**14. Needham-Schroeder protocol (probably)**

Q 1: Why is it so important to use secure protocols in addition to secure encryption algorithms?

A:

Ü Protocol family to support secure data exchange

Ü Providing key exchange and authentication mechanism

Q 2: What principles (in addition to the security aspect) are considered when designing a security protocol?

A:

Ü Developed by Rodger Needham and Michael Schroeder at the Xerox Palo Alto Research Center (MIT) in 1978

Ü Development of diﬀerent variants for symmetric and asymmetric encryption systems

**Remarks**

* The NSP family is not only interesting for historical reasons, but also forms the basis for modern security protocols
* Note that the asymmetric encryption variant had a design ﬂaw that was found 17 years later

Q 3: Which notation is usually used to specify a security protocol?

**A:**

**Preliminary Speciﬁcations for Symmetric Varient**

A: Identity of Alice, B: Identity of Bob

KAB: Symmetric session key of Alice and Bob

AS: Authentication server, is trustworthy, generates and distributes the session key KAB

KA: Symmetric key between AS and A

KB: Symmetric key between AS and B

NA and NB: Nonces (number used one or number once), random numbers used for only one protocol session

**Preliminary Speciﬁcations for Asymmetric Varient**

Given Keys

1 PKAS: Public key of the authentication server AS

2 SKAS: Secret key of the authentication server AS

3 PKA and PKB: Public keys of Alice and Bob

4 SKA and SKB: Secret keys of Alice and Bob

**Assumptions**

* AS knows the public keys of all participants
* All participants only know the public key PKAS before the protocol is started

Ü Participants must request all other required keys from AS

Q 4: Specify the steps of the traditional Needham-Schroeder protocol (symmetric variant)? Why is this protocol vulnerable?

Ans:

Protocol vulnerable : Attack Types

**Man-in-the-Middle Attack**

* The attacker places himself between the communication partners Alice and Bob
* He has full control over the data traﬃc between Alice and Bob
* He can see/modify any information
* Attack is not detectable

**Replay Attack**

* Assumption: The attacker has found old keys and/or old tickets
* Attacker reuses old tickets from a previous session to manipulate the current communication

|  |  |
| --- | --- |
| F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP.PNG |  |

Q 5: Specify the asymmetric variant of the Needham-Schroeder protocol. What attack for NSP has not been detected for many years?

A:

|  |  |
| --- | --- |
| F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP_asys.PNG | F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP_asys_Attack.PNG |
| F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP_asys_Attack 1.PNG | F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP_asys_Attack_Final.PNG |

Q 6: Specify a man-in-the-middle and a replay attack for the NSP example.

|  |  |
| --- | --- |
| A: Man-in-the-middle for NSP and  F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP_MITM.PNG | Replay attack for the NSP  F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP v2_Replay Atttack.PNG |

Q 7: Which countermeasures exist to prevent these attacks?

A:

|  |  |
| --- | --- |
| F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP v2.PNG | F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP v2_MITM.PNG |
| F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP v2_Replay Atttack_NA.PNG | F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP v2_Replay Atttack_NA_Detection.PNG |
| F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP v2_Replay Atttack_Bob.PNG | F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP v2_Replay Atttack_Bob_Handshake.PNG |
| F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP v2_Replay Atttack_Bob_NB.PNG | F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP v2_Replay Atttack_Time Stamp 2.PNG |

We assume that ...

* the local clock of the target system can be manipulated or
* a time service (e.g. of a time server) can be manipulated

**Procedure**

1 Modify the time of your target system

2 Perform a replay attack

**How to protect?**

Ü Use of previously negotiated nonces also for Bob

Ü Disadvantage: The protocol is getting more complicated

|  |
| --- |
| F:\First Semester\Software Security_6 Credit\Exam Preparation\NSP v2_Replay Atttack_Time Stamp_Final.PNG |

**15. Kerberos protocol, v4 & v5 (probably)**

Q 1: What are the diﬀerences between Kerberos and Needham-Schroeder protocol? What are the common features?

A:

**Requirements:**

* Repeated authentication in decentralized networks should be possible by entering a password only once (singlesign-on)
* Bidirectional identity veriﬁcation, i.e. both client and server must be authenticatable
* Optionally, conﬁdentiality and integrity of communication data should be supported (note Kerberos originally only supported authentication)
* The explicit intention is to prevent attacks by fake identities (man-in-the-middle attack)
* Other attacks, e.g. based on retransmitting old messages (replay attack) should be mitigated by the protocol design

**Common Feature:** Authentication service based on the symmetric variant of the Needham-Schroeder protocol

Q 2: What was the motivation for introducing the Kerberos protocol? Which concepts are used to avoid the frequent request for passwords during operation?

A:

**Motivation**

* Protecting resources from unauthorized access
* Each network connection can be potentially insecure
* Network connections should not only be protected outside a subnet, but also within a subnet

**Problem**

* Protection mechanisms based only on passwords are not practical for all network connections, e.g. if each contact between a user and a server requires entering a password

Ü Instead, a centralized key management should be introduced, supported by a secure protocol

Q 3: What are the main diﬀerences between Kerberos v4 and v5?

A:

**Lifetime of Tickets**

* v4: maximum 21 hours
* v5: maximum until 31.12.2099
* In addition, v5 allows to renew tickets and to deﬁne the validity of a ticket into the future

**Encryption & Hash Functions**

* v4: DES & MD5 are ﬁxed
* v5: selectable (e.g. AES & SHA1)

**Pre-Authentication**

* v4: not supported Ü useable for active dictionary attacks
* v5: request step 1. is encrypted and is used for authentication

Q 4: Which protocol step is particularly vulnerable to replay attacks on Kerberos? What would be a successful scenario from an attacker’s point of view?

A:

**Weak Point 2: Replay Attacks**

1 Log the messages of Step 5.

2 Replay these messages at a later time

3 Use a service (e.g. print service) as a diﬀerent person

**Assumptions**

* Lifetime of messages from Step 5. has not expired at the time of replay, or
* Clock of the print server was manipulated

Ü Kerberos is not secure against replay attacks, because the attack can only mitigated by using time stamps!

Q 5: What other vulnerabilities of the Kerberos protocol could an attacker use?

A:

**Weak Points for Attacks**

**1** Key management of KDC

[ Note : Key-Distribution Center (KDC)

* Provides all necessary services for Kerberos
* Services of AS/TGS are often on the same server deployed

Ü Assumption: Services of the KDC are trustworthy (Trusted Third Party) ]

**Weak Point 1: Key management of KDC**

* KDC is a single point of failure
* KDC keys are only protected with a single master key
* DoS attacks highly eﬀective (denial of service attack)

**2** Time synchronisation and lifetime of tickets

**3** Weak passwords (dictionary attack)

**4** Session keys on the client

**5** One-time authentication only (single sign on)

Q 6: How was it possible to avoid an explicit storing of passwords?

A:

**Cross-Realm Authentiﬁcation (only Kerberos v5)**

* Using services of other realms without entering password
* Assumption: There exists a trust relationship between the user realm and the other realm

[ Note:

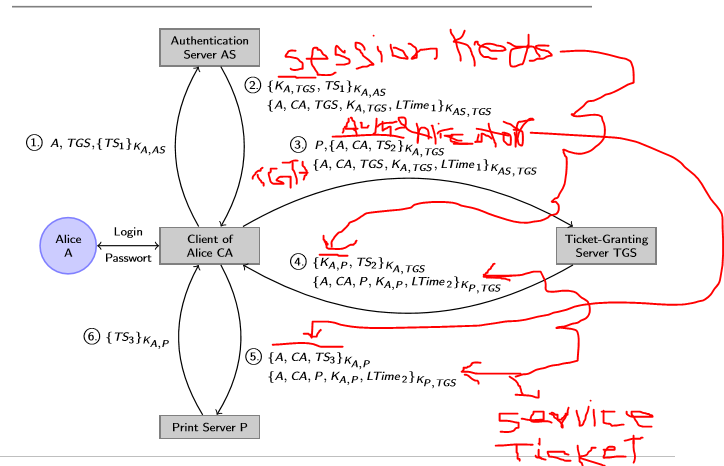
**Realm**

* + Consists of a KDC and the users assigned to it
  + Realm name is often based on DNS names
* name[/instance]@REALM (e.g. helke/admin@TU-COTTBUS.DE) ]

Q 7: Why can’t timestamps provide absolute protection against replay attacks?

A:

Kerberos is not secure against replay attacks, because the attack can only mitigated by using time stamps!



**16. Analysis using BAN logic (probably not)**

**General Remarks**

Q 1: Why does it make sense to formalize protocols using BAN logic?

Q 2: How much can you rely on a protocol that has been proven to be correct?

**How to apply BAN logic?**

Q 1: What are the most important syntactical elements of this logic?

Q 2: What is critical about using the BAN logic? Why is the idealization step often a source of errors?

Q 3: How to derive a new proposition with the BAN logic?

Q 4: Can you specify two or three deduction rules of your own choice?

Q 5: What exactly is to be proven? What are the diﬀerences between ﬁrst-order and a second-order goals?

**17. Veriﬁcation using CSP/FDR (probably not)**

**Foundations**

Q 1: What is the modeling idea of CSP? Which important CSP operators do you know? What are processes, channels and events?

Q 2: Which CSP semantics did we use in the course? Illustrate the semantics using a small example

**Protocol Speciﬁcations**

Q 1: How modeled Gavin Loewe the NSP protocol using CSP? Describe only the rough idea behind this model.

Q 2: Which processes communicate with each other? Which events are used to synchronize?

Q 3: How is the attacker modeled? What knowledge does he have and how can he learn new information?

Q 4: What reﬁnement proof was used to verify the Needham-Schroeder protocol’s vulnerability by FDR?