A secure archive for Voice-over-IP conversations

An efficient archive securing the integrity of VoIP-based two-party conversations is presented. The solution is based on chains of hashes and continuously chained electronic signatures. Security is concentrated in a single, efficient component, allowing for a detailed analysis.

Christian Hett, Nicolai Kuntze, Andreas U. Schmidt
Fraunhofer-Institute for Secure Information Technology SIT
Darmstadt, Germany

Third Annual VoIP Security Workshop 2006 – Berlin, 2th June 2006

Fraunhofer SIT Institut Sichere Informations-Technologie

Third Annual VoIP Security Workshop

Introduction and State of the Art

- VoIP is becoming the prevalent form of voice communication in commercial environments.
- In security sensitive application domains like telephony brokerage, calls need to be archived to ensure non-repudiation. (assuming the consent of the user)
- State of the art in many cases is still analogue recording. Some digital recording solutions exist, but security remains on the transport layer.
- Yet voice communication has some inherent security – Forensic analysis of voice is advanced

- Archived digital documents are always susceptible to undetected forgery
- They need special protection of their integrity.
- Ideally an integrity protection should be applied already during the ongoing conversation

Requirements for secure archiving of VoIP-conversations

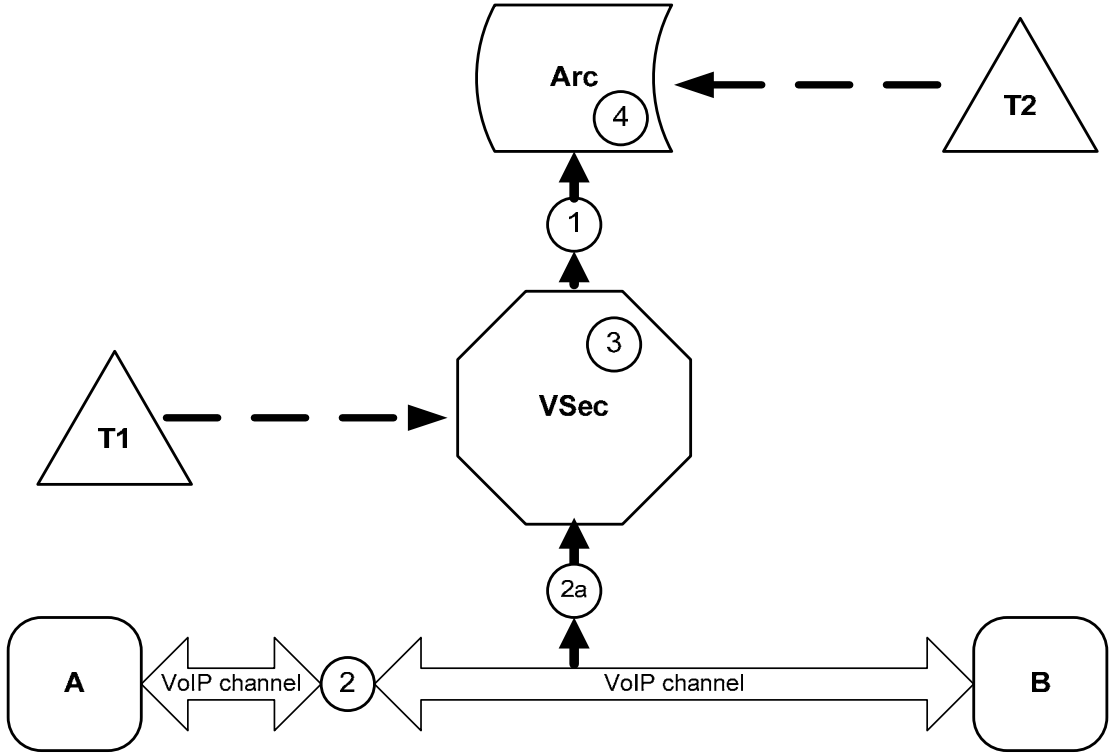
- The medium of communication here consists in a linearly time-based full duplex channel enabling inter- and transactivity

- Security-Requirements for providing non-repudiation.
  - **Cohesion**: Each archived conversation has to provide a proof of cohesion: The ordering, temporal sequencing, and completeness of the stored communication packets of both channels must be verifiable.
  - **Integrity**: Assures that the communication was not changed at any point during or after archiving. Also refers to cohesion: Simply storing raw VoIP-streams is amenable to forgery by cutting.
  - **Creation time**: Each conversation has to be reliably associated with a certain time, which must be as close as possible to the conversation's start and the initiation of the archiving. Serves as base for cohesion by providing a temporal context.

- Efficiency and Scalability:
  - Efficient use of memory: Buffering the whole conversation and securing it afterwards would pose a hard upper threshold for the number of concurrent calls and their length.
  - Computational resources: Signing each RTP-packet is too expensive

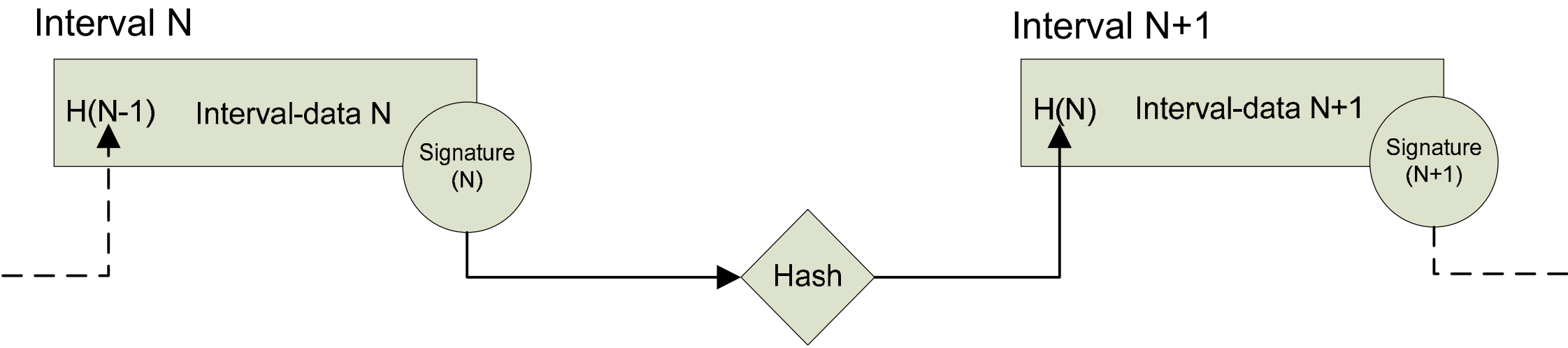
High Level architecture and implementation details

- Main design principle: Minimal requirements at the communication clients. Component VSec can be placed at the site of either of the parties A or B or anywhere in between, as long as at least some part of the communication is based on SIP/RTP.
- ARC is the component to which the secured VoIP-communication is submitted. It handles long-term storage of archived conversations
- VSec listens to the communication and secures it. It can either have a passive role or a dual role, also enforcing policies on the quality of the conversation. It then plays the role of a reference monitor.
- T1 und T2 are additional time-stamping authorities which come into play to raise resilience against attacks exerted by attackers situated in positions (1)-(4)



- This was implemented as an demonstrator. Here VSec is an outbound proxy substituting A's original outbound proxy. The proxy modifies RTP ports and IP addresses contained in the SIP packets/SDP bodies redirecting them to itself and in turn forwards them to the original recipients.

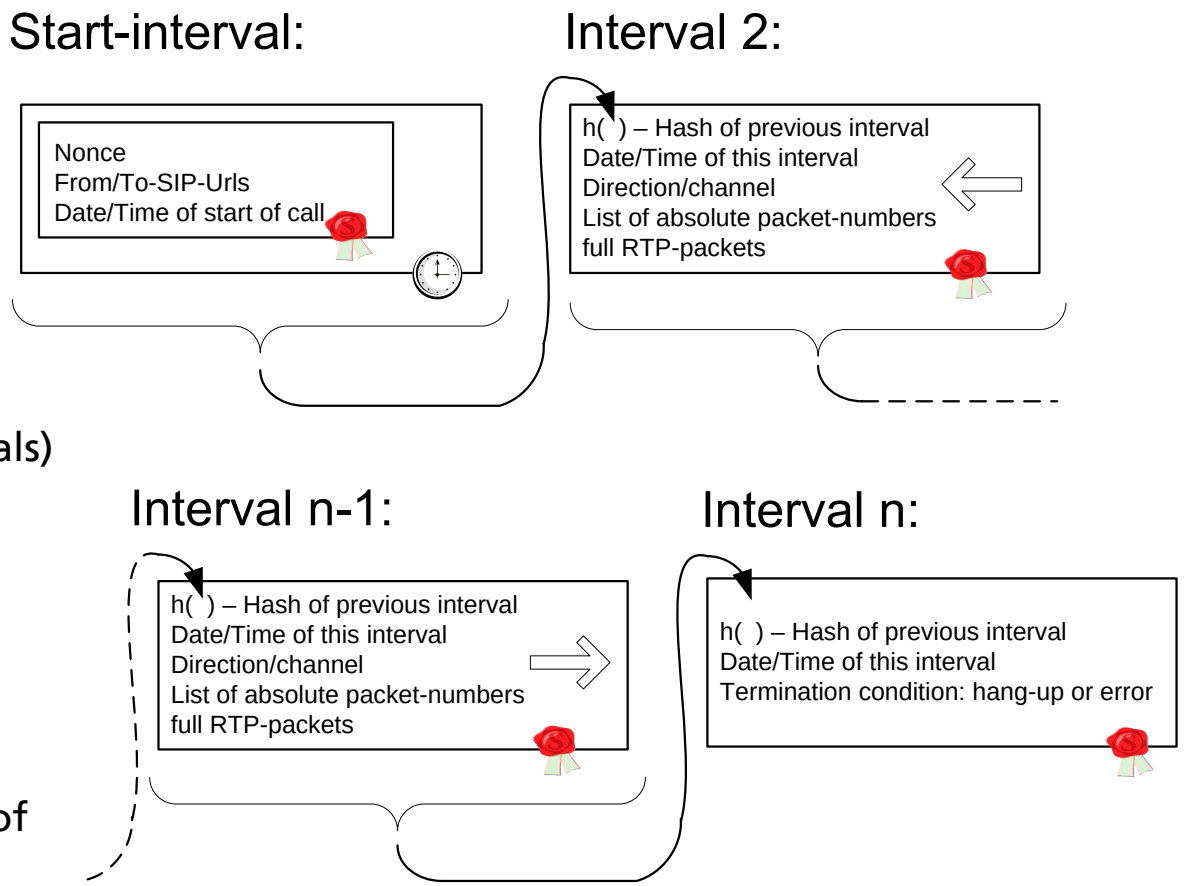
Central Concept, providing cohesion



- In order to provide efficient streaming of conversation to ARC, the packets of each audio channel are grouped into **intervals**. Each interval is digitally signed with the private key of VSec.
- Intervals can contain a dynamic or fixed amount of packets, e.g. 1 second.
- Cohesion is provided by building a hash chain over all intervals by including the hash value of one interval in the data used to compute its successor. An attacker cannot remove a single interval without invalidating the subsequent hashes and thus being detected. Any manipulation of a interval's content as well as addition or deletion of intervals is excluded.
- Because phone calls are full-duplex channels, both channels alternately contribute an interval for the chain. Thus both channels and directions of communication are tightly interweaved.

Data structure for streaming and long-term storage

- Intervals are signed using PKCS#7 with private key of VSec.
- First interval:
  - Contains SIP-URLs of caller and callee
  - Contains date and time of start of call
  - Contains mapping of RTP-payload numbers to codecs
  - PKCS#7 container contains whole certificate chain (can be omitted in all following intervals)
  - Is additionally time-stamped by T1
- Regular intervals:
  - Stems from either channel A→B or other direction.
  - Stores flag for direction/channel
  - Stores time of interval
  - Contains list of absolute sequence numbers of packets that VSec received
  - The complete RTP packets referenced by this list, including their payload type and the truncated timestamps and sequence numbers.
- Final interval:
  - Reason for termination: e.g. regular hangup by A or B, QOS-underrun, tamper-detection



QoS requirements and policies

- Some attacks based on Quality of Service are feasible: If enough packets are removed by an attacker or an compromised VSec, parts of the communication could be omitted or mangled, changing the meaning of the archived conversations.
- During archiving, VSec and Arc constantly monitor the quality of service (QoS) of the voice connection, e.g packet loss and jitter.
- If a certain QoS threshold is under-run then either the connection quality is poor and the participants cannot understand each other with a sufficient quality, or there is an ongoing attempt to attack the communication.
- It is a matter of policy how to deal with this QoS under-run:
  - Ignoring it completely and continue to archive
  - Notify the user while continuing the archiving
  - Aborting the archiving, but not the conversation.
  - Terminating the call.
    (Favoured policy for maximum security and because the QoS threshold is seldom reached without a breakdown of the connection, insufficient understandability, hang-ups or software timeouts.)
- Forced termination of calls was implemented in VSec by injecting a BYE command terminating SIP and RTP forwarding.
- The length of a interval and the QoS threshold are the main free parameters in the concept, to adjust and tune scalability, computational resources for signing and time to attack.
- Because RTP-packets only contain truncated sequence number, VSec contains a component that creates absolute sequence numbers. It is based on the replay windows algorithm of SRTP and also detects replayed RTP-packets.

Security Checks

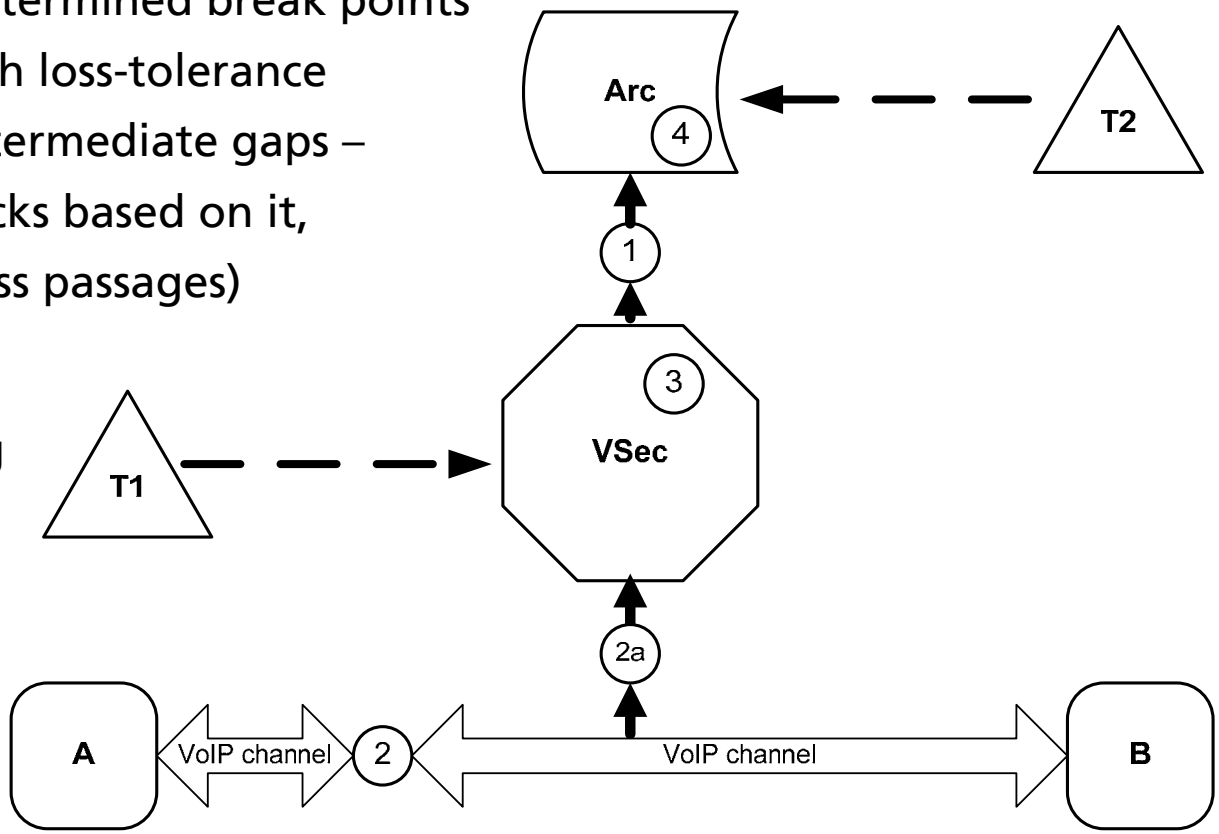
- These tests can be used in forensic, when a proof of correct archiving must be provided.
- They are also performed online while a conversation is streamed from VSec to Arc

- **CHK1**: Checking whether the first interval with the meta data is correctly signed by the external **time-stamping** service T1 and that the timestamp matches the time from the metadata.
- **CHK2**: Validating the **PKCS#7 signature** of each interval, which also authenticates VSec to Arc and ensures that no other person can submit streams to Arc.
- **CHK3**: Verifying interval **chaining**. Arc stores the SHA1 hash of the last interval and compares it to the embedded hash value in the current one.
- **CHK4**: Checking **packet loss** by checking the absolute sequence numbers in the interval structure.
- **CHK5**: Checking the time embedded by VSec in the intervals whether it **drifted** not more than two times the interval duration from the internal clock of Arc or a trusted time source.
- **CHK6**: Checking the **temporal integrity** of the RTP packets, i.e., whether the time-stamps and sequence numbers stored in the RTP protocol, which can suffer from overflows and rollovers, are consistent with the time recorded in the interval.

Security considerations 1/5

- Forgery of voice is generally difficult, requires significant resources and is still detectable
- Attacks by cutting and voice synthesis are generally out of scope of our approach, but the system design still impedes *insertion* of forged communication to a certain extent
- Generally, f*ragility* may be seen as essential:
  - QoS under-run policies provide predetermined break points
  - Non-repudiation can be balanced with loss-tolerance
  - No archived calls with (significant) intermediate gaps – mitigate semantic ambiguity (and attacks based on it, e.g., induction of packet loss to suppress passages)

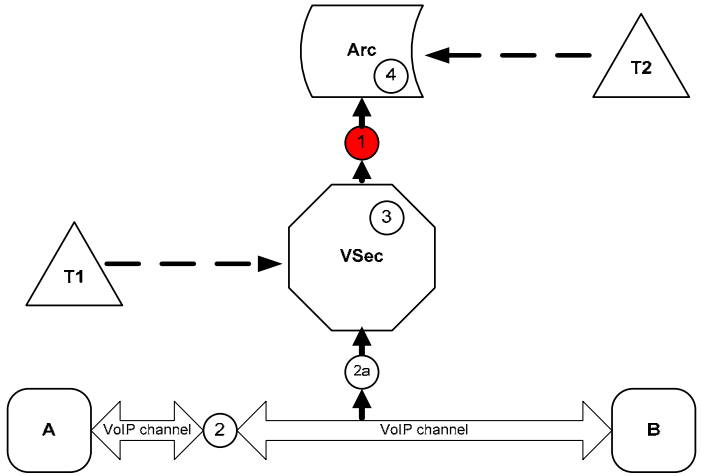
- Attack analysis is performed by positioning (1) – (4) in the overall system
- Protection targets are mentioned aside
- Analysis is essentially implementation-independent



Security considerations 2/5

(1) Man-in-the-middle

- Intercepts and manipulates communication between VSec and Arc (assumes TLS broken)
- Attacker (1) *either*
  - interrupts ongoing submission by suppressing VSec's sendings after interval n and continuing himself, *or*
  - starts his own submission, pretending to be VSec
- Threatens cohesion and therefore archive integrity



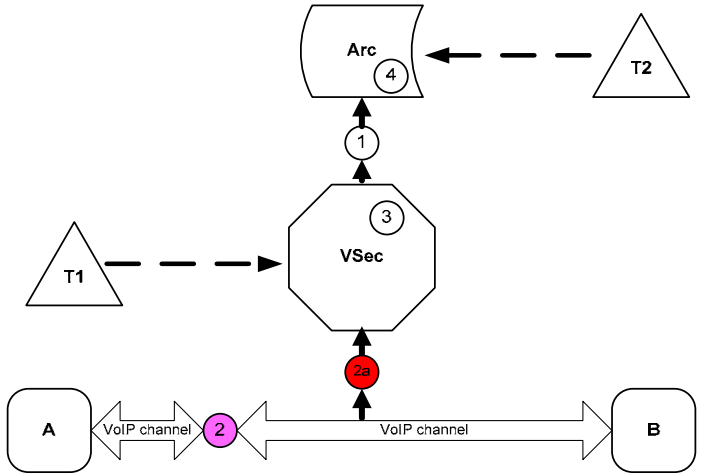
Countermeasures

- **Chaining** intervals with secrets (of VSec, verifiable by Arc) inhibits first variant
- **Securing initial data** prevents second variant, e.g.,
  - secure the first interval including some speech data (simple)
  - use certificate-based authentication of VSec w.r.t. Arc (advanced)

This represents security features inherent in the original concept of interval chaining

Security considerations 3/5

(2), (2a) Replay forgery



- (2) listens to VoIP channel at some point between A and B
- Records some initial portion of an original communication
- Replays this initial piece and continues it himself
  (synthetically or in collaboration with A or B)
- (2a) listens directly to VSec's data source and is thus
  more dangerous
- **Consequence:** Two archived calls with very similar beginnings
- Even bitwise identity and identity of meta-data (routing, system times) in case of (2a)
- Repudiation that the original call is the authentic one becomes possible and the call might loose probative force!

Countermeasures

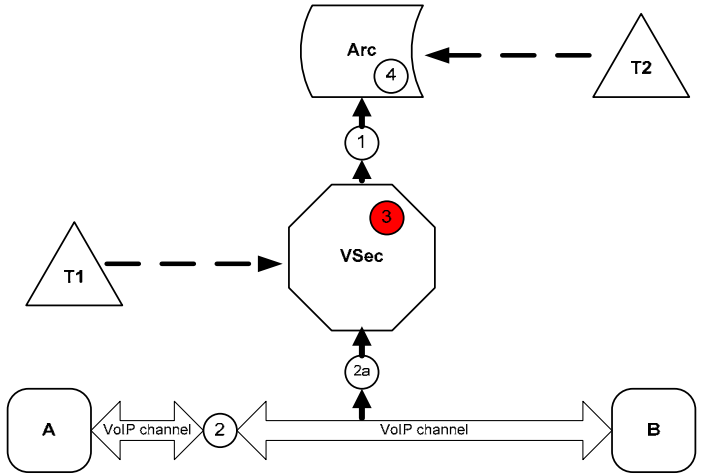
- **Simple:** exploit channel randomness, e.g., packet loss.
  Secure a large first interval that is 'unique enough'. Is effective against (2), not (2a)
- **Strong:** VSec includes random nonce to make first interval unique

Thus at least the archived calls are distinguishable
(the task of determining which is forged still remains)

Security considerations 4/5

(3) Compromised VSec secret

- Attacker knows the secret used to secure and chain intervals
- **Consequence:** can insert forged calls like (1), but maintain the cryptographic chaining



Even if (3) can authenticate to Arc so as to appear as VSec, there are still two

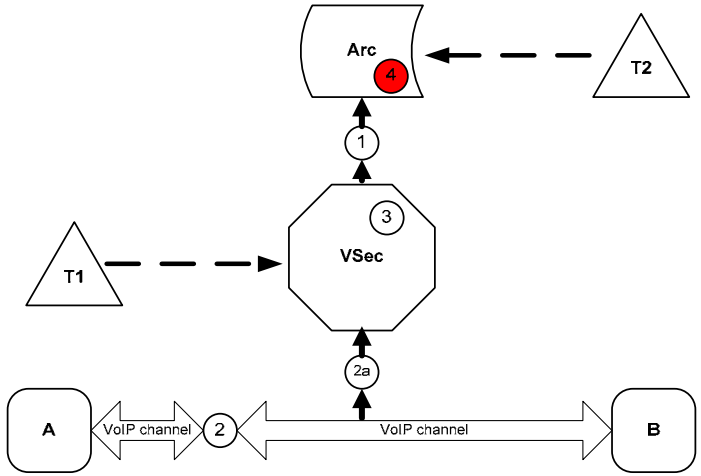
Countermeasures

- **VSec Internal:** VSec uses rotation of secrets
  - e.g. one-time keys to secure each conversation differently
  - if generated from master secret, the latter needs higher protection
- **External source of trust:** Use, e.g., a time-stamping authority T1 to initiate the chain (assuming that (3) cannot obtain time-stamps of his own). Then the attacker would need to synthesize the conversation in realtime which is infeasible.

Security considerations 5/5

(4) Forgery by the archivist

- is prevented by design principles, i.e., separation of duties
- Arc should not know the secret used by VSec to chain intervals, but must be able to verify it (information asymmetry)
- An external source of trust (T1) further impedes this attack



- Yet, Arc has control over the data for long time-spans, so can perhaps still manipulate it, given sufficient resources
- Common prevention method: Archive time-stamps by another authority T2 (efficient implementations using hash trees are well-known)

Conclusions

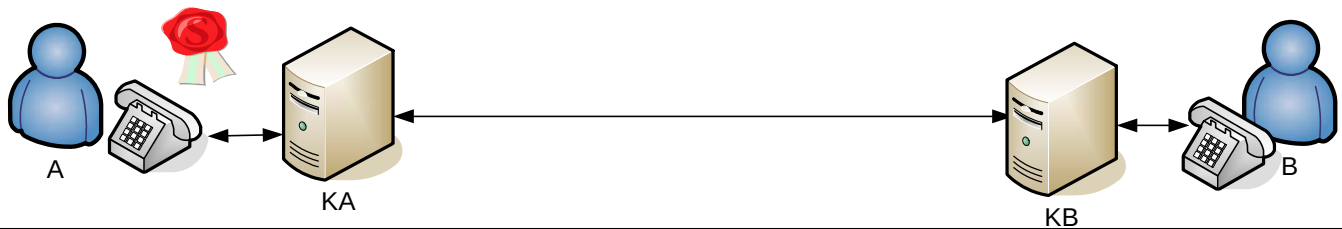
What was achieved:

- We have presented a system for secure archiving VoIP-based communication.
- In contrast to state of the art in call recording it secures cohesion, creation time and temporal context of bidirectional conversations in real-time.
- The presented solution is implemented stand-alone and offers a high degree of scalability, ease of integration, and efficiency

The method was filed as **patent application** on Wednesday

What is next to be done:

- A real-world implementation also needs to consider conditions for, and signaling and negotiation of recording of a conversation. Here use of the 'SIP Preconditions Framework for Media Privacy' could be handy.
- Signaling of archiving status and (reasons for) termination of the archiving, respectively, the call are desirable future features. A device independent way using speech synthesis can be envisaged.
- The method of interweaved signing bidirectional conversations can be extended to provide non-repudiation for telephone-calls and to establish an continuous caller-authentication.



Outlook

It seems possible to extend the present concept to a full-fledged electronic signature over VoIP based conversations.

In particular it seems possible to implement a trusted signing device as envisioned in CEN CWA 14170.

- Utilisation of trusted platforms (TP) as specified by the trusted computing group
- A TP can be used for various security-related tasks in VSec, e.g., storing secrets, securing data channels and interfaces, or providing a trustworthy computing environment.
- As another instance of trusted computing usage, the time-stamping by VSec could be implemented using an internal trusted clock of VSec seeded daily, in order to reduce the cost of purchasing timestamps.

The interessting point here is that this kind of signature terminal only needs to provide protection for the audio channel.