only authorized users should be able to modify the information being exchanged.

Authentication The identity of the sender can be verified by the receiver.

Nonrepudiation The sender cannot deny the message he sent.

Reliability The service and resources are available and not denied to authorized users.

4.2 Authentication in TETRA

Like in other communication systems, authentication is the fundamental security service in TETRA. Authentication is the process that parties participated in the communication proving they are who they claimed to be. If public key certificates are used, one's identity could be proved by the certificate signed by an authority. When using symmetric key cryptography, one can only trust parties share the same secret with him, and only with the same secret can they communicate. The authentication in TETRA is based on proving knowledge of the same secret shared between a mobile station and the authentication centre (AuC).

4.2.1 Authentication Key

Basic Concepts

Two classes of algorithms for cryptography are symmetric key algorithms and asymmetric key algorithms. In symmetric key algorithms, sender and receiver share the same secret key. When apply symmetric key algorithms in systems with multiple nodes, a leakage of secret key at any node will cause the whole system to be insecure. Therefore periodically update of the secret key is desirable. Block ciphers and steam ciphers are two types of symmetric encryption

schemes. Block cipher scheme divided plaintext into fixed-length "blocks" and encrypt one block at a time [11]. A stream cipher operates with one plaintext digit at a time. A secret key initials the creation of a pseudorandom sequence (keystream) which will be used to combine with the plaintext [12].

In asymmetric key (or public key) algorithm, the encryption and decryption processes are using different keys. Public-key cryptography makes use of mathematical functions instead of substitution and permutation and it is computationally infeasible to derive one of the keys from the cryptographic algorithm and the other key [7]. Each user has a pair of keys, one public key and one private key. Public key is made public while private key is kept private. When two parties try to communicate, the sender needs to know the receiver's public key and encrypt message with it. The receiver can decrypt the message with its private key.

Authentication Key Generation

Authentication in TETRA uses symmetric keys. The mobile station will be assigned an User Authentication Key (UAK) when register to the network for the first time. The UAK is stored in terminal equipment's SIM card as well as in the authentication centre database. The authentication key, K , is the knowledge of which has to be demonstrated for authentication [8]. It could be generated in three ways shown in Figure 4.2.1. K may be generated from Authentication Code (AC), which is a pin code entered by the user, using the TB1 algorithm. K may also be generated from UAK stored in

the SIM card with the algorithm TB2. Using both AC and UAK to generate K is the third method labeled TB3. The length of K, KS and KS' are all 128 bits. K will not be used directly in the authentication process but to generate session keys: KS and KS'.

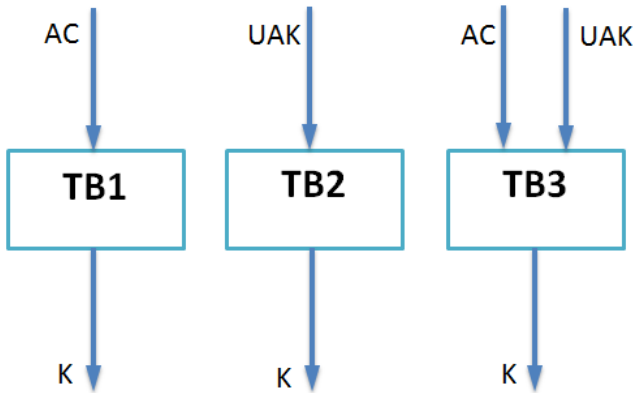


Figure 4.2.1: Generation of authentication key to MS[8]

4.2.2 Authentication Procedures

In TETRA, authentication services include authentication of MS by SwMI, authentication of SwMI by MS and mutual authentication [8]:

Figure 4.2.2 shows the procedure of infrastructure authenticates a mobile station. The infrastructure may include authentication centre and base station. RS is a random seed used together with the authentication key K to generate a session key KS. The algorithm used is TA11, and this procedure will be performed by the authentication centre of the home system.

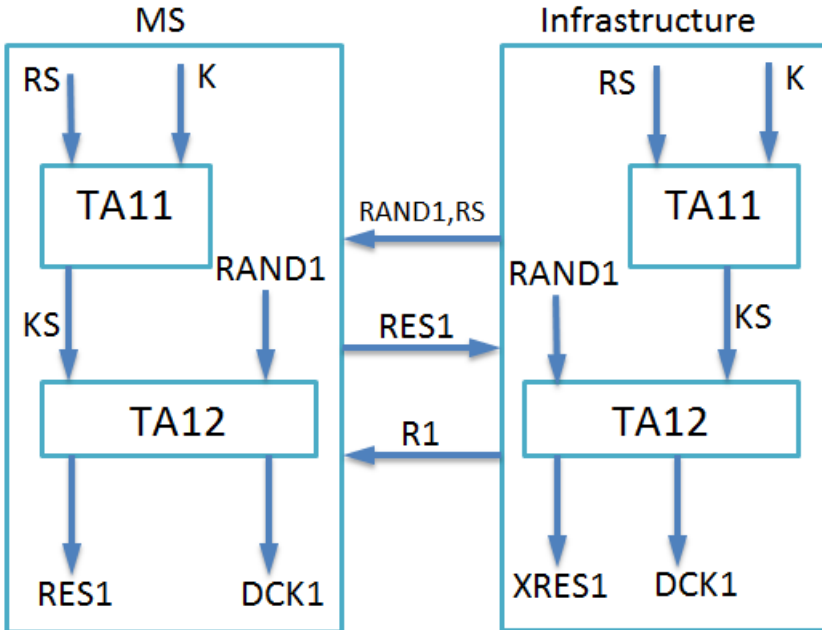


Figure 4.2.2: Authentication of a user[8]

A random number RAND1 is generated by the infrastructure and sent to the MS as a challenge. The MS will compute its response RES1 using the session key KS and the algorithm TA12. This procedure also generated DCK1 which is a part of the derived cipher key (DCK). The infrastructure shall compare RES1 with the expected response XRES1, and the authentication result R1 will be set to TRUE or FALSE based on whether RES1 equals XRES1.

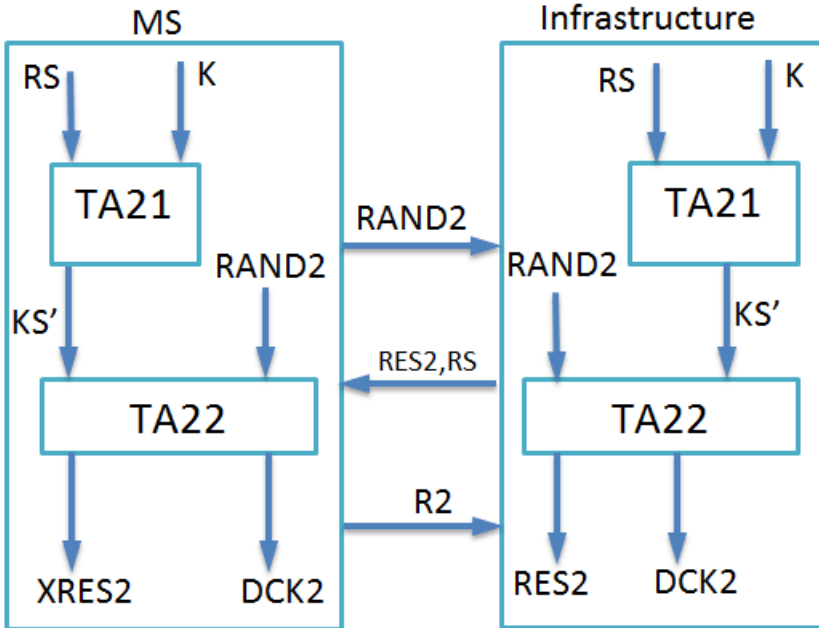


Figure 4.2.3: Authentication of the infrastructure[8]

Figure 4.2.3 indicates the procedure of a MS authenticates the infrastructure which is similar to the procedure described above. The algorithms TA11 and TA12 will be replaced with TA21 and TA22 respectively. The session key KS' also is different from KS . The other part of the derived cipher key $DCK2$ will be generated.

The TETRA system supports mutual authentication between the MS and the infrastructure. The mutual authentication will start as an one way authentication and the challenged party will decide whether to make the authentication mutual. The second authentication will only perform when the first authentication is successful.

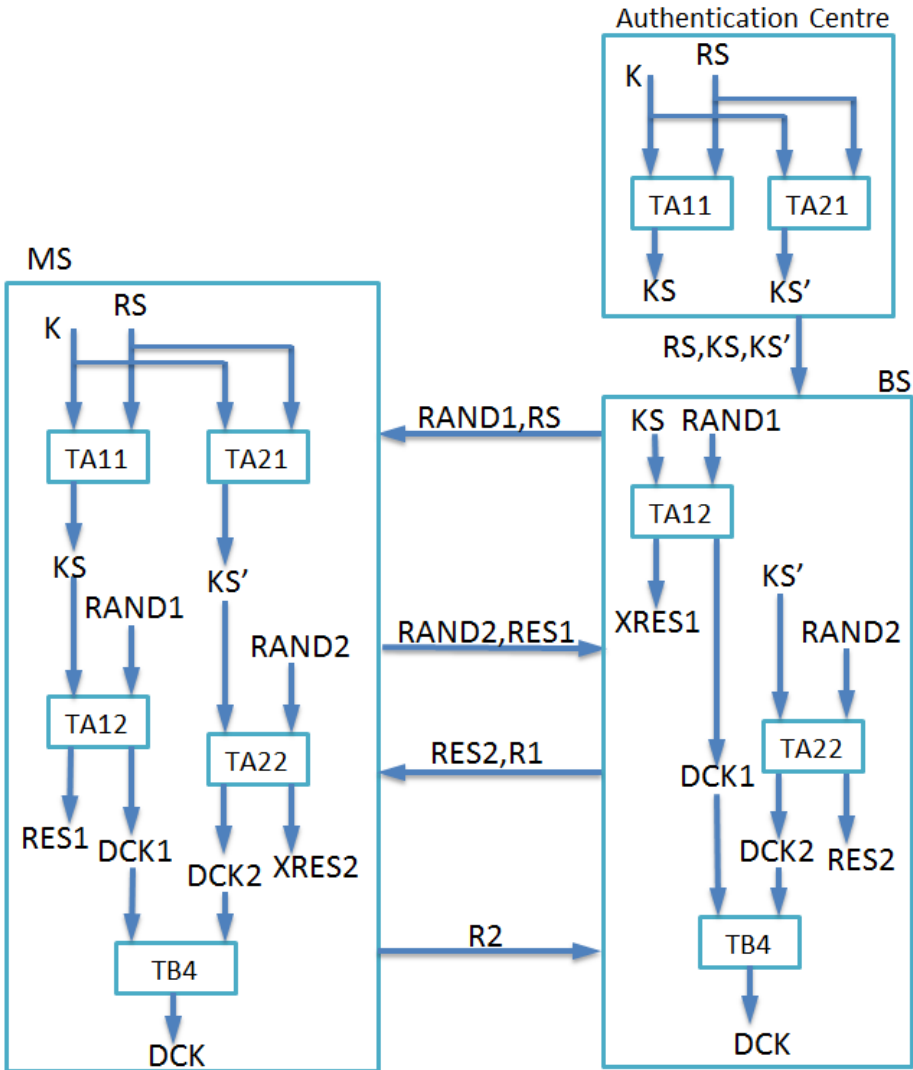


Figure 4.2.4: Mutual Authentication Initiated by Infrastructure[8]

Figure 4.2.4 shows the authentication procedure of mutual au-

thentication initiated by the infrastructure.

- Authentication centre matches user's authentication key K with its identity $ITSI$. The authentication key K and a random seed RS generate a pair of session keys KS and KS' through algorithms $TA11$ and $TA21$.
- The session keys and RS are then sent to the base station. Base station generates a random number $RAND1$ and sends it to the MS together with RS .
- MS generates session keys and compute the response $RES1$ which shall be sent back to the base station. If the user decided to make the authentication mutual, it will also generate and send a random number $RAND2$ to the base station.
- The base station compared $RES1$ with $XRES1$, and if the two values are equal the base station shall compute $RES2$ using $TA22$. Return $RES2$ and $R1$ equals $TRUE$.
- $RES2$ is compared with $XRES2$ by the MS and if the same, the MS will return $R2$ equals $TRUE$. Mutual authentication is completed.
- $DCK1$ and $DCK2$ produced during the procedure shall be inputs to the algorithm $TB4$ to generate the derived cipher key (DCK).

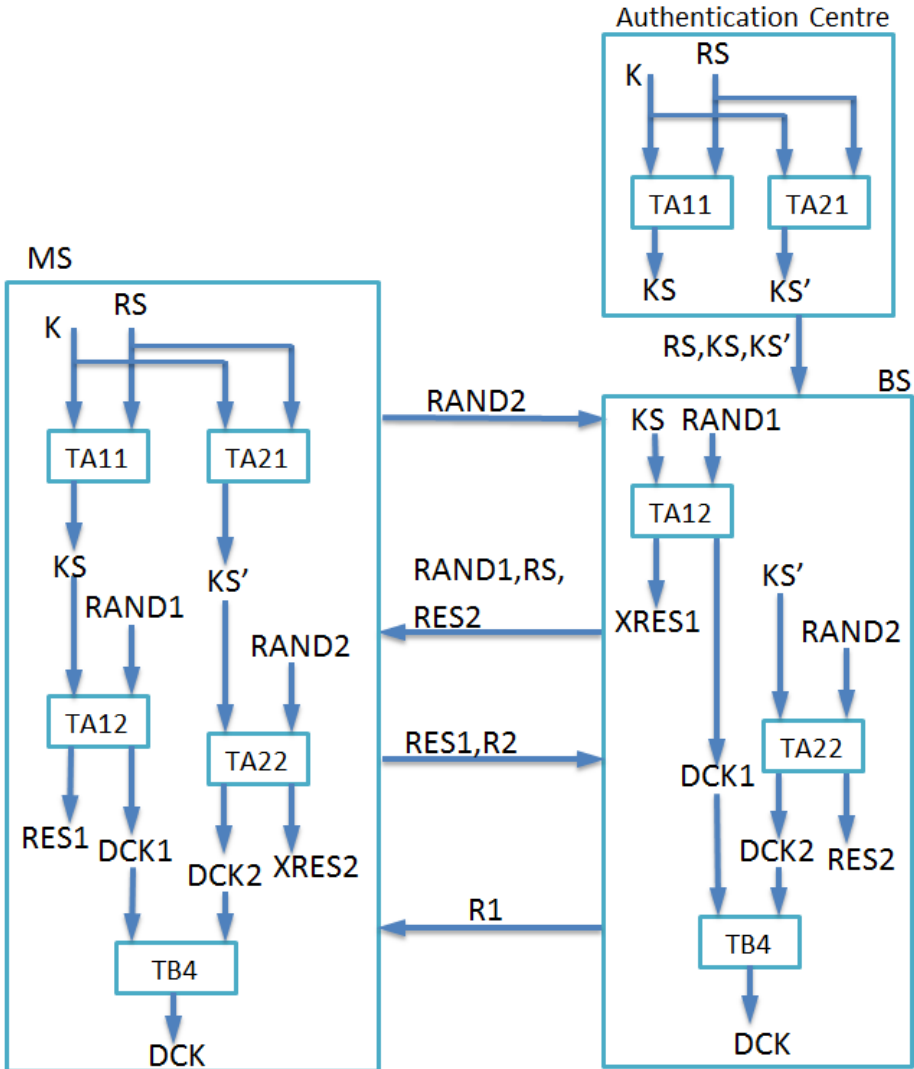


Figure 4.2.5: Mutual Authentication Initiated by a User[8]

The mutual authentication initiated by the MS indicated with figure 4.2.5. It is very similar with the procedure described above.

4.2.3 Authentication PDUs

As data passing from the user application layer down to layers of protocols, each layer adds a header containing protocol information. These headers are called Protocol Data Units (PDUs). The PDUs used in the TETRA authentication process are shown below [8]:

PDU Type	Authentication sub-type	Random Challenge (RAND1)	Random Seed (RS)	Proprietary element
4 bits	2 bits	80 bits	80 bits	...

Figure 4.2.6: D-AUTHENTICATION DEMAND[8]

The PDU in figure 4.2.6 is used by the infrastructure to initiate an authentication.

PDU Type: D-AUTHENTICATION DEMAND=0001₂

Authentication sub-type: D-AUTHENTICATION DEMAND=00₂

Random Challenge (RAND1): an 80-bit number

Random Seed: an 80-bit number

Proprietary element: an optional with variable length for proprietary defined information.

PDU Type	Authentication sub-type	Response Value (RES1)	MAF	Random Challenge (RAND2)	Proprietary element
4 bits	2 bits	32 bits	1 bit	80 bits	...

Figure 4.2.7: U-AUTHENTICATION RESPONSE[8]

The PDU used by the user to response an authentication demand is shown in figure 4.2.7.

PDU Type: D-AUTHENTICATION RESPONSE=0000₂

Authentication sub-type: D-AUTHENTICATION DEMAND=01₂

Response Value (RES1): an 32-bit value calculated from the challenge

MAF (Mutual Authentication Flag): a flag indicates the PDU includes (1) mutual authentication elements or not (0)

Random Challenge (RAND2): an 80-bit number

PDU Type	Authentication sub-type	Authentication Result (R1)	MAF	Response Value (RES2)	Proprietary element
4 bits	2 bits	1 bits	1bits	32 bits	...

Figure 4.2.8: D-AUTHENTICATION RESULT[8]

The PDU used by the infrastructure to return the result of authentication is shown in figure 4.2.8.

PDU Type: D-AUTHENTICATION RESPONSE=1110₂

Authentication sub-type: D-AUTHENTICATION DEMAND=10₂

Authentication Result (R1): a flag indicates success (1) or failure (0) of an authentication

Response Value (RES2): a 32-bit value response to the challenge when MAF being set

4.2.4 Authentication Algorithm

One available standard set of Authentication and Key Management Algorithms from the TETRA MoU is TAA1 [16]. It was developed by the security algorithm group of experts (SAGE) and can

be obtained under a 'Non-disclosure and restricted usage license' from ETSI. The rules for the management of the TETRA standard encryption algorithm TAA1 is specified in [17].

4.2.5 Analysis of Authentication Protocol

During authentication, the authentication key K is never directly used or transmitted over the air. Instead, session keys are used in the authentication. This mechanism protects the authentication key K .

There are three random numbers involved in the mutual authentication process: RS , $RAND1$ and $RAND2$. The use of three random numbers makes it difficult to perform a message replay attack.

A vulnerability of the authentication protocol is that there is no data integrity protection of the authentication messages. A false base station could simply intercept and modify authentication messages which will cause the authentication fail. Based on the intention of the attacker, the same basic attack can have different effects to the system and users.

Setting up false base station and modifying authentication messages of individual users might not seem to be an effective way of attack. Probably a simpler denial of service attack would be jamming the radio path by sending high power signal on the frequencies used by TETRA. However, such kind attacks can be easily detected, false base station could be used to isolate target mobile stations and prevent them from communication.

The security level of authentication depends highly on algorithms:

TA11, TA12, TA21 and TA22. As stated in [13] when a MS roams to another TETRA network, it is not a wise solution to transfer the authentication key to the visited network. If transfer certain information that can be used for one single authentication, it might cause too much overhead in TETRA system. The suggestion is to transfer a session key that can be used for repeated authentication. If the transmission channel between authentication centre and the base station is insecure, the session keys KS and KS' could be intercepted by an attacker. The random seed RS transmitted between the base station and the mobile station could also be intercepted. Then a known plaintext attack could be performed where the plaintext is RS and the ciphertext is KS and KS'. As the attacker collected enough pairs of RS and KS/KS', he might be able to gain a good knowledge of the algorithms TA11 and TA21.

4.3 Key Management

4.3.1 Air Interface Key management mechanisms

Keys managed in the TETRA system include [8]:

Derived Cipher Key (DCK)

As mentioned in section 4.2.2, parts of derived cipher key DCK1 and DCK2 are generated after successful authentications. As inputs of the algorithm TB4, DCK1 and DCK2 derive the DCK. In case the mutual authentication is not performed, either DCK1 or DCK2 will be missing. The missing value shall be set to zero.

DCK used to protect the data, voice and signaling transmitted between the MS and the infrastructure after authentication.

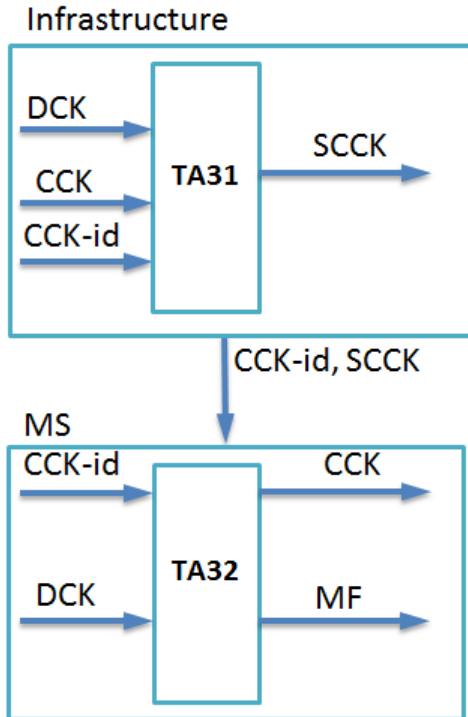


Figure 4.3.1: Distribution of a common cipher key[8]

Common Cipher Key (CCK)

For every Location Area (LA), a common cipher key shall be generated and distributed to each MS by the infrastructure. DCK is used in this process as the sealing key; together with algorithm TA31 the Sealed Common Cipher Key (SCCK) is generated. CCK Identifier (CCK-id) is distributed along with the key since the in-

infrastructure may update the CCK from time to time. The value of CCK-id shall be incremented for each new key. The manipulation flag MF indicates whether a sealed cipher key has been manipulated. The process is shown in the figure 4.3.1.

Group Cipher Key (GCK)

The group cipher key is generated and distributed to mobile stations in a group by the infrastructure. The GCK will not be used on the air interface directly. It will be modified by CCK or SCK to provide a Modified GCK (MGCK). The algorithm used in this process is TA71. The MGCK is used for encryption of group calls.

Static Cipher Key (SCK)

The static cipher key is known to both infrastructure and the MS, the value of SCK shall never change. A terminal could store up to 32 SCKs. The SCK could be used in systems that do not implement authentication. It could also be used for encryption in the Direct Mode operations.

4.3.2 Over The Air Re-keying

Over The Air Re-keying (OTAR) is a way the infrastructure transfer sealed cipher key (CCK, SCK or GCK) to mobile stations over the air interface. The transfer of CCK and GCK are both protected by DCK while SCK is sealed with the KSO (Session Key for OTAR) [8]. KSO is generated from a user's authentication key and a random seed.

4.4 Encryption

4.4.1 Classes of Security

The TETRA system provides three different security classes:

- class 1: No encryption
- class 2: The Static Cipher Key encryption
- class 3: The Dynamic Cipher Key encryption

4.4.2 Air Interface Encryption

Since anyone could listen to air channels during the communication of MS and BS, it is important to encrypt information transmitted over the air. Encryption is a method to make sure the intercepted information is not intelligible to anyone other than intended receiver. Air interface encryption handled in the upper part of the MAC layer, and the MAC headers left unencrypted [8].

Air Encryption Process

Air interface encryption realized using an encryption algorithm implemented in a Key Stream Generator (KGS) [8]. As shown in figure 4.4.1, the KGS has two inputs, an Initial Value (IV) and a cipher key and one output as a Key Stream Segment (KSS). The ciphertext obtained by modulo-2 addition (XORed) the KSS bits with plaintext.

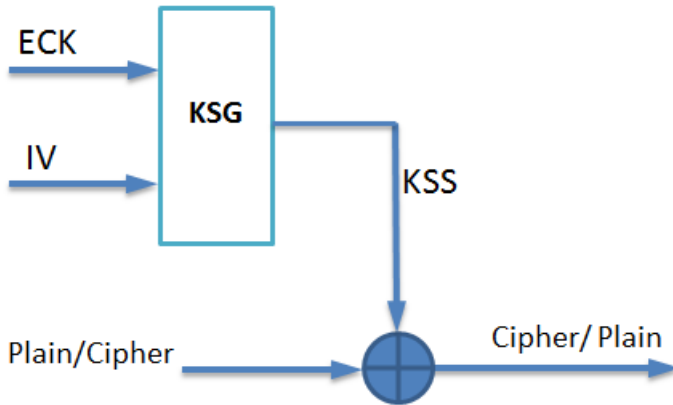


Figure 4.4.1: Speech and control information encryption[8]

The Initial Value (IV) is 29-bit data with composition of slot number (2 bits), frame number (5 bits), multiframe number (6 bits), hyper-frames (15 bits) and a final bit indicates downlink transmission (0) or uplink transmission (1).

As shown in figure 4.4.2 the Encryption Cipher Key (ECK) is derived from a selected Cipher Key (CK) which could be one of SCK, DCK, MGCK or CCK [8]. The CK will be modified by the Carrier Number (CN), LA-id, and Colour Code (CC) using the algorithm TB5.

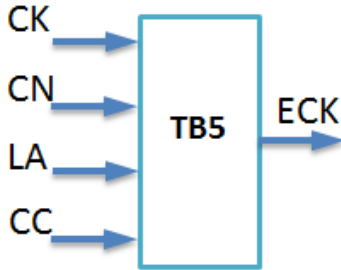


Figure 4.4.2: Generation of ECK[8]

Air Encryption Algorithm The TETRA encryption algorithm (TEA) is used on the air interface. The TETRA MoU recommends a number of possible algorithms [18, 19, 20, 21] for different commercial requirements. TEA2 and TEA3 are restricted export algorithms that primarily designed for public safety organizations, while TEA1 and TEA4 are readily exportable algorithms [16]. The algorithm specifications can be obtained under a 'Non-disclosure and restricted usage licence' from ETSI.

4.4.3 End-to-End Encryption

Air interface encryption described above protects the communication between mobile stations and the base station. The end-to-end encryption also protects the transmission of the information through networks (BS to MSC, MSC to MSC and links within the TETRA infrastructure). Information encrypted by the sender and only be decrypted by the receiver.

End-to-end encryption and key management are not specified in

the TETRA standard. The specification [9] only describes the mechanism for synchronization shown in the figure 4.4.3. The TETRA system does not participate in key generation and management; it only provides the transmission channels.

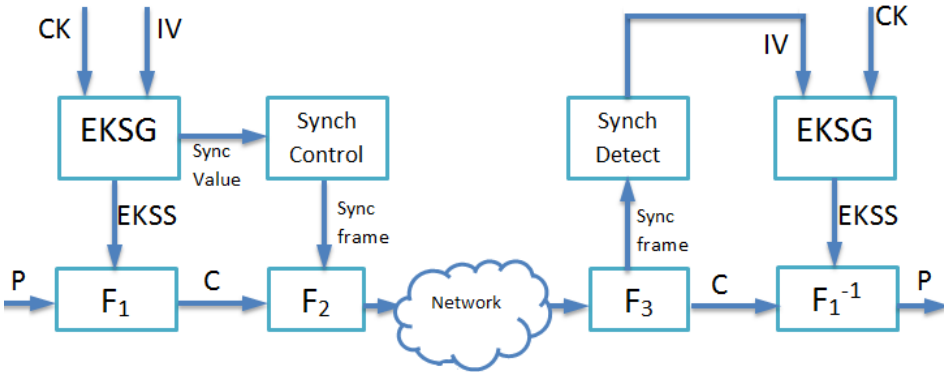


Figure 4.4.3: Synchronization [9]

The End-to-end Key Stream Generator (EKSG) generates a key stream segment EKSS with inputs: CK (cipher key) and IV (initialization value). To prevent "recording and replay", the IV should be a time variant parameter used to initialize synchronization of the encryption units.

The function F_1 combines the plaintext bit stream with EKSS producing an encrypted ciphertext bit stream. The inverse of this process is combined ciphertext bit stream with EKSS through the function F_1^{-1} to get plaintext bit stream.

The function F_2 replaces a half slot of the ciphertext bit stream with a synchronization frame generated from the "Synch Control" functional unit.

The function F_3 recognizes the synchronization frame and transmits it to the "Synch Detect" functional unit.

4.5 Replay Prevention

Replay attack is a form of attack when adversary intercepts other user's message and retransmits it [15]. In the wireless network, it is possible for attacker to sniff packets. Even the attacker could not get any information from the encrypted messages; he can still perform attacks by replaying old messages from eligible users. The most common countermeasures include using sequence numbers or timestamps. Synchronization of the network needs to be achieved first.

In TETRA, the standard [8] states the importance of having protection mechanism against replay attack. A time variant initialization value or a time variant cipher key is suggested to be used. Examples of using call-id or a shared real time clock to prevent recording and replaying an entire call also mentioned in the standard. Specification of approaches is outside the scope of this standard.

Part IV

Analysis of the Authentication Protocol

Chapter 5

Analysis of Security Protocol

Cryptographic protocols use cryptographic primitives to provide security communication over insecure networks. It might be easy to understand a protocol from an informal level; it is extremely difficult for human to correctly verify a protocol due to the complexity of protocol analysis. Automatic tools are preferred in protocol analyzing. Some of the best known protocol analysis methods include BAN logic [23], applied pi calculus [32], strand space approach [34] and multiset rewriting [35]. Also, there are several automatic tools for verification of security protocols, such as proVerif [36], avispa [37], CPSA tool [38] and NRL analyser [39]. The Scyther tool is chosen to analysis the security of TETRA authentication protocol in this thesis.

5.1 The Scyther Tool

The Scyther tool is designed for automatic verification of security

protocols, and it is freely available for Windows, Linux, and Mac OS X. It can be downloaded from [24]. The components required to be installed first include the GraphViz library [25], Phthon [26] and wxPython libraries [27]. The graphical user interface is written in Python and Scyther starts when executing the `scyther-gui.py` file [28].

5.2 Security Protocol Specification

A security protocol specification describes the communication parties, the protocol events to be executed, the order of the events and initial knowledge required for communication parties. The use of the protocol must follow these "rules" states in a protocol specification. Some basic concepts used in protocol specification are [29]:

Roles The protocol analysis model defined in Scyther is a role-based security protocol model, roles are defined as a number of behaviours. There might exist several communication agents in a system and each agent executes instances of one or more roles. Each instance is called a run. To describe roles, Role Terms shall be used. The basic term sets are shown in Table 5.1. Variable used to store received value, and `fresh` denotes freshly locally generated value. The `sk(i)` and `pk(i)` denotes the secret key and public key of role `i` in asymmetric encryption, and `k(i,r)` denotes the symmetric key shared between role `i` and role `r`.

Table 5.1: Basic Term Sets[29]

Description	Set	Typical elements
Role terms	RoleTerm	rt_1, rt_2
Variables	Var	X, Y, Z
Fresh values	Fresh	sessionkey
Roles	Role	i, r, s
Functions	Func	h
Function application		$h(m)$
Long-term keys		$sk(i), pk(i), k(i,r)$

Events The role events are events that can be executed by a role. Events include sending and receiving of messages and security claims. For the role R, $Send_{Label}(R, R', rt)$ denotes R sending rt to R' . $Recv_{Label}(R, R', rt)$ denotes R' receive rt sent by R. Each send and receive event has a label that marks corresponding send and receive events. $Claim_{Label}(R, c, rt)$ or $claim_{Label}(R, c)$ denotes the security goal c is expected to hold with optional parameter rt .

Runs Roles could be executed any number of times by agents, and execution of a role called a run. Turning a role description into a run refers to as instantiation. Each run is assigned with a unique run identifier. Fresh value, roles, and variables that are local to a run are extended with the run identifier. Table 5.2 indicates basic run term sets.

Table 5.2: Basic Run Sets [29]

Description	Set	Typical elements
Run terms	RunTerm	$t1, t2$
Instantiated constants		$ni\#1, nr\#2, sessionkey\#1$
Agents	Agent	A, B, C, S, E

Traces The semantics of a security protocol is expressed as a set of traces where each trace is an interleaving of a number of runs [31].

5.3 Security Properties

In Scyther, security properties are integrated into the protocol specification by claim events [29]. Claim events happened based on the local view of each agent, security properties include [29]:

Secrecy $\text{Claim}_L(\mathbf{R}, \text{secret}, \text{rt})$ is a secrecy claim event which denotes for the role \mathbf{R} , rt should not be known to the adversary. The definition of a secrecy claim is true if and only if roles are mapped to honest agents for each run, and the claimed secret term should not be inferable from the knowledge of the adversary.

Aliveness The least requirement in authentication is there exist a communication partner in the network. Generic Aliveness in a claim event is written as $\text{claim}(\mathbf{R}, \text{alive}, \mathbf{R}')$ with role \mathbf{R} and \mathbf{R}' . To satisfy generic aliveness with the role \mathbf{R}' , the agent executing the role \mathbf{R} thinks he is communicating with an trusted agent and the intended communication partner has actually executed an event [30].

Synchronisation Base on the basic requirement of aliveness in communication, synchronisation requires the entire message exchange exactly as specified by the protocol description. Messages were indeed sent and received by the communication part-

ner. The cast function was introduced to identify which runs perform which roles since run events associated with different roles may belong to the same agent.

Ni-Synch Ni-Synch stands for Non-injective Synchronisation. Ni-Synch property requires that the corresponding send and receive events (1) are executed by the runs indicated by the cast function, (2) happened in the correct order, and (3) have the same contents. Protocol satisfying non-injective synchronisation may still be suffered from message replay attacks in which case property Injective Synchronisation is introduced.

Ni-Agree Ni-Agree stands for Non-injective Agreement. Agreement ensures the communication parties agree on the value of variables after execution of the protocol. The difference between Non-injective agreement and Non-injective synchronisation is that Ni-Agree focus on the correct contents of the message while Ni-Synch also requires messages executed in the expected order. If a protocol satisfies synchronisation it satisfies agreement also.

5.4 Verification Algorithm

According to Cremers and Mauw in the book [29], the algorithm used by Scyther to analysing the security properties of a protocol is based on the analysis of trace patterns. Trace patterns are introduced to capture the concept of similar behaviours from the perspective of property verification. Trace patterns are defined as

partially ordered set of symbolic events. The required properties of traces to evaluate security properties include event order, the equivalence of events and messages, positive occurrence of agent events and contents of the adversary knowledge.

If there exist traces exhibit attack pattern and violates the security property, the claimed security property is fail. If there is no trace exhibits the attack pattern, the security property is hold. Verification procedure determines if any trace contents attack patterns.

5.5 Adversary Models

A formal security protocol analysis model should include a description of the adversary's capabilities, and one of the mostly used threat model is the Dolev and Yao network threat model [40]. In this model the entire communication network is under complete control of the adversary. The adversary can replay, remove, split, reroute, reorder, intercept and learn the content of any messages passing through the network. The honest communication parties can only send and receive messages through the adversary. Cryptographic primitives are assumed to be black boxes, which means only with the knowledge of keys, can the adversary encrypt and decrypt messages.

The adversary model of Scyther has some additional capabilities that mentioned in [29]. The adversary is possible to learn the long-term keys, session keys and random values of an agent.

Chapter 6

Verification of TETRA Authentication Protocol

6.1 TETRA Authentication Protocol Specification

Authentication in TETRA involves three parties: the authentication centre (AuC), the mobile station (MS) and the base station (BS). A detailed description of the authentication protocol could be found in the section 4.2. To summaries the messages exchanged between parties in a mutual authentication:

1. MS→AuC: UserID
2. AuC→BS: RS, KS, KS'
3. BS→MS: RAND1, RS
4. MS→BS: RES1, RAND2

5. BS→MS: RES2, R1
6. MS→BS: R2

Assume the communication channel between AuC and BS is secure, the party AuC and BS can be combined as one role (SwMI). The simplified process then becomes like:

1. MS→SwMI: UserID
2. SwMI→MS: RAND1, RS
3. MS→SwMI: RES1, RAND2
4. SwMI→MS: RES2, R1
5. MS→SwMI: R2

Figure 6.1.1 illustrates the protocol specification using Message Sequence Charts (MSC).

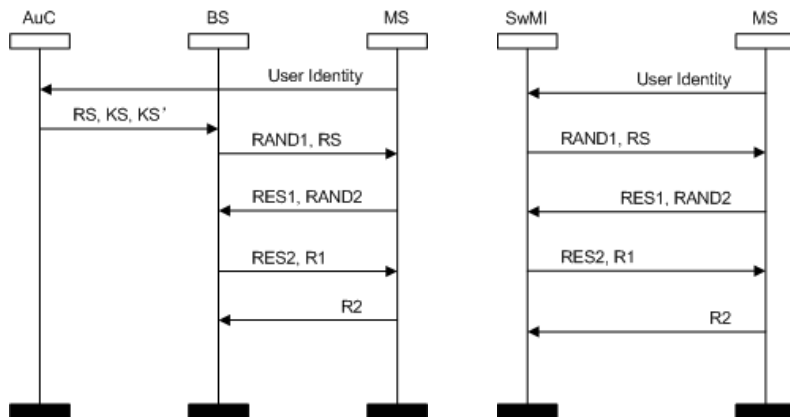


Figure 6.1.1: MSC of TETRA Authentication Protocol

The security requirements for authentication :

1. For both roles, the long-term shared symmetric key between MS and SwMI should not be revealed to an adversary. Model as claim events:

```
claim_SwMI(SwMI,Secret,k(MS,SwMI));
claim_MS(MS,Secret,k(MS,SwMI));
```

2. The session keys KS and KS' should not be revealed to an adversary. Model as claim events:

```
claim_SwMI(SwMI,Secret,TA11(k(MS,SwMI),RS));
claim_SwMI(SwMI,Secret,TA21(k(MS,SwMI),RS));
claim_MS(MS,Secret,TA11(k(MS,SwMI),RS));
claim_MS(MS,Secret,TA21(k(MS,SwMI),RS));
```

3. The derived cipher key (DCK) should not be revealed to an adversary. Model as claim events:

```
claim_SwMI(SwMI,Secret,TB4(TA12b(TA11(k(MS,SwMI),RS),RAND1),
    TA22b(TA21(k(MS,SwMI),RS),RAND2)));
claim_MS8(MS,Secret,TB4(TA12b(TA11(k(MS,SwMI),RS),RAND1),
    TA22b(TA21(k(MS,SwMI),RS),RAND2)));
```

4. For both roles, the claim of aliveness should hold.
5. The MS and the SwMI should agree on all the value of variables exchanged.
6. For both roles the requirement of non-injective synchronisation should be satisfied.

Figure 6.1.2 and figure 6.1.3 on the next page specify one possible input to the Scyther tool.

```

usertype AuthenticationResult;
const R1: AuthenticationResult;
const R2: AuthenticationResult;
hashfunction TA11, TA12, TA12b, TA21, TA22, TA22b, TB4;

protocol TETRA (SwMI,MS)
{
  role SwMI
  {
    fresh RS: Nonce;
    fresh RAND1: Nonce;
    var RAND2: Nonce;

    recv_1(MS,SwMI, MS);
    send_2(SwMI,MS, RAND1, RS);
    recv_3(MS,SwMI, TA12(TA11(k(MS,SwMI),RS), RAND1), RAND2);
    send_4(SwMI,MS, TA22(TA21(k(MS,SwMI),RS),RAND2), R1);
    recv_5(MS,SwMI, R2); claim(SwMI,Running,MS,R1,R2);

    claim_SwMI1(SwMI,Secret,k(MS,SwMI));
    claim_SwMI2(SwMI,Secret,TA11(k(MS,SwMI),RS));
    claim_SwMI3(SwMI,Secret,TA21(k(MS,SwMI),RS));
    claim_SwMI4(SwMI,Niagree);
    claim_SwMI5(SwMI,Nisynch);
    claim_SwMI6(SwMI, Alive);
    claim_SwMI7(SwMI, Weakagree);
    claim_SwMI8(SwMI,Secret, TB4(TA12b(TA11(k(MS,SwMI),RS),RAND1),
    TA22b(TA21(k(MS,SwMI),RS),RAND2)));
    claim_SwMI9(SwMI,Commit,MS,R1,R2);
  }
}

```

Figure 6.1.2: Example input to the Scyther tool


```

role MS
{
  var RS: Nonce;
  var RAND1: Nonce;
  fresh RAND2: Nonce;

  send_1(MS,SwMI, MS);
  recv_2(SwMI,MS, RAND1, RS);
  send_3(MS,SwMI, TA12(TA11(k(MS,SwMI),RS), RAND1), RAND2);
  recv_4(SwMI,MS, TA22(TA21(k(MS,SwMI),RS),RAND2), R1);
  claim(MS,Running,SwMI,R1,R2);
  send_5(MS,SwMI, R2);

  claim_MS1(MS,Secret,k(MS,SwMI));
  claim_MS2(MS,Secret,TA11(k(MS,SwMI),RS));
  claim_MS3(MS,Secret,TA21(k(MS,SwMI),RS));
  claim_MS4(MS,Niagree);
  claim_MS5(MS, Alive );
  claim_MS6(MS, Weakagree );
  claim_MS7(MS,Nisynch);
  claim_MS8(MS,Secret,TB4(TA12b(TA11(k(MS,SwMI),RS),RAND1),
  TA22b(TA21(k(MS,SwMI),RS),RAND2)));
  claim_MS9(MS,Commit,SwMI,R1,R2);
}
}

```

Figure 6.1.3: Example input to the Scyther tool

6.2 Verification of TETRA Authentication Protocol

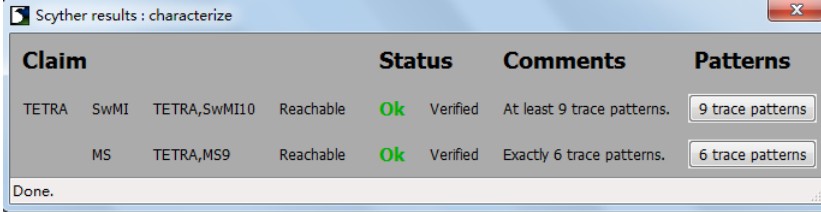
The security properties are modeled as local claim events, and the verification result from Scyther is shown in figure 6.2.1. The long-term shared symmetric key K , the session keys and the derived cipher key DCK are secret. The claim of generic aliveness is satisfied

which means the intended communication partner is alive and actually executing events. The security property weak agreement is hold means that the intended communication partner executes an event during the run in which the claim event occurs [29].

The Scyther tool also has found at least 6 attacks which means 6 security claims do not hold, it also provide user a list of trace patterns shown in figure 6.2.2.

Claim	Status	Comments	Patterns
TETRA_SwMI1 Secret k(MS_SwMI)	Ok	Verified	No attacks.
TETRA_SwMI2 Secret TA11{k(MS_SwMI),RS}	Ok	Verified	No attacks.
TETRA_SwMI3 Secret TA21{k(MS_SwMI),RS}	Ok	Verified	No attacks.
TETRA_SwMI4 Nagree	Fail	Falsified	At least 1 attack. <input type="button" value="1 attack"/>
TETRA_SwMI5 Nsynch	Fail	Falsified	At least 1 attack. <input type="button" value="1 attack"/>
TETRA_SwMI6 Alive	Ok	Verified	No attacks.
TETRA_SwMI7 Weakagree	Ok	Verified	No attacks.
TETRA_SwMI8 Secret TB{TA12b(TA11{k(MS_SwMI),RS),RAND1},TA22b(...	Ok	Verified	No attacks.
TETRA_SwMI9 Commit MS,R1,R2	Fail	Falsified	At least 1 attack. <input type="button" value="1 attack"/>
MS_TETRA_MS1 Secret k(MS_SwMI)	Ok	Verified	No attacks.
TETRA_MS2 Nsynch	Fail	Falsified	At least 1 attack. <input type="button" value="1 attack"/>
TETRA_MS3 Secret TA21{k(MS_SwMI),RS}	Ok	Verified	No attacks.
TETRA_MS4 Nagree	Fail	Falsified	At least 1 attack. <input type="button" value="1 attack"/>
TETRA_MS5 Alive	Ok	Verified	No attacks.
TETRA_MS6 Weakagree	Ok	Verified	No attacks.
TETRA_MS9 Secret TA11{k(MS_SwMI),RS}	Ok	Verified	No attacks.
TETRA_MS7 Secret TB{TA12b(TA11{k(MS_SwMI),RS),RAND1},TA22b(...	Ok	Verified	No attacks.
TETRA_MS8 Commit SwMI,R1,R2	Fail	Falsified	At least 1 attack. <input type="button" value="1 attack"/>

Figure 6.2.1: Scyther Verification Result



Claim				Status	Comments	Patterns
TETRA	SwMI	TETRA,SwMI10	Reachable	Ok Verified	At least 9 trace patterns.	9 trace patterns
	MS	TETRA,MS9	Reachable	Ok Verified	Exactly 6 trace patterns.	6 trace patterns

Done.

Figure 6.2.2: Trace patterns

In the perspective of role SwMI, security properties Ni-Synch and Ni-Agree do not hold, Scyther provides one example attack illustrated in figure 6.2.3. In this attack the intruder intercept the message sent from the MS to the SwMI contenting variables RES1 and RAND2. The intruder selects RES1 and replaces the value of RAND2 with nonce number generated by him. The SwMI received the modified message and believed it was sent from the MS. The intruder then block the message send from the SwMI to the MS with parameters RES2 and R1, and forge a reply R2 to the SwMI. This attack also makes the claim event MS and SwMI agree over the value of data R1 and R2 fail.

In this attack, the intruder makes the SwMI believe the authentication process has successfully completed while the MS still in the half way of authentication process waiting for the reply message from the SwMI. This kind of attack might not be easily identified and could affect the availability of the system.

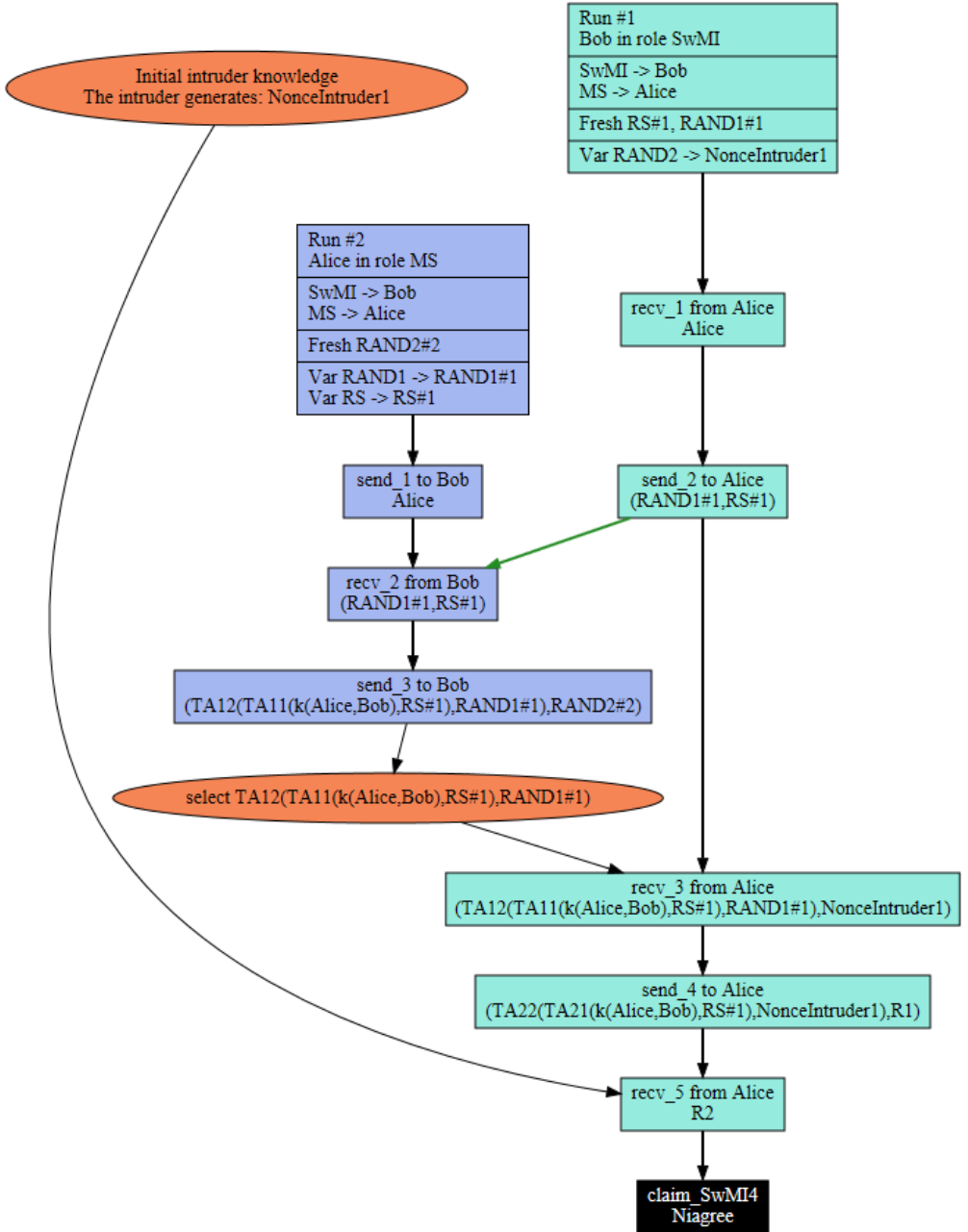


Figure 6.2.3: Attack 1

In the perspective of role MS, the security properties Ni-Synch and the data agreement over R1 and R2 do not hold. The Scyther tool describes one of the attacks illustrated in figure 6.2.4. This is a cutting the final message attack where the last message, the result of authentication in this case, is blocked. If the SwMI under a mass cutting the final message attack, missing notifications from terminals, it shall reserve resources for a large number of terminal users. Such cutting the final message attacks are unpreventable.

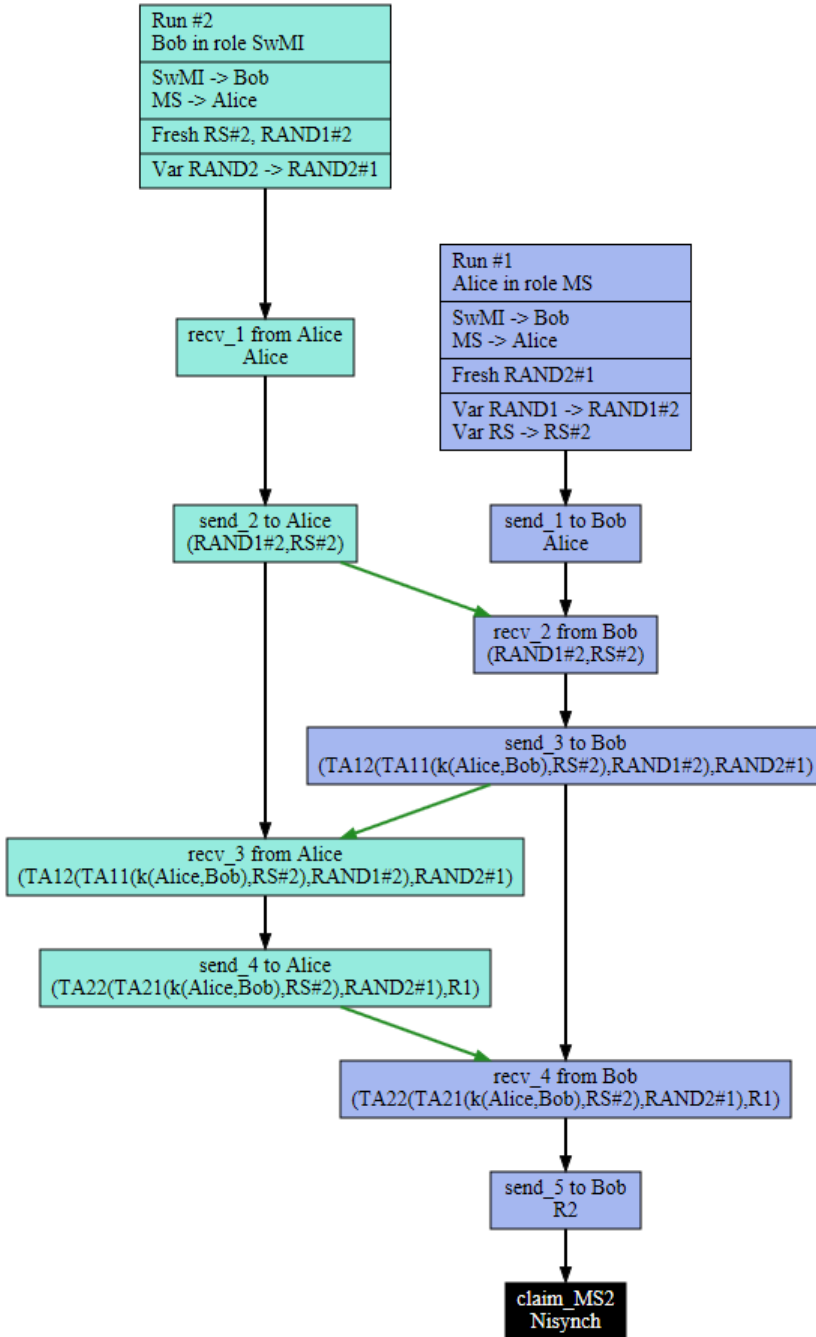


Figure 6.2.4: Attack 2

In attack 3 shown in figure 6.2.5, the Ni-agree claim is violated. The intruder modifies data between different sessions of the same MS user that make the SwMI and the MS believe they have agreed over the same value when they are actually not. The intruder select the correct RES1 from the reply message of MS in the first session, and start the second session with MS with the original RS and an intruder generated random number. Then the intruder shall reply to the SwMI with the correct RES1 from the first session and the RAND2 from the second session. The SwMI compute the RES2 with RS and RAND2, and send it back to the MS. Both the MS and the SwMI believe the authentication has successfully complete but they actually do not have agreement over the value of RAND1 and RES1/XRES1.

Recall the authentication procedure illustrated in figure 4.2.4, the processes of compute RES1 and XRES1 at the MS and SwMI side respectively also generate a value called DCK1. DCK1 shall be used as one of the inputs to the algorithm TB4 to generate the derived cipher key (DCK). As mentioned in the section 4.3.1, the DCK shall be used as air interface encryption key between MS and SwMI after authentication. Since the value of RES1 is not equals to the value of XRES1, the values of DCK1 are also different. Communication between the MS and the SwMI after authentication shall be impossible due to the fact they do not share the same DCK.

The MS and the SwMI might realize they do not share the same DCK, and they may start another authentication procedure.

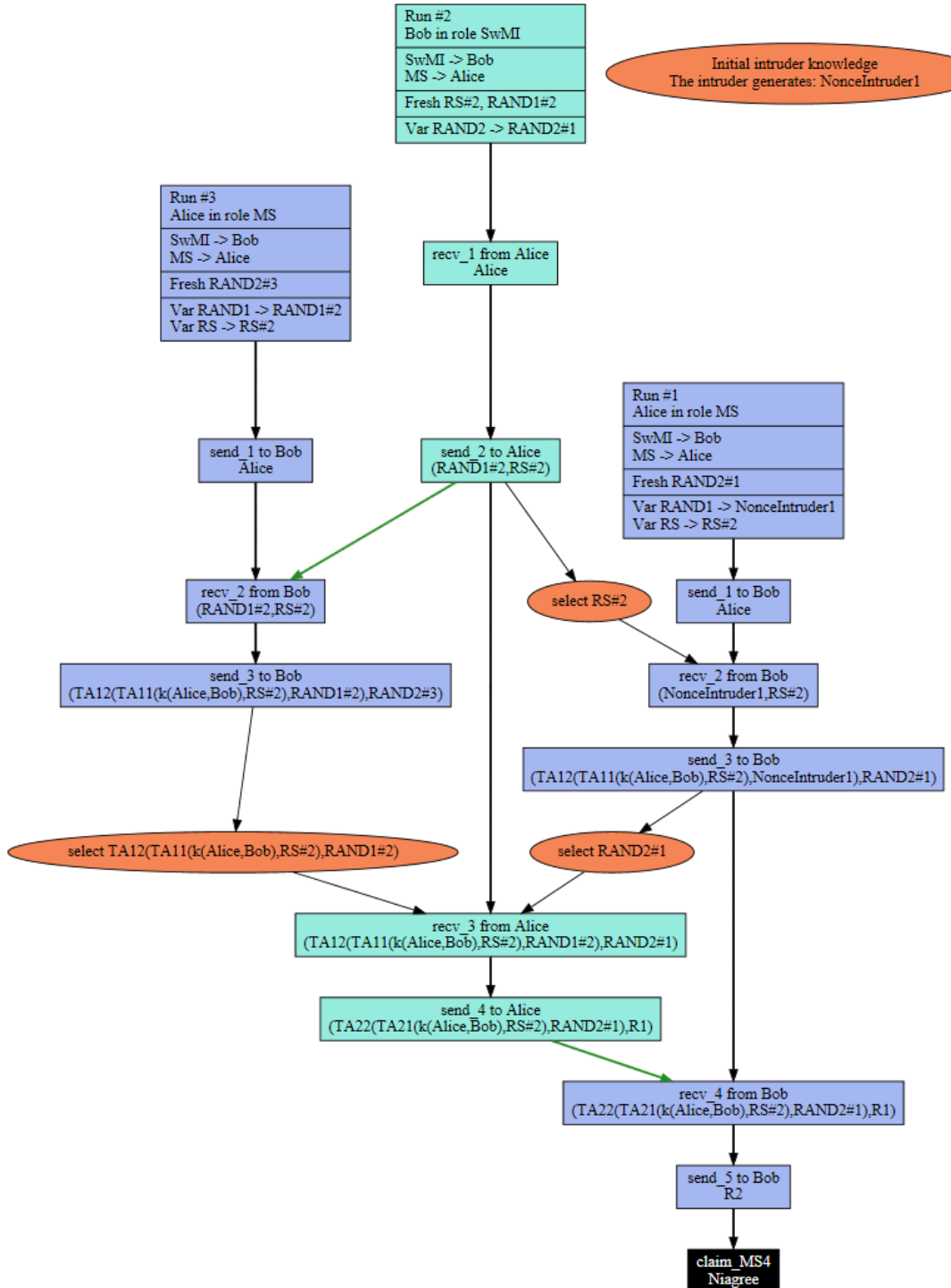


Figure 6.2.5: Attack 3

Chapter 7

Attack Scenarios

Inspired by the result from Scyther, this chapter covers some possible attacks against TETRA networks.

7.1 Denial of Service (DoS) Attacks

In a Denial of Service (DoS) attack, the attacker attempts to make the network resources unavailable to its honest users, and one common method is saturating target with external communication request to exhaust its capability [10].

Jamming

Radio jamming is a way of disrupt communication by transmitting radio signals and decreasing the signal to noise ratio [46]. The most direct way of jamming is generating high power noise over the bandwidth using by TETRA. With some knowledge of the authentication protocol, jamming could be more energy efficient and harder

to be detected.

On the MAC layer, TETRA using a variation of CSMA called Data Sense Multiple Access (DSMA) [42]. In DSMA, the base station transmits a signal periodically on the control channel indicates whether the channel is busy or free. The mobile station transmits a data packet on the reverse channel only when it senses the channel is free [43]. If the attacker jamming the downlink control channel periodically with certain signal indicating "busy", the terminal mobile station shall always find the network busy. Jamming other control signals sent from the base station to terminal users also effectively disrupt network services.

It is possible to make jamming scenarios similar as legitimate scenarios, random jammer is one of the models could be employed. Instead of continuously jamming the communication channel, the jammer switches from jamming and sleeping modes.

Certain amount of security could be provided by spread spectrum using Direct Sequence Spread Spectrum (DSSS) and Frequency Hopping Spread Spectrum (FHSS). The DSSS spreads data on a broad range of frequencies using a key, while using FHSS data packets will be transmitted on different channels in a random pattern [47].

Authentication Request Flooding

In this scenario, the attacker send massive authentication request to the network with intercepted ISSIs. Recall the procedure of TETRA authentication, authentication achieved by the communication parties proving the knowledge of shared secret key to each

other. The infrastructure shall compare the result of RES1 sent from the terminal user with its own calculated XRES1. Massive user registration request generated by an attacker utilized enormous computational capacity of the infrastructure, and affect service availability to legitimate users.

If there is no effective method to protect against flooding attack, partial of the buffer capacity of the infrastructure shall be wasted. The quality of service to legitimate user shall be decreased. It is important for the infrastructure to be able to detect dishonest users and ignore their request. For example, if there are frequent authentication requests from the same user identity, the user might be suspicious.

7.2 Man-in-the-Middle Attacks

Man-in-the-middle attack is the attack where the attacker makes independent connections with the victims and relays messages between them [48].

False Base Station Threat

In the ETR technical requirements specification published in 1994 [44], the risk of false base station threat is mentioned. According to the specification, in 1994 building a false base station is considered to be very expensive. This kind attack is not likely to happen unless some terrorist or criminal organizations are involved. Nowadays the equipment used to build a false base station is cheaper and easier to get, and the attackers are not required to have a broad knowledge of

the network. The false base station attack becomes more practical. One of the possible ways to set up a false base station is using USRP software radio device, antennas, power amplifier and a laptop working GNU Radio and OpenBTS stack. The total cost is around \$2000 [14].

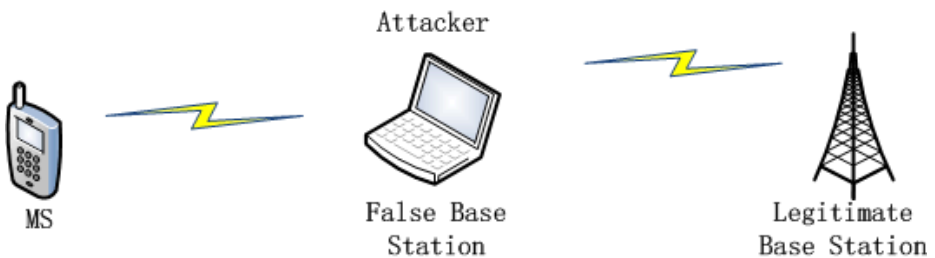


Figure 7.2.1: False base station attack

After the attacker has set up a false base station like shown in figure 7.2.1, the attacker then have the control of air channel between MS and SwMI. The attacker is able to intercept traffic, manipulate or delete user messages, insert messages as the attacker's will, impersonate of a legitimate user or network. The false base station could also be used as a jammer.

Manipulating of the messages exchanged between MS and the SwMI shall easily cause failure of authentication. However, it won't take so long for the communication parties to start another round of authentication. A more intelligent manipulation attack makes both or one of the communication parties unaware of the failure of authentication.

For a particular target, the attack shown in figure 6.2.5 makes

both the MS and the SwMI believe they have successfully completed authentication while they were agreed over different values and would not be able to communicate. The complexity of this attack limits the ability of attacker to intrude massive users.

The attacker can always manipulate parameters RS or random numbers, and MS or the SwMI shall return the incorrect RES1 or RES2. The authentication is then failed but the attacker can forge the reply messages R1/R2 to make one of the communication parties believes the authentication is successful. If the MS believes the authentication has complete when the SwMI actually send $R2 = \text{false}$, the MS starts send out its traffic which will never be forwarded by the SwMI. If the SwMI believes the authentication is successful when the MS send $R1 = \text{false}$, the SwMI shall reserve resources for the MS.

Intercept and delete the last acknowledgment message in the authentication process is cutting the final message attack discussed in section 6.2. In TETRA authentication, the last message is not a simple ACK message but the result of whether authentication is successful or not. This attack leaves one of the communication parties waiting for the last message to complete authentication and start communication. How should the infrastructure deal with the situation of missing the last message is important. The infrastructure could ignore the last message from MS and assume the authentication is successful, but it is possible that the random number RAND2 from the MS has been modified by an attacker or error happened during transmission over the air. The authentication result might be $R2 = \text{false}$. In this case, the last message should not be ignored.

The infrastructure could also request MS to resent the last message. The resend message might be successfully received or might be intercepted and deleted by the attacker again. Another approach is that the infrastructure could initial a new authentication, but it makes the system inefficient. A better way may be adding a final “acknowledgment” from the initiator to the responder as suggested in [33] although additional message increases the complexity of the authentication protocol.

7.3 Summary

TETRA has been widely adopted in police, emergency and rescue service organizations, so guarantee the availability of the system is crucial. Those attempting to compromise TETRA networks might be criminals or terrorists, and they may be interested in launching DoS attacks.

Jamming is the most directly and simple way of denial of service attack, but it is easy to be detected. Within the area covered by a jammer, the terminal user shall notice their equipment cannot receive any signal. The victim shall try to change his position until the equipment can connect to the network, which means he get out of the jamming area. However, it is possible to make jamming scenarios similar as legitimate scenarios and very hard to be detected by employ different jammer models.

The denial of service attack by flooding authentication requests is easier to detect since many requests are from same user identity. Manipulate the challenge parameters during the authentication process

as well as the authentication result R1/R2 is harder to be detected. Although involves intercept, identify and modify messages which is more complicated than simply flooding, the later attack is a better DoS attack for the attacker. Because it more likely to cause serious problem for the system due to the fact that it is harder for the MS or the infrastructure to aware that they are under attack. In jamming attack, the victim realizes there is some problem with the network when his equipment cannot get any signal. In manipulation attack, the victim could not communicate when his equipment seems like working fine. The victim might not realize he is under attack.

Part V

Conclusion

Chapter 8

Conclusion

TETRA mostly used by emergency and rescue service organizations, military, and as a general national safety communication network. The system is then required to provide higher level of security than public mobile radio system. Guarantee confidential of communication, availability and reliability of the system is crucial.

This thesis identifies key security features: encryption, authentication and key management. Analyzing authentication protocol is the focus of this thesis. Theoretical analyzing of the authentication protocol, a good feature is that the authentication key K is protected by never directly using in the communication. Instead, fresh generated session keys K_S and K_S' are used in the authentication. The random numbers involved in the mutual authentication process: R_S , $RAND1$ and $RAND2$ make it difficult to perform a message replay attack. A vulnerability of the authentication protocol is that there is no data integrity protection of the authentication messages. A

false base station could simply intercept and modify authentication messages which will cause the authentication process fail.

In a formal security analysis of the authentication protocol, the secrecy of the long-term shared key, the session keys and the DCK is proved by the Sycther tool. The communication channel between authentication centre and base station is assumed to be secure. The TETRA authentication protocol may suffer from cutting the last message attack. The manipulation attack against different sessions of the same user makes the SwMI and the MS believe they have agreed over the same value when they are actually not. Manipulate challenge parameters and forge the authentication result R1/R2 message makes one of the communication parties believes the authentication is successful. All of these attacks shall seriously affect the availability of the system and waste computational and storage resource of the infrastructure.

In a practical sense, low cost software defined radio (SDR) solutions make it easier nowadays to attack mobile networks [45]. Greg Jones, a director of wireless security specialist said "SDR devices typically use a standard PC to capture and manipulate radio spectrum potentially allowing an attacker to capture and demodulate advanced radio systems which were previously inaccessible to the hacking community". The tools USRP (Universal Software Radio Peripheral) and open source software like GNU were also mentioned by Jones. TETRA is on the list of the "advanced radio system" that could be a target.

Attacks found by the Scyther tool mostly aimed at compromising the availability of the system. There exist more simple and direct at-

tacks like jamming and will attackers even consider complex attacks like those found by Syther? Attacks like attack 3 in figure 6.2.5 is too complex. It seems that for a DoS attack it is not worth the work. Manipulation challenge parameters and forge the authentication result R1/R2 message attack, on the other hand, is implementable since setting up false base station is easier today. There are many possible ways of attack, different consequences depend on attacker's different intentions and how is he using his knowledge of the system. Attacks found by Scyther might not necessarily be the most efficient ones.

Part VI

Appendix

Bibliography

- [1] Reeves, C. M. (1980, September). An overview of trunking techniques in mobile radio systems. In *Vehicular Technology Conference, 1980. 30th IEEE* (Vol. 30, pp. 537-541). IEEE.
- [2] Dunlop, J., Girma, D., & Irvine, J. (1999). *Digital mobile communications and the TETRA system* (Vol. 1). Wiley.
- [3] *TETRA - Terrestrial Trunked Radio* .(n.d.). Retrieved from <http://www.willtek.com/english/technologies/tetra>
- [4] Walke, B. H. (2002). *Mobile Radio Networks: Networking, Protocols and Traffic Performance*. Wiley.
- [5] Edney, J., & Arbaugh, W. A. (2004). *Real 802.11 security: Wi-Fi protected access and 802.11 i*. Addison-Wesley Professional.
- [6] Webb, W. (1999). *The complete wireless communications professional: A guide for engineers and managers*. Artech House, Inc..
- [7] William, S. (2007). *Cryptography and Network Security Principles and Practices*, (pp. 267-269). Prentice Hall.
- [8] ETSI Technical Standard ETSI EN 300 392-7 V2.1.1 (2001-02): *Terrestrial Trunked Radio (TETRA); Voice plus Data; Part 7: Security*.

- [9] ETSI Technical Standard ETSI EN 302 109 V1.1.1 (2003): *Terrestrial Trunked Radio (TETRA); Security; Synchronization mechanism for end-to-end encryption.*
- [10] Denial-of-service attack. (n.d.). In Wikipedia. Retrieved June 25, 2013, from http://en.Wikipedia.org/wiki/Denial-of-service_attack.
- [11] Block Cipher. (n.d.). In Wikipedia. Retrieved March 10, 2013, from http://en.Wikipedia.org/wiki/Block_cipher
- [12] Anne Canteaut. *Stream cipher*. Retrieved March 10, 2013 from <https://www.rocq.inria.fr/secret/Anne.Canteaut/encyclopedia.pdf>
- [13] TETRA MoU Association. (2006): TETRA Security.
- [14] Is it possible to use a fake BTS acting as a transparent proxy for a man in the middle attack against GSM? Retrieved June 21, 2013 from <http://www.quora.com/Is-it-possible-to-use-a-fake-BTS-acting-as-a-transparent-proxy-for-a-man-in-the-middle-attack-against-GSM>
- [15] Replay Attack. (n.d.). In Wikipedia. Retrieved March 11, 2013 from http://en.Wikipedia.org/wiki/Replay_attack
- [16] TETRA Association SFPG Information document (2008). *Overview of Standard TETRA Cryptographic Algorithms and their rules for management and distribution.*
- [17] ETSI Technical Report TR 101 052 V1.1.1 (1997): *SAGE Rules for the management of the TETRA standard authentication and key management algorithm set TAA1.*
- [18] ETSI Technical Report TR 101 053-1 V1.1.2 (2006): *SAGE Rules for the management of the TETRA standard encryption algorithms Part1 TEA1.*

- [19] ETSI Technical Report TR 101 053-2 V2.2.4 (2012): *SAGE Rules for the management of the TETRA standard encryption algorithms Part2 TEA2*.
- [20] ETSI Technical Report TR 101 053-3 V1.1.3 (2007): *SAGE Rules for the management of the TETRA standard encryption algorithms Part3 TEA3*.
- [21] ETSI Technical Report TR 101 053-4 V1.1.2 (2006): *SAGE Rules for the management of the TETRA standard encryption algorithms Part4 TEA4*.
- [22] ETSI Technical Standard ETSI EN 300 392-1 V1.4.1 (2009): *Terrestrial Trunked Radio (TETRA); Voice plus Data; Part 1: General network design*.
- [23] Burrows, M., Abadi, M., & Needham, R. M. (1989). *A logic of authentication*. Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences, 426(1871), 233-271.
- [24] C.J.F. Cremers, The Scyther tool: automatic verification of security protocols. Retrieved May 10, 2013 from <http://people.inf.ethz.ch/cremersc/scyther/>
- [25] Graphviz - Graph Visualization Software. Retrieved May 10, 2013 from <http://www.graphviz.org/Download..php>
- [26] Python Download. Retrieved May 10, 2013 from <http://www.python.org/download/>
- [27] wxPython Download & Stuff. Retrieved May 10, 2013 from <http://www.wxpython.org/download.php>
- [28] C.J.F. Cremers, Generic installation instructions. Retrieved May 10, 2013 from <http://people.inf.ethz.ch/cremersc/scyther/install-generic.html>

- [29] Cremers, C., & Mauw, S. (2011). *Operational semantics and verification of security protocols*. Springer.
- [30] Cremers, C. J. F. (2006). Scyther: Semantics and verification of security protocols. *Dissertation Abstracts International*, 68(02).
- [31] C.J.F. Cremers, S. (2003). Mauw, E.P. de Vink, Defining authentication in a trace model, in *1st International Workshop on Formal Aspects in Security and Trust (FAST'03)*, ed. by T. Dimitrakos, F. Martinelli, Pisa, Italy pp. 131–145. IITT-CNR technical report
- [32] Abadi, M., & Fournet, C. (2001, January). Mobile values, new names, and secure communication. In *ACM SIGPLAN Notices* (Vol. 36, No. 3, pp. 104-115). ACM
- [33] Canetti, R., & Krawczyk, H. (2002). Security analysis of IKE's signature-based key-exchange protocol. In *Advances in Cryptology—CRYPTO 2002* (pp. 143-161). Springer Berlin Heidelberg.
- [34] Ryan, P. Y., Ryan, P., & Schneider, S. A. (2001). *The modelling and analysis of security protocols: the csp approach*. Addison-Wesley Professional.
- [35] Cervesato, I., Durgin, N. A., Lincoln, P. D., Mitchell, J. C., & Scedrov, A. (1999). A meta-notation for protocol analysis. In *Computer Security Foundations Workshop, 1999. Proceedings of the 12th IEEE* (pp. 55-69). IEEE.
- [36] Blanchet, B. (2001, June). An efficient cryptographic protocol verifier based on Prolog rules. In *Proceedings of the 14th IEEE workshop on Computer Security Foundations* (p. 82).
- [37] Armando, A., Basin, D., Boichut, Y., Chevalier, Y., Compagna, L., Cuéllar, J., ... & Vigneron, L. (2005, January). The AVISPA tool for the automated validation of internet security protocols

- and applications. In *Computer Aided Verification* (pp. 281-285). Springer Berlin Heidelberg.
- [38] Ramsdell, J. D., & Guttman, J. D. (2009). CPSA: A cryptographic protocol shapes analyzer. *Hackage. The MITRE Corporation*, 2(009).
- [39] Meadows, C. (1996). The NRL protocol analyzer: An overview. *The Journal of Logic Programming*, 26(2), 113-131.
- [40] Dolev, D., & Yao, A. (1983). On the security of public key protocols. *Information Theory, IEEE Transactions on*, 29(2), 198-208.
- [41] Chosen-plaintext attack. (n.d.). In Wikipedia. Retrieved June 20, 2013 from http://en.Wikipedia.org/wiki/Chosen-plaintext_attack
- [42] Pahlavan, K., & Levesque, A. H. (2005). *Wireless information networks* (Vol. 93, p.547). Wiley-Interscience.
- [43] Molisch, A. F. (2010). *Wireless communications* (Vol. 15, p.376). Wiley
- [44] ETSI Technical Report ETR 086-3 (1994): *Trans European Trunked Radio (TETRA) system; Technical requirements specification Part 3: Security aspects*.
- [45] Tony Dennis. (2012). *Warning of increased GSM + TETRA attacks* Retrieved June 25, 2013 from <http://www.gomonews.com/warning-of-increased-gsm-tetra-attacks/>
- [46] Radio jamming. (n.d.). In Wikipedia. Retrieved June 25, 2013, from http://en.Wikipedia.org/wiki/Radio_jamming
- [47] (2007). *Frequency Hop Spread Spectrum vs. Direct Sequence Spread Spectrum*. Theory and Terminology Note Banner Engineering Corp.

- [48] Man-in-the-middle attack. (n.d.). In Wikipedia. Retrieved June 25, 2013, from http://en.Wikipedia.org/wiki/Man-in-the-middle_attack.