MATHEMATISCHES FORSCHUNGSINSTITUT OBERWOLFACH

T a g u n g s b e r i c h t 17/1990

Mathematical Concepts of Dependable Systems

15.4. bis 21.4.1990

This meeting dedicated to the area of **Mathematical Concepts of Dependable Systems**, was the first to be dedicated to this new topic in applied mathematics. Thus it was a special honour and pleasure for the organizers Gustavus Simmons (Albuquerque) and Thomas Beth (Karlsruhe) to convene the 22 participants of six countries at Oberwolfach.

The people invited had been carefully selected from the increasingly important area of Dependable Systems Research, with emphasis on special mathematical and proof theoretic questions associated with software engineering, protocol design, hardware development and their system aspects for future dependable information processing engines. The selection of both the topics to be covered and of the international list of specialists who were invited to take part in the workshop - all of whom had contributed outstanding research results during the last few years - resulted in one of the most productive workshop atmospheres at the Mathematisches Forschungsinstitut, ever experienced by either organizer who are longstanding "Oberwolfachers".

At most Oberwolfach workshops devoted to well-established topics, the participants already know each other personally, or else know of each other through their acquaintance with each other's research. In this case, where the subject matter cut across several disciplines, many of the participants (and their areas of research) were new to each other. The relaxed and positive atmosphere at the Mathematische Forschungsinstitut Oberwolfach and the beautiful surrounding landscape - as well as the well-known hospitality of all staff - combined to achieve a very close and warm working relationship among the participants, in spite of the warm (but wet) April-like weather. All participants, quite a few of whom had been visiting Oberwolfach the first time, were also impressed by their ability to work and interact in this meeting. It was especially noted that without the special support and the dedication of the director of the institut, Professor Barner, this meeting setting off a new topic of Applicable Mathematics could not have take place.

Introduction to the scope of the Workshop

The objective of this workshop was to examine the techniques that have been devised to analyse the correctness of function for systems whose functioning is so complex as to preclude an exhaustive search of all inputs and of all system states. The initiating talks entitled *Mathematical Concepts of Dependable Systems* and *Pioneering a New Topic in Applied Mathematics* given by the organizers were devoted to defining this new area of applied mathematics from two vantage points.

As this meeting was the first of its kind it is reasonable to expect the organizers to say something about what was achieved towards the stated objective. We will do this by describing briefly some of the more surprizing results reported which are specifically of the sort that prompted the organization of the workshop. Since the purpose of these examples is to illustrate as clearly as possible the scope of the workshop – no attempt will be made to summarize the entire workshop or to evaluate the contributions made by the participants.

Amongst the mathematical concepts adressed a central notion is that of a security protocol, which can be thought of as a black box that operates on several inputs (text, master key, session key, IDs, etc.) and produces one or more outputs per time step. There is an order for inputs that should produce an (expected) output output that satisfies the intended function: secrecy, authentication, private key distribution, controlled access, etc. The choice of input information is external to the black box (protocol) and hence the specified orders may not be followed. To prove that a protocol is secure would require that any possible sequence of inputs and outputs avoids the states that are failures for the protocol. C. Meadows *(Applying Formal Methods to the Analysis of a Selective Broadcast Protocol)* modeled this operation as essentially a word problem in an appropriate semi-group – so that if a string reduced to a disallowed state, the system failed. Additionally, if at some stage all strings reduced to shorter strings already seen, then the analysis could be terminated as a rigorous proof that the protocol was sound if no string had reduced to a disallowed word. This technique when applied to one of the most thoroughly studied key distribution protocols first "proved" that the key distribution protocol was correct, but also found a string that reduced to a failure for the authentification protocol! Given the scrutiny this protocol has been subjected to, this result was truly a surprise. Having exposed the failure, it was then easy to fix the fault after which the authentification protocol was also "proven" to be secure. The relevance of these

results to the workshop is obvious – formal methods provide a powerful tool for the general analysis of the correctness of the functions of complex systems. The papers by Berson, Millen, Kemmerer and Simmons addressed other aspects of proof of correctness for protocols.

The second example is from truly a counterintuitive area: Zero-Knowledge Proofs. These are two person protocols between a prover and a verifier that are supposed to make it possible for the prover to convince a sceptical verifier either that the prover knows something that he does not wish to disclose, or the class membership of a statement in a way that will not permit the verifier to produce a convincing proof to a third party. These notions are vital to schemes for proofs of identity or authority. Many such protocols depend on proving the quadratic residuosity of a test number. Desmedt and Burmester have shown that contrary to what was so far believed to have been proven, in the Fiat-Shamir protocol, a cheating prover could convince, i.e. "prove", to a sceptical verifier that a non-quadratic residue was a quadratic residue. Again this was a proof technique that had been subjected to very intensive and thorough analysis, so that finding a flaw was a true surprize. The contribution presented *System Security of Identification* by Desmedt refers to a new result on *Identification Tokens - or: Solving The Chess Grandmaster Problem* by Desmedt and Beth which shows that identification and authentification protocols based on mathematical (logical) proofs in general can be circumvented by a dedicated intruder as the Main Theorem of Game Theory shows. The inescapable conclusion is that different models and theories might to have be developed that take into account the relative time at which events (inputs and/or outputs) occur. Such a model for system function based on the absoluteness of time introduces an interesting aspect that touches on the foundations of mathematical proof theory, as does the technique of Zero-Knowledge Proofs as mentioned above.

These three examples are illustrative of the intent of the workshop organizers to bring together the principal researchers working on the problem of divising ways to analyze and prove the correctness and completeness of function of complex systems – too complex to examine exhaustively.

A fourth topic that is a central problem of research of today's complex systems is connected with the completely new and essential problems of trust and authenticity management in distributed systems. The longstanding tools of information theory do not give the equivalent notions and theorems needed to handle this problem. The preceding examples of protocols have shown the close connection with probability theory, algorithmics and logic. A more surprising result has been presented by Ingemarsson in

the paper *Democratic Shared Control Schemes* by Ingemarsson and Simmons showing how to devise systems for shared responsibility, capabilities and access control in many respects the information theoretic equivalent to the mechanistic security of bank safety-deposit boxes that require two out of three keys for the box to be opened in an authenticated way. The surprising result of Ingemarsson and Simmons is that the theory of finite geometries and error correcting codes, especially the properties of Maximum-Distance-Seperable-codes are providing just the right tool to enable system designers to construct shared access and management control systems in a provably secure way.

These four results, summarized here quite briefly, indicate why the initial expectation of the organizers to have a workshop devoted to frontline research has been more than met. The other contributions were of equally high quality and in complete accord with the intent to develop a broad understanding of the new topic in applied mathematics: Mathematical Concepts of Dependable Systems.

G. J. Simmons, Albuquerque
Thomas Beth, Universität Karlsruhe

Abstracts

Pioneering a New Topic in Applied Mathematics

The intent of this talk was twofold: to explain why organizers felt that this subject was appropriate one for an Oberwolfach workshop and to put forward some objectives which we hoped would be retained. The complex systems of interest may be software, protocols, hardware or combinations of these into information based systems in general. By dependable we mean that the system realizes the desired function(s) and that there are no surprizes. A "surprize" can be the result of

- failure, i.e. the consequence of an actionly and important nature. This is the subject matter of reliability and quality control.
- subversion, i.e. the consequence of an action by someone who wishes to subvert the system functions (to act in a way that it should not).
- unexpected function, i.e. nothing is wrong with the system. It is performing as designed - but not as desired (expected).

At the time this talk was given, we said that while all these points were appropriate topics to the theme of the workshop that (2) and (3) would be the main ones to be treated here. Subsequent discussions by the participants have been balanced between all three however. A list of eighteen information integrity functions (identification, signature, authentification, access control, shared capability, etc.) was given and discussed as a mean of illustrating what dependable system functioning meant in several/different settings. Based on this discussion a list of five information integrity primitives (identification, signature, verification, access-control and shared capability) was given and the importance of identifying and formally deciding the primitives to information integrity was emphasized.

G. J. Simmons, Albuquerque

Mathematical Concepts of Dependable Systems

This introduction is an attempt to describe the aim of systems dependability in form of a mathematical model. The dependability region is described as an admissible domain in the space parametrized by variables of efficiency, reliability, safety and security under the conditions imposed by an cost-objective function. Different paradigms of systems design are being discussed especially w.r.t. the view of interdependence of the above few parameters, the latter two of which require an intrinsically more involved

model of description. Mapping the paradigms into the shell model of man-system-interface layers this intrinsic relation between description languages, algebraic modelling, and logical theories is used there to derive an understanding of the problem of system development methodologies w.r.t. both formal and behavioural criteria. Analogies with information theory lead to giving a mathematical model of security safety violation in a layered system model. A construction for a dependability processor architecture is then derived using insights from algebraic coding theory and complexity theory.

Thomas Beth, Universität Karlsruhe

An introduction to zero-knowledge proofs

Proofs, in the classical sense, can be reproduced by the reader. So the reader can claim authorship falsely. Zero-Knowledge, informally, is a property of a protocol such that the "verifier" does not learn anything new related to some public number (called the input). In classical proofs the reader is in fact the verifier. Goldwasser-Micali-Rackoff (SIAM J. Comput., Feb. 1989) introduced and formally defined this concept. When the input of interactive proof (of membership) belongs to a set (the language) the prover will overwhelmingly convince the verifier. When the input does NOT belong to the language, except by luck, no prover whatever he tries will succed in convincing the verifier. When a protocol satisfies the last two properties it is an interactive proof (of membership). When additionally the verifier, whatever he tries to perform does not learn anything new the interactive proof is zero-knowledge. An introduction to a formal definition of zero-knowledge was given.

Yvo Desmedt, University of Wisconsin-Milwaukee, USA

Many zero-knowledge proofs are wrong!

The proof of soundness for many zero-knowledge proofs has been given in an incomplete way. We discussed the Fiat-Shamir-, Feige-Fiat-Shamir- and the Guillou-Quisquater-Scheme. All induce languages other than these claimed. Essentially they include "sporadic" numbers which are not of the prescribled type (not quadratic residues, n-th mods, etc.). These numbers are present however large the input is. This problem can be sorted out by adjusting appropriately the protocols. In general,

1. the assumption that the only way that a fraudulant prover can provide the verifier with a correct answer is try guessing, is false.

2. If a proof of knowledge is sound and the prover knows some, but not all, secrets than the verifier should not be convinced.

3. A zero-knowledge protocol induces a formal language: a complete description of the protocol is necessary to avoid vagueness, and a correct proof for soundness has to be given. The concept of soundness is very subtle.

Mike Burmester, Egham

SELANE (SEcure Local Area Network Environment)

I presented a system developed at the E.I.S.S. (European Institute for System Security) in Karlsruhe. This system is designed to cover the security needs especially for computer networks but also seems to be useful for signature and authentication systems. The basic protocol is based on the DLP (discrete logarithm problem) and could best be explained as a tricky combination of Diffie-Hellman Key Exchange together with El-Gamal-Signatures. It allows for authentication and key exchange and for the setup of mail and file encryption - electronic signatures also being possible. One basic concept of SELANE is the concept of a SKIA (Secure Key Issuing Authority) which acts as a kind of passport authority and has not to be "alive" during the operation of the system. Thus after the complete setup, the system does not need any priviliged authority, which on the one hand shows to be useful concerning performance and security, as fewer things can be attached to the running system. On the other hand this allows easily interconnecting secure LANs to form a secure WAN.

Fritz Bauspieß, Karlsruhe

Proving identity in an hostile environment?

Current authentication protocols prove the identity of a human user. To be used for program-program-authentication, the programs must be protected against reading and execution monitoring. As an example a checksum approach against intrusion using a network was given. As a solution idea, I proposed compiling the secret to the instruction stream and merging it randomly with the program proper. This was regarded infeasible by the audience.

Rainer Glaschick, Paderborn

Cryptography and Provability

Cryptography aims to make an information system "secure" against various types of subversion by an intruder. It is argued that a proof of security demands a clear statement of the cryptanalytic assumptions (what does the intruder know about the system? What messages does the intruder observe? What other actions can the intruder perform?) and a clear definition of security. It is further argued that this definition of security, if it is to be useful in practice, will inherently be stochastic in nature. The example of a "one-time-pad" (generalized so that encryption is any finite-group operation) is used to illustrate a sound proof of security. The example of a two-stage cascade cipher is used, together with the Even-Goldreich proof that the cascade is at least hard to break as either of its component ciphers, is used to illustrate the danger of failing to make the cryptanalytical assumption explicit. The first example concerns unconditional (or information-theoretic) security. The recently proposed proof scheme of Ueli Maurer, viz to find an event A such that the system is unconditionally secure when A occurs and that $P(A^c)$ is necessarily very small unless a large amount of computation is performed, is described and advocated as a powerful way to combine the information-theoretic and complexity-theoretic approaches. The scheme is illustrated by applying it to an unpublished system proposed by Whitfield Diffie in which the secret key is the telephone number of a telephone that provides the running key for an additive stream cipher.

James L. Massey, ETH Zürich

Security Analysis of Cryptographic Protocols

Cryptographic protocols are used for identification, authentification, key management, distribution of keys, confidentiality of messages, etc. Experience teaches that these protocols often fail to provide the service they intend. Mathematical analysis of the security of cryptographic protocols, especially when machine-aided, is expected to improve this situation. A survey is given of four current approaches to the analysis. Leading practitioners of the four approaches had been invited to this workshop. [See especially the talks by participants whose name is starred.]

1. Existing tools developed for the analysis of systems in general. This is exemplified by the work of Kemmerer*, with a mechanized state-transition model.

2. Expert systems with "knowledge of attacks". This is exemplified by the work of Millen*, et al., with the *Interrogator* system.

3. Model the cryptosystem as an algebra and use existing algebraic unification techniques. This is examplified by the resolution proof work of Meadows*, which has been used to find previously unknown protocol failures.

4. Modal logics of belief and authentication. This is examplified by the work of Burroughs, Abadi, Needham, which is being extended by Gong, Needham and Yahalom. Although these logics are not yet completely developed they are being productively used.

Thomas A. Berson, Palo Alto

Protocol Failures

Information integrity is broadly concerned with protocols or systems in which the ability to access, use, control, distribute, etc., a valuable resource or information, or to be able to show or to delegate these capabilities depends on the availability and integrity of pieces of information known (only) by some of the participants - a participant is any functional element in the system which may be either an individual or a device with the specified function to perform - in the system. This could be as simple as requiring a potential user to produce a fixed password to gain entry, or as complex as a nondeterministic interaction protocol among a group of participants in which this individual responses are functions of the prior responses, some of which may have been random. The main point is that if in such an information based system one can conceive of an illicit objective that can be farthered by cheating, i.e. by tampering with information. Then one has identified an information integrity problem - or in other words a problem in the dependable operation of the system as defined in my introductory remarks. This talk was devoted to an especially important and interesting (to applications) type of information integrity problem known as protocol failures. These were illustrated with a half dozen examples of cryptographic protocols (with the functions of secrecy, key distribution, manipulation detection, authentication, digital signatures, etc.) which were shown to be flawed in such a way that the protocol totally fails to deliver the intended security function, even though the integrity of the cryptographic element in the protocol is not impeached at all. Most of these failures are dependent on

the manipulation of homomorphic images through the cryptographic function to achieve a usable (to the cheater) result even though he can not invert (i.e. break) the cryptographic operation. The relevance of this topic to the subject matter of this workshop is that protocol failures provide a drastic example of the difficulty of specifying the dependable function of complex (information based) systems. These are essentially the result of doing cryptanalysis at the system and instead of the algorithm level - and hence one apropos to a discussion of system dependability.

G. J. Simmons, Albuquerque

Inconsistencies in the File Access Mechanisms of UNIX

The primitive access privileges of UNIX are overloaded. "Read", "write", and "execute" are interpreted differently when applied to different file types: directories, and ordinary files. To execute a file, different privileges are needed depending on the fact wether the file is an executable, a shellscript or a command file to be interpreted. Some examples of commands were presented where the outcome is - or should be - no access to the file at all. To handle files inodes are used. They are handled by the file manager though it does not exist as a process of its own. They are handled differently by different commands: chown, chgrp and ln. The ln-command can be misused such that files exist in the system of which the owner does not know anything, and which he can no longer access. This happens if the owner P1 of a file removes it after another person P2 has made a link to it. P2 can continue to use the file of P1 though P1 can no longer access it having deleted it. These examples show what kinds of failures can happen, when the primitive concepts used in a protocol are not well defined.

Winfried Gleißner, München

Analyzing Encryption Protocols Using Formal Verification Techniques

Encryption protocols are used for sending and receiving messages in a secure manner over a possibly insecure network. These protocols often fail to provide the expected security. By using formal specification techniques to represent the protocol one can analyze the protocol and either prove that it meets its security objectives or discover flaws in its logic. With this approach the actions of an active intruder are also modeled formally. When

using this approach nothing is proved about the encryption algorithms. That is with this approach the encryption algorithms are represented by giving axioms that characterize their properties. For example, the commutativity of encryption and decryption can be represented as: $\forall t : Text, K_1, K_2 : Key$, $(Encrypt(K_1, Decrypt(K_2, t))) = (Decrypt(K_2, Encrypt(K_1, t)))$. A tool for symbolically executing the formal specification, called *Inatest*, has been developed. Using the *Inatest tool* to test formal specifications written in *Ina Jo* a weakness in the IBM SNA protocol was demonstrated (this involved the two master keys being equal). A weakness in this same protocol when two semi-weak keys are used was also demonstrated. The weakness in the Newman, Tatebayashi, and Matsuzaki protocol that was discovered by Gus Simmons could also be duplicated using *Inatest*. Although this approach was successful in demonstrating previously known weaknesses, its true value will not be demonstrated until a flaw in a previously assumed secure protocol is found.

Richard A. Kemmerer, University of California, Santa Barbara, CA

Applying Formal Methods to the Analysis of a Selective Broadcast Protocol

In this work we develop methods for analyzing key management and authentication protocols using techniques developed for the solutions of equations in a term rewriting system. In particular, we describe a model of a class of protocols and possible attacks, and we describe a software tool based on the narrowing algorithm that can be used in their analysis. We formally model an already published protocol (Simmons, IEEE Symposium on Research in Security and Privacy, 1985) and describe the results of using these techniques to analyze various security properties. We describe a security flaw that was found using these techniques, and show how a corrected scheme was formally modeled and verified.

Catherine Meadows, Washington D.C.

The Interrogator

The *Interrogator* is a Prolog program to detect certain active wire tapping vulnerabilities in cryptographic protocols for key distribution, authentication and similar functions. Given a state transition specification of each

communicating party, it defines a relation among a data item, a message history, and a network state: namely, the attacker has obtained the data item (which may be a key or message field to be kept secret) when the given state is reached, after the message history. The message history shows where the attacker has interfered with messages in transit, and is found through the normal Prolog search. The *Interrogator* rediscovered the Denning-Sacco vulnerability in the Needham-Schroeder authentication protocol, given that the attacker had the necessary initial knowledge (a previously used key) and given the appropriate final state. The *Interrogator* contains very little algebraic knowledge and is thus not yet suitable for analyzing misuse of "secure crypto modules".

Jonathan K. Millen, Bedford, MA

Using Trace Specifications to Prove Noninterference

Noninterference is a trace-based definition of "security". The standard approach to proving that a system satisfies Noninterference is to develop a state machine model of the system and prove that the state machine satisfies unwinding conditions sufficient to guarantee Noninterference. We show how to prove that a trace specification of system behaviour satisfies Noninterference directly without having to develop a state machine model of the system. We go on to show to prove that a program correctly implements a trace specification of system behaviour.

John McLean, Washington D.C.

Evaluation or how to gain confidence

For any system to be called dependable there exists the need to establish a high degree of confidence that it fulfils its intended operational capability. This confidence can be gained in 3 ways. One is observation or use, the second by reference, the third by a so called evaluation. The first two give only very limited confidence as they are equal to testing or believe. Only the third form is felt adequate. Evaluation can be defined as: use evidence delivered and/or produce evidence that the created system when in use will show the effects described in the requirements document. This evidence shall be created with respect to known criteria. It shall also allow for peer review not only by experts. Needless to say that the evaluation shall be performed by an independable body. In the system development process requirements

are transformed until finally they can be interpreted by a machine. For higher levels of confidence the notations used to describe the transformations should be as precise as possible. Today's notions and the tools to prove certain properties about the transformations are not far enough developed so that high levels of confidence cannot be reached when systems are complex.

Christian Jahl, München

Democratic Shared Control Schemes

In an autocratic shared control scheme a trusted authority is distributing shares to the participants in a way that only preselected subsets of participants can reconstruct the secret (perform the control). If, for example, the shares are points on a line in a plane it takes two of them to reconstruct the line. We then have a 2 of l-scheme. In a democratic shared control scheme there is no trusted authority. Instead each participant randomly selects a point in some space. (This is the point protocol.) The (secret) desired control is effected by the sum of the points. Each participant distributes his point, or rather shares of the point, to the other participants using an autocratic shared control scheme. The distributer himself selects the groups of participants that should be able to act in his stead, i.e. to reconstruct his point. As an alternative the plane protocol can be used. Here each of the l participants randomly chooses a hyperplane in a l-dimentional space. The common secret is the intersection of these planes. With high probability this is a point. The simplest forms of shared control schemes (the k of l-schemes), also called threshold schemes.) are conveniently implemented using (n, k) Reed-Solomon codes.

Ingemar Ingemarsson, Linköping University, Sweden

What to do when we cannot depend on time?

In many problems time appears as a substitute for natural variables, e.g. in cooking a soft-boiled egg, "three minutes" is a substitute for a desired white/yolk consistence. Moreover, there is no way in which time could be repesented by programmatic means in a provably correct fashion. This does not prevent useful theories of a single-processor computation from being developed. For concurrent processes, the classical approach requires a kind of time-related concept (synchronisation, secure message-exchange protocols, "next" global state, etc.). A possible solution of the design dilemma

is presented by the pre- and post-guarded action specification. The construct $(P,Q) \rightarrow A$ specifies an action that can be undertaken in a state satisfying P and accepted only if the state at its completion satisfies Q. If the effects of A are not included in Q, we get a notational device for representing specifications for computations in concurrently changing environment. In particular, $(P,P) \rightarrow A$ represents a specification for action A executable and acceptable only if the world does not change during A's execution (or is restored to a state indistinguishable from initial), $(P,Q) \rightarrow A$ with $P \wedge Q \equiv \mathit{false}$ represents an action that cannot be accepted in a single processor environment. The specification by pre/post guarded actions can be applied to many classical problems in concurrent computations yielding novel and unorthodox solutions.

Wladyslaw M. Turski, Inst. of Informatics, Warsaw University

Less surprises in UNIX - extended Access Control

In standard UNIX V the access control is not object oriented. Instead of this we have to use s-bit-programs switching the effective UIDs/GIDs to other. Access operations are not fixed as data types. There is hardly any user support in administration of access-control. Tools visualizing access situations are missing, thinking in logical units and using semantic links between objects/subjects is not supported. As a solution to improve the current discretionary access control in an UNIX V compatible manner context related ACLs are proposed. Possible contexts: group, access-program (access operation), access time, etc. Some additional ACL-features: flexible administration of rights by using wildcards with different priority and comments, using of user-types or user-roles and user-competencies as logical units. Further possibilities of extensions to the ACL-concept are shown.

Hermann Strack, E.I.S.S.

The definition of "zero-knowledge" does not say anything

Concerning zero-knowledge there are two definitions - an informal one (saying: you do not learn anything during the protocol) and a formal one (being based on polynomially bounded Turing machine). After the talks and discussions about that topic I got the impression that these two definitions are independent. In fact I think that there are protocols that can be proven to

be zero-knowledge in the formal sense but which are not zero-knowledge in the informal definition. I tried to give some reasons for this and in addition presented an example of such a protocol - but the discussion is still going on ...

Fritz Bauspieß, Karlsruhe

System Security of Identification

The (Feige-)Fiat-Shamir zero-knowledge scheme was proposed as an identification scheme. The scheme is however insecure due to the so called *mafia fraud* in which a middleperson forwards the information. In the *terrorist fraud* a carrier of an "identity card" helps the terrorist identifying himself fraudulatly. A link with game theory was made. A solution based on time was proposed. The receiver (verifier) and the prover agree on an exact response time which is checked. The system need to take the laws of physics into consideration such as relativity theory.

Yvo Desmedt, University of Wisconsin-Milwaukee, USA

A Building Blocks Approach to Network Security

The problem of evaluating the security of systems, in particular distributed systems and networks, are considered. It is argued that the modular structure built into most systems is not properly reflected by current approaches to security evaluation, which stick to a global view of system analysis. The point is made that an improvement can be achieved by developing models that better suit the purpose of representing the security of building blocks to systems and their relations to each other. An extended notion of security domain is proposed to serve this purpose. As an additional advantage it provides a way of combining methods of COMSEC and COMPUSEC to achieve security for distributed systems. Examples that illustrate the arguments are given.

Hans Peter Rieß, Siemens AG, Erlangen

What you always wanted to know about Public Key Systems - or: The Algebraic Specification and Implementation of a Security Primitive

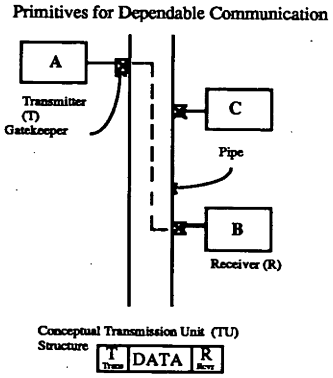
Based on a taxonomy of basic security functions the properties of a "primitive" security engine are derived from requirements of higher layers, such as

(commutative) key exchange, one-way features, (non-)homomorphism properties and performances. We give the specification of an algebraic function which realizes this concept in the arithmetics of finite groups. An implementation in abelian groups with emphasis on a VLSI-layout for groups of rational points on an elliptic curve over $GF(2^n)$ is presented.

Thomas Beth, Karlsruhe

Defining Primitives for Dependable Communication

Already in the first sessions of this meeting the need for defining basic notions and mechanisms for the **Mathematics of Dependable Systems** was identified in spontaneous and intensive discussions. During the course of the meeting, a draft proposal for coincing such "primitives" has been derived through close interaction between the participants. We present the outcome as follows:



Basic Notions:
A transmitter emits transmission units for delivery over the pipe to the indicated receiver. A message consists of some sequences (possibly fractional) of data files from TU's. Gatekeepers which may or may not exist between

each transmitter or receiver and the transmission medium (pipe) are under control of external authority not represented above. Various objectives may be stated for a given pipe, transmitter, receiver, and gatekeeper. It is the statement of an useful set of "primitive objectives" that we strive for here. Most objectives may be stated in either a positive (X should occur) or negative (X should not occur) form; only one form of each is stated below, but the other is permitted. Further, wherever the term "message" occurs below, the phrase "some function of a message" can be submitted. This permits, for example, the objective that "A cannot deny having sent the last page of a message." The function of the message required may be decided by the Transmitter, Receiver or the Authority in a particular case. "The authority" acts via the gatekeeper.
Possible objective for a Pipe:

- P1: TU should eventually arrive at R no more than specified number of times.
- P2: TU should not be revealed (completely or in part) to any but R.
- P3: TU existence should not be revealed to any but T and R.
- P4: TU should not be modified completely or partially.
- P5: TU should only arrive at R.

Possible objective for a Transmitter:

- T1: A cannot deny having sent TU that B receives with T=A relative to ?
- T2: A is obliged to send a message to B or C as decided by himself, an authority, or others. (Example: Notary cannot refuse to respond to notarization request.)
- T3: A cannot send a TU to B or C as decided by (himself, receiver, an authority).

Possible objective for a Receiver:

- R1: B receives a message as decided by (himself, an authority, the transmitter, or randomly).
- R2: B cannot receive a message as decided by (himself, an authority).

- R3: B acknowledges receipt of a message, as decided by (himself, an authority, the sender).
- R4: Complement of T1.

Examples are needed to show how systems meeting combinations of these "primitive objectives" can provide more general functions such as:

Simple Authentication:	P1+P4+?
Confidential Authentication:	P2 + ?
Notarization:	P1+P4+T1+T2+R3+?
Signature:	P4?
Non-repudiation:	T1?
Broadcast:	

Yvo Desmedt, University of Wisconsin-Milwaukee, USA
Christian Jahl, München
Carl Landwehr, Washington D.C.
Hermann Strack, Karlsruhe

Editor: Andreas Klar, Karlsruhe

Participants

Fritz Bauspieß
European Institut for System Security
Kaiserstr.8
7500 Karlsruhe 1, West Germany
E-Mail: f_bauspiess@IRAVCL.IRA.IKA.DE

Thomas Berson
Anagram Laboratories
Post Office Box 791
Palo Alto, CA 94302-0791
E-Mail: berson@SRI.COM
Fax: +1 (415) 324-0120

Thomas Beth
Institut für Algorithmen und Kognitive Systeme
Universität Karlsruhe
Am Fasanengarten
7500 Karlsruhe 1, West Germany
Tel.: ++49-721-608 4205
Fax: ++49-721-661 908

Mike Burmester
Department of Mathematics
Royal Holloway & Bedford New College
Egham Hill, Egham, Surrey TW20 OEX, England
E-Mail: uhah205@vaxa.rhbnc.ac.uk

Yvo Desmedt
Department of EE & CS
Post Office Box 784
Milwaukee, Wisconsin 53201, USA
E-Mail: desmedt@cs.uwm.edu
Fax: +1 (414) 229-6958

Rainer Glaschick
Nixdorf Computer AG
Leiter der Abt. Tools und Netz-SW
Pontanusstr. 55
4790 Paderborn, West Germany
E-Mail: glaschick@nixpbe.uucp

Wilfried Gleißner
Schleißheimerstr. 209
8000 München 40

Dieter Gollmann
European Institut for System Security
Kaiserstr.8
7500 Karlsruhe 1, West Germany
E-Mail: gollmann@IRAVCL.IRA.IKA.DE
Fax : ++49-721-661 908

Ingemar Ingemarsson
Department of Electrical Engineering
Linköping University 58183 Linköping, Sweden
E-Mail: ingemari@isy.liu.se

Christian Jahl
IABG
Abt. SZT
Einsteinstr. 20
8012 Ottobrunn
E-Mail: hhummel@AJPO.SEI.SMU.EDU

Richard A. Kemmerer
Computer Science Department
University of California
SantaBarbara, Ca 93106, USA
E-Mail: kemm@cs.ucsb.edu
Tel.: +1 (805) 961-4232

Andreas Klar
European Institut for System Security
Kaiserstr.8
7500 Karlsruhe 1, West Germany
E-Mail: s_klar@IRAVCL.IRA.IKA.DE
Fax : ++49-721-661 908

Hans-Joachim Knobloch
European Institut for System Security
Kaiserstr.8
7500 Karlsruhe 1, West Germany
E-Mail: knobloch@IRAVCL.IRA.IKA.DE
Fax : ++49-721-661 908

Carl Landwehr
NRL Code 5542
Naval Research Laboratory
Washington, D.C. 20375-5000, USA
E-Mail: landwehr@itd.nrl.navy.mil
Fax: +1 (202) 404-7942

James L. Massey
ETH-Zentrum
Institut für Signal und Informationsverarbeitung
CH-8092 Zürich
Fax: +41 1 251 21 72

John McLean
NRL Code 5543
Naval Research Laboratory
Washington, D.C. 20375-5000, USa
E-Mail: landwehr@itd.nrl.navy.mil
Fax: +1 (202) 404-7942

Catherine Meadows
NRL Code 5543
Naval Research Laboratory
Washington, D.C. 20375-5000, USA
E-Mail: landwehr@itd.nrl.navy.mil
Fax: +1 (202) 404-7942

Jonathan Millen
Mail Stop K325
The MITRE Corporation
Burlington Road
Bedford, Ma 01730, USa
E-Mail: jkm@mitre.org

Hans Peter Rieß
Siemens AG
AUT E 511
Postfach 3220
8520 Erlangen, West Germany
Tel.: ++49-9131-733381
Fax: ++49-9131-733193

Gustavus J. Simmons
Sandia National Laboratories
Albuquerque, New Mexico 87185-5800, USA
E-Mail: GJSIMMO@SANDIA.GOV
Fax: +1 (505) 846-9493

Hermann Strack
European Institut for System Security
Kaiserstr.8
7500 Karlsruhe 1, West Germany
E-Mail: strack@IRAVCL.IRA.IKA.DE
Fax : ++49-721-661 908

Wladyslaw M. Turski
Institute of Informatics
University of Warsaw
PKiN room 850
00-901 Warsaw, Poland
Fax: ++48-22-268258