Západočeská univerzita v Plzni

Fakulta aplikovaných věd

Více faktorová autentizace v mobilních sítích

RNDr. Libor Dostálek

disertační práce
k získání akademického titulu doktor
v oboru Informatika a výpočetní technika

Školitel: Prof. Ing. Jiří Šafařík, CSc.

Konzultant specialista: Ing. Jiří Ledvina, CSc.

Katedra: Informatiky a výpočetní techniky

Plzeň 2021

Prohlašuji, že jsem tuto disertační práci vypracoval samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Západočeská Univerzita v Plzni má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V Plzni dne . . . .

Podpis autora

University of West Bohemia

Faculty of Applied Sciences

Multifactor Authentication in Mobile Networks

RNDr. Libor Dostálek

Doctoral thesis
submitted in partial fulfillment of the requirements
for a degree of Doctor of Philosophy
in Computer Science and Engineering

Supervisor: Prof. Ing. Jiří Šafařík, CSc.

Consulting specialist: Ing. Jiří Ledvina, CSc.

Department of Computer Science and Engineering

Plzeň 2021

I hereby declare that this thesis has been written only by the undersigned and without any assistance from third parties.

Furthermore, I confirm that no sources have been used in the preparation of this thesis other than those indicated in the thesis itself.

In Plzeň on …..

Author's signature

Content

Picture list

List of Abbreviations

5GC	5G Core Network+
AF	Application Functions
AF	Application Features
AKA	Authentication and Key Agreement
APN	Access Point Name
A-SBC	Access SBC
AuC	Authentication Centre
AV	Authentication Vector
B2BUA	SIP Back-to-Back User Agent
BGCF	Breakout Gateway
CSCF	Call Session Control Function
DNS	Domain Name System
EAP	Extensible Authentication Protocol
EAP-AKA	Extensible Authentication Protocol Method for the 3rd Generation Authentication and Key Agreement
EATF	Emergency Access Transfer Function
E-CSCF	Emergency CSCF
EIR	Equipment Identity Register
eNB	Evolved Node B
ENUM	E164 Number Mapping
EPC	Evolved Packet Core
EPS	Evolved Packet System
FDS	Fraud Detection System
FPS	Fraud Prevention System
HSM	Host (sometimes Hardware) Security Module
HSS	Home Subscriber Server
HTTP	Hypertext Transfer Protocol
IBCF	Interconnection Border Control Function
I-CSCF	Interconnect CSCF
IMS	Internet Multimedia Subsystem
IMS-AGW	IMS Access Gateway
IMS-ALG	IMS Application Level Gateway
IMSI	International Mobile Subscriber Identity
IPX	IP Exchange Network
I-SBC	Interconnect SBC
I-SBC	Interconnect SBC
ISIM	IP Multimedia Services Identity Module
LTE	Long Term Evolution
MAC	Message Authentication Code
MF	Media Gateway
MG	Media Gateway
MGC	Media Gateway Controller
MGCF	Media Gateway Control Function
MGCF	Media Gateway Controller
MME	Mobility Management Entity
NAS	Non-Access Stratum
NAT	Network Address Translation
NE	Nash Equilibrium
PCRF	Policy and Charging Rules Function
P-CSCF	Proxy CSCF
PDN PGW or PGW	Public Data Network Gateway
PIN	Personal Identification Number
PSD2	Directive (EU) 2015/2366
RRC	Radio Resource Control
RTP	Real Time Protocol
SBC	Session Border Controller

S-CSCF	Severing CSCF
SCTP	Stream Control Transmission Protocol
SD	Secure Digital (card)
SDP	Session Description Protocol
SEG	Security Gateway (or IPsec GW) - Ending IPsec Tunnels,
SGW	Serving Gateway
SIP	Session Initiation Protocol
SLF	Subscriber Location Function
SMTP	Simple Mail Transfer Protocol
SRB	Signaling Radio Bearer
SRTP	Secure Real-time Transport Protocol
SS7	Signaling System No. 7
TrGW	Transition Gateway
TTE	Trusted Execution Environment
UICC	Universal Integrated Circuit Card
USIM	Universal Subscriber Identity Module
VoLTE	Voice over LTE
WiFi	Pseudonym of IEEE 802.12

1 Introduction

At present, many Internet applications are accessed from smartphones. If internet applications require strong authentication, there are a number of authentication methods available, e.g.:

1. Native authentication methods in 3G/4G networks based on an AKA mechanism [1]. This authentication is undoubtedly a cryptographically strong authentication. Its disadvantage is that it is used for authenticating a mobile device to the network (often called Equipment Authentication) but it is not used for the user authentication into the application.

2. Password authentication is a typical method used for user authentication. Unfortunately, this authentication method is generally considered to be weak, and thus applications such as eBanking or eGov require other mechanisms.

3. Strong password authentication is a more sophisticated method resistant to known attacks (sniffing or elicitation of password, password-file compromise attacks, guessing attacks, forgery attacks, impersonation attacks, stolen-verifier attacks, replay attacks etc.).
4. Authentication based on public key certificates. The problem with such authentication is the location where the application is meant to securely generate and store the private keys.

5. External devices such as authentication tokens (calculators) generating one-time passwords. The main disadvantage of this solution is that the user has to be concerned with an additional device, which he/she may find disagreeable.

Using multiple authentication methods independently does not increase security. The aim of this thesis is to discuss the combination (“breeding”) of vaious authentication methods in a common multi-factor authentication. For example, the first factor may be an equipment authentication based on AKA mechanisms and the second factor a strong password authentication. (It should be noted that the AKA method itself is a two-factor authentication but under the sole control of Telco operators.)

At present, Telco providers provide the Internet Multimedia Subsystem (IMS). IMS enables them to provide multimedia services (VoLTE, videoconferencing, etc.). Moreover, it also allows the provision of application services based on HTTP. The well-established service is user access to the portal of Telco operators. The convenience of this solution is that users can use it to authenticate via USIM / ISIM smartcards. It is a strong authentication mechanism using an AKA protocol [1].

In the future, we can expect that more applications will be placed into the IMS environment. These applications will be based either on SIP (video on demand etc.) or on HTTP (e.g. as banking applications or internet access to government applications etc.). Applications will be provided not only by Telco operators but also by independent third parties - the application (content) providers in IMS (hereinafter referred to as application providers). These application providers will use the secure environment of IMS. They will need strong authentication to their applications.

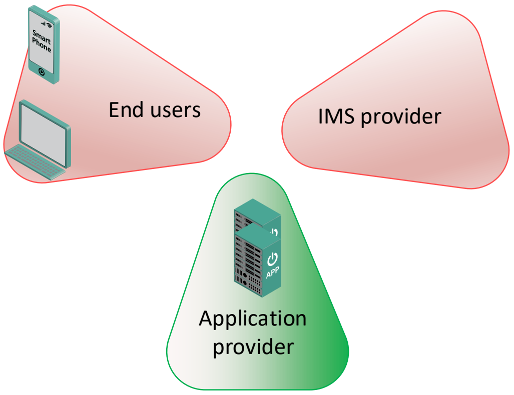

Figure 1.1 Strong USIM/ISIM authentication while each application provider keeps control of authentication to its applications

A good idea is to use the AKA authentication mechanism. The disadvantages of this authentication are not the strength of its cryptographic algorithm, but its organizational disadvantages:

- User management is under the full control of Telco operators.
- After opening the security context (PIN entry), the mobile device repeats authentication silently in the background without user intervention, e.g. when entering the visited network.

The impact of the above-mentioned disadvantages is the following: the authentication method is controlled by Telco operators, i.e. authentication is mutual to all applications provided in the whole environment of one Telco operator, while applications themselves can be provided by a third party (e.g. application providers). Then, a third party can:

- Either use their own independent authentication, which does not include strong authentication using USIM / ISIM, e.g. password authentication. It is, however, too weak e.g. for banking applications or internet access to government applications etc.

- Or to be fully subordinate to an IMS provider and use its user database.

There is the need to find a solution that allows the use of strong authentication while each application provider keeps control over the authentication process inside its own applications. The solution should provide authentication not only for SIP-based applications (Multimedia applications) but for HTTP-based applications (web application) as well. Logically, the status of application providers will become independent of the connection (Figure 1.1).

The question is how to compare the strength of individual authentication methods. A model of quantification of the strength of authentication methods is needed. Based on this model, a comparison of the proposed authentication method with other authentication methods is possible.

2 Authentication

Authentication is the entity's (subject's) identity verification process. This process is carried out by a verifier who guarantees that the entity has a declared identity (Figure 2.1). The quality of this warranty depends on the particular authentication process.

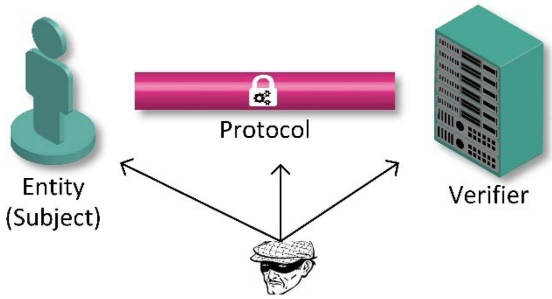

Figure 2.1 Fundamental roles in authentication process

We distinguish between entity authentication and message authentication. The difference is perceived in the time perspective. Authentication of the message (e.g. by an electronic signature) gives no guarantee of when the message was created. Instead, the entity authentication includes a proof of identity of the applicant as a rule through the current communication with the verifier.

An example of an authentication process is a process by which a user logs on to the application using a username and a password.

The secondary effect of the authentication process may be the fact that during the authentication of the entity, cryptographic material for subsequent communication is also generated.

2.1 Authentication methods

The ways in which someone might be authenticated can be divided into three categories, based on what is known as the factors of authentication:

1. The entity (subject) knows something - knowledge factors – e.g. password, private or secret key, shared secret, etc.
2. The subject owns something - ownership factors – e.g. smartcard, one-time passwords token, etc.
3. The subject is or does something - inherence factors – e.g. fingerprint, dynamic biometric signature, digital footprint, etc., or behavioral biometrics such as keystroke dynamics may also be used.

With the development of mobile networks that allow effortless subscriber localization, there is sometimes talk of a fourth authentication factor – localization [2]. We will consider localization as an inherence factor.

2.2 Knowledge factors

The well-known knowledge factors are:

- Password authentication which includes different variations: passphrase, personal identification number (PIN) etc.
- Dialog authentication which includes different variations like secret questions.
- Zero-knowledge authentication.

2.2.1 Password authentication

The password is a memorable character string for the user that is valid for a certain amount of time. In general, password authentication is considered weak. There are also weaker authentication methods such as IP-based authentication (see also subsection 2.4.1).

In addition to passwords, we have one-time passwords, i.e. passwords that can only be used once. There are a number of algorithms for creating one-time passwords. From a simple list of one-time passwords, through algorithms based on a shared secret between the subject and the verifier, e.g. according to the so-called Lamport scheme [3]. However, schemas for generating one-time passwords are typically included in dialogue authentication.

2.2.2 Dialogue authentication

Before dialogue authentication takes place, secret information must have been exchanged between the subject and the verifier in advance. Secret information is e.g. cryptographic keys or shared secrets. This secret information is subsequently used e.g. as encryption keys to encrypt a challenge. In the case of a one-way function, the secret information is concatenated with a challenge before the one-way function is applied.

The dialogue consists of at least two steps: a challenge and a response. The challenge contains a string including e.g. a random number, sequence number of authentication, time, etc. (maximum entropy). The response is a string from the challenge to which a symmetric cipher, an asymmetric cipher, or a one-way function is applied .

Dialogue authentication is sometimes referred to as "strong" - contrary to password authentication. However, the term "strong authentication" is used in various ways, as we will see.

The authentication scheme may be divided into two groups depending on whether they do or do not use the data bearer to store cryptographic material:

- Authentication without a data bearer ("without a smartcard"). This is the Lamport diagram already mentioned [3], but other schemes have also been published, e.g. [4]. Schemes from the "without a smartcard" group are generally considered weak today, and are no longer part of discussion.
- Password Authentication with a data bearer ("with a smartcard"). Instead of the concept of "smartcard", we should rather use the term "data bearer" to avoid confusion with USIM / ISIM smartcards used by mobile devices e.g. [5] [6].

2.2.3 Zero-knowledge authentication

Password or dialogue authentication is based on knowledge of secret information (a password, a shared secret, etc.). Because the secret information is known only by the subject and the verifier, it is assumed that this is a sufficient proof of the client's authenticity. The weakness of these methods is the fact that secret information is revealed in some way during authentication, which may be an opportunity for an adversary.

Zero Knowledge schemes are based on the assumption that the subject has knowledge of a complex problem (his/her own secret). Authentication then proceeds by demonstrating knowledge in solving this complex problem (e.g. NP problem [7]). As a result of authentication, only one-bit information is authenticated/unauthenticated. This is very interesting from a security point of view, because it does not generate cryptographic material to ensure subsequent communication, so we will not elaborate further upon them here.

2.3 Ownership factors

Ownership factors may have a variety of real-world features - such as a plastic entrance card. In mobile networks it may be:

- A smartcard or its variation - i.e. a single-chip computer with stored cryptographic material for the authentication of persons (i.e. a device for storing personal authentication assets). This device communicates electronically with the verifier’s equipment during authentication. Access to personal assets (cryptographic material) is protected by the following:

    - One or more PINs in the case of access by a person.
    - A secure messaging mechanism when accessing applications without a user intervention.
- An authentication token (a calculator) - i.e. a single-chip computer with stored cryptographic material used for person authentication (i.e., a device for storing personal authentication assets), but usually does not communicate electronically with the verifier, the information is presented on the display. The holder then writes the information and through an application process they are passed to the verifier.
- A HSM (Host Security Module, sometimes also Hardware Security Module) is a powerful computer used to store system (server) assets. A HSM directly communicates with the system (subject).
- A smartphone.

This classification is now considered historical. GlobalPlatform has abstracted from a specific physical implementation and has defined a so-called Secure Element to store personal cryptographic assets [8]. Practically speaking, a dedicated one-chip microprocessor is designed to safely store cryptographic data and safely perform operations with them. These operations are performed in the so-called Trusted Execution Environment (TEE) [9].

The secure element can be implemented as part of a smartcard (USIM, ISIM, SD, etc.) or, e.g. as a chip integrated on the motherboard of a mobile device. The security element is fundamentally viewed from the point of view of security as it is on a HSM.

In conclusion, it can be said that personal authentication assets can be stored:

- On the data bearer without protection (or with poor protection).
- In the secure element.
- In a HSM.

When comparing individual methods, we take the following security features into account (it also depends on the specific implementation):

- A device physically stores cryptographic material (and uses it).
- Access to cryptographic material uses a password or PIN.
- Cryptographic material does not leave the device (it is non-exportable).
- A device is physically protected against an unauthorized access.

2.4 Inherence factors

These authentication factors are usually considered as authentication factors based on the biometric properties of a subject, i.e. verification of the person's identity based on measurable physiological or behavioral characteristics, unique and relatively unchanging for the subject.

Authentication is based on the match of the input pattern with the pre-stored pattern in the database. Authentication is confirmed if the matching exceeds a predetermined threshold.

A basic disadvantage of biometric personal characteristics is that they cannot be revoked and subsequently altered in the case of abuse. For example if an attacker obtains a dynamic biometric signature from a subject, then the subject can never use a dynamic biometric signature without the possibility of its misuse.[1]

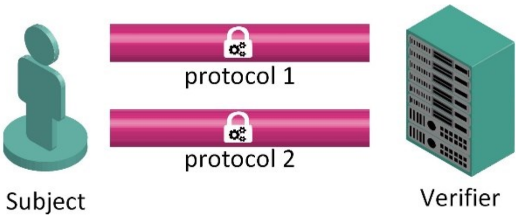

Figure 2.2 Two factor authentication with two protocols

2.4.1 Digital Footprint

Digital Footprint has similar features to biometric characteristics. This is in particular the tracking of the metadata that we use in communication or which we have left behind (more in sections 2.14 and 7.2).

A special case of Digital Footprint is Device fingerprinting, which is used for mobile phones and similar devices (more in section 7.3).

2.5 Multi factor authentication

Multi factor authentication grants access only after successfully presenting two or more factors (Figure 2.2).

[1] In the case of the handwriting signature, it is possible to attach a digit, picture, etc. to the signature. This de facto revokes the previous signature without a digit or image.

It is important to use different authentication factors, e.g. using two passwords does not improve the authentication quality. Authentication factors may be varied in:

- Different cryptographic material.
- A different authentication scheme.
- A different communication protocol.
- A different communication channel.
- Different verifiers.

It is also important that the authentication factors are intertwined. If they are not, then the attacker can firstly deal with breaking one authentication factor and then another. However, this cannot always be achieved. For example if an entity is already authenticated (e.g. it has brought authentication from Facebook) and it turns out that stronger authentication (e.g. smartcard) is required for the operation, then it is usually re-authenticated only with a stronger chip scheme independent of the original authentication.

2.6 Authentication in EU law

The quality of authentication is also defined in EU law. Most important is the eIDAS regulation [10] and directive PSD2 [11].

2.6.1 eIDAS

The eIDAS regulation [10] solves authentication in a more complex way. It introduces so called "electronic identification scheme" which includes the following elements: registration, identity proving, electronic identification means management, authentication mechanism, management and organization, compliance and audit.

The eIDAS regulation [10] introduces assurance levels for electronic identification schemes. For each element of the electronic identification scheme, it defines assurance levels Low, Substantial and High.

The following definitions are introduced in [12]:

1. An "authoritative source", meaning any source irrespective of its form that can be relied upon to provide accurate data, information and/or evidence that can be used to prove identity.
2. An "authentication factor", meaning a factor confirmed as being bound to a person, which falls into any of the following categories:

(1) a “possession-based authentication factor”, meaning an authentication factor where the subject is required to demonstrate possession of it;

(2) a “knowledge-based authentication factor”, meaning an authentication factor where the subject is required to demonstrate knowledge of it;

(3) an “inherent authentication factor”, meaning an authentication factor that is based on a physical attribute of a natural person, and of which the subject is required to demonstrate that they have that physical attribute.

3. A “dynamic authentication”, meaning an electronic process using cryptography or other techniques to provide a means of creating on demand an electronic proof that the subject is in control or in possession of the identification data and which changes with each authentication between the subject and the system verifying the subject's identity;

We will now focus on the assurance level of *High*. This assurance level according to [12] requires:

- The release of person identification data is preceded by the reliable verification of electronic identification means and its validity.
- Where person identification data are stored as part of the authentication mechanism, this information is secured in order to protect them against loss and against being compromised, including offline analysis.
- The authentication mechanism implements security controls for the verification of electronic identification means, so that it is highly unlikely that activities such as guessing, eavesdropping, replay or the manipulation of communication by an attacker with enhanced-basic attack potential can subvert the authentication mechanism.
- The release of person identification data is preceded by reliable verification of electronic identification means and its validity through a dynamic authentication.
- The authentication mechanism implements security controls for the verification of electronic identification means, so that it is highly unlikely that activities such as guessing, eavesdropping, replay, or the manipulation of communication by an attacker with high attack potential can subvert the authentication mechanisms.

Electronic identification means with the assurance level of *High* need [12]:

- The electronic identification means utilize at least two authentication factors from different categories.
- The electronic identification means are designed with the assumption of being used only under the control of the person they belong to.

- The electronic identification means are proof against duplication and tampering as well as against attackers with high attack potential.
- The electronic identification means are designed so that they can be reliably protected against others' usage by the person they belong to.

The entity (subject) may communicate with the government authority if it uses an identification scheme with the assurance level of High.

2.6.2 PSD2

The EU directive PSD2 [11] on payment services in the internal market introduces strong customer authentication. It means an authentication based on the use of two or more elements categorized as knowledge (something only the user knows), a possession (something only the user possesses) and an inherence (something the user is) that are independent, thus the breaking of one does not compromise the reliability of the others, and is designed to protect the confidentiality of the authentication data.

This directive enforces "strong customer authentication": Where the payer's payment service provider does not require strong customer authentication, the payer shall not bear any financial losses unless the payer has acted fraudulently. Where the payee or the payment service provider of the payee fails to accept strong customer authentication, it shall refund the financial damage caused to the payer's payment service provider.

Strong authentication is specified in more detail [13] in Article 4 of the Authentication Code:

1. Where payment service providers apply strong customer authentication in accordance with Article 97(1) of Directive (EU) 2015/2366 [11], the authentication based on two or more elements categorized as knowledge, possession and inherence shall result in the generation of an authentication code.

   The authentication code shall be accepted only once by the payment service provider when the payer uses the authentication code to access its payment account online, to initiate an electronic payment transaction or to carry out any action through a remote channel which may imply a risk of payment fraud or other abuses.

2. For the purpose of paragraph 1 payment, service providers shall adopt security measures ensuring that each of the following requirements is met:

a) no information on any of the elements of the strong customer authentication categorized as knowledge, possession and inherence can be derived from the disclosure of the authentication code;
b) it is not possible to generate a new authentication code based on the knowledge of any other authentication code previously generated;
c) the authentication code cannot be forged.

2.7 Session Initiation Protocol

A Session Initiation Protocol (SIP) [14] is an application protocol that can be used to establish, modify, and terminate multimedia sessions (multimedia calls, teleconferences, etc.). SIP can be used to invite other participants to already-running sessions (such as multimedia conferences).

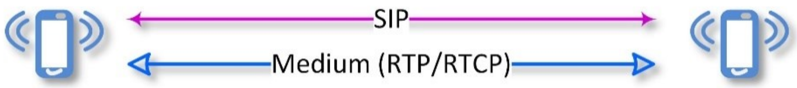

Figure 2.3 SIP is generally used in conjunction with other protocols

SIP is used along with other network protocols to create a communication architecture. Typically (Figure 2.3), this architecture also includes protocols such as the Real-time Transport Protocol (RTP) [15] which provides transfer services for applications that require real-time data transfer, e.g. audio, video, simulation, etc.

An SIP is a client/server protocol like SMTP [16], however, we use the term SIP agent because most SIP entities act once as a client (caller) and at the same time as a server (callee). The end user often uses an SIP agent in a SIP phone that provides traditional phone services such as dialing, rejecting, answering, hanging up, and forwarding, etc. Furthermore, an SIP allows additional services such as Instant Messaging.

For entity identification, an SIP uses the Uniform Resource Identifier (URI) [17] referred to as SIP URI or SIPS URI [14].

Table 2.1 contains an overview of individual SIP protocol entities.

Table 2.1 Entities of SIP

Entity	***Meaning***
User Agent (UA)	A logical entity that can act as both a user agent client and a user agent server. A typical example is an SIP phone.
User Agent Client (UAC)	A user agent client is a logical entity that creates a new request, and then uses the client transaction state machinery to send it. The role of a UAC lasts only for the duration of a transaction. In other words, if a piece of software initiates a request, it acts as a UAC for the duration of that transaction. If it receives a request later, it assumes the role of a user agent server for the processing of that transaction
User Agent Server (UAS)	A user agent server is a logical entity that generates a response to an SIP request. The response accepts, rejects, or redirects the request. This role lasts only for the duration of a transaction. In other words, if a piece of software responds to a request, it acts as a UAS for the duration of the transaction. If it generates a request later, it assumes the role of a user agent client for the processing of the transaction.
Server	A server is a network element that receives requests in order to service them and sends back responses to the requests. Examples of servers are proxies, user agent servers, redirect servers, and registrars.
Proxy, Proxy Server	An intermediary entity that acts as both a server as well as a client for the purpose of making requests on behalf of other clients. A proxy server primarily plays the role of routing, which means its job is to ensure that a request is sent to another entity "closer" to the targeted user. Proxies are also useful for enforcing policy (e.g. making sure a user is allowed to make a call). A proxy interprets, and, if necessary, rewrites specific parts of a request message before forwarding it.
Outbound Proxy	A proxy that receives requests from a client, even though it may not be the server resolved by the Request-URI. Typically, a UA is manually configured with an outbound proxy, or can learn about one through auto-configuration protocols.
Back-to-Back User Agent (B2BUA)	The Back-to-Back user agent (B2BUA) is a logical entity that receives requests and processes them as a user agent server (UAS). In order to determine how the requests should be answered, it acts as a user agent client (UAC) and generates requests. Unlike a proxy server, it maintains dialogue state and must participate in all requests sent on the dialogues it has established. Since it is a concatenation of a UAC and UAS, no explicit definitions are needed for its behaviour.
Registrar	A registrar is a server that accepts REGISTER requests and places the information it receives in those requests into the location service for the domain it handles.
Redirect Server	A redirect server is a user agent server that generates 3xx responses to requests it receives directing the client to contact an alternate set of URIs.

2.7.1 Authentication in SIP

SIP headers used for authentication are as following:

- *WWW-Authenticate* – authentication challenge. The SIP message with this header is mostly associated with the result code *401 Unauthorized*.
- *Authorization*, which carries a subsequent authorization response.

A SIP admits several authentication methods, which are mainly taken from the HTTP protocol:

- Basic authentication, e.g. using the subscriber's name and the password as part of the SIP URI.
- Authentication with the Kerberos Protocol [18], which is especially useful for intranets where Windows domains are used. The Kerberos ticket is packaged in the SPNEGO envelope [19] and then into the Authorization header of the Negotiate authentication method.
- Digest Authentication, which also uses password authentication. However, the password in plaintext is not transmitted via the network (it is hidden in the hash). This is a challenge-response authentication. The query is located in the *WWW-Authenticate* header and the answer in the *Authorization* header.

  The 3GPP Digest method uses the *WWW-Authenticate* and *Authorization* header for Digest. However, the similarity to digest authentication ends there. In the above mentioned SIP headers there contains AKA authentication.

Notice: The 3GPP Digest method was originally designed for SIP, but it may be applicable to HTTP. For example for web communication (HTTP) from the mobile device to the operator portal, a reference point Ut with 3GPP Digest Authentication was introduced (see hereafter).

2.7.2 Session Border Controller

A Session Border Controller (SBC) is a B2BUA entity of SIP frequently combined with security and other functions (Figure 2.4). SBC functionality can be compared to a firewall which separates a network of different security zones.

Basic features of a SBC:

- A SBC is a B2U entity of the SIP protocol, which accepts not only the SIP protocol but also the transfer medium. The accepted requests are thereafter passed to the destination server on behalf of the original client.

- Security features ("firewalling"), which are typically different for input and output.
- Other features (not shown in Figure 2.4), such as the authentication of subscribers to external databases, billing, etc.

A SBC consists of the following parts:

- The Media Gateway Controller (MGC), which accepts SIP, requests and passes them to the destination SIP server.
- The Media Gateway (MG), which accepts and transmits medium (RTP).

What is important is that the MGC and MG must mutually agree on what to accept and to pass on. This is usually ensured by the protocol H.248 (H.248 is a historical name only, current standard is H.248.1 [20]).

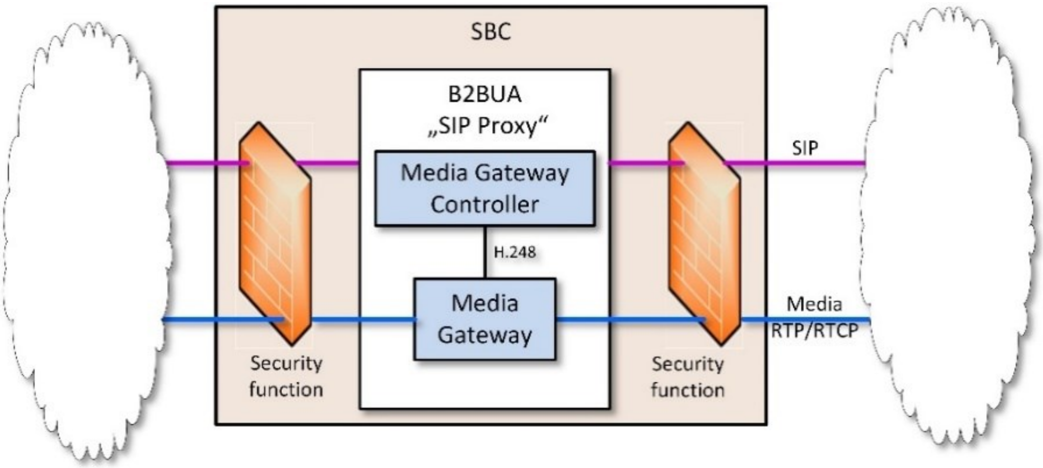

Figure 2.4 Session Border Controller (SBC)

From the point of view of mobile network providers, SBC is a security feature that protects the core of their networks from other mobile network operators or subscribers. They use specialized SBCs that are called:

- Access SBC (A-SBC), which is specialized for the defense against end-user accesses (subscribers). Its role is mainly the authentication of accessing subscribers, billing, and often the creation of IPsec tunnels among accessing participants and the A-SBC. The A-SBC also hides the kernel topology from participants.
- Interconnect SBC (I-SBC), which connects providers to other mobile network providers (operators). Again, it provides e.g. the creation of IPsec tunnels between I-SBC and other providers (or IPX providers), hiding the core network topology.

Since we consider the core of the network a "safe zone", it can minimize security functions at SBC interfaces towards the core of the network. However, this does not mean they cannot be implemented there.

SIP communication between core entities is also considered safe in the sense that entities may not authenticate each other. The A-SBC entity authenticates the subscribers and other entities trust this authentication. This is true even if the participant has been authenticated in the visited (foreign) network and his/her request came through I-SBC.

2.7.3 Securing SIP

An SIP can be secured in several ways:

- IPsec secures IP layer communication. IPsec endpoint public key certificate usually includes the name (not the IP address) of the network interface (similar to the certificate of a web server running on the same computer). A weak point is that a private key and a web server running on the same computer can be misused for the IPsec tunnel of the SIP protocol and vice versa. The solution is to use dedicated servers for each purpose separately e.g. an SBC is such a specialized computer.
- S/MIME [21] is provided only by the body of the message (not the headers bearing the "localization data") between the endpoints. The use of this security in an SIP-based networks is very limited.
- TLS/DTLS [22] [23] appears to be the most secure because it provides the entire message on the application layer. For IPsec reasons, it is not appropriate to combine TLS/DTLS with IPsec.

There are often ask why it is necessary to secure the SIP body. The reason is that we often transmit cryptographic security material for media (RTP) in the SIP message body. The medium (RTP) can then be secured by the SRTP [24].

2.8 Mobile networks

The mobile network attempts to cover the service area as much as possible. It uses base stations to cover the territory. In LTE (Long Term Evolution) [25] networks (the 4th generation) are called “evolved Node B” (eNB). (The 5th generation is gNB.)

The basic technologies in the 4th generation are:

- Evolved Packet System (EPS) that provides subscribers with an IP layer connection e.g. to connect Internet subscribers. Evolved Packet Core (EPC) [26] consists of two parts:
  - The LTE network, which provides the last mile of communication, i.e. the connection between the mobile device and the base station.
  - The EPC that manages the LTE access network and provides mobile subscribers with access to the internet and other IP-based networks, e.g. Internet Multimedia Subsystem (IMS) and IP Exchange network (IPX). EPC has already been introduced with 3G networks. 5G networks have a redesigned core called 5GC [27].
- Internet Multimedia Subsystem (IMS) [28], which forms the next core of the network - the core on the application layer (SIP, RTP, etc.). The IMS ensures multimedia services, i.e. "to make it phone". IMS can be operated by both the same operator operating the EPS and another operator (an operator independent of EPS).

As mentioned above, while EPS provides an IP layer connection, IMS provides services on the application layer. This is an analogue to the Internet where Internet providers provide an IP layer connection and content providers provide services on the application layer.

2.8.1 EPS Entities

The Mobility Management Entity (MME) authenticates subscribers when logging into the network, it monitors the movement of idle participants, determines the Serving Gateway (SGW) which the subscriber's data flow through, etc.

The MME (Figure 2.5) is also responsible for managing mobile-to-network communications, for subscribers' authentication, for generation of cryptographic material, etc. Interestingly, there is an eNB base between the mobile device and the MME. The eNB only forwards communication between the mobile device and the MME. The exception is the initial part of the subscriber's network login, at the beginning (before network login) there is a communication between the mobile device and eNB only.

The Serving Gateway (SGW) is responsible for the management of the subscriber's data flows, the so called data bearers. Through this entity, the subscriber's data flow are passed. Occasionally, the SGW is called an anchor in the mobile network (from the point of view of the subscriber), because if the subscriber moves across the mobile network, all of his/her packets pass through the anchor SGW.

Home Subscriber Server (HSS) contains subscriber's data. Each subscriber is maintained in the HSS under the International Mobile Subscriber Identity (IMSI)

[29]. A part of HSS is the Authentication Centre (AuC). The AuC keeps information about subscribers: personal data, the contracted services, and then the secret K shared by AuC with the USIM smartcard. The shared secret K serves to authenticate the subscriber and to generate cryptographic material to secure the subscriber's communication with the mobile network (K is of course different for each USIM).

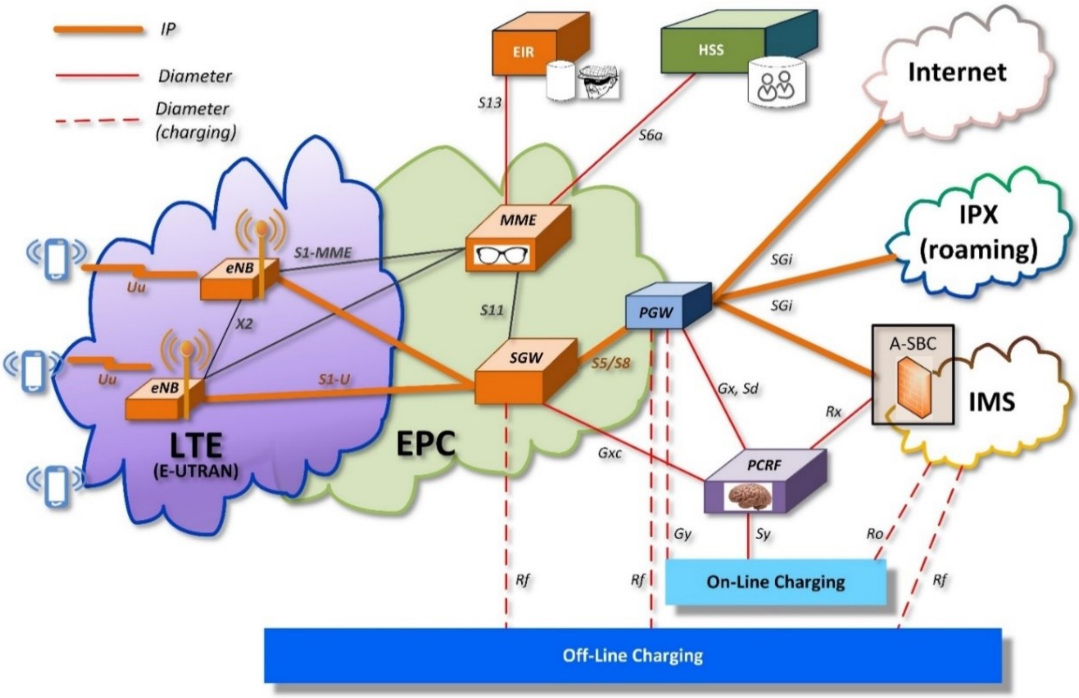

Figure 2.5 Evolved Packet System (EPS) with marked reference points

A moderator of the entire network is the entity Policy and Charging Rules Function (PCRF), which defines network policies in real time. It automatically creates rules for each subscriber enrolling in the network. Through the Sd reference point (Figure 2.5), the PCRF can monitor subscriber traffic and, based on monitoring, it can change its network policies, if necessary. In the case of incoming VoLTE calls (incoming from the IMS), the PCRF may request the SGW via the Gxc reference point to allocate the appropriate data bearer for an incoming call (this request is originated by A-SBC using the reference point Rx).

The Public Data Network Gateway (PDN GW or simply PGW) is a gateway, which forwards subscriber's data packets to external networks. The external network can be Internet, Internet Multimedia Subsystem (IMS), or IP Exchange network (IPX) in the case of roaming. The subscriber rather knows the term Access Point Name (APN), which we can simply imagine as an identification of the external interface of PGW (Figure 2.6).

From the point of view of an external user, the entire EPS (EPS = LTE + EPC) appears as if all LTE subscribers were on the local network behind PGW (Figure 2.6). As in the case of household connections to the Internet, similarly mobile network subscribers are behind the access modem (in this case PGW) i.e. the LTE provides a mobile subscriber IP connectivity. The way the subscriber communicates with the external networks specifies the Access Point Name (APN) of the PGW. For different APNs, the mobile operator may have different quality of services.

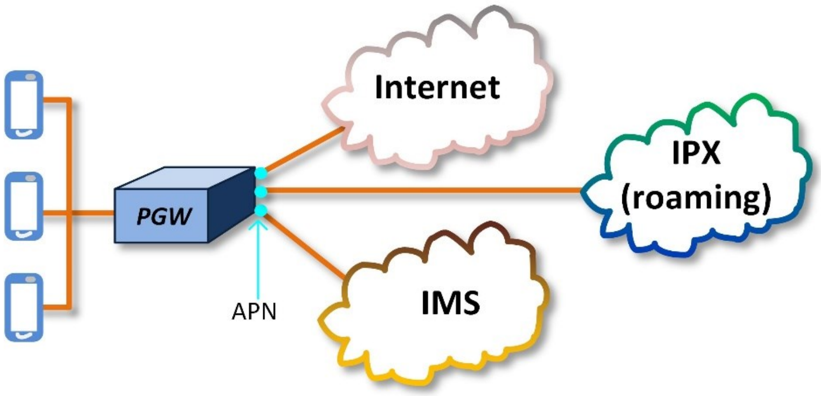

Figure 2.6 Viewing EPS from external networks

The Equipment Identity Register (EIR) – a register of stolen devices. A part of the subscriber's login procedure is checking that the device has not been stolen. It is detected using the International Mobile Equipment Identity (IMEI). The MME entity in the EIR confirms that the subscriber's device has not been stolen.

We cannot forget about charging. If participants only had contracts with the operator and the operator sent them an invoice at the end of the month, then we would be able to make off-line charging. With the introduction of credit coupons, on-line charging has to be introduced so that it can be detected in real time whether the subscriber has sufficient credit for the requested service.

2.8.2 Control Plane and Media Plane

When looking at EPS from a higher level, we can see that it consists of two planes (Figure 2.7):

- The Control Plane, which ensures that everything works. From the point of view of the subscriber, this layer cannot be seen.
- The Media Plane, or User Plane the role of which is to ensure transferring subscriber's IP datagrams, often from the mobile device to external packet networks (Internet, IPX, IMS, etc.). For each subscriber the Media Plane creates one Default Bearer and optionally one or more Dedicated Bearers for a

particular data service. Data bearers are of different categories. For example for audio we require a bearer of guaranteed bandwidth (Quality of Services). For mail transfer, we do not require guaranteed Quality of Services.

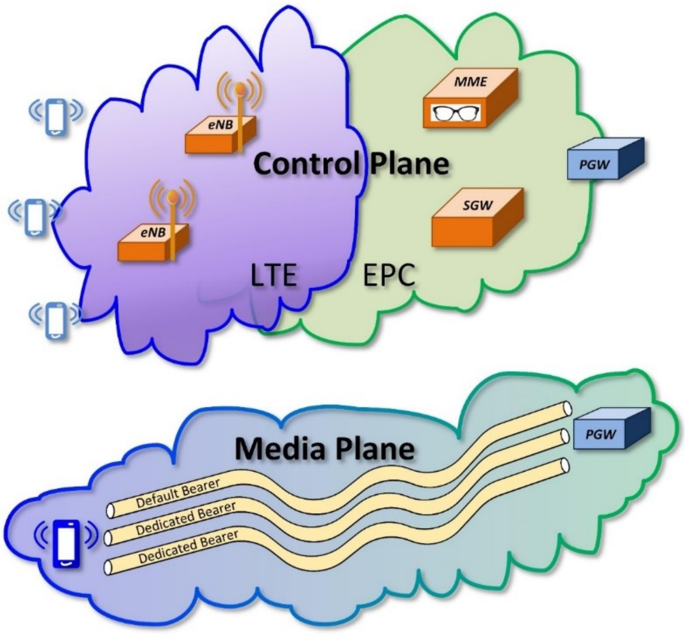

Figure 2.7 Top level view of EPS

2.8.3 Radio Bearers and Data Bearers

The term bearer is used in two different network layers (Figure 2.8):

1. In the physical layer we have radio bearers between the mobile device and the eNB base station:
   - A Signaling Radio Bearer 0 (SRB0), which is used for communication between the mobile device and the base station. It transmits the initial communication (RRC Connection Request [30], RRC Connection Setup [30], etc.)
   - A Signaling Radio Bearer 1 (SRB1), which transports NAS [31] packets before the transmission security is enabled.
   - A Signaling Radio Bearer 2 (SRB2), which transfers NAS packets after security activation.
   - The Data Radio Bearer, which serves to create data carriers. There may be more than one data radio bearer. Similar to the NAS protocol, security will be based on cryptographic material K_{ASME} (see section 2.11).
2. In the application layer, there are GTP-U [32] tunnels which encapsulate subscriber's IP datagrams (IP over IP), i.e.:
   - The GTP-U tunnel between eNB and SGW. All communication between eNB and SGW is usually secured by IPsec.

- ○ The GTP-U tunnel between SGW and PGW. All communication between SGW and PGW is usually secured by IPsec.

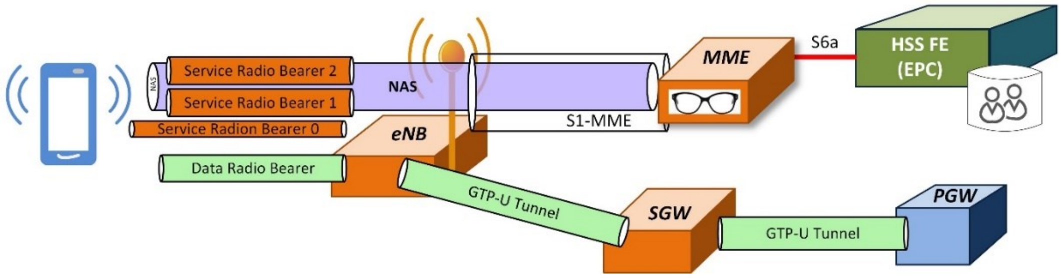

Figure 2.8 Radio Bearers and Data Bearers

2.8.4 EPS Reference Points

Interconnections between entities are described by using reference points. The individual EPC reference points and their network models are shown in Figure 2.9.

After the Radio Bearer (Figure 2.9) is allocated, eNB starts data forwarding between the mobile device and the MME. A Non-Access Stratum (NAS) protocol is used for forwarding this data.

The Uu reference point has one Network Plane model and another for the User (Media) Plane. The Control Plane uses the Radio Resource Control (RRC) [30]. The User Plane uses IP. Together, they use the following protocols: the Packet Data Convergence Protocol (PDCP) [33], Radio Link Control (RLC) [34], and Medium Access Control (MAC) [35].

EPC reference points consist of two groups:

- Control Plane reference points - these reference points provide network control (network signaling). These are reference points:
  - ○ S1-MME (sometimes referred to as S1-C). This reference point transfers data between eNB and MME. The S1AP protocol is defined as the application protocol.
  - ○ S11 which provides communication between MME and SGW,
  - ○ The portion of the X2 reference point designed for network signaling that provides mutual communication between individual eNBs. The application protocol is defined as the X2AP protocol.
  - ○ The part of the S5/S8 reference points that provide network signaling.

  - Diameter reference points for communication with HSS, EIR, and PCRF (Figure 2.5).

- User (Media) Plane reference points for the transmission of a subscriber’s data (subscriber IP datagrams). These reference points use the IP protocol that carries the GTP-U tunnel protocol packets for communication between entities. The GTP-U then tunes the subscriber's IP datagrams. This is IP over IP tunneling. The tunnel may lead to foreign networks which is important in roaming.

  The lower IP is the communication within the EPC, the upper IP is the user's communication. These are reference points:

  - The part of the S5/S8 reference point that provides user data transfer. If the user data are transmitted through the S5/S8 reference point, then the GTP-U protocol is used.
  - The S1-U reference point that transmits data between eNB and the SGW anchor.
  - The part of the X2 reference point for transmitting user data to the mutual communication between eNBs.
  - The part of the S5/S8 reference point that transmits the user data. If the S5/S8 reference point is transmitted by the user data, then the GTP-U protocol is used

A special case is the Uu reference point, which ensures communication between the mobile device and eNB. The Uu reference point does not use TCP/IP family protocols for data transport only, unlike other reference points.

2.8.5 IMS

We have two worlds: one is the EPS (EPS = LTE + EPC) [26] and the other is Internet Multimedia Subsystem (IMS) [28]. While the EPS provides IP connectivity, the IMS allows us to "make calls". In other words, the EPS provides IP layer services; the IMS provides services in the application layer.

The IMS mainly uses network protocols the Session Initiation Protocol (SIP) [14], Real Time Protocol (RTP) [15] and Diameter Base Protocol [36]. The Diameter Base Protocol [36] is a next-generation protocol for the interposition of authentication, authorization, and accounting.

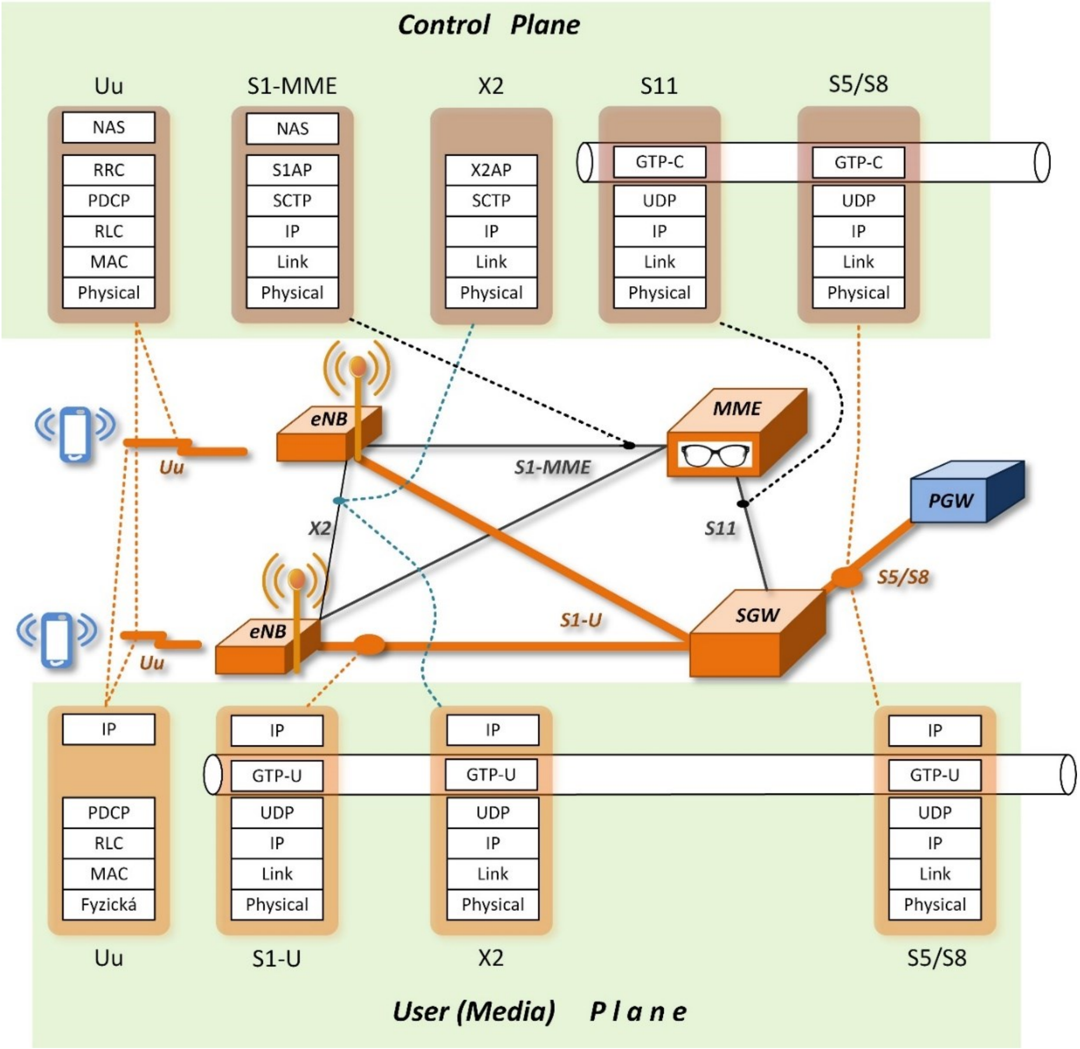

Figure 2.9 Reference points of EPS

The SIP and RTP protocols usually use underlying unacknowledged transport protocols, e.g. UDP [37] or especially SCTP [38].

Figure 2.10 shows the IMS architecture. IMS consists of the following entities:

- The A-SBC (Access SBC) accessed by subscribers.
- The Interconnect SBC (I-SBC) that makes interconnection to other operators or obsolete networks.
- The Call Session Control Function (CSCF) that is responsible for processing SIP requirements. CSCF consists of a number of SIP proxies (B2BUA entities respectively) that are described below. CSCF can be compared to a telephone exchange.

- Application services that provide application features (AF). Individual AFs can be equally operated by third parties. Examples of application features include Presentation Services, Conference Servers, Instant Messaging Servers, etc.
- The media resource that provides multimedia messaging, or mixing of media streams e.g. Session Control can provide audio: "The called party is temporarily unavailable".
- Entities that may be shared with EPC:
    - The Home Subscriber Server (HSS) again contains information about the subscribers, their services and the secret K that is shared with the subscriber's USIM/ISIM smartcard. On the basis of this secret, the participant authenticates against the P-CSCF which will hand over this request to S-CSCF, which handles them through the HSS. (The HSS can be common to both the EPS and IMS if both networks are operated by the same operator.)
    - Policy and Charging Rules Function (PCRF) moderate networks in real time.

The S-CSCF entity i.e. the core of CSCF. S-CSCF handles requirements but does not contain any security features. It is supposed to be protected by SBC (A-SBC and I-SBC).

Entity E-CSCF provides emergency calls (e.g. "112"). Emergency calls are interesting from the point of view of authentication. The subscriber does not have to be authenticated for emergency calls (applies to public mobile networks).

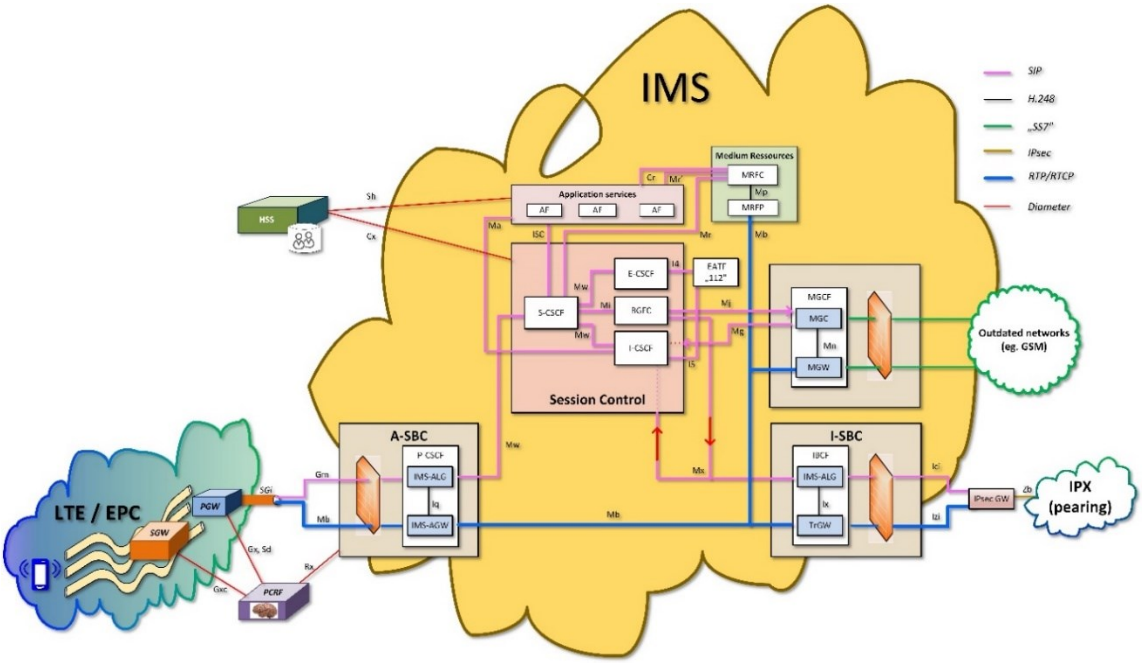

Figure 2.10 Internet Multimedia Subsystem (IMS)

If the called party is not on the same network (in the same IMS), then the request is forwarded to the BGFC entity to find out whether the called party is a subscriber of another IMS provider and afterwards the request is passed to the IBCF entity. If the subscriber is a subscriber of outdated network (e.g. based on SS7[2] protocols family), then the request is passed to the MGFC entity.

On Figure 2.10 the following entities are not shown:

- The Subscriber Location Function (SLF), which is important if we have more HSS entities in the network. The SLF can find in which HSS the subscriber's data are located.
- Charging entities.
- The Domain Name System (DNS).

In VoLTE, subscribers are not primarily addressed by classic phone number but by a SIP URI that is reminiscent of an email address. The DNS is therefore important for finding the SIP server of the domain which it is being called to.

If classic phone numbers are used then there is a need to use DNS ENUM [39] to convert the phone number to SIP URI. Records of DNS ENUM also solve the portability of phone numbers. the For DNS ENUM, it should be emphasised that the DNS in the IMS uses IPX root DNS servers, not root Internet DNS servers.

Table 2.2 List of IMS entities

CSCF	There are Call Session Control Function (CSCF).
P-CSCF	It is the first IMS entity to contact the subscriber mobile device. Although it has P for Proxy in the name, in the SIP protocol terminology it is B2BUA.
S-CSCF	Processing calling requests and maintaining session status. It provides the authentication of subscribers (on behalf of the P-CSCF) using the HSS. In the HSS, it also searches for the subscribers who are called.
E-CSCF	Processing emergency calls ('112').
I-CSCF	It is a contact point for incoming calls from foreign networks.
MGCF	The Media Gateway Control Function is a gateway to networks based on protocols other than the SIP (GSM, etc.). In particular, it provides conversion of network protocols and media conversion format (transcoding).
BGCF	The Breakout Gateway Control Function is a dispatcher of outgoing requests. It decides what external network the request is going to be transmitted to: whether to SIP-based networks or obsolete networks (switching circuits) based on SS7.
IBCF	An Interconnection Border Control Function makes appropriate modifications to SIP/SDP requests between networks of different operators. It can make IPv4 and IPv6 conversions, hiding the operator topology, generating charging, etc.

[2] Obsolete telephone signalling protocol introduced in 1975 [80]

TrGW	The Transition Gateway performs IP address/port translation, IPv4 and IPv6 conversion, etc. Under certain circumstances, it can also convert media (transcoding).
EATF	The The Emergency Access Transfer Function is a gateway to the integrated rescue system. Requirements are received both from the E-CSCF network itself and from external networks (I-CSCF).
IMS-ALG	The IMS Application Level Gateway performs SIP/SDP level adjustments e.g. the authenticated subscriber may be in the appropriate SIP header marked as trusted applying to communication with other entities. The IMS-ALG also converts IP addresses etc.
IMS-AGW	IMS Access Gateway performs IP address / port translation, IPv4 and IPv6 conversion, etc. Under certain circumstances, it can also convert media format (transcoding).
SEG	The Security Gateway (or IPsec GW) ending IPsec tunnels.
AF	Application Functions – application based on SIP/RTP protocols.
SLF	The Subscriber Location Function – if there is more than one HSS in a network, then the network must ask the SLF which HSS should be requested for the subscriber's information. More than one HSS can be used, e.g. if the network shares more operators, if the network is large, etc.

2.9 USIM and ISIM

The USIM and ISIM are applications embedded in the Universal Integrated Circuit Card (UICC), which shares secret *K* with Authentication Centre (AuC) for AKA authentication as described in section 2.11. USIM / ISIM can be made as a smartcard (colloquially “SIM card”) or generally as a secure element [8].

The LTE (Network Long Term Evolution) contains the mile of communication between the mobile device and the base station (eNB). The base stations (eNB) are connected to the core network EPC (Evolved Packet Core) that will ensure that mobile devices can communicate to IP networks. Access into the network is controlled by the Mobility Management Entity (MME).

If we do not only require IP connectivity from mobile networks but application services, then we need to authenticate on the application layer. At the application layer, the mobile device communicates with the IMS network which provides multimedia services.

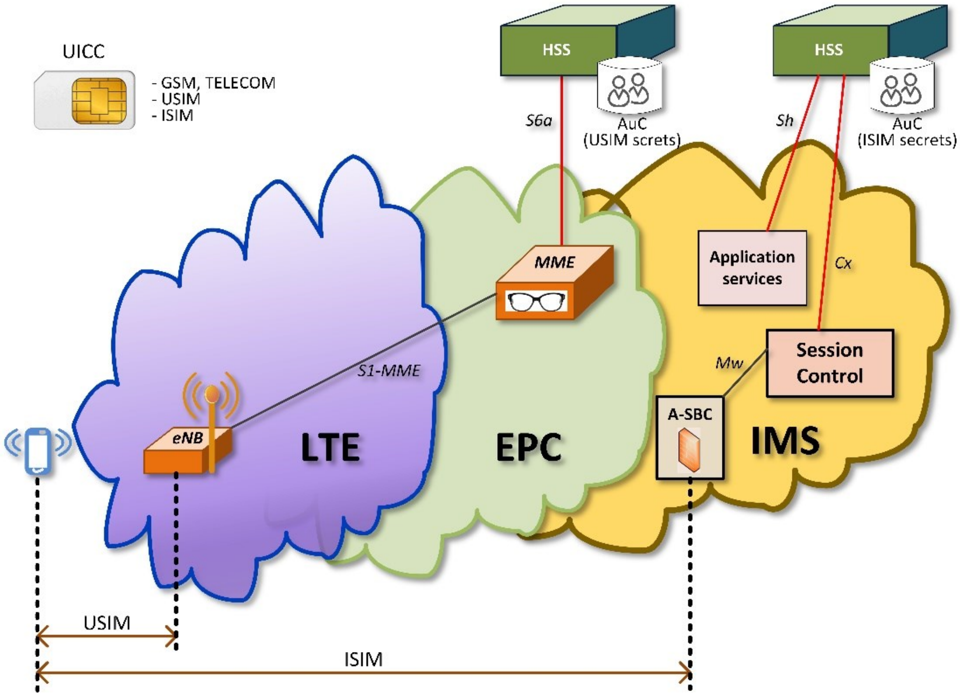

Figure 2.11 Authentication in mobile networks

The client authenticates towards the LTE/EPC using the shared secret K (described in section 2.11), which he/she has stored in the Universal Subscriber Identity Module (USIM) application on his/her Universal Integrated Circuit Card (UICC) via the AKA mechanism. The network has the shared secrets stored in the Authentication Center (AuC) of LTE/EPC networks. AuC is jointly operated with the subscriber database by the Home Subscriber Server (HSS).

In general, we can have another HSM for LTE and another for IMS (some telco operators use the same shared secret *K* for both LTE and IMS, so they don't have to implement the ISIM, only the USIM.)

At the application layer, the client authenticates towards the IMS using the shared secret K that has been stored in the IP Multimedia Services Identity Module (ISIM) application to his/her Universal Integrated Circuit Card (UICC). The network has shared the secret K stored in the Authentication center (AuC) of the IMS network. The authentication has also used AKA mechanism, however be encapsulated into another protocol.

Generally, the shared secret *K* for LTE/EPC is other than the IMS network. In practice, however, the client's smartcard often contains only the USIM application and does not contain the ISIM application, so they use the same shared secret for the LTE/EPS and for the IMS network too. In that case, on the Telco side AuC is common for the LTE/EPC and for the IMS.

2.10 EAP

The Extensible Authentication Protocol (EAP) [40] brought the possibility for the client and the server to first agree on the authentication method and then to authenticate. EAP is used to select a specific authentication method, typically after the authenticator requests more information in order to determine the specific authentication method to be used. Rather than requiring the authenticator to be updated to support each new authentication method, EAP permits the use of a backend authentication server, which may implement some or all authentication methods, with the authenticator acting as a pass-through for some or all methods and peers.

The AKA (Authentication and Key Agreement) mechanism described in the next section is one of the authentication methods introduced in the EAP.

The Extensible Authentication Protocol Method for the 3rd Generation called Authentication and Key Agreement (EAP-AKA) was defined by [41]. The standard [42] improves the EAP-AKA to EAP-AKA'. The EAP-AKA' is a small revision of the EAP-AKA. The change is a new key derivation function that binds the keys derived within the method to the name of the access network. In addition, the EAP-AKA' employs the SHA-256 hash algorithm [43] instead of SHA-1 [44].

5G networks use EAP-AKA' and new 5G AKA which is another revision of AKA [45].

2.11 AKA mechanism

AKA (Authentication and Key Agreement) mechanism is a security protocol used in the 3G/4G/5G mobile networks for mutual authentication and cryptographic material (Figure 2.12) agreement.

There are three communication parties:

- A (mobile equipment), usually equipped with the USIM/ISIM containing a shared secret K.
- B (network), the MME in LTE or A-SBC (P-CSCF) in IMS.
- Authentication Centre (part of a Home Subscriber Server - HSS).

Table 2.3 Notation of AKA

Symbol	***Meaning***
K	*Shared secret (permanent key stored in USIM/ISIM and at the same time in AuC)*
AMF	*Authentication Management Field. Well known string defined by standard* [1]
SQN	*Sequence number of authentication*
RAND	*Random number*
MAC-A	*One time password of network authentication*
RES	*RESponse that is the output of the function f2 (one time password of user equipment authentication)*
CK, IK	*Cyphering key, Integrity key (both keys derived in AuC and on USIM/ISIM during AKA authentication)*
K_{ASAME}	*Is a key derived by mobile equipment and AuC during an AKA authentication*
AK	*Anonymization key (anonymized SEQ)*
f 1 - f 5	*One way functions*

Parties A (mobile) and AuC:

- share a secret *K* (shared secret) different for each smartcard,
- maintain sequence number *SEQ* of authentication.
- Both parties support the AKA mechanism using one way functions *f1, f2, f3, f4,* and *f5* (Figure 2.13) which are defined in [46] [47].
- In addition, there is the well-known *AMF* string.

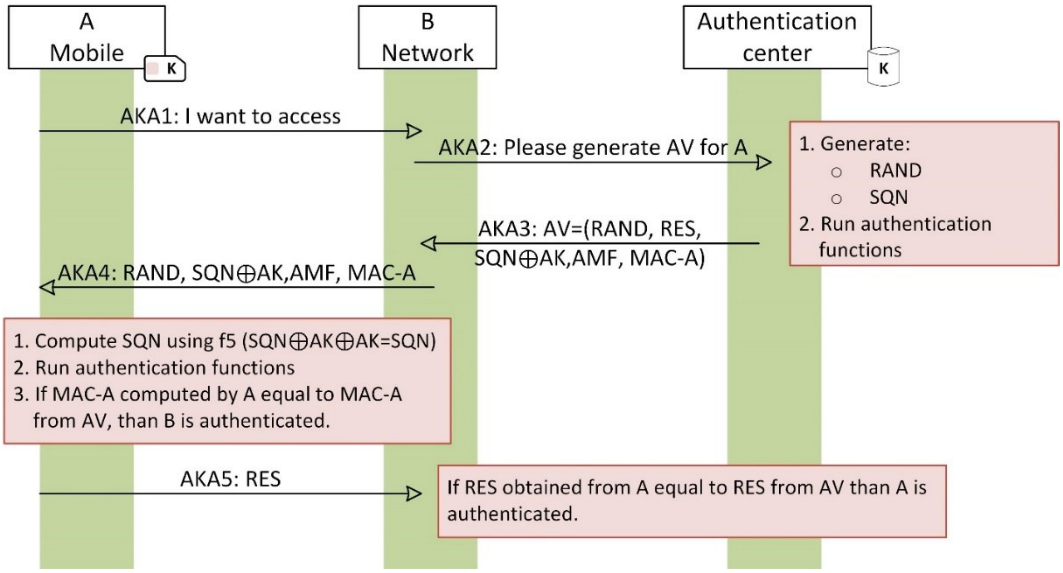
A
Mobile
K
B
Network
Authentication
center
K
AKA1: I want to access
AKA2: Please generate AV for A
1. Generate:
○ RAND
○ SQN
2. Run authentication functions
AKA3: AV=(RAND, RES, SQN⊕AK,AMF, MAC-A)
AKA4: RAND, SQN⊕AK,AMF, MAC-A
1. Compute SQN using f5 (SQN⊕AK⊕AK=SQN)
2. Run authentication functions
3. If MAC-A computed by A equal to MAC-A from AV, than B is authenticated.
AKA5: RES
If RES obtained from A equal to RES from AV than A is authenticated.

Figure 2.12 AKA mechanism

AKA mechanism (Figure 2.12) is the following:

AKA-1: *A* (mobile) will grant access, then sends its identity to B. Concrete: A wants mutual authentication – its own authentication and also the authentication of *B* (network).

AKA-2: *B* (network) sends identification of *A* (mobile) to the Authentication center (AuC).

AKA-3: The Authentication center on behalf of *B* (network):

- Generates a *RAND* random number.
- Generates the next *SEQ* sequence number of authentication.
- Runs Authentication functions *f1, f2, f2, f4,* and f5 (Figure 2.13) and generates the Authentication vector *AV = (RAND, RES, CK, IK, SQN ⊕ AK, AMF, MAC-A*). While *RES* is a one-time password for authentication of *A* (mobile) and *MAC-A* is a one-time password for authentication of *B* (network).
- Sends *AV* to *B* (network).

AKA-4: *B* (network) sore *RES* form Authentication vector *AV.* Next, *RAND, SQN ⊕AK, AMF* and *MAC* sends to *A* (mobile).

AKA-5: *A* (mobile) know: *SEQ* and *AMF* and next from *B* (network) receives *SQN ⊕AK, AMF* and *MAC-A. A* at first by function *f5* verifies *SQN*. Next *A* runs the rest of authentication functions and:

- If the *MAC-A*, computed by *A* (mobile), is equal to the *MAC-A* from *AV*, then *B* (network) is authenticated.
- Generates *RES* and sends it to *B* (network).

AKA-6: *B* (network) compares the *RES*, obtained from *A* (mobile), with saved *RES* from *AV* (step *AKA-4*). If equal, *A* (mobile) is authenticated.

AKA should be used for mutual "silent" authentication. The word "silent" means, that the user single sign on (after switching on the mobile equipment, a user inserts his/her PIN for opening access to the shared secret *K*) and the application runs subsequent authentication without any user intervention, e.g. when entering the visited network.

Cryptographic key K_{ASAME} is a result of AKA authentication. It is known by both *A* (mobile) and *B* (network). Cryptographic material is then derived from K_{ASAME} to ensure communication between *A* and *B*.

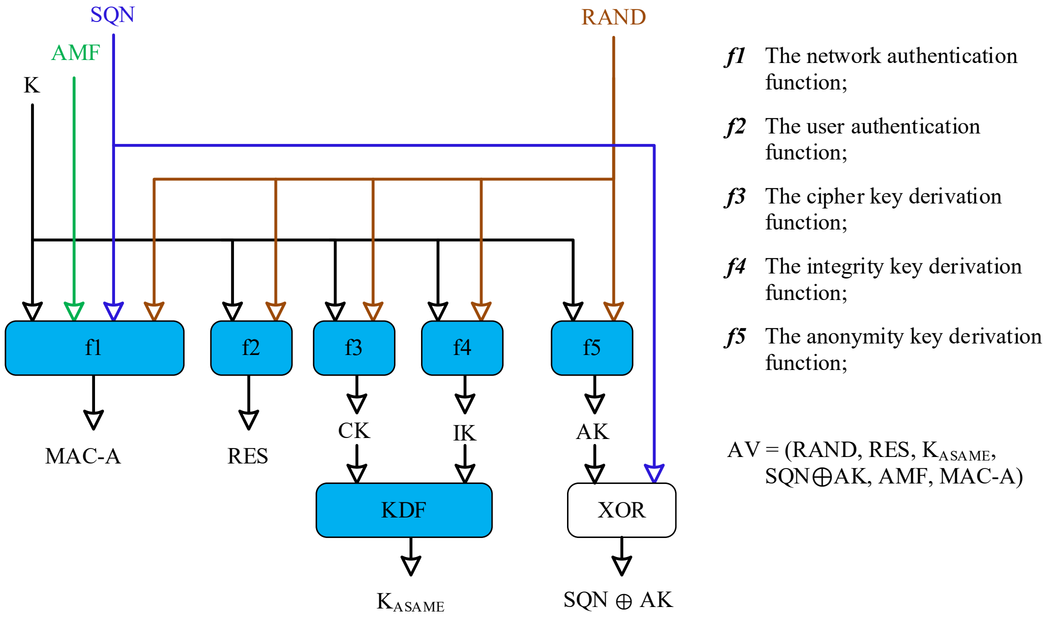
SQN
AMF
K
RAND
f1
f2
f3
f4
f5
MAC-A
RES
CK
IK
AK
KDF
XOR
K_{ASAME}
SQN ⊕ AK
f1 The network authentication function;
f2 The user authentication function;
f3 The cipher key derivation function;
f4 The integrity key derivation function;
f5 The anonymity key derivation function;
AV = (RAND, RES, K_{ASAME}, SQN⊕AK, AMF, MAC-A)

Figure 2.13 AKA Authentication function f1-f5

It is also necessary to define functions *f1, f2, f3, f4,* and *f5*. A concrete example of functions *f1, f2, f3, f4,* and *f5* is defined by the “MILENAGE algorithm set” [46] [47] protocol.

2.12 Use of AKA in EPS

Before the subscriber login to the network, the mobile device first searches for the network. Next, the mobile device will choose a network and starts communicating with the base station of the chosen network on a random access channel. The next step is to create a connection at the RRC protocol level (specifically using SRB0). This communication is unsecured.

Subsequently, the subscriber authenticates. The MME entity will already used in this step (the eNB will only mechanically pass NAS packets between the subscriber and the MME). The AKA mechanism is used for authentication (Figure 2.14). If the authentication is successful, then the subscriber is logged on to the network, and IK and CK (K_{ASAME}) cryptographic materials are also provided, which will then be used to secure the subsequent communication.

Now a secure channel is created between the mobile device and the MME. Subsequently subscriber authorization (SRB2) can be performed, i.e. MMEs obtain localizing data and the specifications of contracted services from HSS. All information to create a data carrier is now available.

Once these three parts have been created, they are welded into a single data bearer between the mobile device and the PGW. The mobile device will behave as if it were one hop from the PGW (Figure 2.6).

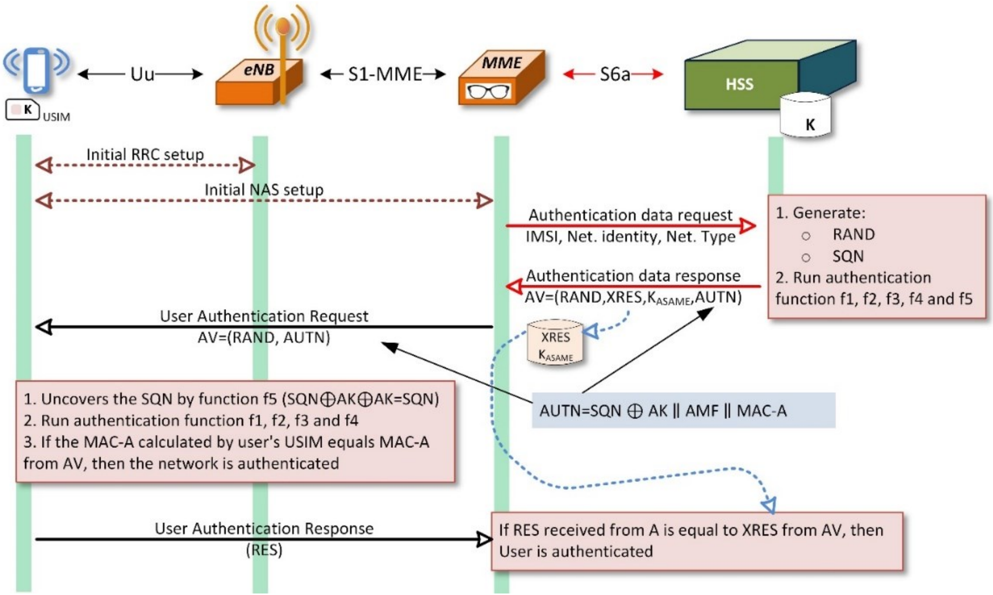
K
USIM
Uu
eNB
S1-MME
MME
S6a
HSS
K
Initial RRC setup
Initial NAS setup
Authentication data request
IMSI, Net. identity, Net. Type
1. Generate:
○ RAND
○ SQN
2. Run authentication function f1, f2, f3, f4 and f5
Authentication data response
AV=(RAND,XRES,K_{ASAME},AUTN)
User Authentication Request
AV=(RAND, AUTN)
XRES
K_{ASAME}
1. Uncovers the SQN by function f5 (SQN⊕AK⊕AK=SQN)
2. Run authentication function f1, f2, f3 and f4
3. If the MAC-A calculated by user's USIM equals MAC-A from AV, then the network is authenticated
AUTN=SQN ⊕ AK || AMF || MAC-A
User Authentication Response
(RES)
If RES received from A is equal to XRES from AV, then User is authenticated

Figure 2.14 Using AKA in EPS (IC are IK are included in K_{ASAME})

2.13 Use of AKA in IMS

This registration is again based on the challenge-response method. Figure 2.15 shows the process in a simplified way (not all entities through which communications are shown, but only those that are important to understanding the process).

In general, a subscriber is in a visited network. Therefore, the registration request is sent to the P-CSCF (part of the A-SBC) of the visited network. The visited network does not have cryptographic material for subscriber authentication, so it sends the request to the subscriber's home network. In the home network, the request goes to the S-CSCF entity that is responsible for handling the request. The S-CSCF entity asks for the appropriate cryptographic material of the HSS entity (only the home HSS shares the K secret with the USIM / ISIM subscriber). The HSS generates the AV Authentication Vector and passes it to the S-CSCF, which retrieves the RES from the authentication vector and passes the rest to the response.

The security architecture in 3GPP networks specifies 3GPP TS 33.203 [48]. 3GPP networks use IPsec security for the SIP (TLS / DTLS or other mechanisms are also

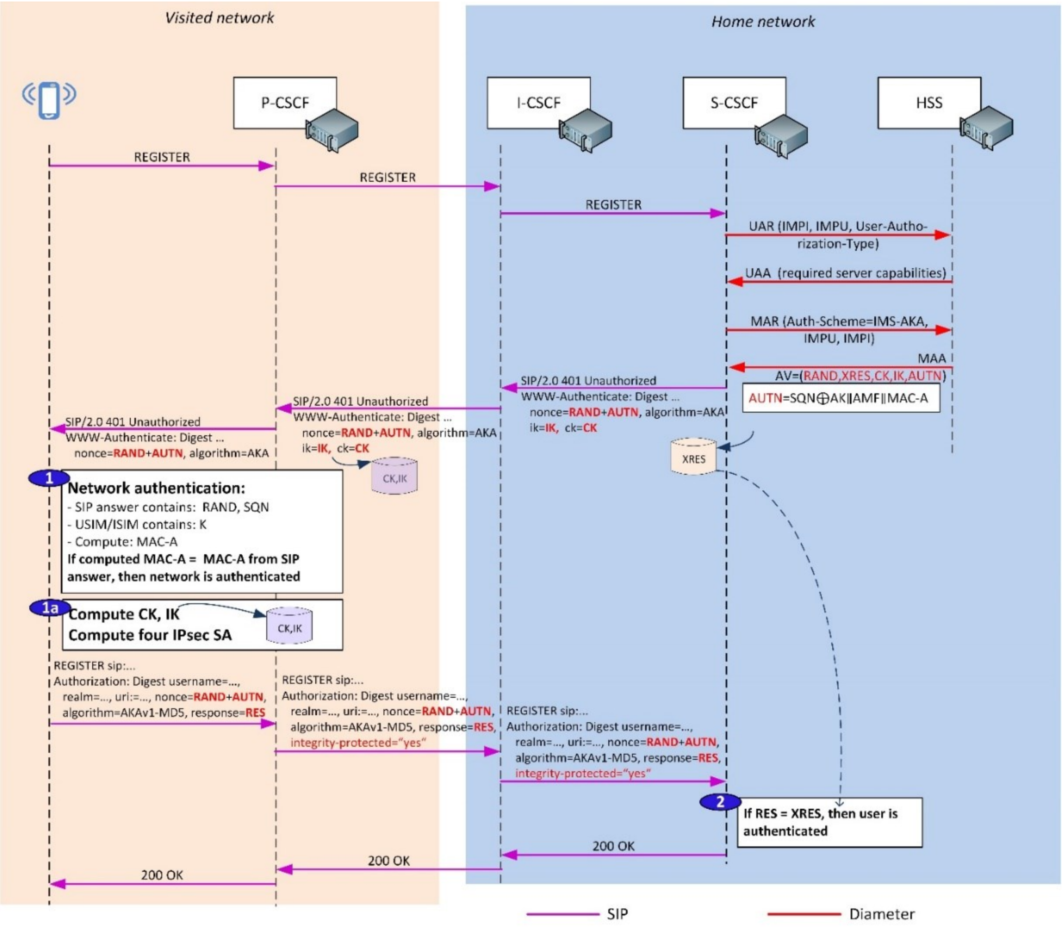
Visited network
Home network
P-CSCF
I-CSCF
S-CSCF
HSS
REGISTER
REGISTER
REGISTER
UAR (IMPI, IMPU, User-Autho-rization-Type)
UAA (required server capabilities)
MAR (Auth-Scheme=IMS-AKA, IMPU, IMPI)
MAA AV=(RAND,XRES,CK,IK,AUTN)
AUTN=SQN⊕AK‖AMF‖MAC-A
SIP/2.0 401 Unauthorized WWW-Authenticate: Digest ... nonce=RAND+AUTN, algorithm=AKA ik=IK, ck=CK
SIP/2.0 401 Unauthorized WWW-Authenticate: Digest ... nonce=RAND+AUTN, algorithm=AKA ik=IK, ck=CK
SIP/2.0 401 Unauthorized WWW-Authenticate: Digest ... nonce=RAND+AUTN, algorithm=AKA
XRES
CK,IK
1
Network authentication:
- SIP answer contains: RAND, SQN
- USIM/ISIM contains: K
- Compute: MAC-A
If computed MAC-A = MAC-A from SIP answer, then network is authenticated
1a
Compute CK, IK
Compute four IPsec SA
CK,IK
REGISTER sip:... Authorization: Digest username=..., realm=..., uri:=..., nonce=RAND+AUTN, algorithm=AKAv1-MD5, response=RES
REGISTER sip:... Authorization: Digest username=..., realm=..., uri:=..., nonce=RAND+AUTN, algorithm=AKAv1-MD5, response=RES, integrity-protected="yes"
REGISTER sip:... Authorization: Digest username=..., realm=..., uri:=..., nonce=RAND+AUTN, algorithm=AKAv1-MD5, response=RES, integrity-protected="yes"
2
If RES = XRES, then user is authenticated
200 OK
200 OK
200 OK
SIP
Diameter

Figure 2.15 3GPP SIP Registration

allowed on the application layer). IPsec is used both as a security between the subscriber and the A-SBC (Gm reference point), as well as the security between I-SBC and the IPX network provider (Ici and Izi reference points will be IPsec secured and the Zn reference point will be created).

Figure 2.16 illustrates schematically the security of the entire path between the subscriber's device and the A-SBC. We see that A-SBC fulfills two roles: the IPsec tunnel portal and the B2UA agent. The security is done in two layers:

1. The SIP (Gm reference point) is secured using the IPsec across the path from the participant to the A-SBC. The medium (reference point Mb) is usually secured by the SRTP SDES method also throughout the path between the participant and the A-SBC. This is the security on the entire route between the participant and the A-SBC.

   If the encryption was not enabled in the IPsec protocol at the Gm reference point, and the cryptographic material for the SDES was transmitted in the SDP packet in the body of the SIP protocol, SRTP activation would be meaningless as the cryptographic material would be transmitted in an unsecured way.

2. Securing LTE communication between the subscriber and the base station eNB (mobile network radio interface). It also uses the AKA mechanism that shares the K secret between the USIM participant and the HSS.

   EPC security, i.e. secure communications between the eNB and the EPC core. Here, as a rule, the IPsec tunnel is configured.

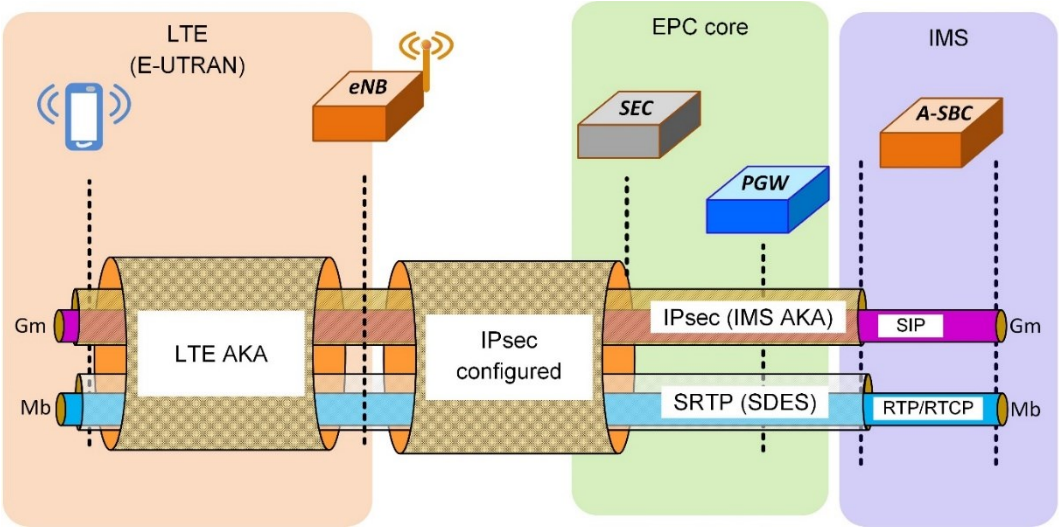

Figure 2.16 Securing the path between the subscriber's device and the A-SBC

2.14 System and Method for Fraud Monitoring, Detection, and Tired User Authentication

The System and Method for Fraud Monitoring, Detection, and Tired User Authentication is patented in [49]. The functions of this patent are configured to provide consistent methods of checking the authenticity and security of a user-initiated request made to a service-provider application, e.g., an online store application, an online banking application, and the like. The methods receive a copy of the request itself or information describing and abstracting the substance of a current request. The input information is processed, and the methods output risk scores, risk alerts, and actions (or action recommendations). Risk scores and alerts are indicia of the likely risks that the current request is incorrect, or malicious, or fraudulent, and so forth. More specifically, the risk scores output are products of number fraud detection inputs which are weighted and analyzed in real time using analytic processes customizable for individual service providers.

In Figure 2.17 processes are structured as shown and include: Footprint processing, Analysis and Authenticator service.

Authentication services are invoked when an online application (services provider system) receives a user request that needs authentication.

Footprint processing (Figure 2.17) is the first authentication process invoked with input data describing the user request. The footprint processing then gathers identifying information describing the device from which the user request originated and creates a device identifier. ("ID Device ").

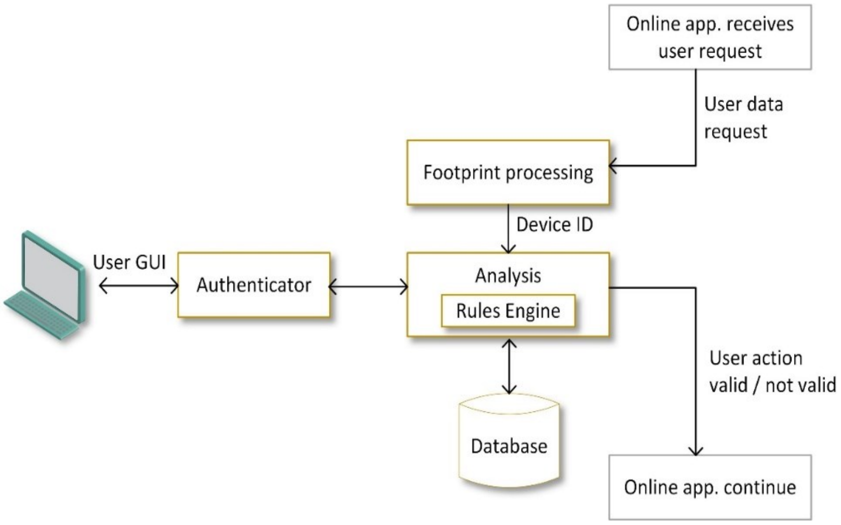

Figure 2.17 Authentication process

Analysis process is invoked with the ID Device (and optionally other devices and/or user identifying information). This process evaluates its input identifying information and can either, e.g., recommend to the online application (Service Provider) or system that the request should be processed further or blocked from the system.

The rules engine (Figure 2.17) analyses models and policies and their data elements in advance of their use during authentication servicing.

Process Analysis begins with retrieving forensic information related to the characteristics of the current request that are apparent in the input request information. Information sources can include services of Database, which stores past authentication results in the authentication system, and third party data services, which can provide a wide range of data, e.g., geolocation data services providing the likely geographical location of the requestor (user). The input data and the retrieved forensic data is then analyzed by a rules-based decision process in order to determine output actions and risk scores.

The System and Method for Fraud Monitoring, Detection, and Tired User Authentication [49] is not the only patent in this field. The patented [50] uses so-called Application Fingerprinting.

In the field of network security, a patent [51] was patented. The time-activity footprints in IP traffic is described in [52]. However, network attack detection based on fingerprints is still evolving today [53].

Systems for fraud monitoring and detection [54] are divided into two types:

- The fraud prevention system (FPS) is the first level of protection to stop fraud from occurring. The mechanisms are restrict, suppress, destruct, destroy, control, remove, or prevent the occurrence of cyber-attacks.
- The fraud detection system (FDS); the next layer of protection. The FDS tries to discover and identify fraudulent activities as they enter the systems and report them.

3 Aims of the Dissertation Thesis

A number of authentication schemes with a data bearer have been published. The publication of some schemes were followed by the publication of their weaknesses, usually supplemented by a change of scheme (or by designing a new scheme) in order to remove the weaknesses.

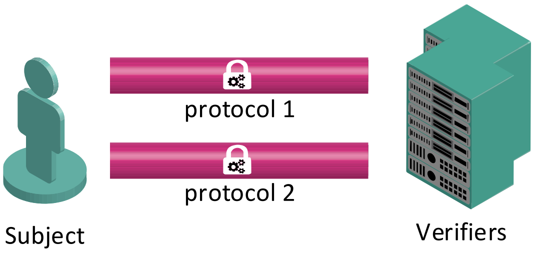

Figure 3.1 Two factor authentication with two different verifiers

In section 0 we mentioned that authentication factors may vary with different cryptographic materials, different authentication schemes, different communication protocols, different communication channels, or different verifiers.

One of the aims of this thesis is to design new authentication algorithms such as those used by more independent verifiers (Figure 3.1). However, it is important that this does not relate to more than one factor authentication used in succession, but simultaneous multifactor authentication i.e. that authentication be intertwined. The basic requirements for the new proposed authentication methods are listed in Table 3.1.

Multi-factor authentication methods are usually more laborious for users and therefore less user friendly. The question is whether it is always necessary to use laborious authentication methods. In which situations is it safe to use even less laborious authentication methods? Another aim of the dissertation is to create a model that would suggest when it is appropriate to use less demanding authentication methods and when it is necessary to use more demanding authentication methods.

Modeling the use of the demanding of authentication methods is the view of the service provider or user (defender). The attacker solves the opposite problem: in what situation is it advantageous for him to attack. This view was another aim of the thesis.

List of aims:

1. To design algorithms that are used by multiple independent verifiers.
2. To analyze and compare proposed solutions
3. To design a model for multi factor authentication.
4. Modelling game attacker with defender (service provider or user).

Table 3.1 Basic requirements for the proposed authentication methods

1. Mutual authentication of subject and verifier
2. Multifactor authentication
3. User anonymity - the secrecy of user identity, means that any third party except the communicating agents cannot knows the identity of the user with whom he is interacting
4. User un-traceability - a stronger property than user anonymity, which requires that any third party should be not only unable to infer the identity of the user but also to link one user session to another one. That is, the adversary is not able to tell whether he has seen the same user twice
5. No time synchronization - a scheme should not require additional clock synchronization mechanisms
6. Anti-desynchronization - the user and server cannot be desynchronized on their shared secrets or values, which would result in that the user being denied any future access to the server
7. In case of password using: a. Password change i. Password reset ii. Password protection against well-known attacks: iii. Password Guessing Attack - stands for an adversary attempting to guess the user's private information (including directory attack, brute force attack etc.). iv. Sniffing attack (incl. network sniffing, keystrokes) v. Replay attack vi. Elicit password attack (forgery verifier attack, phishing etc.).
8. In case of data bearer ("smartcard") using: a. Data bearer revocation. b. In the case of data bearer (token) loss, invalidation of the further use of the lost data bearer (token) should be provided to prevent an adversary from impersonating the registered user.
9. In case of generating cryptographic material for securing subsequent communication:

a. Key freshness - neither party can predetermine the shared session key being established.
b. Perfect forward secrecy - the attacker cannot know any information about a previously established session's key even when the long-term keys of the server and the user are disclosed.
c. Forward and backward secrecy. Forward secrecy means that even though some session keys are exposed for some reason, the secrecy of any earlier session key should still be maintained. Backward secrecy means that even though some previous session keys are exposed, the secrecy of any future session key should still be maintained.

4 Strong Authentication for Internet Mobile Application

This chapter discusses the possibility of strong authentication into applications running on mobile devices. It deals with the possibility of combining the AKA mechanism with the Secure Hash-Based Password Authentication Protocol Using Smartcards [5] .

Table 4.1 Notion of Secure Hash-Based Password Authentication Protocol Using Smartcards

Symbol	***Meaning***
U	*User*
S	*Server*
Uid, ID	*Identification of the user*
XS	*secret key of S*
P	*password of U*
Ku	*Randomly generated key selected by U and shared with the server and stored in secure storage in a smartcard*
rn	*Random nonce generated by U or S*
Ts	*Timestamp*
EKpu	*Public key encryption algorithm with (e.g. RSA) the public key of S. Used whenever user verifiers are stored in the registration phase*
DKpr	*Decryption with the private key of S.*
ESpu(M)	*Encryption of M with the public key of S when U sends M to S*
$D_{Spr}(M)$	*Decryption of M with the private key of S when U's* $E_{Spu}(M)$ *decrypts*
AuthQ, *AuthA*	*Authentication question and answer for the registration, forget password, and password/verifier change Phases.*
h()	*Hash function, such that h(m) signifies the hash of message m.*
h(a, b)	*Hash of concatenated a and b; i.e., h(a, b) = h(a || b)*
$\oplus$	*XOR operation*
||	*Concatenation*

4.1 Secure Hash-Based Password Authentication Protocol Using Smartcards

The Hash-based strong-password authentication protocol was described in [55]. In [56] has been shown that the protocol [55] is vulnerable to impersonation, guessing, and stolen-verifier attacks and an improvement password authentication protocol has been proposed to solve the described problems.

In [5] the security weaknesses of [56] was shown and the Secure Hash-Based Password Authentication Protocol Using Smartcards was proposed. The term "smartcard" is not related to the USIM cards of mobile equipment. In such "smartcard", the user's cryptographic material may be saved anywhere in mobile equipment (theoretically in an elementary file of the USIM).

The Secure Hash-Based Password Authentication Protocol Using Smartcards consists of the following phases [5]:

- Registration.
- Login
- Forgot smartcard (see [5]),
- Password change (see [5]).

4.1.1 Registration Phase

We have two parties: the user U and the server S (Figure 4.1). We assume that the user U has the public key certificate of S. In the registration phase U and S exchange four messages R1, R2, R3 and R4:

R1. $U \rightarrow S$: ID, PV
U inputs his/her ID and password P, generates K_u, and computes the password verifier $PV{=}h(K_u||P) \oplus K_u$. U sends ID and PV to S for a registration request.

R2. $S \rightarrow U$: $Auth_Q$

S computes $IDX{=}E_{Kpu}(h(ID||X_S)) \oplus X_S$ and stores it in S's password file. And S generates random $Auth_Q$ and sends $Auth_Q$ to U.

R3. $U \rightarrow S$: $R = E_{Spu}(K_u,\ P,\ Auth_Q \oplus Auth_A)$

U inputs $Auth_A$ as an answer to the authentication question $Auth_Q$ and computes $Auth_Q \oplus Auth_A$. Next, U encrypts K_u, P, $Auth_Q \oplus Auth_A$ with S_{pu} and sends it to S.

R4. $S \rightarrow U$: $h(ID||X_S)$

When S receives R3, S decrypts it and computes PV' using K_u and P from R. And S compares PV' with the received PV in R1. If they are equal, S stores $XPV{=}E_{Kpu}(h(PV||K_u)) \oplus X_S$,

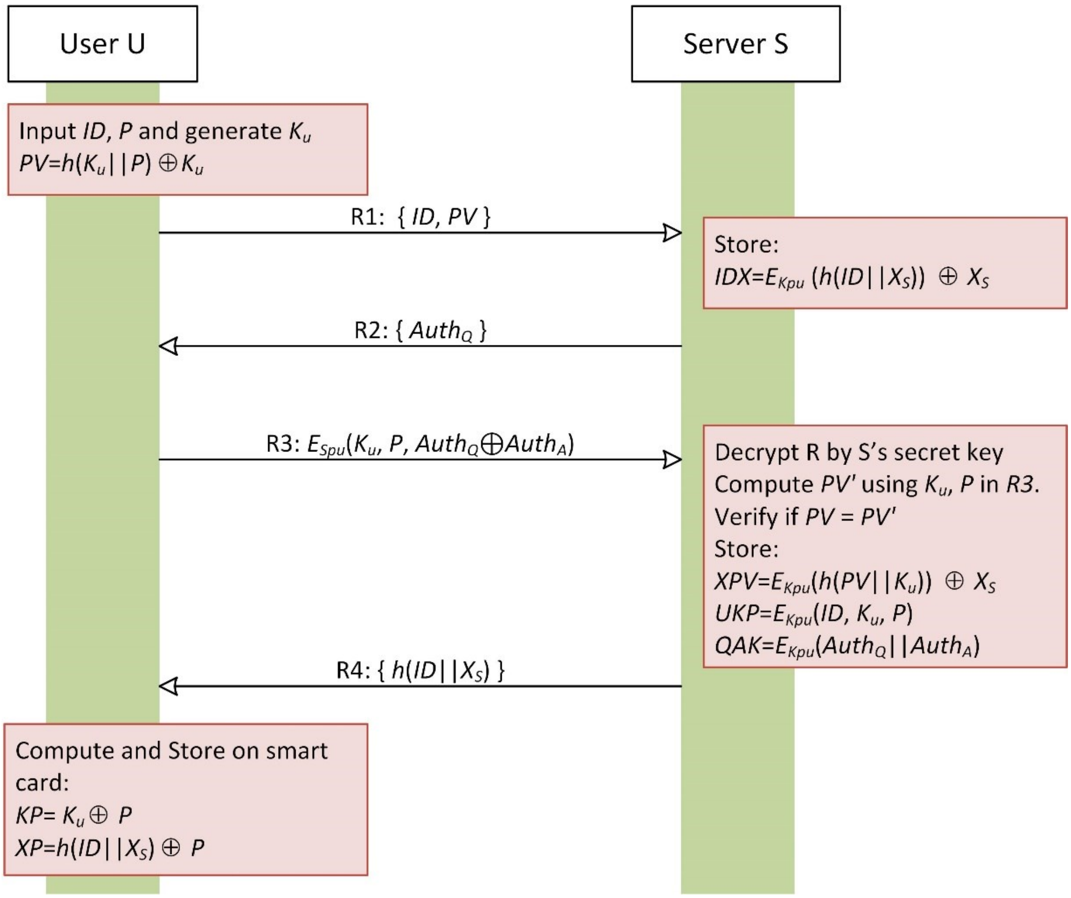
User U
Server S
Input ID, P and generate K_u
$PV=h(K_u||P) \oplus K_u$
R1: { ID, PV }
Store:
$IDX=E_{Kpu} (h(ID||X_S)) \oplus X_S$
R2: { $Auth_Q$ }
R3: $E_{Spu}(K_u, P, Auth_Q \oplus Auth_A)$
Decrypt R by S's secret key
Compute PV' using K_u, P in $R3$.
Verify if $PV = PV'$
Store:
$XPV=E_{Kpu}(h(PV||K_u)) \oplus X_S$
$UKP=E_{Kpu}(ID, K_u, P)$
$QAK=E_{Kpu}(Auth_Q||Auth_A)$
R4: { $h(ID||X_S)$ }
Compute and Store on smart card:
$KP= K_u \oplus P$
$XP=h(ID||X_S) \oplus P$

Figure 4.1 Registration Phase

4.1.2 Login Phase

This phase uses the challenge-response method as protection from replay attack. Figure 4.2 shows the login phase and the detailed steps are as follows:

L1. $U \rightarrow S$: $E_{Spu}(XP \oplus P, ID, r_1)$
U enters his/her smartcard in the card reader (grant access into PV), and inputs *ID* and *P*. Next, *U* generates a nonce r_1 and encrypts *XP*, r_1 and *ID* with S_{pu}. And then *U* sends it to *S* for a login request.

L2. $S \rightarrow U$: CA_1, r_2
When *S* receives L1, *S* decrypts it and computes $G_1=h(ID||X_S)$ and $G_2=D_{Kpr}(IDX \oplus X_S) = h(ID||X_S)$ by decrypting $IDX \oplus X_S$ with *S*'s private key K_{pr}. And *S* compares G_1 with G_2. If they are equal, *S* computes $P = XP \oplus G_2$ and $CA_1=h(ID||P) \oplus r_1$, generates a nonce r_2, and sends them to *U*.

L3. $U \rightarrow S$: $L = h(h(PV||K_u) \oplus r_2) \oplus h(PV||K_u) \oplus r_2$

U computes $CA_2=h(ID||P) \oplus r_1$ and compares CA_2 the received CA_1. If they are equal, *U* computes $L=h(h(PV||K_u) \oplus r_2) \oplus h(PV||K_u) \oplus cr_2$, and sends *L* to *S*. When *S* receives *L*, *S* computes $CB_1=L\oplus r_2$ and computes $CB_2=D_{Kpr}(XPV \oplus X_S) = h(PV||K_u)$ by decrypting $XPV \oplus X_S$ with *S*'s private key K_{pr}. And then *S* computes $CB_3=h(CB_2 \oplus r_2) \oplus CB_2$ and compares CB_3 with CB_1. If they are equal, *U* authenticates *S*.

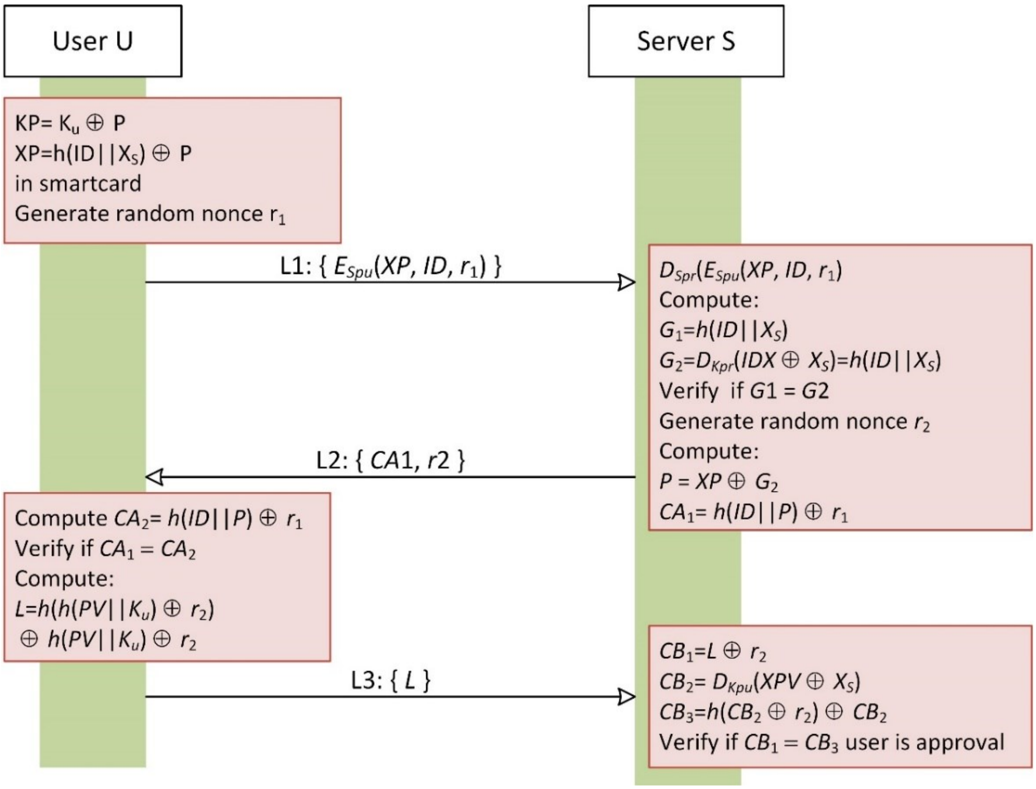
User U
Server S
KP= $K_u \oplus$ P
XP=h(ID||X_S) $\oplus$ P
in smartcard
Generate random nonce r_1
L1: { $E_{Spu}(XP, ID, r_1)$ }
$D_{Spr}(E_{Spu}(XP, ID, r_1)$
Compute:
$G_1=h(ID||X_S)$
$G_2=D_{Kpr}(IDX \oplus X_S)=h(ID||X_S)$
Verify if $G1 = G2$
Generate random nonce r_2
Compute:
$P = XP \oplus G_2$
$CA_1= h(ID||P) \oplus r_1$
L2: { $CA1, r2$ }
Compute $CA_2= h(ID||P) \oplus r_1$
Verify if $CA_1 = CA_2$
Compute:
$L=h(h(PV||K_u) \oplus r_2)$
$\oplus h(PV||K_u) \oplus r_2$
L3: { L }
$CB_1=L \oplus r_2$
$CB_2= D_{Kpu}(XPV \oplus X_S)$
$CB_3=h(CB_2 \oplus r_2) \oplus CB_2$
Verify if $CB_1 = CB_3$ user is approval

Figure 4.2 Login Phase

4.2 *Proposed Solution*

The proposed solution creates multifactor authentication by merging the AKA and Secure Hash-Based Password Authentication Protocol Using Smartcards [5]. The combination of two algorithms creates strong multifactor authentication, which is suitable for applications demanding high secure authentication such as Internet banking or Internet access to the government applications.

The Secure Hash-Based Password Authentication Protocol Using Smartcards [5] consists of registration, login, forget password and password/verifier change phases (section 4.1). Assuming that the user is registered:

– In terms of [5]: the mobile user U and the application function S (server) exchange four messages R1, R2, R3 and R4.

– In terms of the AKA mechanism: User's is equipped by UICC (USIM/ISIM) smartcard, which share secret K with the Authentication Centre (AuC).

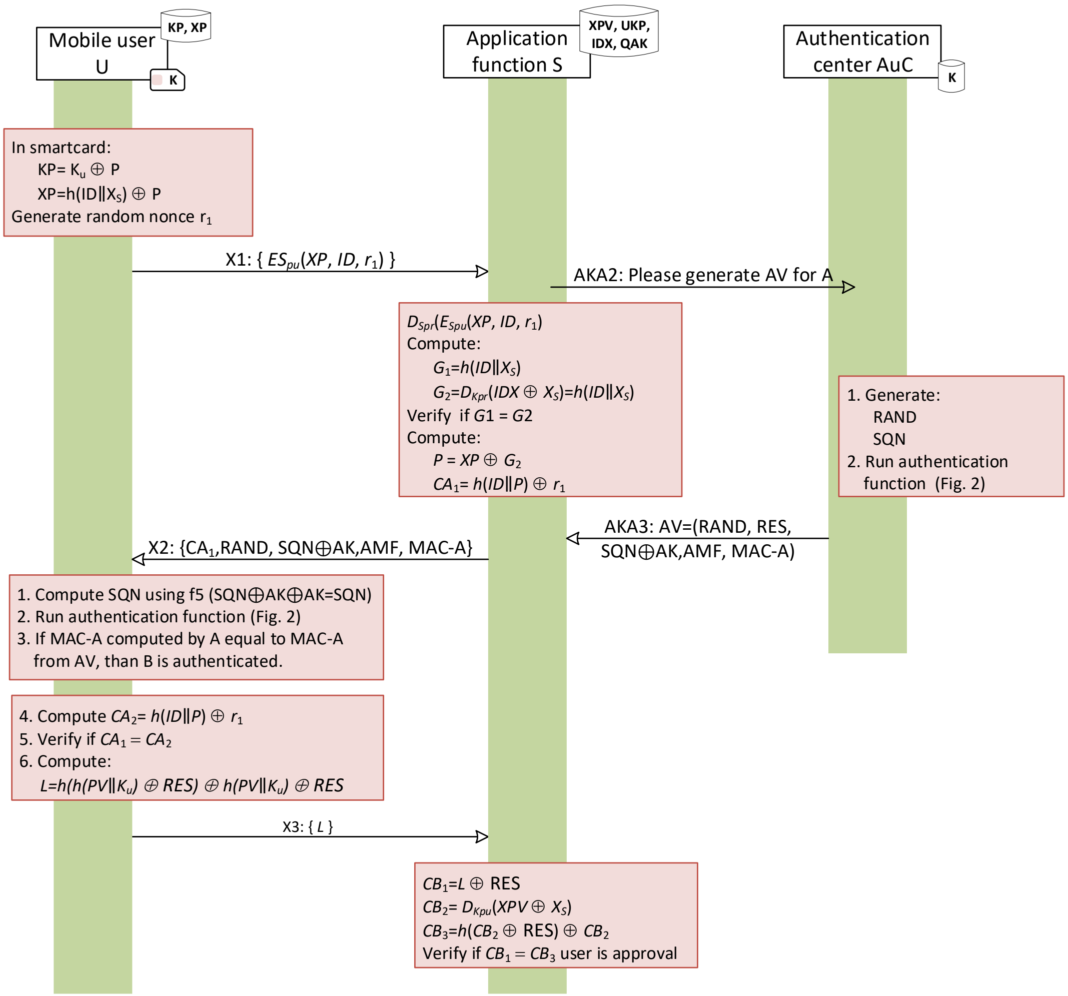
Mobile user U
KP, XP
K
Application function S
XPV, UKP, IDX, QAK
Authentication center AuC
K
In smartcard:
$KP = K_u \oplus P$
$XP = h(ID \| X_S) \oplus P$
Generate random nonce r_1
X1: { $ES_{pu}(XP, ID, r_1)$ }
AKA2: Please generate AV for A
$D_{Spr}(E_{Spu}(XP, ID, r_1)$
Compute:
$G_1 = h(ID \| X_S)$
$G_2 = D_{Kpr}(IDX \oplus X_S) = h(ID \| X_S)$
Verify if $G1 = G2$
Compute:
$P = XP \oplus G_2$
$CA_1 = h(ID \| P) \oplus r_1$
1. Generate:
RAND
SQN
2. Run authentication function (Fig. 2)
AKA3: AV=(RAND, RES, SQN⊕AK,AMF, MAC-A)
X2: {CA_1,RAND, SQN⊕AK,AMF, MAC-A}
1. Compute SQN using f5 (SQN⊕AK⊕AK=SQN)
2. Run authentication function (Fig. 2)
3. If MAC-A computed by A equal to MAC-A from AV, than B is authenticated.
4. Compute $CA_2 = h(ID \| P) \oplus r_1$
5. Verify if $CA_1 = CA_2$
6. Compute:
$L = h(h(PV \| K_u) \oplus RES) \oplus h(PV \| K_u) \oplus RES$
X3: { L }
$CB_1 = L \oplus$ RES
$CB_2 = D_{Kpu}(XPV \oplus X_S)$
$CB_3 = h(CB_2 \oplus \text{RES}) \oplus CB_2$
Verify if $CB_1 = CB_3$ user is approval

Figure 4.3 Strong Authentication for Internet Mobile Application

In the proposed authentication, the mobile user U and the application function S exchange three messages X1, X2, and X3 (Figure 4.3):

Step X1: U → S: $ES_{pu}(XP, ID, r_1)$

This step is identical with step L1 in [5]. In addition, this step ensures step AKA1 (Figure 2.12). Subsequently, the application function S asks the Authentication Centre (AuC) for generating authentication vector AV for the mobile user U (step AKA2). The Authentication Centre returns AV (step AKA3).

Step X2: S → U: CA_1, RAND, SQN⊕ AK, AMF, MAC-A

When S receives X1, S decrypts it and computes $G_1 = h(ID \| X_S)$ and $G_2 = D_{Kpr}(IDX \oplus X_S) = h(ID \| X_S)$. And S compares G_1 with G_2. If they are equal, S computes $PW=XP\oplus G_2$ and $CA_1=h(ID\|PW)r_1$. S does not generate a nonce r_2 [5], instead it will use RES, cut RES from AV from step AKA3 and save it. The rest of AV: CA_1, RAND, $SQN\oplus AK$, AMF, MAC-A are sent to U.

Step X3: U → S: L

U computes sequence number SQN using function f5; runs authentication functions (Figure 4.3) and:

- If MAC-A is computed by U equal to MAC-A from X2, then application function (server) is authenticated.
- Generate RES and use it for subsequent computing L.

U computes $CA_2=h(ID\|PW)\oplus r_1$ and compares CA_2 with the received CA_1. If they are equal, U computes $L=h(h(PV\|K_u)\oplus RES)\oplus h(PV\|K_u)\oplus RES$, and sends L to S. When S receives L, S computes $CB_1=L\oplus RES$, and computes $CB_2= D_{Kpu}(XPV\oplus X_S)$ and $CB_3=h(CB_2\oplus RES)\oplus CB_2$ and compares CB_3 with CB_1. If they are equal, mobile user U authenticates in application function.

The proposed solution is strong authentication with following two-factors:

- Equipment authentication - this itself is two-factor authentication (UICC + PIN). This authentication is controlled by an operator, but can be used by the application service provider [1].
- Strong password authentication. This authentication is fully under the control of the application service provider

5 Strong Authentication for Internet Application

The Strong Authentication for Internet Mobile Application proposed in chapter 4 does not generate cryptographic material for securing subsequent communication and does not support:

- Session key agreement - A session key is established between the user and the server during the authentication process, which is known only to the user and the server. Then, the session key is used to create a secure communication channel between the user and the server.
- Perfect forward secrecy - a potential attacker cannot know any information about the previously established session key, even when the long-term keys of the server and the user are disclosed.
- Forward and backward secrecy - Forward secrecy means, that even though for whatever reason some session keys are exposed, the secrecy of any previous session key will be still maintained. Backward secrecy means, that even though some previous session keys are exposed, the secrecy of any future session key will be still maintained.
- Key freshness. Neither party can guess the next shared session key.

For these reasons, we were looking for a new solution. The new proposed solution creates multifactor authentication by merging the AKA (Figure 2.12) authentication mechanism and Robust Two-factor Authentication [6].

5.1 Robust Two-factor Authentication and Key Agreement Preserving User Privacy

The Robust Two-factor Authentication and Key Agreement Preserving User Privacy [6] scheme consists of 5 phases: parameter generation, registration, authentication, password change, and data bearer token revocation. The cipher-block-chaining mode is engaged to protect against unauthorized data modification such as insertion or deletion. We mention only a phase parameter generation, registration and authentication.

5.1.1 Parameter Generation Phase

S chooses an elliptic curve E over a finite field F_p with a large prime number p. S also chooses generator point G with a large order n. Finally, S publishes the parameters (p, E, G, n).

Table 5.1 Notion of Robust Two-factor Authentication

Symbol	**Meaning**
p	*A large prime*
E	*An elliptic curve equation over Z_p*
F_p	*A finite field with notions of addition (+), subtraction (-), multiplication (x), and division*
G	*A generator point of large order*
S	*Server*
U	*User*
ID	*Login ID of U*
PW	*PW Password of U*
$E_{key}(m)$	*Encryption of message m with key*
$D_{key}(m)$	*Decryption of message m with key*
h1(); h2(); h3()	*Cryptographic hash function*
H()	*Cryptographic map-to-point (on elliptic curve) hash function, e.g.* [57]
⊕	*XOR operation*
‖	*Concatenation*

5.1.2 Registration Phase

In this phase, *U* registers with *S* by performing the following steps via a secure channel, as shown in Figure 5.1.

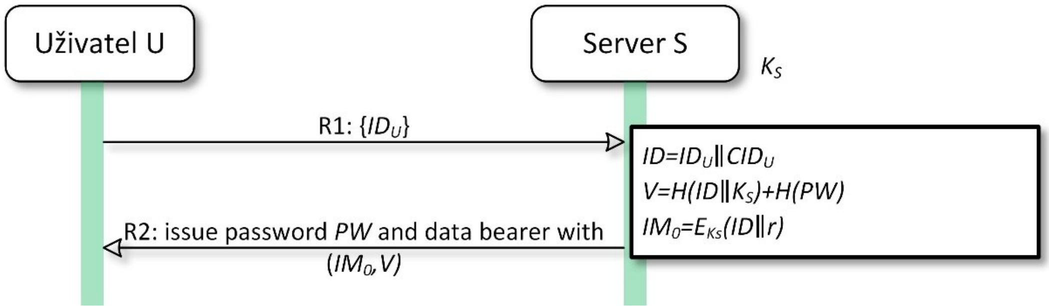

Figure 5.1 Registration Phase (Robust Two-factor Authentication)

Step R1: *U* selects the sub-identifier ID_U following the appointed format and submits it to *S*.

Step R2: After receiving ID_U, if it is valid, S generates the identifier $ID=ID_U\|CID_U$ for U. Here, CID_U is the identifier of the data bearer token for U. Then, S computes $V=H(ID\|K_S)+H(PW)$ and $IM_0=E_{K_S}(ID\|r)$, where K_S is the mast secret key, PW is the initial password selected by *S*, and *r* is a random number to provide the identity protection.

Step R3: *S* issues the password *PW* and the data bearer token to *U*, the data bearer token contains the parameters (IM_0, V).

5.1.3 Authentication Phase

In this phase, U and S authenticate each other and establish the session key KSU for the subsequent secret communication, as is shown in Figure 5.2. The steps involved are the following.

Step A1: *U* inserts his data bearer token in his equipment and inputs his password *PW*. Next, a random integer r_C from [1, n − 1] is generated and computes $G_C=r_C \times G$. Then it continues to compute $V'=V–H(PW) = H(ID\|K_S)$ and $G'_C=G_C+V'$. Finally, *U* sends $\{IM_0, G'_C\}$ to *S*.

Step A2: Upon receiving $\{IM_0,G'_C\}$, *S* decrypts the parameter IM_0 by K_S and obtains the value $ID\|r$. Then, *S* verifies whether the identifier *ID* is valid using the *ID* table maintained. If the verification is false, *S* terminates the session. Otherwise, *S* computes $V'=H(ID\|K_S)$ and recovers $G_C=G'_C-V'$. After that, *S* generates $G_S=r_S\times G$, where r_S is a random integer from [1, n−1] , and then computes $IM_1=E_{K_S}(ID\|r')$, $K_{SU}=h_1(H(ID\|K_S)(r_S\times G_C))$, $IM'_1=h(K_{SU}) \oplus IM_1$ *and* $M_S=h_2(K_{SU}\|G_C\|G_S\|IM'_1)$. *S* sends $\{M_S,G_S,IM'_1\}$ to *U*.

Step A3: Upon receiving $\{M_S,G_S,IM'_1\}$, *U*'s equipment computes the session key $K_{SU}=h_1(V'\|(r_C\times G_S))$ and further checks whether the value M_S is equal to $h_2(K_{SU}\|G_C\|G_S\|IM'_1)$. If not, the session is terminated. Otherwise, $IM_1=h(K_{SU}) \oplus IM'_1$ is computed and IM_0 is replaced by IM_1, then it computes $M_U=h_2(K_{SU}\|G_S)$ and sends it to S.

Step A4: Upon receiving $\{M_U\}$, *S* checks whether the value M_U is equal to $h_2(K_{SU}\|G_S)$. If it is, *U* and *S* successfully authenticate each other and share the session key. Otherwise, *S* terminates.

5.2 Proposed Solution

Assume that the user is registered:

- In terms of Robust Two-factor Authentication: the parameter generation phase is complete. The mobile user *U* and the application function **S** (server) exchange messages R1 and R2 (Figure 5.1).
- In terms of the AKA mechanism: A user equipped with USIM / ISIM shares the secret K with the Authentication Centre (AuC).

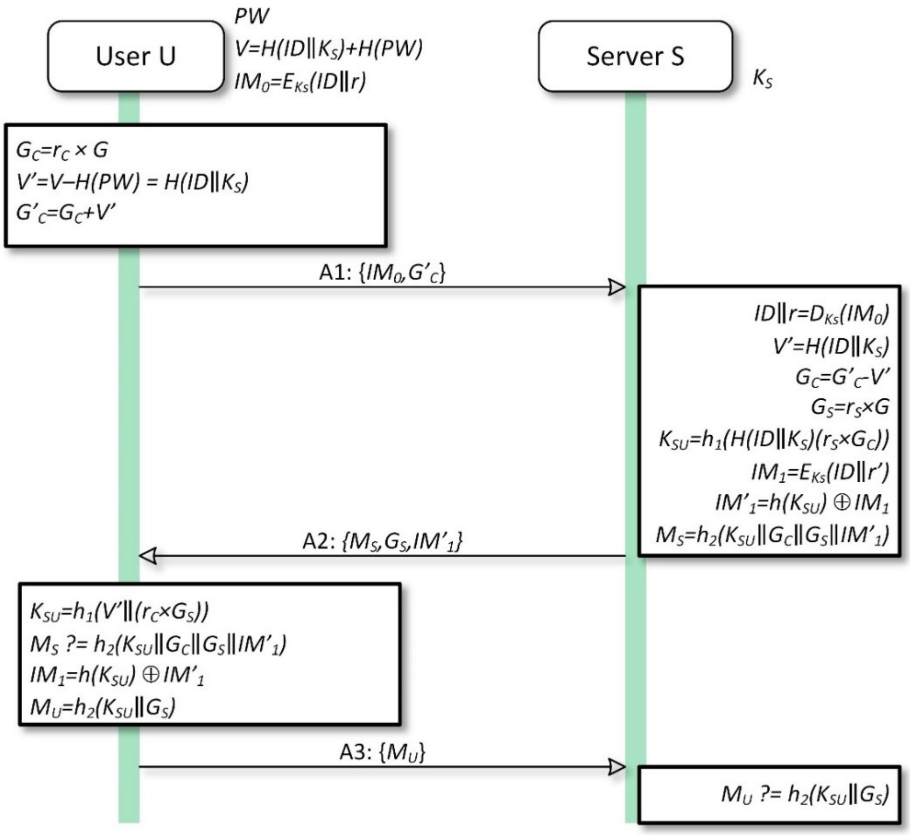

Figure 5.2 Authentication Phase (Robust Two-factor Authentication)

In the proposed solution, the mobile user *U* and application function *S* during authentication exchange three messages Y1, Y2, and Y3 (Figure 5.3):

Step Y1: *U* inserts its data bearer token with parameters IM_0 and V into its equipment and inputs its password *PW*. Next, a random integer r_C from $[1, n - 1]$ is generated and $G_C=r_C \times G$ is computed. Then, it continues to compute $V'=V–H(PW) = H(ID\|K_S)$ and $G'_C=G_C+V'$. At the end, *U* sends its identity (for AKA mechanisms) and $\{IM_0,G'_C\}$ to *S*.

Step Y2: This step consists of step A1 (Robust Two-factor Authentication) and steps AKA1, AKA2, and AKA3. Upon receiving message Y1, *S* asks AuC for AV generation for the user *U*. AuC sends AV to *S* and *S* cuts off RES from AV and stores it.

In the meantime, *S* handles message $\{IM_0,G'_C\}$, *S* decrypts the parameter IM_0 with K_S and obtains the value $ID\|r$. Then, *S* verifies whether the identifier *ID* is valid. If the verification fails, *S* terminates the session. Otherwise, *S* computes $V'=H(ID\|K_S)$ and recovers $G_C=G'_C-V'$. After that, *S* generates $G_C=G'_C-V'$, where r_S is a random integer within range [1, n − 1], and then computes

$IM_1=E_{K_S}(ID\|r')$, $K_{SU}=h_1(H(ID\|K_S)(r_S\times G_C))$, $IM'_1=h(K_{SU})\oplus IM_1$ *and* $M_S=h_2(K_{SU}\|G_C\|G_S\|IM'_1\|MAC\text{-}A)$.

S sends to U: M_S, G_S, IM'_1, $RAND$, $SQN\oplus AK$ and AMF .

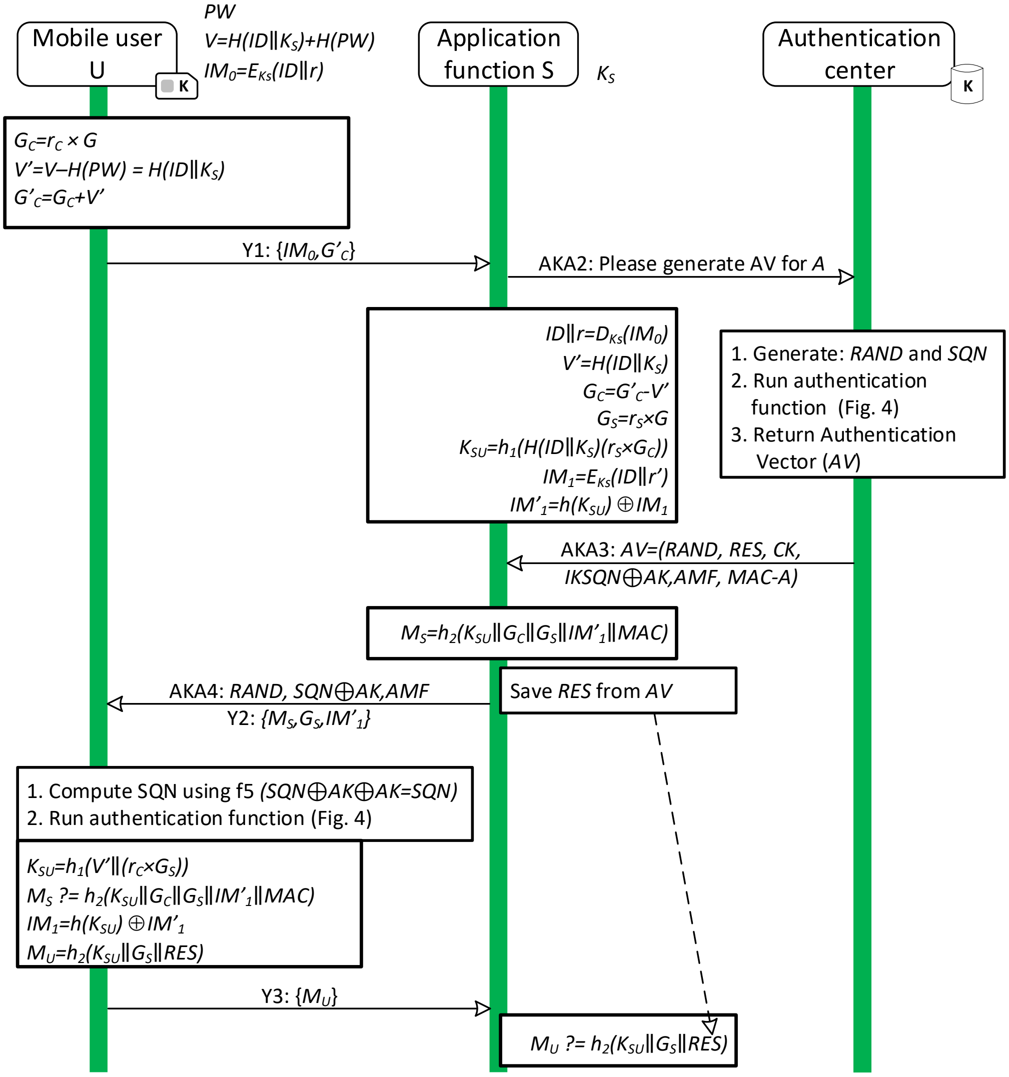
Mobile user U
K
PW
V=H(ID||K_S)+H(PW)
IM_0=E_KS(ID||r)
Application function S
K_S
Authentication center
K
G_C=r_C × G
V'=V–H(PW) = H(ID||K_S)
G'_C=G_C+V'
Y1: {IM_0,G'_C}
AKA2: Please generate AV for A
ID||r=D_KS(IM_0)
V'=H(ID||K_S)
G_C=G'_C-V'
G_S=r_S×G
K_SU=h_1(H(ID||K_S)(r_S×G_C))
IM_1=E_KS(ID||r')
IM'_1=h(K_SU) ⊕IM_1
1. Generate: RAND and SQN
2. Run authentication function (Fig. 4)
3. Return Authentication Vector (AV)
AKA3: AV=(RAND, RES, CK, IKSQN⊕AK,AMF, MAC-A)
M_S=h_2(K_SU||G_C||G_S||IM'_1||MAC)
Save RES from AV
AKA4: RAND, SQN⊕AK,AMF
Y2: {M_S,G_S,IM'_1}
1. Compute SQN using f5 (SQN⊕AK⊕AK=SQN)
2. Run authentication function (Fig. 4)
K_SU=h_1(V'||(r_C×G_S))
M_S ?= h_2(K_SU||G_C||G_S||IM'_1||MAC)
IM_1=h(K_SU) ⊕IM'_1
M_U=h_2(K_SU||G_S||RES)
Y3: {M_U}
M_U ?= h_2(K_SU||G_S||RES)

Figure 5.3 Strong Authentication for Internet Application

Step Y3: First, *U*'s equipment runs function f5 and obtains SEQ. Subsequently, it runs function f1, f2, f3 and f4 and obtains MAC, RES and the cryptographic material IK and CK.

Upon receiving $\{M_S,G_S,IM'_1\}$, U's equipment computes the session key $K_{SU}=h_1(V'\|(r_C\times G_S))$ and then checks whether the value M_S is equal to $h_2(K_{SU}\|G_C\|G_S\|IM'_1\|MAC\text{-}A)$. If it is not, it terminates the session. Otherwise, it computes $IM_1=h(K_{SU})\oplus IM'_1$ and replaces IM_0 with IM_1, computes $M_U=h_2(K_{SU}\|G_S\|RES)$ and sends $\{M_U\}$ to *S*.

Step Y4: Upon receiving $\{M_U\}$, S checks whether the value M_U is equal to $h_2(K_{SU}||G_S||RES)$. If it is, U and S successfully authenticate each other and share the session key. Otherwise, S terminates this session.

The proposed solution enables applications, running on IMS environments, using of authentication, which brings:

- Evidence that the user is authenticated in the device that he/she uses, with the particular USIM / ISIM (proof of ownership).
- Evidence that a concrete person who knows the password performed the authentication.

Such authentication can be used in banking applications or Internet access to government applications. In addition, it may be used e.g. in applications for selling tickets in public transport. It is important that in the case of such applications, the ticket cannot be duplicated to another mobile device.

The proposed method supports:

- Multi-factor security - the security of the scheme is guaranteed when either the user's password or his data bearer token or the USIM/ISIM is compromised, but not all.
- The content provider keeps control over authentication to its applications.
- Authentication is associated with the USIM/ISIM i.e. AKA mechanisms used.
- Roaming support - authentication is possible both in a home network and in a visited network.
- A password change - a user can freely update his/her password.
- A password change without communication with a server - a user can freely update his/her password without any interaction with a server.
- Mutual authentication - a user and a server are sure about the identity of each other. Both the server and the user can verify the legality of its counterpart.
- No-time synchronization - the scheme does not require additional clock synchronization mechanisms
- Anti-desynchronization - both a user and a server cannot be desynchronized against their shared secrets or values, which could result in the denial of any future access to the server.
- Data bearer revocation. In the case of data bearer token loss, invalidating the further use of the lost data bearer token should be provided to prevent an adversary from impersonating the registered user with the lost s data bearer token.

- Password protection (eavesdropping, elicitation etc.) – apart from the user, no other party may obtain any information on the user's password. More specially, the user's password will not be revealed to the server during registration, and there are no variation tables such as plain text or hashed passwords stored in the server.
- Session key agreement - a session key is established between the user and the server during the authentication process, which is known only to the user and the server. Then, the session key is used to create a secure communication channel between the user and the server.
- Perfect forward secrecy - a potential attacker cannot know any information about the previously established session key, even when the long-term keys of the server and the user are disclosed.
- Forward and backward secrecy - forward secrecy means, that even though for whatever reason some session keys are exposed, the secrecy of any previous session key will be still maintained. Backward secrecy means that even though some previous session keys are exposed, the secrecy of any future session key will still be maintained.
- Key freshness - neither party can guess the next shared session key.

6 Comparison of proposed authentication method

The proposed method Strong Authentication for Internet Application (described in chapter 5) meets all the requirements for the authentication method we are looking for, i.e. fulfils all requirements that have been specified in the aims of thesis (chapter 3). Table 6.1 compares both proposed verification methods with the AKA method itself and the Secure Hash-Based Password Authentication Protocol [5] and Robust Two-factor Authentication [6] methods mentioned in previous chapters.

Table 6.2 summarizes the computation costs between those authentication methods.

Table 6.1 Features of the described algorithms

		AKA	Secure Hash-Based Password Authentication Protocol [5]	Robust Two-factor Authentication [6]	Strong authentication for mobile application	Strong Authentication for Internet Application
1	Multi-factor security	Yes	Yes	Yes	Yes	Yes
2	Content provider maintains control over authentication to its applications	No	Yes	Yes	Yes	Yes
3	Authentication is associated with USIM/ISIM	Yes	No	No	Yes	Yes
4	Roaming support	Yes	Yes	Yes	Yes	Yes
5	Password change	n/a	Yes	Yes	Yes	Yes
6	Password change without any interaction with the server	n/a	No	Yes	No	Yes
7	Mutual authentication	Yes	Yes	Yes	Yes	Yes
8	No time synchronization	Yes	Yes	Yes	Yes	Yes
9	Anti-de-synchronization	Yes	Yes	Yes	Yes	Yes
10	Data bearer revocation	n/a	Yes	Yes	Yes	Yes
11	Password protection (eavesdropping, elicitation etc.)	n/a	Yes	Yes	Yes	Yes
12	Session key agreement	Yes	No	Yes	No	Yes

13	Perfect forward secrecy	n/a	n/a	Yes	n/a	Yes
14	Forward and backward secrecy	n/a	n/a	Yes	n/a	Yes
15	Key freshness	n/a	n/a	Yes	n/a	Yes

Table 6.2 Computation costs

	Secure Hash-Based Password Authentication [5] Protocol	Robust Two-factor Authentication [6]	Strong authentication for mobile application	Strong Authentication for Internet Application
Computation costs of the user	4h, 1A	3h, 1H, 2PM	4h, 1A, AKA	3h, 1H, 2PM, AKA
Computation costs of the server	4h, 2A	4h, 1H, 2S, 2PM	4h,2A, AKA	4h, 1H, 2S, 2PM, AKA
Communication round between the user and server	3	3	3	3

Where:

- h is defined as the time complexity of the hash computation;
- H is defined as the map-to-point hash computation;
- S is defined as the time complexity of the symmetric encryption/decryption
- PM is defined as the time complexity of the EC point multiplication;
- A is defined as the time complexity of the asymmetric (e.g. RSA) encryption/decryption.
- AKA is defined as the time complexity of the AKA algorithm

7 Multi-factor authentication modeling

Multi-factor authentication methods are usually more laborious for users and therefore less user friendly. We are proposing a model that would dynamically suggest when it is appropriate to use less demanding authentication methods and when it is necessary to use more demanding authentication methods.

Authentication is the process of verifying the identity of the subject. This process makes it possible to identify a person or to confirm the origin of the data message. This process is performed by a verifier who guarantees that the entity or origin has a declared identity (Figure 7.1). The quality of this guarantee depends on the specific authentication process. And also depends on participants of the authentication process.

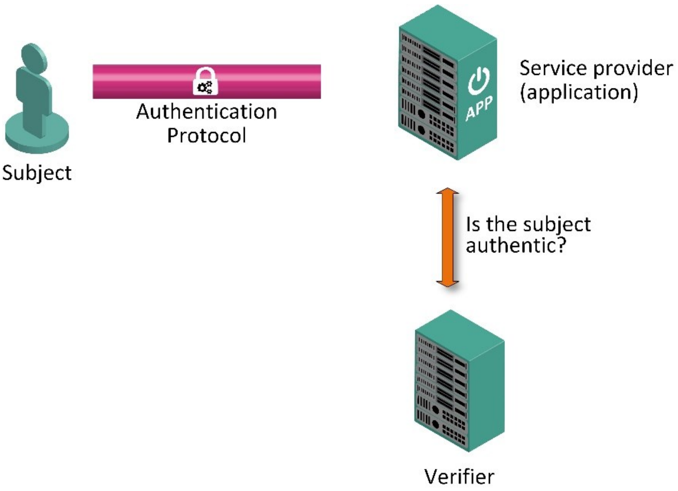

Figure 7.1 Participants in authentication process

The person is usually authenticated at the beginning of the session. The difference between session and data message authentication is in the time perspective. When a subject authenticates itself at the beginning of the session, e.g. with the help of name and password, then usually the authentication is valid for the duration of the whole session. We can then investigate if over time this method of authentication retains its quality or not. By contrast, the data message is authenticated or not.

The side effect of the authentication may be the generation of cryptographic material for securing subsequent communication.

More recently, authentication by external providers has been used as well used (e.g. Google). External authentication often uses OAuth 2.0 [58] and OpenID Connect [59] protocols. In the case of this external authentication, we will not investigate what factors it is based on. We will look at external authentication as a black box, and for the purposes of the model described below we will consider it as a separate forth authentication factor.

In the case of multifactor authentication, it is also important that used authentication factors are mutually interlinked. If authentication factors are completely independent, an attacker can attempt to break each of the factors used independently of each other.

In our model, however, we will work with the individual authentication methods. We will consider interlinked authentication as another (stronger) type of authentication. In the event that an authenticated subject during its session requires access to resources that require stronger authentication, then the subject must re-authenticate itself.

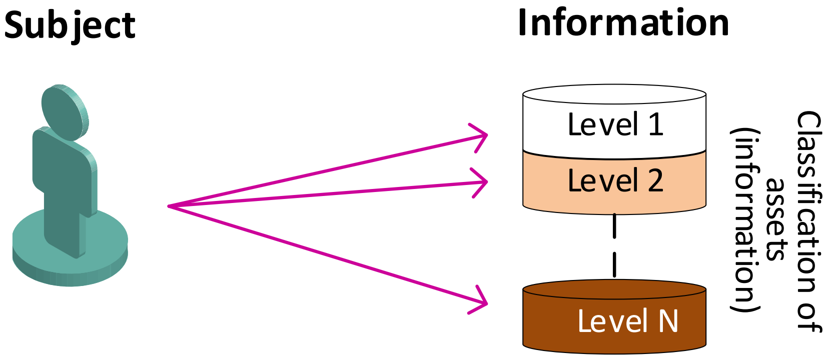

Figure 7.2 An subject wants to access information (assets) classified to level information I, where $I \in \{1, \dots N\}$

Re-authentication can be performed in two ways:

- Session termination and new authentication in new session.
- Additional authentication in the running session

If the application provider allows the user to authenticate themselves with several different authentication tools. Then we talk about omni-factor authentication [60]. The goal of our model is to choose optimal authentication tools for accessing specifically classified resources provided by the application.

7.1 Why compare authentication methods?

The subject uses authentication to gain access to assets (e.g. information assets) that will be classified I to levels 1 to N for the needs of our model (Figure 7.2). Assets can be classified, e.g. by the asset's carrying amount. However, it is more likely to be based on the risk analysis mentioned below.

The question is, what authentication method is sufficient to access the valued (classified) level information $I \in \mathbb{N}$, where $I = 1, \dots N$? The solution described below is to use the Risk Based Authentication principle. We introduce an authentication model that determines whether the authentication is sufficient for a specified level of classification based on the input parameters.

7.2 Digital Footprints

So far, we have considered authentication as it is used, i.e. when logging in to FTP or Telnet server. However, at present, logging is part of wider communication, i.e. when a user logs in to a web server. The communication for displaying the login page transmits a large amount of data and the logged-in user leaves the digital footprint (see also section 2.14). A digital footprint can be used for authentication itself or can serve as another authentication factor. Interestingly, the information extracted from the digital footprint can not only amplify but also weaken the resulting authentication.

If we want to use the digital footprint for authentication, then:

1. From a digital track, we must be able to identify users. The easiest way to do this is to save the user's identification to cookies.
2. Upon subsequent authentication, we can determine the degree of match information in the current digital footprint with information in the previous digital footprints of the same user.
3. If we find a mismatch of the current digital footprint with the previous digital tracks, then we can ask the user for additional (secondary) authentication.

The key is to be able to identify the user from the digital track. However, even when digital footprint are able to identify users with a certain probability, it can be useful in practice.

7.3 Device Fingerprinting

A special case of Digital Footprint is Device fingerprinting, which is used for mobile phones and similar devices.

Device fingerprinting means collecting information about a computing system and its communication that may lead to device identification or at least partial identification. In practice, however, device fingerprint typically detects a user communicating from a specific device, i.e. gathering information not only about the hardware and software installed on the device, but also about its user configuration and, if applicable, the behavior of a particular user.

The server can collect a wide range of client communication information based on various methods that can characterize a particular device, and therefore assume that the device is being used by a specific user. Individual methods are referred to as device fingerprinting vectors. Such fingerprinting vectors can be, e.g. information about the device software, operating system version type including its version, or information provided by the browser (e.g., cookie) or the time zone set, etc. For example [61] defines 29 fingerprinting vectors that are classified into four categories: browser provided information, inference based on device behavior, extensions and plugins and network and protocol techniques. It also states five basic types of attacks on device fingerprinting. The paper [62] uses fingerprinting vectors based on sensors motion. There are a number of such articles. Their deficiency is that they identify individual authentication factors, but they do not specify to qualify their authentication strength.

7.4 Fraud Detection Systems (FDS)

The FDS (see also section 2.14) principle is the opposite of authentication. An FDS calculates the likelihood of a successful attack against authentication. It is not a wholly new approach. Patent [49] has already applied this idea to Internet applications (see section 2.14).

We mention FDS here, because the model proposed below in this section can be useful for an information system of the FDS type.

An FDS typically introduces a client behavior model and classifies deviations from this model. Article [63] classifies different approaches in FDS (Data Mining, Artificial Intelligence, Machine Learning, Genetic Programming, Reinforcement Learning, Transformed-domain-based and Combined Criteria).

7.5 Risk based authentication

By the term risk based authentication, we mean authentication taking into account the risk of a successful attack against this authentication. Historically, patent [49] assumed that Fraud Detection Systems (FDS) methods and authentication methods may sometimes use the same principles. More recent work has already been mentioned [61] which defines 29 fingerprinting vectors that are classified into four categories.

A risk-based authentication scheme is illustrated in Figure 7.3. The communication between the subject and the application is intercepted. The duplicated communication stream is evaluated by a risk engine. The risk engine uses a knowledge database to evaluate the behavior of the subject and calculate the device fingerprint. Sometimes it does not evaluate the behavior of individual subjects, but a group of subjects that contains equivalent subjects in terms of this evaluation (e.g. domestic customers, foreign customers, corporate customers etc.). The result of the evaluation is a risk score value. |The value of the risk score depends on the decision of the verifier as to whether the subject is authentic or not, i.e. whether the authentication is successful or not.

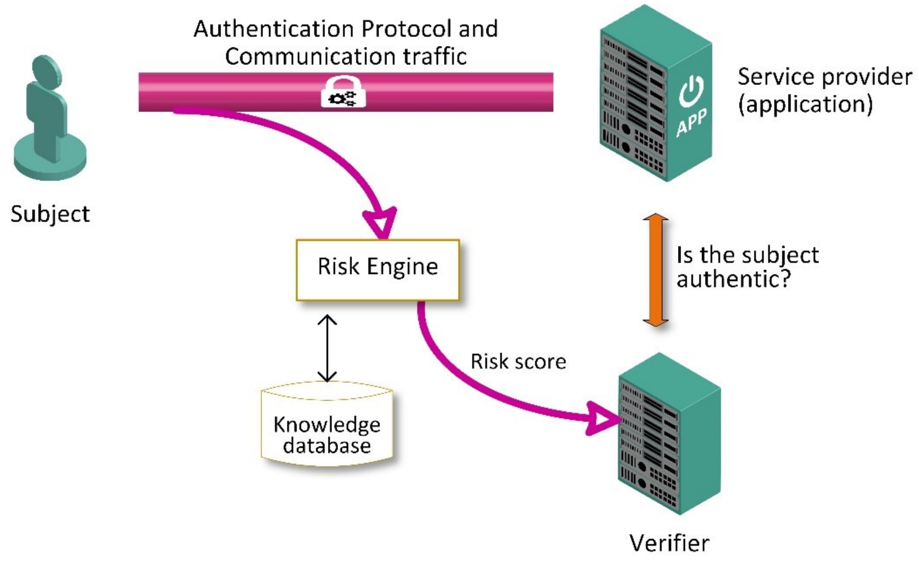
Authentication Protocol and Communication traffic
Subject
Service provider (application)
APP
Risk Engine
Is the subject authentic?
Risk score
Knowledge database
Verifier

Figure 7.3 Risk Based Authentication

The Patent [49] of the authentication assessment essentially carries out a decision based on something that that is reminiscent of a decision tree. The article [61], in turn, makes the assessment on the basis of a penalty. We will make the decision on the basis of a standardized risk analysis built on the standard [64], because it includes a range of options, including the above.

7.6 Risk analysis

Risk analysis is today a very common technique. This can be done, i.e. based on the standard [64]. This standard:

- Performs the identification of assets. Which can lay down the classification of information.
- Evaluates the risks. The resulting risk value is the product of the threat assessment, vulnerabilities assessment, and impacts of assessed risks.

Table 7.1 Notion of Multi-Factor Authentication Modelling

Symbol	***Meaning***
$\{\,\}$	*Braces denote sequence*
A	*Asset*
F^K_{know}, F^K_{own} F^K_{inh}, F^K_{ext}	*Authentication factors of the authentication method K based on knowledge, ownership, inheritance, and external factors*
i	*Index of authentication factors features*
I	*Valued level of Asset*
j	*Index of authentication method K_j*
K	*Authentication method*
m	*Number of authentication methods supported by application (in institution)*
n^K_{know}, n^K_{own}, n^K_{inh}, n^K_{ext}	*Number of features of the authentication method K based on knowledge, ownership, inheritance, and external factors*
N	*Maximal value of valued level of asset*
$\mathbb{N}$	*Natural numbers*
$p^K_{know_i}$, $p^K_{own_i}$, $p^K_{inh_i}$	*1 – feature is relevant for authentication method K* *0 – feature is irrelevant for authentication method K*
q^K_{know}, q^K_{own}, q^K_{inh}, q^K_{ext}	*Qualities of Authentication factors of the authentication method K based on knowledge, ownership, inheritance, and external factors*
r_{know_i}, r_{own_i}, r_{poss_i}	*Risk score of authentication factors features i (output from risk analysis)*
r^K_{ext}	*Risk score of the external authentication method K (output from risk analysis)*
$risk_{max}$	*Maximum risk value used in risk analysis*
W^K_{know} ,W^K_{own}, W^K_{inh}, W^K_{ext}	*Weights of authentication factors of the authentication method K*
$w^K_{know_i}$, $w^K_{ext_i}$ $w^K_{poss_i}$, $w^K_{inh_i}$,	*Risk based weights of feature i of authentication factors of the authentication method K*

For evaluation of threats assessment, vulnerabilities assessment, and impacts of assessed risks we will use scale from 1 to $risk_{max}$. For the purposes of this thesis, we will use the scale 1 to 4 (Low, Medium, High or Critical) as an example. The resulting risk is then the product threats, vulnerabilities and impact of risks. In the case of scale 1 to 4 the maximal risk value can be up to 64 (=4.4.4). The range of 64 values is very detailed (to fine) for the high-level decision. Therefore, for the high-level decision, the interval from 1 to 64 will again be divided into 4 parts that again use scale of 1 to 4 corresponded to the risks Low, Medium, High, or Critical.

7.7 Quality of authentication factors

Let us have the asset A valued by $I \in \mathbb{N}$ at levels $1, 2, \ldots, N$, where N is the maximal valued level of any asset of application (organization). Now we have a fixed value for the authentication method K. The question is whether the authentication method K is sufficient to access the asset A of the valued level I of asset, where $1 \leq I \leq N$.

In order to answer this question, we must somehow express the quality (strength) of the authentication method. We express the quality (strength) of the authentication method using the risk analysis mentioned in the previous section.

The authentication method K is generally a multifactor authentication method, which may consist of authentication factors: F_{know}^{K}, F_{own}^{K} F_{inh}^{K} and F_{ext}^{K}. Where:

- Authentication factor F_{know}^{K} is based on knowledge. Quality (strength) of this authentication factor we evaluate by value $q_{know}^{K} \in < 0, 1 >$.
- Authentication factor F_{own}^{K} is based on ownership. Quality (strength) of this authentication factor we evaluate by value $q_{own}^{K} \in < 0, 1 >$.
- Authentication factor F_{inh}^{K} is based on inherence. Quality (strength) of this authentication factor we evaluate by value $q_{inh}^{K} \in < 0, 1 >$.
- Other authentication factor F_{ext}^{K}, e.g. external authentication. Quality (strength) of this authentication factor we evaluate by value:
  $$q_{ext}^{K} \in < 0, 1 >.$$

Overall, we evaluate authentication method K by value q^{K}:

$$q^{K} = W_{know}^{K} q_{know}^{K} + W_{own}^{K} q_{own}^{K} + W_{inh}^{K} q_{inh}^{K} + W_{ext}^{K} q_{ext}^{K}$$

Wight W_{know}^{K}, W_{own}^{K}, W_{inh}^{K} and W_{ext}^{K} we choose zero when the authentication factor is not supported and non-zero for the categories of technology, algorithms and parameters, which ensure the increasing quality of authentication. Use of these weights allows us to take into account the dependency or independency of various authentication factors. Weights W_{know}^{K}, W_{own}^{K}, W_{inh}^{K}, and W_{ext}^{K} will be used for an attack simulation. We will mention this weights W_{know}^{K}, W_{own}^{K}, W_{inh}^{K}, and W_{ext}^{K} in section 7.11.

We will now look at how to determine the values: q_{know}^{K}, q_{own}^{K}, q_{inh}^{K}, and q_{ext}^{K}.

7.7.1 Knowledge Risk Based Authentication

We will now think about knowledge-based authentication in general. This authentication is based on the knowledge under which we can imagine a password. However, we will now abstain from a specific authentication method. We determine the risks of this authentication factor based on a risk analysis performed for a specific application (specific institution).

Table 7.2 An example of risk analysis of knowledge based authentication

i	**Security risks** $know_i$	**Risk score** r_{know_i} **(1-4)**	**Weight** w_{know_i}
1	The knowledge has an information entropy lower than the specified limit.	4	0.08
2	The validity of knowledge is not time limited	3	0.06
3	Knowledge can be used multiple times (not one-time)	4	0.08
4	The number of attempts to guess knowledge is not limited	4	0.08
5	Change of knowledge is not supported	4	0.08
6	Reset of knowledge is not supported	4	0.08
7	Knowledge eavesdropping is possible	4	0.08
8	Knowledge guessing is possible	4	0.08
9	Knowledge elicitation is possible	3	0.06
10	Authentication requires time synchronization	1	0.00
11	Authenticated subject anonymity is not guaranteed	2	0.03
12	Authenticated subject traceability is possible	2	0.03

Risks of authentication factor are the features (characteristics) of specific authentication factors. We assume n_{know} risks $know_i$ $(i = 1, 2, \dots\ n_{know}\ )$ of the knowledge authentication factor. An example is in Table 7.2[3].

In Table 7.2 there is a risk assessment of the individual features of the authentication factor (risks) on a scale 1 through 4 (column Risk score r_{know_i}). The risk assessment means that we ask what the risk is for a specific application. The higher the risk,

[3] Other risks (not included in the table) can be e.g. in the case of generating cryptographic material for securing subsequent communication:

- Key freshness - neither party can predetermine the shared session key being established.
- Perfect forward secrecy – an attacker cannot know any information about the previously established session key even when the long-term keys of the server and the user are disclosed.

the higher the risk score. The value of risk score is based on a subjective assessment of a specific situation as is common when performing risk analysis [64].

Table 7.3 Examples of values q^{K}_{know} *for a standard password and a one-time password*

i	**Security risks**	**Risk score**	**Weight**	**Password**		**HW based one-time password**	
	$know_i$	r_{know_i} (1-4)	w_{know_i}	$p^{Pass}_{know_i}$	$w_{know_i} p^{Pass}_{know_i}$	$p^{HW}_{know_i}$	$w_{know_i} p^{HW}_{know_i}$
1	The password has an information entropy lower than the specified limit.	4	0.08	0	0.00	1	0.08
2	The validity of the password is not time limited	3	0.06	0	0.00	0	0.00
3	The password can be used multiple times (not one-time)	4	0.08	1	0.08	0	0.00
4	The number of attempts to guess the password is not limited	4	0.08	0	0.00	0	0.00
5	A change of password is not supported	4	0.08	0	0.00	1	0.08
6	A reset of the password is not supported	4	0.08	0	0.00	1	0.08
7	Password eavesdropping is possible	4	0.08	1	0.08	1	0.08
8	Password guessing is possible	4	0.08	1	0.08	1	0.08
9	Password elicitation is possible	3	0.06	1	0.06	1	0.06
10	Authentication requires time synchronization	1	0.00	0	0.00	0	0.00
11	Authenticated subject anonymity is not guaranteed	2	0.03	1	0.03	1	0.03
12	Authenticated subject traceability is possible	2	0.03	1	0.03	1	0.03
Summa			0.75		0.36		0.53
				q^{Pass}_{know}	0.64	q^{HW}_{know}	0.47

Based on the risk score, we determine n_{know} risk weights w_{know_i} $(i = 1,2, \dots n_{know})$ of the individual risks (risk weight expresses the degree of risk, which is the opposite of the strength of the authentication factor). The weight w_{know_i} of specific risk $know_i$ we will define as:

$$w_{know_i} = \frac{r_{know_i} - 1}{(risk_{max} - 1)\, n_{know}}$$

In our case we have ($risk_{max} = 4, n_{know} = 12$):

$$w_{know_i} = \frac{r_{know_i} - 1}{36}$$

This will ensure that the sum of all weight for a particular authentication factor can be at most 1. This is because if we add additional risks to the model, the model does not give diametrically different results. The maximum risk of 1 corresponds to the situation as if the authentication factor was not supported at all.

We will now deal with a specific authentication factor F_{know}^{K}, i.e. we will want the authentication factor F_{know}^{K} to be valued by q_{know}^{K} expressing the strength of the authentication factor. We determine the value q_{know}^{K} based on the risks of the risk-based factor of a concrete authentication method.

Generally, for the authentication method K, not all identified risks are relevant. Therefore, we define the variable $p_{know_i}^{K}$, which for each risk is $know_{\boldsymbol{i}}$:

- will get value 1 if the risk r_{know_i} for the authentication factor F_{know}^{K} is relevant.
- will get value 0 if the risk r_{know_i} for the authentication factor F_{know}^{K} isn't relevant.

Value $q_{know}^{K}$of authentication factor F_{know}^{K} will be defined as:

$$q_{know}^{K} = 1 - \sum_{i=1}^{n_{know}} w_{know_i} p_{know_i}^{K}$$

The maximum value of q_{know}^{K} is one. From value one we subtract the sum of weight of specific risks, i.e. specific risks reduce assessment of the strength of the concrete authentication factor.

The Table 7.3 shows two simple examples of setting value q_{know}^{K} for a password ($K = Pass$) and one-time password ($K = HW$) generated by a hardware token (only for knowledge base factor - does not include a possession-based factor).

Table 7.4 An example of risk analysis of possession based authentication ($1 \leq i \leq n_{poss}^{K}$)

i	**Security risks** $poss_i$	**Risk score** r_{poss_i} **(1-4)**	**Weight** w_{poss_i}	**HW based one-time password**	
				$p_{poss_i}^{HW}$	$w_{poss_i}p_{poss_i}^{HW}$
1	Cryptographic material is not stored on data bearer in secured environment	4	0.17	0	0.00
2	Access to cryptographic material without any authentication	4	0.17	0	0.00
3	Whole secure environment is not physically protected	4	0.17	1	0.17
4	Cryptographic material is exportable	3	0.11	1	0.11
5	Cryptographic material does not physically protected against unauthorized access	3	0.11	0	0.00
6	Data bearer revocation not supported	2	0.06	1	0.06
Suma					0.33
				$q_{poss}^{K} =$	0.67

7.7.2 Possession Risk Based Authentication

Possession Risk Based Authentication we deal like the Risk Knowledge Based Authentication, i.e. we will use a risk analysis to assess the risks of the authentication factor F_{own}^{K}. The possession based authentication factor is often based on the possession of an environment of cryptographic material including a data bearer of cryptographic material (e.g. smartcard, hardware security module etc.).

For the category of possession authentication we similarly define the n_{poss} of security risk r_{poss_i}.

Again we perform a risk assessment of individual risks, e.g. on a scale of 1 through 4 (otherwise may be different) to determine the risk score r_{poss_i} for specific risk $poss_i$ asking what the risk is for a specific application (institution). The higher the risk, the higher the risk score. The value of risk score is based on a subjective assessment of a specific situation as is common when performing risk analysis [64].

Similarly to knowledge based authentication, we will establish risk weights w_{poss_i}:

$$w_{poss_i} = \frac{r_{poss_i} - 1}{(risk_{max} - 1)\, n_{poss}}$$

Next, similarly to what is seen in knowledge based authentication we establish variable $p^K_{poss_i}$ of authentication factor F^K_{poss}.

We will be define q^K_{poss} as:

$$q^K_{poss} = 1 - \sum_{i=1}^{n_{poss}} w_{poss_i} p^K_{poss_i}$$

Table 7.4 contains an example of the risk analysis of possession based authentication for hardware based one-time password calculator.

7.7.3 Inherence Risk Based Authentication

The biometric characteristics of the person are traditionally included in this category. In the case of biometric authentication the coefficient q^K_{inh} can be determined, e.g. as the percentage of match of actually captured biometric pattern with the saved pattern in the database. Such agreement may be e.g. 0.93 (i.e., 93%).

However, the use of biometric characteristics of persons has many disadvantages. Biometric features cannot be revoked, so have many common features with traditional passwords. In addition, biometric authentication brings complications with the protection of personal data.

Next, we will deal with inheritance authentication based on device fingerprinting. Inheritance risk based authentication is dealt with in the same way as risk knowledge based authentication, i.e. we will use risk analysis to assess the risks of the authentication factor F^K_{inh}. For the category of inheritance authentication we similarly define a set of security risks, e.g. most device fingerprint vectors from [61] are listed in Table 7.5.

Again we perform the risk assessment of individual risks, e.g. on a scale 1 through 4 (through this scale may be different) to determine the risk score r_{inh_i} for the specific risk inh_i $(i = 1, \dots, n_{inh})$ we ask what the risk is for a specific application. The value of risk score is based on a subjective assessment specific situation as is common when performing risk analysis.

Table 7.5 An example of risk analysis of inheritance based authentication $(1 \leq i \ \leq \ n^K_{inh})$

i	**Security risks** inh_i **(device fingerprint vector doesn't match)**	**Risk score** r_{inh_i} (1-4)	**Weight** w_{inh_i}	**Device fingerprint**	
				$p^{HW}_{inh_i}$	$w_{inh_i}p^K_{inh_i}\mathrm{w}_{\mathrm{inh_i}}\mathrm{p}^{\mathrm{HW}}_{\mathrm{inh_i}}$
1	Major software and hardware details	3	0.04	1	0.07
2	System time and clock drift	4	0.07	1	0.07
3	Battery information	4	0.07	1	0.07
4	Evercookies	4	0.07	1	0.07
5	Password autofill	4	0.07	1	0.02
6	Hardware sensors	2	0.02	1	0.02
7	CSS feature detection	2	0.02	0	0.00
8	JavaScript standards conformance	2	0.02	0	0.00
9	URL scheme handlers	2	0.02	0	0.00
10	Video RAM detection	3	0.04	0	0.00
11	Browser plugin fingerprinting	2	0.02	0	0.00
12	IP address	2	0.02	0	0.00
13	Geolocation	2	0.02	0	0.00
14	Counting hosts behind NAT	2	0.02	1	0.00
15	Transaction information is suspicious	1	0.00	1	0.07
				Suma	0.31
				$q^K_{inh} =$	0.69

Similarly to knowledge based authentication, we will establish risk weights w_{inh_i}:

$$w_{inh_i} = \frac{r_{inh_i} - 1}{(risk_{max} - 1)\, n_{inh}}$$

Next we establish variable $p^K_{poss_i}$ and value $q^K_{poss}$of the authentication factor F^K_{poss} will be define as:

$$q^K_{inh} = 1 - \sum_{i=1}^{n_{inh}} w_{inh_i} p^K_{inh_i}$$

Table 7.5 contains an example of risk analysis of inheritance based authentication for device fingerprint.

7.7.4 External Risk Based Authentication

External authentication is provided by various providers. A concrete subject will use external authentication from one provider at most. We perform risk assessment of individual external authentication providers, e.g. on a scale 1 through 4 (practically 1 to 3 because 4 is critical and it is not acceptable) to determine the risk score r^K_{ext} of the specific provider, specific authentication method K.

Table 7.6 External Risk-based authentication method (fictional example)

K **(authentication of the below listed providers will be risk assessed)**	**Risk score** $\boldsymbol{r^K_{ext}}$ (1-4)	**Weight** $\boldsymbol{w^K_{ext}}$ $\mathrm{w}^{\mathrm{K}}_{\mathrm{ext_i}}$	$\boldsymbol{q^K_{ext}}$
Google	2	0.33	0.67
Facebook	1	0.00	1.00
Provider 3	2	0.33	0.67
Provider 4	3	0.67	0.33

Since the subject uses at most one authentication provider at a time, then we define the risk weight $\boldsymbol{w^K_{ext}}$:

$$\boldsymbol{w^K_{ext}} = \frac{\boldsymbol{r^K_{ext}} - 1}{risk_{max} - 1}$$

In this case, we do not need to use the variables p. Value q^K_{ext} of authentication factor F^K_{ext} will be defined as:

$$q^K_{ext} = 1 - \boldsymbol{w^K_{ext}}$$

Table 7.6 contains a fictional example of evaluating external providers.

7.8 Omni-factor authentication modelling

In Omni-factor authentication [60] we assume that a user from a set of authentication methods has chosen method K. The resulting quality q^K is the weighted sum of individual categories:

$$q^K = W^K_{know} q^K_{know} + W^K_{own} q^K_{own} + W^K_{inh} q^K_{inh} + W^K_{ext} q^K_{ext}$$

Weight W^K_{know} , W^K_{own}, W^K_{inh}, and W^K_{ext} we choose zero when the category is not relevant and non-zero for the categories of technology, algorithms and parameters, which ensures an increasing quality of authentication.

7.9 Multi-factor authentication modelling

An application (institution) using m authentication method K_j, where $1 \le j \le m$. Individual authentication factors can be combined, but q^{K_j} can not exceed the sum of the highest possible values of each authentication factor. This sum we denote S as:

$$\max_{1\le j\le m} q^{K_j} \le \max_{1\le j\le m} q_{know}^{K_j} + \max_{1\le j\le m} q_{own}^{K_j} + \max_{1\le j\le m} q_{inh}^{K_j} + \max_{1\le j\le m} q_{ext}^{K_j} = S$$

Asset (information) A is valued (classified) by I at levels 1 through N (Figure 7.4). The question is whether the concrete authentication method K is sufficient to access the asset A.

We define non-decreasing function f from interval $< 0, S >$ to sequence $\{1, 2, \dots N\}$. Function f assigns a concrete authentication method K represented by the quality $q^K \in < 0, S >$ the valued (classified) level I, where $I \in \{1, 2, \dots N\}$.

It can be said about authentication method K that is sufficient for valued (classified) level I and all lower valued (classified) levels, if:

$$f(q^K) \ge I.$$

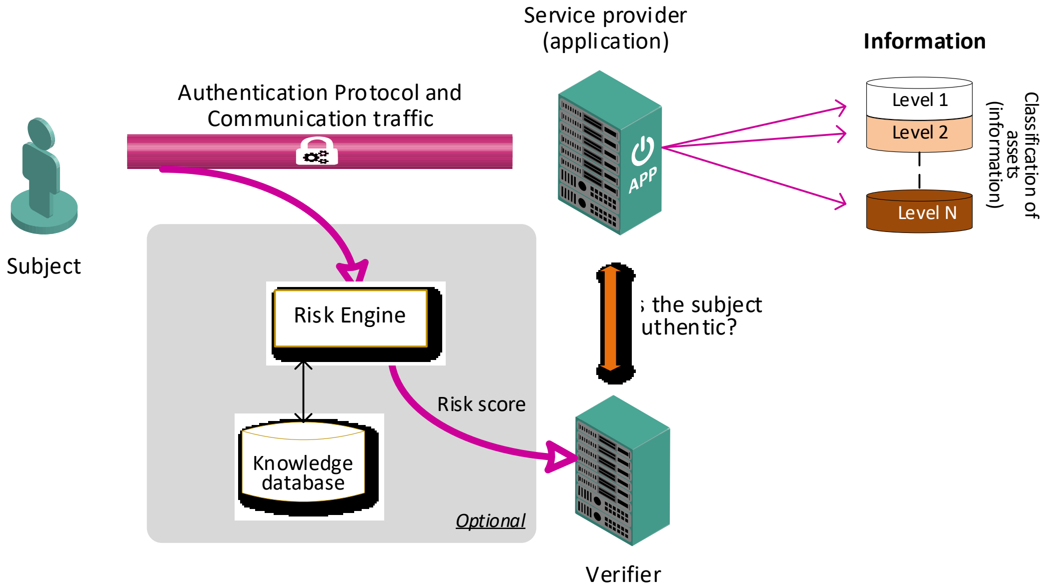

Figure 7.4 The subject requests access to the information of a particular classification N

7.10 Multi-factor authentication model

In our model, we divide interval $< 0, S >$ in N parts, where the number of parts corresponds to classified levels). Intervals may not be the same length, but for simplicity, we will consider equally long intervals (it depends on the assessment of assets [64]).

In previous described examples we have $S = 2.99$ (its sum of blue fields on Table 7.7). If in this case we will use eight valued (classification) levels (and intervals of the same length), then for each classification level we will need authentication methods of least quality, as shown in Table 7.8.

Table 7.7 Simulation: data used from the above examples (blue fields are max of authentication factor – max of table row)

	Password	HW based one-time password	Device fingerprint	Facebook	Password + Device fingerprint	Password + Device fingerprint + Facebook
q_{know} =	0.64	0.47			0.64	0.64
q_{poss} =		0.67				
q_{inh} =			0.69		0.69	0.69
q_{ext} =				1.00		1.00
q =	0.64	1.14	0.69	1.00	1.33	2.33

Table 7.8Clasification levels

$q^K \geq$	Classification level
2.62	8
2.25	7
1.87	6
1.50	5
1.12	4
0.75	3
0.37	2
0.00	1

7.11 Experiments

We performed the experiment using a simulation. The application provider (the institution) supplies the subjects with the following authentication means:

- Password
- Hardware based one-time password
- Device fingerprint
- External authentication from Facebook
- Combination of: Password + Device fingerprint
- Combination of: Password + Device fingerprint + External authentication from Facebook

We can imagine electronic banking as an experimental application. Then the individual valued (classification) levels can, e.g. correspond to:

- 1 for public information.
- 2 for basic information provided for authenticated users.
- 3 for accounts balance.
- 4 for a limited payment order.
- 5 for an unlimited payment order.
- 6 for operations on the finance market.
- 7 for worldwide interbank financial operations.
- 8 for other operations.

Table 7.9 Experiment I (no attack)

				Password	HW based one-time password	Device fingerprint	Facebook	Password + Device fingerprint	Password + Device fingerprint + Facebook
$W_{know}^{K} q_{know}^{K} =$				0.64	0.47			0.64	0.64
$W_{own}^{K} q_{own}^{K} =$					0.67				
$W_{inh}^{K} q_{inh}^{K} =$						0.69		0.69	0.69
$W_{ext}^{K} q_{ext}^{K} =$							1.00		1.00
$q =$				0.64	1.14	0.69	1.00	1.33	2.33
Classification level	8	>	2.62	No	No	No	No	No	No
	7	>	2.25	No	No	No	No	No	Yes
	6	>	1.87	No	No	No	No	No	Yes
	5	>	1.50	No	No	No	No	No	Yes
	4	>	1.12	No	Yes	No	No	Yes	Yes
	3	>	0.75	No	Yes	No	Yes	Yes	Yes
	2	>	0.37	Yes	Yes	Yes	Yes	Yes	Yes
	1	>	0.00	Yes	Yes	Yes	Yes	Yes	Yes

For our experiment we will use the classification of individual resources from the above examples. We will use weight W_{know}^{K} , W_{own}^{K} , W_{inh}^{K} , and W_{ext}^{K} for subsequent modeling.

The question is by what means the subject accesses which information (Figure 7.4).

7.11.1 Experiment I

In this case, experimental application is under standard security conditions, when no attack campaign is indicated.

We do not consider any attack on our application, therefore for the weights W^K_{know} , W^K_{own} , W^K_{inh} , and W^K_{ext} we choose zero when the authentication factor is not supported and one if it is supported. In Table 7.9 we can see the result. "Yes" means that for a given level of classification specific authentication is sufficient. "No" means that it is insufficient.

Table 7.10 Experiment II: First step in campaign attacking knowledge base authentication.

				Password	HW based one-time password	Device fingerprint	Facebook	Password + Device fingerprint	Password + Device fingerprint + Facebook
$W^K_{know}q^K_{know} =$									
$W^K_{own}q^K_{own} =$					0.67				
$W^K_{inh}q^K_{inh} =$						0.69		0.69	0.69
$W^K_{ext}q^K_{ext} =$							1.00		1.00
$q =$				0.00	0.67	0.69	1.00	0.69	1.69
Classification level	8	>	2.62	No	No	No	No	No	No
	7	>	2.25	No	No	No	No	No	No
	6	>	1.87	No	No	No	No	No	No
	5	>	1.50	No	No	No	No	No	Yes
	4	>	1.12	No	No	No	No	No	Yes
	3	>	0.75	No	No	No	Yes	No	Yes
	2	>	0.37	No	Yes	Yes	Yes	Yes	Yes
	1	>	0.00	No	Yes	Yes	Yes	Yes	Yes

7.11.2 Experiment II

In this experiment attacks are conducted in campaigns. The application provider, through the exchange of information through the Computer Security Incident Response Team (CSIRT) [65], learns that the current campaign is attacking knowledge base authentication (e.g. password fishing). Therefore in the first step, for all means, K sets:

$$W_{know}^{K} = 0.$$

We can see the result in Table 7.10.

7.11.3 Experiment III

In this experiment, the application provider will find out exactly what kind of attack it is going to face. It was found that the attack was a tapping of passwords. A risk analysis was carried out. While using long-term passwords is risky, the institution is willing to accept a 30% risk, i.e. W_{know}^{K} will be changed to 0.30 for password based algorithms. (It should be noted that there are also so-called strong password-based authentication methods (mentioned in [66] [67]) that will be immune to the actual attack, and the values may come back as they were before the attack.)

After decreasing W_{know}^{K} from 1 to 0.30 we get the values listed in Table 7.11.

Table 7.11 $W_{know}^{Password}$ decreased to 0.30

				Password	HW based one-time password	Device fingerprint	Facebook	Password + Device fingerprint	Password + Device fingerprint + Facebook
$W_{know}^{K} q_{know}^{K} =$				0.30	0.30			0.30	0.30
$W_{own}^{K} q_{own}^{K} =$					0.67				
$W_{inh}^{K} q_{inh}^{K} =$						0.69		0.69	0.69
$W_{ext}^{K} q_{ext}^{K} =$							1.00		1.00
$q =$				0.19	1.14	0.69	1.00	0.88	1.88
Classification level	8	>	2.62	No	No	No	No	No	No
	7	>	2.25	No	No	No	No	No	No
	6	>	1.87	No	No	No	No	No	Yes
	5	>	1.50	No	No	No	No	No	Yes
	4	>	1.12	No	Yes	No	No	No	Yes
	3	>	0.75	No	Yes	No	Yes	Yes	Yes
	2	>	0.37	No	Yes	Yes	Yes	Yes	Yes
	1	>	0.00	Yes	Yes	Yes	Yes	Yes	Yes

7.12 Use of the proposed model

Previous experiments have shown that the proposed model probably faithfully models a real authentication that can be used, in Fraud Detection Systems (FDS) as is mentioned in section 7.4. An FDS is able to detect an attack campaign. Based on this detection, measures must be taken quickly to minimize losses to this campaign. The basic measure is to increase the demands on authentication in a targeted manner. The decisive factor is the speed of the reaction.

Using the above-mentioned model, it is possible to react automatically, i.e. without delay.

At present, the most common incident detected by the FDS is that the user buys a new computer/mobile. Alternatively, that he/she travels on vacation in a distant country and suddenly his approach seems suspicious. At this point, the user is usually contacted to verify if an attack is occurring or not. This entails considerable cost. Using this model, applications can automatically request stronger authentication from the user. This reduces user service costs.

8 The Cyber Security Game

The Multi-factor Authentication Model we presented in [68]. Immediately afterwards, we were contacted by colleagues who explained to us that authentication is a security tool to protect assets from potential attackers. They warned us that the attacker's perspective was somewhat different. Based on this discussion, we came to the conclusion that it would be appropriate to create a model that would model the situation from the perspective of the attacker as well as the defender.

One of the questions related to cybercrime is to understand whether cyber-attacks focus more on low-value assets or high valued assets such as those mentioned above. One may expect that low valued assets are less well protected which may attract more attacks.

Game theory has been used to describe and model cyber-attacks (see e.g. [69], [70]). These games capture the basic features of resource allocation decisions to prevent data loss and defend an organization's strategic assets because developing efficient defending and attacking strategies is costly for the defender as well as for the attacker. In this chapter, we develop a game theoretic model that we call the Cyber Security Game.

Using methods from the chapters above we designed a game that meets our requirements, adding an evolutionary dynamics perspective. We develop a non-cooperative, two-player two-strategy game between a defender and an attacker that we call the Cyber Security Game. We assume that the goal of the defender is to defend an asset, X, that can be attacked by an attacker. The defender has two strategies:

1. *Defend:* Invest in defense amount c_1. In this case, the asset X resists the attacker.
2. *Not Defend:* Do not invest anything in defense. In this case, an attack will cause a loss v_1 ($\geq c_1$) to the defender, i.e. the defender's payoff will be $-v_1$.

The attacker has two strategies:

1. *Attack*: Invest in an attack amount c_2. In this case, if the defender does not defend, the attacker wins v_2 (we assume that $v_2 \geq c_2$). However, if the defender defends, the attacker wins nothing.
2. *Not Attack*: If the attacker does not attack his payoff will be 0 independent of the defender's strategy.

These payoffs are written in the form of a payoff bimatrix:

Attacker (A)

		Attack	*Not Attack*
Defender (D)	*Defend*	$-c_1, -c_2$	$-c_1, 0$
	Not Defend	$-v_1, v_2-c_2$	$0, 0$

Matrix games are solved using the concept of the Nash equilibrium (hereinafter we use the abbreviation NE) [71]. The Nash equilibrium is a strategy (pure or mixed) where none of the players can improve their situation by unilaterally changing their chosen strategy.

It is easy to see that the Cyber Security Game given by payoff bimatrix has no NE in pure strategies. Since every game in normal form has a NE [71] there must exist a NE in mixed strategies in which the attacker plays Attack with probability q and the defender plays Defend with probability p as shown in the following bimatrix:

A

			Attack	*Not attack*
		Probability	*q*	*1–q*
D	*Defend*	*p*	$-c_1, -c_2$	$-c_1, 0$
	Not Defend	*1–p*	$-v_1, v_2-c_2$	$0, 0$

To determine the NE, we write payoff function W_D for the defender in his two pure strategies, i.e.,

$$W_D\ (Defend) = q\ (-c_1) + (1-q)\ (-c_1) = \ -c_1$$

$$W_D\ (Not\ defend) = q\ (-v_1) + (1-q)\ (0) = \ -qv_1.$$

Similarly, payoff function W_A for the attacker in his two pure strategies is

$$W_A\ (Attack) = \ -pc_2 + (1-p)(v_2 - c_2)\ = \ v_2\ (1-p) - \ c_2$$

$$W_A(Not\ attack) = 0.$$

A mixed strategy (p, q) $(0 < p < 1, 0 < q < 1)$ is a NE provided it satisfies:

$$W_A\,(Attack) = W_A\,(Not\ Attack)$$

$$W_D\,(Defend) = W_D\,(Not\ Defend)$$

i.e. NE is:

$$p = 1 - \frac{c_2}{v_2}, \quad q = \frac{c_1}{v_1}.$$

We observe that in NE

- As the reward of the attacker (v_2) when it Attacks increases, the probability (p) that the defender defends his asset increases.
- As the value of the defender asset (v_1) increases, the probability that the attacker will Attack decreases.

Example: For parameters $c_1 = c_2 = 10$; $v_1 = v_2 = 100$ *the payoff bimatrix of the Cyber Security Game is:*

		A	
		Attack	*Not attack*
D	*Defend*	−10,−10	−10,0
	Not Defend	−100,+90	0,0

The game in this example has no Nash equilibrium in pure strategies. Indeed, there is no strategy such that the payoff to the defender is the highest in the corresponding column of the payoff matrix and the payoff to the attacker is highest in the corresponding row. The Nash equilibrium is:

$$p = \frac{9}{10}, \quad q = \frac{1}{10}.$$

8.1 Replicator Dynamics

Replicator dynamics are used to express the evolutionary dynamics of an entity called a replicator, which has the means of making more or less accurate copies of itself. The replicator equations models replication in time [72]. We use replicator equations for modeling the Cyber Security Game in time.

To see that an evolutionary outcome must involve a mixture of behaviors for both attacker and defender, notice that a non-attacking attacker would do better than in attacking when the defender is defending. Similarly, when the attacker is not-attacking, the defender that does not defend can invade. However, when the system is (*Not defend, Not attack*) the attacking attacker can invade. Finally, when (*Not defend, Attack*) defending, the defender can invade. That is, there is a cyclic pattern to how the mixture of behaviors is expected to evolve. This is reflected in the trajectories of such evolutionary dynamics as the replicator equation (system of two ordinary differential equations) [72]:

$$\frac{dp}{dt} = p\,(1-p)\;\big(W_D(Defend) - \;W_D(Not\;Defend)\big)$$

$$\frac{dq}{dt} = q\,(1-q)\;\;(W_A(Attack) - \;W_A(Not\;Attack)).$$

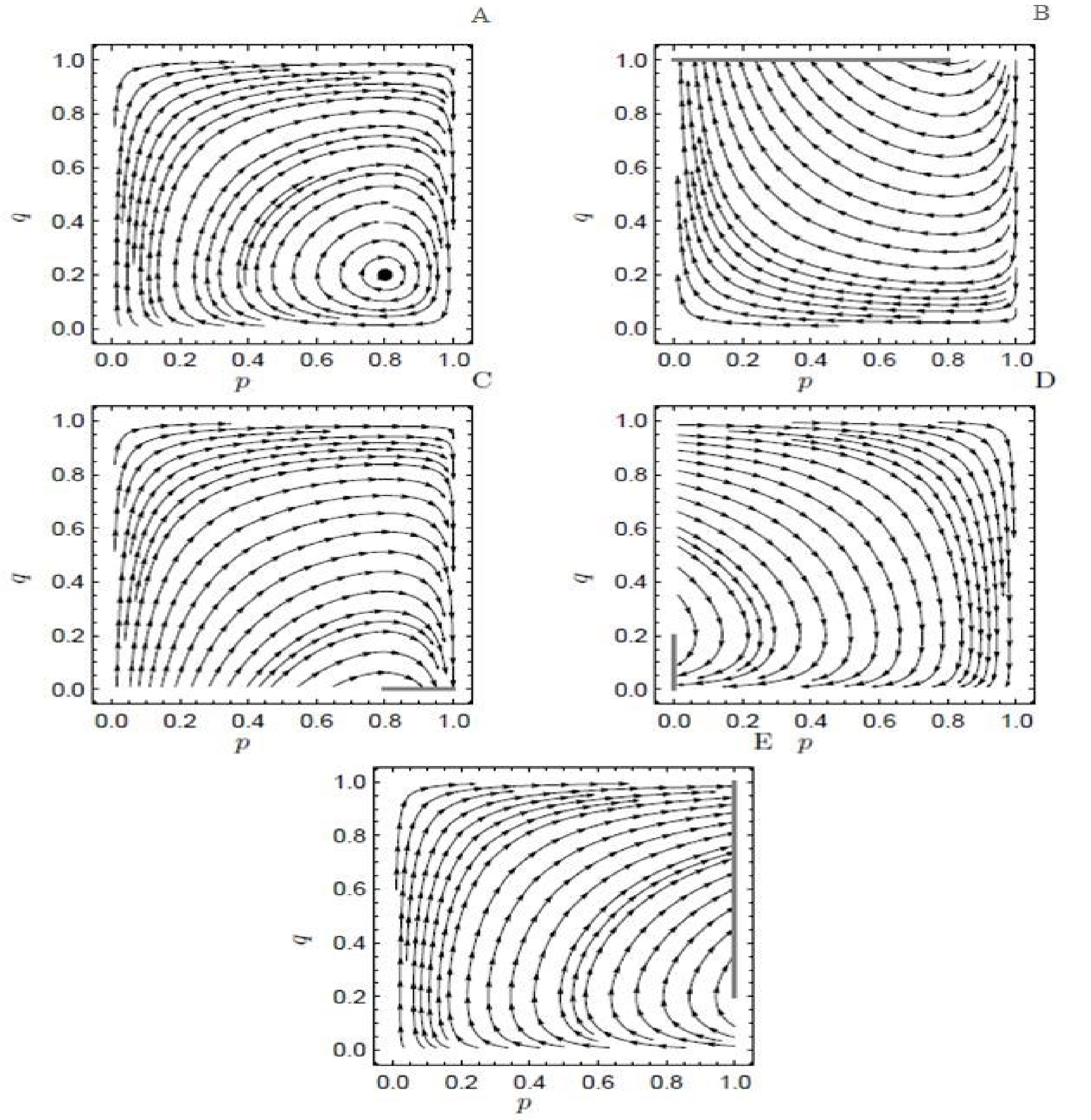

Figure 8.1 Phase portraits of the replicator equation for the Cyber Security Game

Panel A in Figure 8.1 shows the corresponding trajectories of the replicator dynamics. In this figure, there is a unique rest point that corresponds to the game's only Nash equilibrium:

$$(p, q) = (1 - \frac{c_2}{v_2}, \frac{c_1}{v_1}).$$

For some threshold parameter values there are infinitely many Nash equilibria forming Nash equilibria components (shown as gray line segments in Figure 8.1) on the boundary of the unit square. In particular,

- if $c_1 = v_1$ the segment of the upper boundary where $q = 1$ and $0 \leq p \leq 1 - \frac{c_2}{v_2}$ is a NE component (Panel B in Figure 8.1).
- When $c_1 = 0$, the segment of the lower boundary where $q_1 = 0$ and $1 - \frac{c_2}{v_2} \leq p \leq 1$ is a NE component (Panel C in Figure 8.1).
- When $c_2 = v_2$, the line segment on the left boundary where $p = 0$ and $0 \leq q \leq 1 - \frac{c_1}{v_1}$ is a NE component (Panel D in Figure 8.1).
- Finally, when $c_2 = 0$, the segment of the right boundary where $p = 1$ and $\frac{c_1}{v_1} \leq q_1 \leq 1$ is a NE component (Panel E in Figure 8.1).

9 Conclusions and Future Work

In chapter 3 have been set following aims:

1. To design algorithms that are used by multiple independent verifiers.
2. To analyze and compare proposed solutions
3. To design a model for multi factor authentication.
4. Modelling game attacker with defender (service provider or user).

We have published two new authentication schemes [66] [73] described in chapters 4 and 5 which design algorithms that are used by multiple independent verifiers. Thus, aim 1 has been fulfilled.

We have published comparison of this proposed method [74] described in chapter 6. Thus, aim 2 has been fulfilled.

However, we consider the model presented in chapter 7 to be an even more interesting result. If it is not possible to dynamically model the use of authentication methods, then in the event of an attack, applications must be stopped and re-configured. This leads to application failures.

In [68] we published the first model that assumed that risk analysis would be performed for each method of authentication. In this paper, we have already concluded that it is enough to do a risk analysis within the application (institution) for each authentication factor. The risk analysis result is then common to each authentication method. This greatly simplifies modeling.

The work [75] deals with cloud services authentication. We believe that the method we propose will be suitable for this type of service.

The model allows users to work effectively with user authentication when the application provider provides multiple authentication means of different strengths. In addition the model allows quickly respond to emerging security situations.

The novelty of our approach lies in the idea of modeling the use of authentication methods. In [68] [76] presenting the model, we simplified the risk analysis only for the whole application (institution) and not for each authentication factor.

If we do not use the model, but the individual authentication mechanisms are implemented in a "hard" manner, then in the case of an attack the corresponding countermeasures will take a long time and will be clumsy. This model allows dynamic response to various emergencies.

We simulated an attack on knowledge based authentication. The model may be used in a similar manner for other types of attack. Thus, aim 3 has been fulfilled.

The proposed model turns out to be suitable in the case of defense against organized crime. However, in the case of cybernetic warfare, other factors need to be taken into account. The goal of organized crime is to maximize profits. On the contrary, the goal of cybernetic warfare is to maximize the probability of winning.

The described model turns out to be suitable in the case of defense against organized crime. However, in the case of cybernetic warfare, other factors need to be taken into account.

Finally, we have published the Cyber Security Game [77] [78] described in chapter 8. It is a model based on game theory in which foreign power attacks the defender's asset. Thus, aim 4 has been met. This fulfilled the last goal - goal number 4.

We see further research mainly in two directions:

1. In multifactor authentication modeling: The value q^K is evaluated at the moment of authentication, i.e. at time 0 after authentication. However, we must be aware that with increasing time, the risk of a successful attack on an authenticated session increases. We should not see W_i^K as a one value (scalar), but as a function $W_i^K\ (t)$ of time t from the start of authentication. The question is how does this function changes over time?

2. In the Cyber Security Game: In the next research work, we would like to focus on a scenario where the defender has several assets but is not able to concurrently defend all of them. In this case, it is necessary to distinguish whether the attacker is able to attack one or more targets simultaneously.

10 Abstract

The thesis deals with multi factor authentication in mobile networks. The first part of thesis describes two new multifactor authentication algorithms: „Strong Authentication for Internet Mobile Application“ [66] [73] a „Strong Authentication for Internet Application“ [74].

Protocol „Strong Authentication for Internet Mobile Application“ combines protocol „Secure Hash-Based Password Authentication Protocol Using Smartcards“ [5] with AKA mechanism. On the other hand protocol „Strong Authentication for Internet Application“ combines AKA mechanism with protocol „Secure Hash-Based Password Authentication Protocol Using Smartcards“ [5].

Furthermore, the thesis introduces a model that allows you to dynamically change the requirements for the authentication method depending on the strength of authentication based on the current level of security risk [68] [76]. The possibility of practical use of the proposed model in practice is shown in three experiments.

The last part describes the model „Cyber Security Game“ [77] [78] based on game theory in which a foreign power attacks the defender's asset. This is a game involving a defender who has to decide whether to leave his asset unprotected or to invest some resources to defend the asset against a threat posed by an attacker who may or may not attack. We showed that, generically, a single mixed Nash equilibrium exists.

Keywords: multi factor authentication, AKA, risk based authentication, security game

11 Shrnutí

Práce se zabývá více faktorovou autentizací v mobilních sítích. V první části popisuje dva nové více faktorové algoritmy: „Strong Authentication for Internet Mobile Application“ [66] [73] a „Strong Authentication for Internet Application“ [74].

Protokol „Strong Authentication for Internet Mobile Application“ kombinuje protokol „Secure Hash-Based Password Authentication Protocol Using Smartcards“ [5] s mechanismem AKA. Protokol „Strong Authentication for Internet Application“ naproti tomu kombinuje mechanismus AKA s protokolem „Secure Hash-Based Password Authentication Protocol Using Smartcards“ [5].

Dále práce zavádí model, který umožňuje dynamicky měnit požadavky na autentizační metodu v závislosti na síle autentizace založení na aktuální míře bezpečnostního rizika [68] [76]. Na třech experimentech je pak ukázána možnost praktického využití navrženého modelu v praxi. V těchto experimentech jsou rovněž využity autentizační metody navržené v první části práce.

V poslední části je popsán model „Cyber Security Game“ [77] [78] založený na teorii her, kde cizí moc útočí na aktiva obránce. Jedná se o hru mezi obráncem, který se musí rozhodnout, zda ponechá své aktivum nechráněné, nebo bude investovat do obrany aktiva před hrozbou útočníka, který může nebo nemusí zaútočit. Ukázali jsme, že v tomto modelu existuje jediná smíšená Nashova rovnováha.

Klíčová slova: více faktorová autentizace, AKA, autentizace závislá na míře rizika, bezpečnostní hra

12 References

[1] „3G security; Security architecture,“ 3GPP TS 33.102 v 16.0.0, 2020. [Online]. Available: http://www.3gpp.org.

[2] F. Zhang, A. Kondoro a S. Muftic, „Location-Based Authentication and Authorization Using Smart Phones,“ v *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, Liverpool, UK, 2012.

[3] L. Lamport, „Password Authentication with Insecure Communication,“ *Communications of the ACM,* sv. 24, č. 11, pp. 770-772, 1981.

[4] W. C. Ku, „A hash-based strong-password authentication, scheme without using smart card,“ *ACM Operating Systems Review, 38(1),* p. 29–34, 2004.

[5] H. Jung, H. S. Kim, B. Murgante, O. Gervasi a A. Iglesias, „Secure Hash-Based Password Authentication Protocol Using Smartcards,“ v *11th International Conference on Computational Science and Its Applications (ICCSA), PT V Book Series: Lecture Notes in Computer Science, Volume: 6786, Pages: 593-606*, 2011.

[6] Q. Jiang, J. Ma, G. Li a L. Yang, „ Robust Two-Factor Authentication and Key Agreement Preserving User Privacy,“ *IJ Network Security, 16(4),* pp. 321-332, 2014.

[7] S. A. Cook, „The complexity of theorem-proving procedures,“ v *STOC '71: Proceedings of the third annual ACM symposium on Theory of computing*, 1971.

[8] „Secure Element Access Control,“ GlobalPlatform Device Technology, 2012. [Online]. Available: http://www.globalplatform.org.

[9] „TEE Protection Profile, Version 1.0,“ GlobalPlatform Device Committee, 2013. [Online]. Available: http://www.globalplatform.org/.

[10] „REGULATION (EU) No 910/2014 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC,“ 23 July 2014, THE EUROPEAN PARLIAMENT AND THE COUNCIL OF THE EUROPEAN UNION.

[11] „DIRECTIVE (EU) 2015/2366 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL on payment services in the internal market, amending Directives 2002/65/EC, 2009/110/EC and 2013/36/EU and Regulation (EU) No 1093/2010, and repealing Directive 2007/64/EC,“ THE EUROPEAN PARLIAMENT AND THE COUNCIL OF THE EUROPEAN UNION, 25 November 2015.

[12] „COMMISSION IMPLEMENTING REGULATION (EU) 2015/1502,“ THE EUROPEAN PARLIAMENT AND THE COUNCIL OF THE EUROPEAN UNION, 8 September 2015.

[13] European Commision, „Supplementing Directive 2015/2366 of the European Parliament and of the Council with regard to regulatory technical standards for strong customer authentication and common and secure open standards of communication (Draft),“ Brussels, 2017.

[14] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley a E. Schooler, „SIP: Session Initiation Protocol,“ IETF RFC 3261, June 2002. [Online]. Available: http://www.rfc-editor.org/rfc/rfc3261.txt.

[15] H. Schulzrinne, H. Schulzrinne, R. Frederick a V. Jacobson, „RTP: A Transport Protocol for Real-Time Applications,“ IETF RFC 3550, July 2003. [Online]. Available: http://www.rfc-editor.org/rfc/rfc3550.txt.

[16] J. Klensin, *Simple Mail Transfer Protocol,* RFC 5321: Network Working Group , 2008.

[17] T. Berners-Lee, R. Fielding a L. Masinter, *Uniform Resource Identifier (URI): Generic Syntax,* RFC 3986: Network Working Group, 2005.

[18] L. Zhu, K. Jaganathan a S. Hartman, „RFC 4121, The Kerberos Version 5, Generic Security Service Application Program Interface (GSS-API), Mechanism: Version 2," Network Working Group, 2005. [Online]. Available: https://tools.ietf.org/html/rfc4121.

[19] K. Jaganathan, L. Zhu a J. Brezak, „ SPNEGO-based Kerberos and NTLM HTTP Authentication in Microsoft Windows," Network Working Group, 2006. [Online]. Available: https://tools.ietf.org/html/rfc4559.

[20] *ITU-T H.248.1, Gateway control protocol: Version 3,* International Telecommunication Union, 2013.

[21] J. Schaad, B. Ramsdell a S. Turner, „RFC 5721, Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 4.0, Message Specification," Internet Engineering Task Force (IETF), April 2019. [Online]. Available: https://www.rfc-editor.org/rfc/rfc8551.txt.

[22] E. Rescorla, „The Transport Layer Security (TLS) Protocol Version 1.3," Internet Engineering Task Force (IETF), August 2018. [Online]. Available: https://www.rfc-editor.org/rfc/rfc8446.txt.

[23] E. Rescorla a N. Modadugu, „Datagram Transport Layer Security Version 1.2," Internet Engineering Task Force (IETF), January 2012. [Online]. Available: https://www.rfc-editor.org/rfc/rfc6347.txt.

[24] M. Baugher, D. McGrew, M. Naslund, E. Carrara a K. Norrman, „RFC 3711, The Secure Real-time Transport Protocol (SRTP)," Network Working Group , 2004. [Online]. Available: https://www.rfc-editor.org/rfc/rfc3711.txt.

[25] 3GPP, *Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN),* 3GPP TS 36.300: Release 15, 2019.

[26] 3. T. 23.002, Network architecture (release 16), 2020.

[27] *3GPP TS 23.501 System architecture for the 5G System (5GS),* 3GPP, 2020.

[28] „IP Multimedia Subsystem (IMS); Stage 2; 3GPP TS 23.228 v14.2.0," 3GPP TS 23.228 v14.2.0, 12 2016. [Online]. Available: http://www.3gpp.org.

[29] *E.212 The international identification plan for public networks and subscribers,* International Telecommunication Union, 2016.

[30] „Radio Resource Control (RRC); Protocol specification," 3GPP TS 25.331, January 2015. [Online]. Available: http://www.3gpp.org.

[31] „Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3," 3GPP TS 24.301, December 2014. [Online]. Available: http://www.3gpp.org.

[32] „General Packet Radio System (GPRS) Tunnelling Protocol User Plane (GTPv1-U)," 3GPP TS 29.281, January 2015. [Online]. Available: http://www.3gpp.org.

[33] „Evolved Universal Terrestrial Radio Access (E-UTRA); Packet Data Convergence Protocol (PDCP) specification," 3GPP TS 36.323, January 2015. [Online]. Available: http://www.3gpp.org/.

[34] „Radio Link Control (RLC) protocol specification," 3GPP TS 25.322, September 2014. [Online]. Available: http://www.3gpp.org/.

[35] „Medium Access Control (MAC) protocol specification," 3GPP TS 25.321, January 2015. [Online]. Available: http://www.3gpp.org/.

[36] V. Fajardo, J. Arkko, J. Loughney a G. Zorn, „Diameter Base Protocol," IETF RFC 6733, October 2012. [Online]. Available: http://www.rfc-editor.org/rfc/rfc6733.txt.

[37] J. Postel, „User Datagram Protocol,“ IETF RFC 768, August 1980. [Online]. Available: http://www.rfc-editor.org/rfc/rfc768.txt.

[38] R. Stewart, „Stream Control Transmission Protocol,“ IETF RFC 4960, September 2007. [Online]. Available: http://www.rfc-editor.org/rfc/rfc4960.txt.

[39] P. Faltstrom a M. Mealling, *The E.164 to Uniform Resource Identifiers (URI), Dynamic Delegation Discovery System (DDDS) Application (ENUM),* RFC 3761: Network Working Group, 2004.

[40] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson a H. Levkowetz, *RFC 2284, Extensible Authentication Protocol (EAP),* Network Working Group, 2004.

[41] J. Arkko a H. Haverinen, *RFC 4187, Extensible Authentication Protocol Method for 3rd Generation, Authentication and Key Agreement,* Network Working Group, 2006.

[42] J. Arkko, V. Lehtovirta a P. Eronen, *RFC 5448, Improved Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA'),* Network Working Group, 2009.

[43] „"Secure Hash Standard", FIPS PUB 180-2,“ National Institute of Standards and Technology, August 2002. [Online]. Available: http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf.

[44] „"Secure Hash Standard", FIPS PUB 180-1,“ National Institute of Standards and Technology, April 1995. [Online]. Available: http://www.itl.nist.gov/fipspubs/fip180-1.htm.

[45] *3GPP TS 33.501 Security architecture and procedures for 5G system (Release 16),* 3GPP, 2020.

[46] „3G Security; Specification of the MILENAGE algorithm set: An example algorithm set for the 3GPP authentication and key generation functions f1, f1*, f2, f3, f4, f5 and f5*; Document 1: General; 3GPP TS 35.205,“ 3GPP TS 35.205 v14.0.0, March 2017. [Online]. Available: http://www.3gpp.org.

[47] „3G Security; Specification of the MILENAGE algorithm set: An example algorithm set for the 3GPP authentication and key generation functions f1, f1*, f2, f3, f4, f5 and f5*; Document 2: Algorithm specification; 3GPP TS 35.206,“ 3GPP TS 35.206 v 14.0.0, March 2017. [Online]. Available: http://www.3gpp.org.

[48] „3G security; Access security for IP-based services,“ 3GPP TS 33.203, December 2014. [Online]. Available: http://www.3gpp.org.

[49] T. E. Varghese, J. B. Fisher, S. L. Harris a D. D. Boseo, *System and Method for Fraud Monitoring, Detection, and Tired User Authentication,* Patent N0.: US 7,908,645 B2, 2001.

[50] T. E. Varghese, *Techniques for Fraud Monitoring and Detection using Application Fingerprinting,* US Patent No. 8,739,278 B2, 2014.

[51] Pierson et al., *Network Security and Fraud Detection System and Method,* US Patent No. 7,272,728 B2, 2007.

[52] F. Iglesias a T. Zseby, „Time-activity footprints in IP traffic,“ *Computer Networks,* sv. 107, pp. 64-75, 2016.

[53] P. Ferrari, E. Sisinni, A. Saifullah, R. C. S. Machado, A. O. De Sá a M. Felser, „ork-in-Progress: Compromising Security of Real-time Ethernet Devices by means of Selective Queue Saturation Attack,“ v *IEEE International Workshop on Factory Communication Systems - Proceedings, WFCS*, Porto; Portugal, 2020.

[54] A. Abdalah, M. A. Maarof a A. Zainal, „Fraud detection system: A survey,“ *Journal of Networkand Computer Applications,* pp. 90-113, 68 (2016).

[55] K. Kim, „A secure hash-based strong-password authentication protocol using one-time,“ *Journal of Computer and Systems Sciences International 45,* p. 623–626, 2006.

[56] H. Jeong, D. Won a S. Kim, „Weaknesses and improvement of secure hash-based strongpassword,“ *Journal of Information Science and Engineering 26,* p. 1845–1858, 2010.

[57] T. Icart, „How to hash into elliptic curves,“ v *CRYPTO 2009*, Santa Barbara, California, USA, 2009.

[58] D. Hardt, *RFC 6749, The OAuth 2.0 Authorization Framework,* Internet Engineering Task Force (IETF) , 2012.

[59] „OpenID Connect Core 1.0,“ 2014. [Online]. Available: https://openid.net/specs/openid-connect-core-1_0.html.

[60] L. Dostálek a I. Dostálková, „Omnifactor Authentication,“ v *Advanced Computer Information Technologies ACIT 2018*, České Budějovice, 2018.

[61] F. Alaca a P. C. v. Oorschot, „Device fingerprinting for augmenting web authentication: classification and analysis of methods,“ v *ACSAC '16 Proceedings of the 32nd Annual Conference on Computer Security Applications*, Los Angeles, California, USA, 2016.

[62] Z. Yang, R. Zhao a C. Yue, „Effective Mobile Web User Fingerprinting via Motion Sensors,“ v *2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering*, 2018.

[63] S. Carta, G. Fenu, D. R. Recupero a R. Saia, „Fraud detection for E-commerce transactions by employing a prudential Multiple Consensus model,“ *Journal of Information Security and Applications,* sv. 46, pp. 13-22, 2019.

[64] „ISO/IEC 27005:2018 Information technology -- Security techniques -- Information security risk management,“ ISO/IEC, 2018.

[65] EU, „DIRECTIVE (EU) 2016/1148 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL, of 6 July 2016, concerning measures for a high common level of security of network and information systems across the Union,“ EU, Strasbourg, 2019.

[66] L. Dostalek a J. Ledvina, „Strong Authentication for Mobile Application,“ *International Conference of Applied Elecronics,* č. IEEE CFP1569A-PRT, pp. 23-26, September 2015.

[67] L. Dostálek, „Srong Authentication for Internet Application,“ v *16th European Conference on Cyber Warfare and Security*, Dublin, 2017.

[68] L. Dostalek, „Multi-factor Authentication Modeling,“ v *Advanced Computer Information Technologies ACIT'2019*, České Budějovice, 2019.

[69] E. C. Amadi, G. Eheduru, F. Eze a C. Ikerionwu, „Game Theory Application in Cyber Security; A Review,“ v *2017 IEEE 3rd International Conference on Electro-Technology for National Development (NIGERCON)*, Owerri, Nigeria, 2017.

[70] A. Zarreh, C. SayginHung, D. Wan, Y. Lee a A. Bracho, „A game theory based cybersecurity assessment model for advanced manufacturing systems,“ v *46th SME North American Manufacturing Research Conference, NAMRC 46*, Texas, USA, 2018.

[71] J. Nash, „Non-cooperative games,“ *Annals of Mathematics,* p. 54:286–295, 1951.

[72] D. P. Taylor a B. L. Jonker, „Evolutionarily stable strategies and game dynamics,“ *Mathematical Biosciences,* p. 40:145–156, 1978.

[73] L. Dostálek a J. Šafařík, „Strong autentication for internet application,“ v *uropean Conference on Information Warfare and Security, ECCWS*, Dublin, 2017.

[74] L. Dostálek a J. Šafařík, „Strong password authentication with AKA authentication mechanism," v *International Conference on Applied Electronics*, Pilsen, 2017.

[75] V. J. Chandra, N. Challa a K. S. Pasupuletti, „Authentication and authorization mechanism for cloud security," *International Journal of Engineering and Advanced Technology,* pp. 2072-2078, August 2019, E-ISSN:2249-8958.

[76] L. Dostálek a J. Šafařík, „Multifactor authentication modelling," *Radio Electronics, Computer Science,* 2020.

[77] S. Mazepa, L. Dostálek a O. Shyarmar, „Cybercrime and Vulnerability of Ukrainian Critical Information Infrastructure," v *10th International Conference on Advanced Computer Information Technologies, ACIT 2020*, Deggendorf, 2020.

[78] S. Mazepa, L. Dostálek, V. Křivan a S. Banakh, „Cybercrime in Ukraine and the Cyber Security Game," v *10th International Conference on Advanced Computer Information Technologies, ACIT 2020*, Deggendorf, 2020.

[79] L. Dostalek, „Comparison of authentication mechanisms for mobile devices," v *SPI*, Brno, 2017.

[80] *Introduction to CCITT Signaling System No. 7, ITU-T Recomendation Q.700,* International Telecommunication Union, 1993.

13 Author's publications

13.1 Publications related to the dissertation

- Dostálek L., Šafařík J: Multifactor authentication modelling, Radio Electronics - Computer Science - Control, Issue 2, Pages 106-116, 2020, volume 2, DOI: 10.15588/1607-3274-2020-2-11.
  - Indexed in WoS
- Mazepa S., Dostálek L., Shyarmar O.: Cybercrime and Vulnerability of Ukrainian Critical Information Infrastructure, 10th International Conference on Advanced Computer Information Technologies, ACIT 2020 Proceedings, art. no. 9208965, pp. 783-786, Deggendorf, Germany, DOI: 10.1109/ACIT49673.2020.9208965
  - Indexed in Scopus
- Mazepa S., Dostálek L., Křivan V., Banakh S.: Cybercrime in Ukraine and the Cyber Security Game, 10th International Conference on Advanced Computer Information Technologies, ACIT 2020 Proceedings, art. no. 9208942, pp. 787-790, Deggendorf, Germany, DOI: 10.1109/ACIT49673.2020.9208942
  - Indexed in Scopus
- Dostalek, L.: Multi-Factor Authentication Modeling, 9th International Conference on Advanced Computer Information Technologies, ACIT 2019 – Proceedings, pp. 443-446.
  - Indexed by Scopus, WoS
  - Cited in WoS:
    - Fang, D., Ye, F.: Identity Management Framework for E-Health Systems over 5G Networks, IEEE International Conference on Communications, 2018-May
- Dostálek, L., Dostálková, I.: Omnifactor authentication, CEUR Workshop Proceedings, 2300, pp. 228-231, 2018, 228-231.
  - Indexed in Scopus
  - Cited in Scopus:
    - Goncharenko, A.: Optimal Price Choice through Buyers' Preferences Entropy, 2020 10th International Conference on Advanced Computer Information Technologies, ACIT 2020 –

        Proceedings, September 2020, Article number 9208892, Pages 537-540
        - Goncharenko, A. V.: Relative Pseudo-Entropy Functions and Variation Model Theoretically Adjusted to an Activity Splitting, 9th International Conference on Advanced Computer Information Technologies, ACIT 2019 - Proceedings, pp. 52-55

- Dostálek, L., Šafařík, J.: Strong password authentication with AKA authentication mechanism. In Proceedings of 22nd 2017 International Conference on Applied Electronics, art. no. 8053581, Plzeň 2017. s. 49-54, ISBN 978-80-261-0641-8, ISSN 1803-7232, DOI: 10.23919/AE.2017.8053581
    - Indexed in Scopus, WoS
    - Cited in WoS:
        - Fang, D., Ye, F.: Identity Management Framework for E-Health Systems over 5G Networks, IEEE International Conference on Communications, 2018-May
    - Cited in Scopus:
        - Vijaya Chandra, J., Challa, N., Pasupuletti, S.K.: Authentication and authorization mechanism for cloud security, International Journal of Engineering and Advanced Technology, 8(6), pp. 2072-2078
        - Fang, D., Ye, F.: Identity Management Framework for E-Health Systems over 5G Networks, IEEE International Conference on Communications, 2018-May
- L. Dostalek: Strong Authentication for Internet Application, 16th European Conference on Cyber Warfare and Security, ECCWS, pp. 102-108, Dublin, 2017.
    - Indexed in Scopus, WoS
    - Cited in WoS:
        - Fang, D., Ye, F.: Identity Management Framework for E-Health Systems over 5G Networks, IEEE International Conference on Communications, 2018-May
- L. Dostalek: Comparison of authentication mechanisms for mobile devices SPI, Brno, 2017.
- L. Dostalek: Authentication and authorization applications in 4G networks, v Security and protection of information, ISSN 2336-5587, ISBN 978-80-7231-997-8, Brno 2015, 2015.

- L. Dostalek a J. Ledvina: Strong Authentication for Mobile Application, International Conference of Applied Electronics, art. no. 7301048, IEEE CFP1569A-PRT, pp. 23-26, September 2015.
    - Indexed in Scopus
- L. Dostálek: Velký průvodce protokoly TCP/IP: Bezpečnost, 2001, Computer Press
    - 99 Citation in Google Scholar

13.2 Other publications

- M. Vohnoutová, L. Dostálek, I. Dostálková, L. Gahurová: Conditional entropy of DNA, 10th International Conference on Advanced Computer Information Technologies, ACIT 2020, Proceedings, art. no. 9208960, pp. 94-97, DOI: 10.1109/ACIT49673.2020.9208960
- Z. Říhová, L. Dostalek: Hard and Soft Information to Achieve the Success of Project, 9th International Conference on Advanced Computer Information Technologies, ACIT 2019 – Proceedings
    - Cited by Scopus, WoS
- K. Hornickova, J. Fesl, L. Dostalek, (...), A. Weinfurtner, L. Zavitkovska: Photostruk-Uniting Science and Humanities for the Reconstruction of Lost Cultural Heritage Sites and Landscape, 2019 9th International Conference on Advanced Computer Information Technologies, ACIT 2019 – Proceedings, pp. 456-460
    - Cited by Scopus, WoS
    - One citation – see Scopus.
- K. Paclíková, A. Weinfurtnar, M. Vohnoutová, W. Dorner, J. Fesl, M. Preusz, L. Dostálek, K. Horníčková: Geoinformatics and crowdsourcing into cultural heritage. A tool for managing historical archives, AGRIS, ISSN 1804-1930, 10 (2), pp. 73-83, 2018, DOI: 10.7160/aol.2018.100207
    - Cited by Scopus, SJR (2016): 0.344
    - One citation – see Scopus.
- M. Dvorak, L. Dostalek, Z. Říhova: Optimizing the Amount of Data to Evaluate the Events of Cyber Security, International Journal of Modern Communication Technologies & Research (IJMCTR), ISSN: 2321-0850, Vol. 3 Issue 7 (July 2015)
- L. Dostálek, M. Novák: The Cryptographic Sensor, Security and Protection of Information, Brno 2013
- L. Dostálek, I. Dostálková: Není PDF/A jako PDF/A, Data Security Management 1/2013: ISSN 1211-8737
- L. Dostálek: Formáty pro zaručené elektronické podpisy – část IV, Data Security Management 3/2012: ISSN 1211-8737
- L. Dostálek, K. Štíchová: Biometrický podpis v PDF – Formát PDF, Data Security Management 2/2012: ISSN 1211-8737
- L. Dostálek: Formáty pro zaručené elektronické podpisy – Formát PDF, Data Security Management 1/2012: ISSN 1211-8737
- L. Dostálek: Formáty pro zaručené elektronické podpisy – Viditelný podpis a podpis ve formátu PDF, Data Security Management 4/2011: ISSN 1211-8737

- I. Dostalková, L. Dostalek: The Impact of Synchronization on the Group Size, ICNAAM 2011, AIP Volume 1389, 1389, pp. 1644-1647, 2011 Halkidiki, GREECE
    - Cited in Scopus, WoS
- L. Dostálek: Formáty pro zaručené elektronické podpisy – část II,CAdES, Data Security Management 3/2011: ISSN 1211-8737
- L. Dostálek: Formáty pro zaručené elektronické podpisy – část I, Data Security Management 2/2011: XAdES, ISSN 1211-8737
- L. Dostálek, M. Vohnoutová: Velký průvodce infrastrukturou PKI a technologií elektronického podpisu, 534 stran, Computer Press, Druhé vydání 2010.
    - 73 Citation in Google Scholar
- L. Dostálek, I. Dostálaková: Password Audit as indicator of security qualaity, Systémová integrace 3/2009, VŠE Praha 2009, ISSN 1210-9479
- L.Dostálek, A.Kabelová: Velký průvodce TCP/IP a systémem DNS, 418 stran, Computer Press 1999, druhé vydání 2000, třetí vydání 2003, čtvrté vydání 2005 už 542 stran, páté vydání 2008.
    - 210 citation in Google Scholar
- Л Досталек, А Кабелова: TCP/IP и DNS в теории и на практике. Полное руководство. Наука и техника 2006. ISBN 5-94387-280-9, Moskva
    - 10 citation in Google Scholar
- L.Dostálek: Bezpieczeństwo protokołu TCP/IP, 768 stran, Wydawnictwo Naukowe 2006, ISBN 1083-01-14959-0, Polsko.
    - 2 citation in Google Scholar
- L.Dostálek, A.Kabelová: A clear and comprehensive guide to TCP/IP protocols, 462 stran, ISBN 1-904811-71-X, Packt Publishing 2006.
    - 21 citation in Google Scholar
- L.Dostálek, A.Kabelová: DNS in Action, 183 stran, Packt Publishing 2006.
    - 6 citation in Google Scholar
- J.M.Kretchmar, L.Dostálek: Administrace a diagnostika sítí, Computer Press 2004
    - 17 citation in Google Scholar
- M.Vohnoutová, L. Dostálek, J. Lapáček: Připojme se k Internetu, Computer Press 2003.

Index