A Reference Model for Firewall Technology

Christoph L. Schuba and Eugene H. Spafford
COAST Laboratory
Department of Computer Sciences
Purdue University
1398 Computer Science Building
West Lafayette, IN 47907–1398
{schuba,spaf}@cs.purdue.edu

Abstract

This paper concentrates on one particular technological aspect of providing communications security, firewall technology*. Currently firewall technology is a specialized engineering solution rather than a scientifically based solution.*

The paper introduces a reference model that captures existing firewall technology and allows for an extension to networking technologies to which it was not applied previously. It can serve as a framework in which firewall systems can be designed and validated. The essential components of the reference model are authentication, integrity assurance, access control, audit, and their enforcement. All components are governed by a centralized security policy, and they can be deployed in a distributed fashion to achieve scaling.

1 Introduction

Data communications networks have become an infrastructure resource for businesses, corporations, government agencies, and academic institutions. Computer networking, however, is not without risks as Howard ([19]) illustrates in his analysis of over 4000 security incidents on the Internet between 1989 and 1995. Firewall technology is one mechanism to protect against network-based attack methods. A balanced approach to network protection draws from several other fields, such as physical security, personnel security, operations security, communication security, and social mechanisms ([20, Part II]).

Classically, firewall technology has been applied to TCP/IP (*transmission control protocol, internet protocol*; [35, 36]) internetworks. Firewalls are used to guard and isolate connected segments of internetworks. "Inside" network domains are protected against "outside" untrusted networks, or parts of a network are protected against other parts. Various architectures for firewalls have been published and built, such as filtering routers, or application level proxy services.

To date there is neither a well designed reference model nor any theoretical background for firewall technology, let alone a definition of the term. Landwehr suggests the application of formal models of security for secure system design (see [24, §1]): by demonstrating that a design to which an implementation corresponds enforces a formal model of security, a convincing argument can be made that the system is secure.

2 Firewall Technology

2.1 Definition of Firewall Technology

Using a study of firewall systems we arrive at the following characterization of the term *firewall technology*: *Firewall technology* is a set of mechanisms that can enforce a network domain security policy $\mathbb{P}$ on communication traffic $\mathbb{T}$ entering or leaving a network policy domain $\mathbb{D}$. A *firewall system*, or *firewall*, is an instantiation of firewall technology.

This characterization covers the current state of firewall technology. Furthermore, it includes the view of firewall technology as a distributed security architecture placed on the locally controlled data transmission path between communication endpoints. Below we define the remaining terms used in the definition.

2.2 Further Terminology

A *network* is a communication system that allows computers and other electronic devices attached to it to exchange data. A *router*, *gateway*, or *switch* is a device that attaches to two or more networks and forwards information from one network to another.

We define *communication traffic* to be the transmission of information over a network. We denote the set of all possible transmissions by $\mathbb{T}$. Any instance of communication traffic, called *a transmission unit*, is a tuple $(ctrl, data) = t \in \mathbb{T}$ consisting of control information ($ctrl$) and data ($data$) either of which may be empty, but not both. The interpretation of what amount of information comprises a transmission unit depends on the protocol layer of observation. For example, in a popular instance of network layer functionality (see *open systems interconnection* (OSI) model [14]), the Internet Protocol ([35]), transmission units are called *datagrams*.

Attribute $t.ctrl$ may contain information, such as source ($t.ctrl.src$) and destination ($t.ctrl.dst$) addresses, reliability ($t.ctrl.reliab$) and flow control ($t.ctrl.flow$) information, access request information ($t.ctrl.acc$), and quality of service parameters ($t.ctrl.qos$). Attribute $t.data$ may contain application-specific payload or a payload that, at a higher layer of abstraction, can be interpreted as a transmission unit in itself. Transmission units do not need to contain all fields of t. For example, some fields may not be necessary at all, such as $t.data$ in control messages; others may be available through established state, such as $t.ctrl.qos$ in an existing connection.

A *security policy* is the definition of the security requirements for a given system. It can be defined as a set of standards, rules, or practices. We define a *network domain security policy* $\mathbb{P}$ as a subset of a security policy, addressing requirements for authenticity and integrity of communication traffic $t \in \mathbb{T}$, authorization requirements for access requests $req(t.ctrl.src, t.ctrl.dst, t.ctrl.acc)\ \forall\ t \in \mathbb{T}$, and auditing requirements.

A *network policy domain* $\mathbb{D}$ is a set of interconnected networks, gateways, and hosts offering services that are governed by a network domain security policy $\mathbb{P}$.

2.3 Firewall Mechanisms

Firewalls are implemented using a variety of security mechanisms, such as packet filtering, packet labeling, network address translation, or proxy forwarding. Several research papers and some text books describe the different approaches (see e.g., [16, §21], [5], [12], [47], [39], [41], [40], [42], [4], [3], and [2]). A subset of these mechanisms may interact to make up a comprehensive firewall system.

Currently, the term firewall is used ambiguously, because it is applied to products implementing a single mechanism as well as a small set of mechanisms that together implement possibly incompletely the network domain security policy of a corporation. Typically, a network security domain is identical to an entire corporate network. Firewalls are often deployed without the existence of a well defined network domain security policy.

Often mechanisms do not *cooperate* in enforcing a network domain security policy. It is a challenge to the designer of a firewall system to ensure that their functionality *collectively* implements the policy. This is a cruicial element in the definition of firewall technology.

2.4 Advantages of Firewall Technology

As the previous sections describe, firewalls can protect deployed computing systems and networked applications. Proponents argue that firewall technology is more than a retrofit patch for shortcomings in systems and protocol design (a survey conducted by the *National Computer Security Association* (NCSA) documents the positive experiences and perception of a small set of American businesses ([32]).) Because of their placement at the network perimeter, firewalls can serve as a centralized focus of security policy and as a place to collect comprehensive security audits, even in the presence of secure hosts. Firewalls address some problems of network security that cannot be addressed by host security mechanisms: they protect the network as a resource as well as the hosts connected to it and provide protection against some denial of service attacks ([44, §4.4]).

The aggregation of security functions in firewalls allows for a simplification of management, installation, and configuration of security functions ([6]). They improve administrative control and network management via controlled exposure of internal network structure, topological flexibility, and transparency to the user ([6]). Security firewalls represent a technology that is widely accepted, available, cost effective, and economically justifiable to management personnel in charge of purchasing decisions ([32]).

2.5 Disadvantages of Firewall Technology

Conversely, firewall technology can provide a false sense of security: it may lead to lax security within the firewall perimeter (see [6, §3]), similar to the way the supposedly impregnable Maginot Line[1] led French

[1] After André Maginot (1877-1932), French minister of war. The Maginot Line was a 150 mile long system of heavy fortifica-

army leaders to ignore the need for provision of additional defense mechanisms further inside their country ([13]). In [6, §3.1.1] this concern is expressed through another analogy: firewalls provide "a hard, crunchy outside with a soft chewy center."

Security firewalls neither provide "perfect security" nor are free of operational difficulties. They do not protect against malicious insiders. There is no protection against connections that circumvent the firewall, such as unauthorized modems attached to computers inside the firewall because the enforcing mechanism is bypassed. There is limited protection against *illicit rendez-vous* (unauthorized tunneled connections) and data-driven attacks, such as malicious executable code in downloaded Java applets ([5]). Because typical practice does not provide a check of firewall system configuration against the security policy, changes in system configurations may produce security holes ([32]).

Firewall technology has been developed for and applied to TCP/IP networks exclusively ([6]). It was never developed according to a reference model and only addressed acute problems at hand. Because of the reactive character of firewall design, there is little reason to expect that effective protection against new attacks is guaranteed. An incentive for advances in the state of the art of firewall technology has been the need to develop defenses against attack scenarios that have initially succeeded through or against firewalls.

3 Reference Model for Firewall Technology

This section presents a reference model for firewall technology. An earlier version of the model was presented in [28] (see also [29]). Computer networking is based on a layered model of communication. Communication protocols are distributed algorithms that execute between peer instances of the same layer or a range of layers. Similarly, the reference model for firewall security services as described in section 3.1 applies to a single layer or a range of layers. The reference model can be applied repeatedly at several layers within a network system as described in section 7.

In general, the analysis, manipulation, and simulation of a modeled system can lead to new knowledge and insight without the risk, cost, or inconvenience associated with its direct manipulation ([21]). The process of modeling a system gives the modeler an improved understanding of the modeled system: Jensen states that modeling as an educational tool is often its primary benefit ([21, §1.7]). One of the limitations system developers face is their own inability to cope with too many details at the same time: models help overcome this limitation ([21]) and can prove beneficial during a system's implementation (as experienced in [45, §4.1]).

The main benefits of the reference model are the provision of an understanding of

- which functions may need to be present in a firewall system,
- their enforcement,
- their interaction on a conceptual level,
- how their distribution can yield scaling benefits,
- their application at various protocol layers, and
- their composition into an overall firewall security architecture.

3.1 The Model

The reference model focuses on functionality required by firewall systems to enforce network domain security policies. For that reason we chose a functional model over other types of models, such as data processing, classification, stimulus-response, or process models ([17]). The idea is that systems are, at a conceptual level, composed of separate, interacting functional components.

Our reference model can be interpreted as a system composed of several types of security components. The components are combined under certain constraints to make up a firewall system. This explains how the functional components interact with each other and the rest of the system. Section 4 describes the components in detail.

Figure 1 displays a high-level view of the reference model of firewall technology (a more detailed representation is presented subsequently in figure 2). The representation in figure 1 is used in this paper to explain the reference model in a stepwise refined manner and as a simplified representation of the model for presentation purposes later in the paper.

Consider the case where a principal *a* outside of a protected network policy domain attempts to communicate with a principal *b* inside that domain. The gap between "out" and "in" can be filled with intermediate networks of any technology and topology so long as data can be transmitted between the sender's and the receiver's networks. Everything between the gap and

tions at the eastern frontier of France built before World War II to protect French territory from Germany. Germany did invade France again, but it went around the Maginot Line to do so. The Line itself was never taken by force.

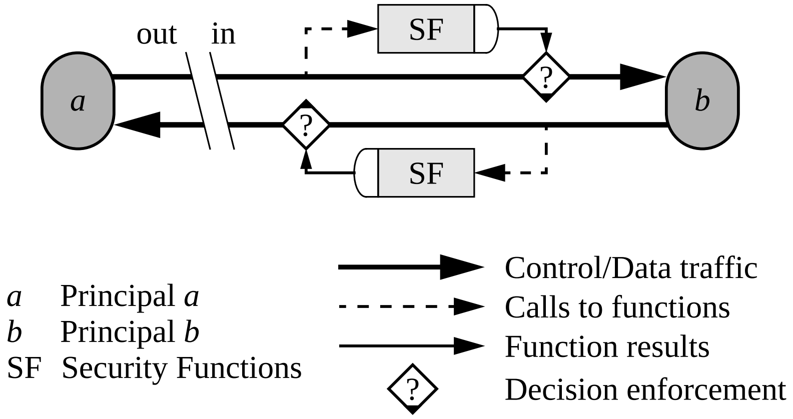

Figure 1. Abbreviated version of the reference model displayed in figure 2.

the representation of principal b is considered part of the protected network policy domain.

All communications are divided into transmission units that are transmitted by the network. The reference model operates on one protocol layer or a range of protocol layers on transmission units at that layer (range, respectively). It operates on inbound as well as outbound communication traffic. Transmission units are handled separately by the firewall, though state may be retained. The heavy, solid line represents the conceptual path that transmission units travel.

Shaded boxes represent functions. In figure 1 the boxes labeled SF represent a collection of *security functions* that are applied to transmission units exchanged between principals a and b. The dashed arrows represent the invocation of this collective function SF. Each SF receives portions or possibly even (a copy of) the entire transmission unit as input arguments. SFs calculate a result PASS or FAIL for each transmission unit. The diamond with the question mark (◇) represents the matching of the decision to its transmission unit and the decision branching and enforcement depending on the result. If the result is PASS, the transmission unit is forwarded to its destination; if the result is FAIL, an exception occurs (represented by the solid triangle in the diamond), and the transmission unit is dealt with accordingly (e.g., recorded to the audit log, and then discarded). The separation of SF into two boxes serves to further illustrate the bi-directionality of communications.

Figure 2 depicts a more detailed representation of the reference model of firewall technology. It further qualifies the structure of SF, includes an access enforcement function, and illustrates possible participation by sender and destination.

The *access enforcement function* (AEF, see section 4.5) located in the communication path between these two principals may request the authentication of each transmission unit, the verification of the integrity of each transmission unit, the access control decision, and enforce the results of these functions.

Transmission units may need to be authenticated to assure that their apparent and actual origins are identical (*authentication function*, AF; see section 4.1). The integrity of the transmission units can be verified by the *integrity function* (IF; see section 4.2).

The *access control function* (ACF; see section 4.3) determines if the transmission unit is to be forwarded further into the protected network and toward its destination. This decision can be based on control information in each transmission unit or on data contents in the case of content filtering, such as the search for Java applets or computer viruses.

Arrows with thin, dashed lines indicate possible invocations of the *audit function* (AudF; see section 4.4). All blocks that are part of the firewall system have invocation access to the audit function to record events and data according to the network domain security policy in force.

For any network transmission unit, functions AF, IF, and ACF can be called in any order. Their results are considered for the decision if the transmission unit should be forwarded toward its destination. There are firewalls that do not implement all these functions at any level of the protocol stack. Although they cannot meet the complete functionality as present in the

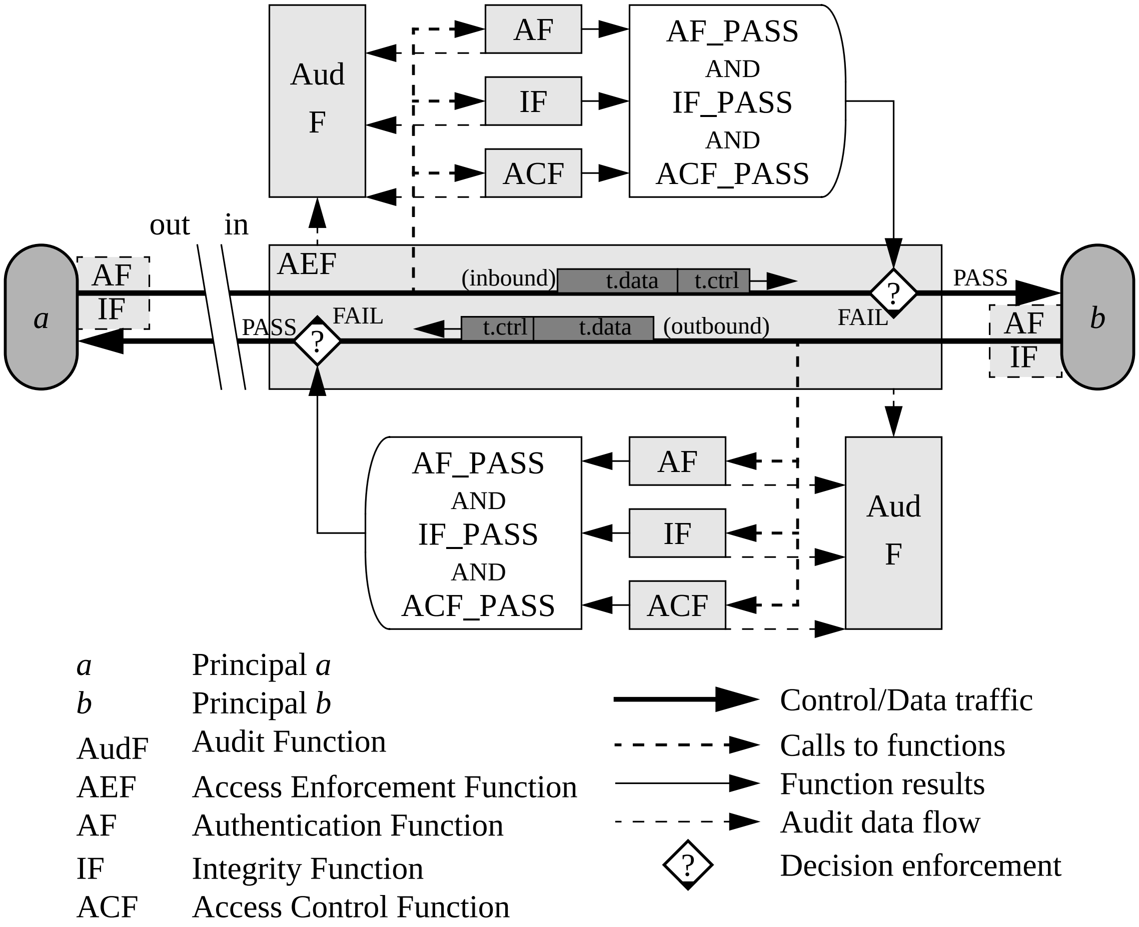

Figure 2. Reference model for firewall technology.

model, they may be sufficient to implement a particular network domain security policy. The logic gate symbol (=D-) indicates the combination of the results of the three functions into a single PASS/FAIL: if a single function generates FAIL as a result, the transmission unit should not be forwarded to its destination.

Outbound communication traffic is subject to the same security functions as inbound communication traffic. However, because of the trust relationship between the firewall and internal principals, it may not be necessary to enforce the same functions as on inbound traffic. For example, if a "trust relationship" exists between internal hosts and the firewall, a firewall designer may choose to omit outbound authentication verification of communication traffic.

Authenticity verification, integrity verification, and access control decisions may be performed close to the network perimeter of the guarded network policy domain or further inside. In both cases it needs to be assured that any possible path that transmission units can take toward the destination in the guarded network policy domain is protected by these functions.

The dashed boxes with labels AF and IF close to principals a and b indicate cooperation by a sender for the authentication and integrity functions. Cryptographic protocols, the primary means in network security to provide authentication and integrity assurance services, may require the participation of the sender (e.g., to provide cryptographic secrets for the generation of session keys). Without this cooperation, cryptographic protocols could not be used to provide the necessary services of AF and IF. The box is dashed to indicate there are authentication procedures that do not require participation of the sender, and to represent that the participation is not under control of the firewall.

There are certain constraints on the interaction of the components. For example, before the result of a call to the access control function results in communication traffic being forwarded toward its destination, the authenticity and integrity of the arguments of the function invocation must be assured. Without great confidence in the authenticity of the arguments for the access control function, its result cannot be trusted.[2] For example, in TCP/IP firewall technology, IP packet filters perform their actions based on the IP header fields present in datagrams, none of which are authenticated. This shortcoming has resulted in the exploitation of system vulnerabilities through, for example, SYN flooding ([44]) or IP address spoofing ([10]).

Figure 2 displays functions as monolithic boxes; however, such a characteristic is not meant to be implied as an implementation requirement. The representation is kept at a high level of abstraction to concentrate on the information flows and functional dependency of its components. The representation of the model is independent of its implementation.

The model allows for unilateral and mutual authentication by choosing the appropriate authentication functions on inbound and outbound communication traffic.

The application of the model is not restricted to an end-to-end, end-to-intermediate, or intermediate-to-intermediate discussion because there is no limitation on the choice of principals a and b. In particular, they do not need to be communication endpoints on destination hosts but can be on intermediate switches.

4 Components of Reference Model

As mentioned in section 3.1 and illustrated in digure 2, the reference model consists of the following functional components: authentication function (AF), integrity function (IF), access (admission) control function (ACF), audit function (AudF), and access enforcement function (AEF). This section describes these functional components.

4.1 Authentication Functions (AF)

Authentication provides assurance of the claimed identity of an entity. Authentication provides corroboration of the identity of a principal, within the context of a communication relationship. A *principal* is an entity having one or more distinguishing identifiers associated with it. Authentication services can be used by entities to verify the purported identities of principals.

A second form of authentication, called *connection authentication*, provides assurance about the authenticity of the sender of data in a connection and the integrity of transmitted data. Integrity assurance is part of connection authentication. Nevertheless, we treat these two services as separate functions.

According to our definition of the term firewall technology, firewall enforcement operates on communication traffic. It is the task of the authentication function to verify the authenticity of communication traffic based on its identifiers and authentication information (such as authentication protocol specific data). The authentication function is a predicate and returns either `AF_PASS` or `AF_FAIL`.

If there is any incoming communication traffic for which no authentication function is performed, attacks, such as address spoofing, become possible (e.g., [10, 11]). Furthermore, access control mechanisms may produce incorrect results if the source identifier of the access request is not authentic. The same arguments apply for outbound communication traffic. Therefore the AF is a necessary component for network access control.

It is necessary that the identifier that is involved in the authentication process be interpretable at any place along the connection establishment where it might be verified. If identifiers have global significance, this requirement is trivially satisfied. However, this property is usually not necessary. If an endpoint cannot be authenticated, or its identifying label cannot be interpreted, its identity is labeled as "unknown." It is the responsibility of the security policy in force to comprehend this case. A policy might allow unauthenticated traffic on its perimeter network (a network added between the protected internal network and the external network; popularly called a *demilitarized zone network* (DMZ)), but not on its internal networks; or it might allow unauthenticated traffic only to reach anonymous network services, such as anonymous ftp. These types of decisions are made by the access control function and are discussed in section 4.3.

The following definitions are consistent with [37]. Distinguishing identifiers are required for unambiguous identification within a network policy domain. They can be distinguished at a coarse level by virtue of group membership or at the finest degree of granularity identifying exactly one entity. The term *claimant* is used to describe a principal for the purpose of authentication. The authentication *verifier* is an entity which is or represents the entity requiring an authenticated identity. Authentication of a claimant to a verifier is called *unilateral* authentication. An entity involved in *mutual* authentication will assume both claimant and

[2] This does not imply that the authentication function must precede the access control function.

verifier roles.

Authentication methods rely on one or a combination of the following principles: something known (e.g., password), something possessed (e.g., security token), or some immutable characteristic (e.g., biometric identifier) (see [52, §91.2]).

There are authentication schemes with and without trusted third party involvement (see [37, figures 1,2]). In the one case no trusted third party is involved. The claimant establishes his identity with the verifier through a direct exchange of authentication information. Third parties can get involved in a variety of ways (see [37, figures 3,4,5]): in-line[3] (a trusted entity intervenes directly in an authentication exchange between the claimant and the verifier, e.g., ftp proxy), on-line[3] (one or more trusted parties are actively involved in every instance of an authentication exchange, e.g., Kerberos; [49]), or off-line[3] (one or more trusted parties support authentication without being involved in each instance of authentication).

Liebl provides a comprehensive bibliography on authentication in distributed systems in [25]. Notable publications investigating the concept of authentication as a basis for other security services are [7, 33, 8], [23], [37], [18], and [54].

4.2 Integrity Function (IF)

The integrity function protects communication traffic from unnoticed and unauthorized modifications, such as insertion, replacement, or deletion ([15, §1.2]). It cannot prevent these violations from happening, but it can detect and flag them after the fact. It is a predicate and returns either `IF_PASS` or `IF_FAIL`.

Connection hijacking, such as the active attack against TCP described in [22], is possible if there is any transmission unit for which the integrity function is not applied. Thus, the integrity function is necessary to protect against network-based active wiretapping.

Although possible, and in cases desirable, to provide an additional data confidentiality service, it is not necessary to assure integrity through encryption of the whole data stream. Integrity and confidentiality services each serve different purposes and have different performance characteristics.

There are a variety of mechanisms to detect modification of data ranging from checksum schemes, such as *cyclic redundancy checks* (CRC), to cryptographically secure digital signatures. Schneier and Stinson describe a number of such mechanisms in [43] and [50]. Keyed MD5 (*message digest 5*; [31]) is an example of such a mechanism to provide data communications integrity assurance.

4.3 Access Control Function (ACF)

The purpose of the network access control function is to generate the answer to the question of whether communication traffic is allowed to be forwarded past the firewall toward its destination, or not. This function is a predicate. Its two possible results are `ACF_PASS` and `ACF_FAIL`.

If there is no access control function on incoming communication traffic, access to arbitrary services is possible (e.g., unauthorized file retrievals via the *trivial file transfer protocol* (TFTP); [9]. A single TFTP packet is sufficient to form a file transmission request.) Furthermore, without an access control function on incoming communication traffic, data-driven attacks cannot be prevented (e.g., transmission of Java applets containing malicious code; [30]). Thus, an ACF is necessary to provide network access control.

The access control function also needs to be enforced on outgoing transmission units. Otherwise, policies, such as "No access to external Web-sites is allowed during business hours," could not be enforced. A second reason is that this function enables the prevention of information leakage (e.g., an ftp transmission of a password file, or trade secrets.)

Figure 3 illustrates the input/output behavior of a generic access control function. Any such function operates on a subset of the following input information: source and destination information, the type of access request, contextual information, and retained *access control decision information* (ADI). A security policy provides access control policy rules to the decision process. The access control function calculates a result that either allows or denies access, based on the policy and the supplied information.

This model of access control includes two main principals: an initiator and a target. Initiators can be human beings or computer-based entities that access or attempt to access targets. Targets represent computer-based or communications entities to which access is attempted. The access enforcement function is located on any possible path between initiator and target. It is part of the trusted computing base.

The object of the decision, communication traffic $t \in \mathbb{T}$, needs to contribute initiator information ($t.ctrl.src$), target information ($t.ctrl.dst$), and the access request ($t.ctrl.acc$). Source and destination address information are part of the control information, be it conveyed through out-of-band signaling messages (in connection-oriented communications), or as part of packet headers

[3] ITU terminology; see [37]

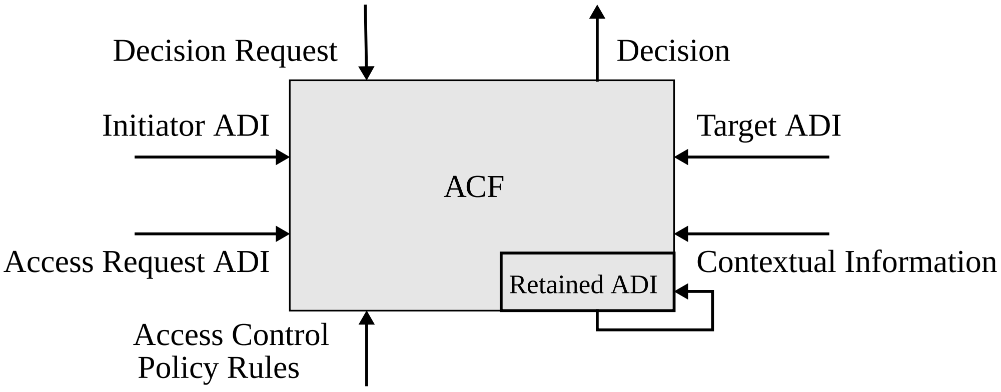

Figure 3. Model of the *access control function* **(ACF).**

(in connectionless communications). The type of access *t.ctrl.acc* may not be explicitly present in *t.ctrl*, but encoded through a combination of source and destination address information. For example, in TCP/IP well-known destination port numbers map to services that offer a certain type of access. We assume that external state can be retrieved whenever necessary and that the access control function can retain ADI for later use.

In the case of application-specific proxy servers, the access control function may be part of the proxy service. In this case, some contents of *t.data* can be input to a fine-grained access control decision. This approach has efficiency drawbacks if applied in a generic fashion at the network perimeter at the network layer. However, the approach can be feasible in specified solutions, e.g., attempting to detect certain types of Web traffic in the data stream to disallow its passing of the firewall. For example, Martin et al. ([30]) explores mechanisms to block possibly hostile external Java applets from passing through a firewall. Such mechanisms can be part of the ACF.

Much research has been performed on the semantics of access control (see e.g., [15, Chap.4], [26], [1], [38], [55], [53], and [56]). Several publications propose languages as tools for the specification of access control policies and their enforcement. A rich set of theories and existing implementations can be used.

4.4 Audit Function (AudF)

The audit function provides the capability of recording an uninterrupted, ordered journal of *significant* system events: what is designated as significant is determined by the security policy in force.

All components of a firewall system need the opportunity to record information in a consistent manner for use by systems, such as notification utilities, audit trail analysis tools, intrusion detection engines, and billing agents ([34]). The information is also provided to authorized personnel for security and system monitoring.

An audit system should be constructed in such a way that if a system violation occurs, the events leading up to and including that violation are reconstructible ([51]). Shimomura and Spafford demonstrate ([46, 48]) how audit information may be used in the aftermath of a system violation for the recovery of its functionality and the investigation of what led to the violation. Furthermore, an audit system might allow for the monitoring of systems prior to a violation. Attempts to violate security may then be noticed and acted upon before a violation occurs ([51]).

Audit does not imply the storage of redundant information beyond what is needed to establish monotonicity. However, to achieve fault tolerance, in particular in adverse situations where portions of audit information are deliberately deleted, the storage of redundant information in several locations is highly desirable: redundancy allows cross-checks for the correctness of information. Detected inconsistencies can be a warning sign of tampering. Recorded data needs to be protected from unauthorized modification, retrieval, and addition. The audit system itself needs to be protected against tampering, or the recorded data cannot be trusted. Audit in distributed systems adds several aspects, such as the problem of chronological synchronization of audit events, consistency of record formats, naming issues, and correlation of events for analysis purposes. They are beyond the scope of this paper.

Picciotto argues in [34] that the inclusion of a comprehensive auditing facility is a necessary security enhancement for any system. Security policies place various levels of emphasis on the importance of audit: in some cases the availability of the audit subsystem may not be necessary, in others required. In the latter case there is a functional dependency among all recording clients and the audit system itself, similar to the dependency of security services, such as access control on authentication. In that scenario, if the audit subsystem is not present, the remaining system is not allowed to make progress until the functionality of the audit subsystem is restored.

4.5 Access Enforcement Function (AEF)

The access enforcement function needs to enforce that the functions explained above (authentication function, integrity function, etc.) are called if required by the network domain security policy. Otherwise the access enforcement function will not receive the necessary indication at the decision points indicated by the diamonds in figures 1 and 2 to make and to enforce its decision. The logic gate symbol (⊐) illustrates that all of the results of the applied functions are part of the decision if the packet is to be forwarded toward its destination (inbound or outbound), or if it is to be discarded.

It is not sufficient to calculate the results of the functions; they need to be enforced. Therefore, without the access enforcement function all of the above attacks (and many more) are possible because of the lack of a guarantee that the results of the functions were enforced.

5 Distributed Enforcement

Figure 4 illustrates the classical view that firewalls are security devices that enforce security policy close to the network perimeter. The box labeled “Firewall” can take on various configurations, but their common characteristic is the existence of a single “choke point” or a small set of such choke points at the network perimeter ([12, §6]). The firewall is a central point of failure and becomes a performance bottleneck in the presence of high performance networking technologies ([27]).

The distribution of functions can offer various benefits. The firewall functions do not all have to be provided at the same location. They can be distributed. Their distribution reduces the performance overhead experienced at the network perimeter because fewer functions need to be computed there. Functions provided further inside the network can be executed concurrently, thus contributing to an overall performance increase of the distributed firewall. In this fashion, firewall security services can be constructed in an architecture that scales better than previous designs. The distribution of the components may be driven not only by criteria, such as performance increase through replication of functions, but also by the goal to improve reliability, availability, and disaster protection through redundant distribution of functions. Single points of failure can be avoided by design.

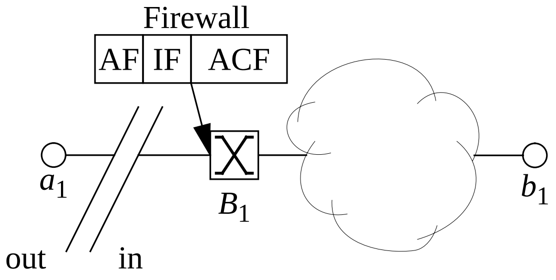

Figure 4. Example of classical approach to firewall technology: central application of a focused security enforcement device.

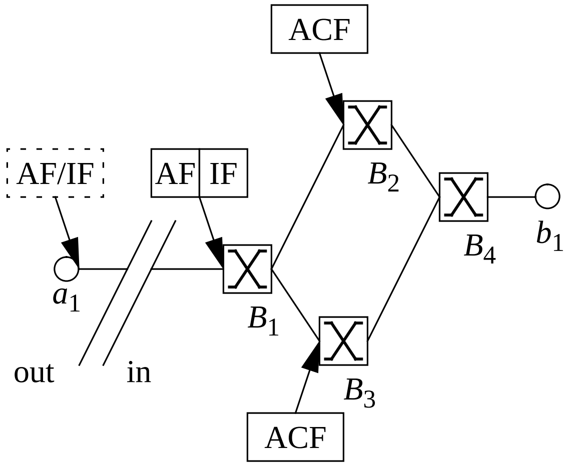

Figure 5. Example of distribution of functional components within one layer

Figures 5 and 6 depict two examples of the distri-

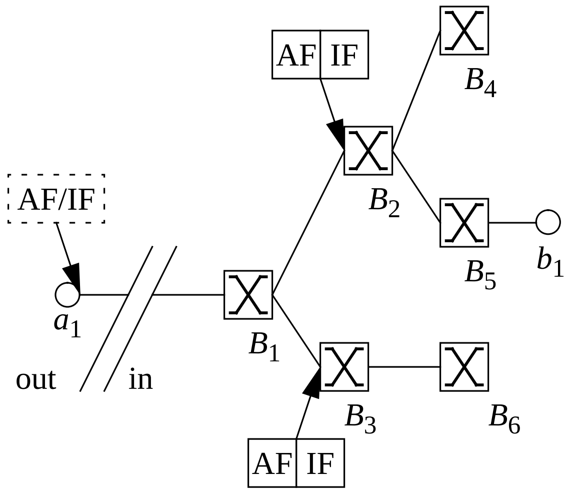

Figure 6. Second Example of distribution of functional components

bution of functions. The example in Figure 5 is a system in which the authentication and integrity functions are enforced at the perimeter switch, and the access control function (ACF) is replicated across several switches within the network. In such a scenario the ACF must be enforced on any possible path between the sender a_1 and the destination b_1, i.e., on path $P_1 := B_1 - B_2 - B_4 - b_1$ and on path $P_2 := B_1 - B_3 - B_4 - b_1$. If it was not, the ACF could be bypassed, and attacks as described in section 4.3 became possible. The example in Figure 6 locates the enforcement of functions AF and IF further inside the network than the example in figure 5.

6 Scope of Reference Model

The reference model operates on a single network policy domain with its associated network domain security policy. It does not need to address the interaction or coordination of several firewalls operating on disjoint network policy domains: if, for example, an organization decides to use firewalls to enforce the division of its internal network into several network policy domains, it can apply this model separately for each individual domain.

The reference model does not impose a specific approach to the identification of communication traffic; rather it requires external naming, addressing, and directory mechanisms that may be used for name translation. The implementation of mechanisms is not addressed by the model.

A system described by the reference model collaborates with other services outside the scope of the model, such as connection management, data forwarding (or switching) agents, and user processes. Although the reference model provides for security services, such as authentication, their implementations are provided externally. The overall system depends on the availability of these services.

We therefore assume the existence of a naming service and a secure key distribution infrastructure (public or private key distribution, depending on the requirements of the used security protocols). Furthermore, we assume the binding between the identifiers of communicating principals and their associated keys are not compromised. The integrity of the trusted computing base and the appropriate strength of used cryptographic algorithms and parameters must be assured.

The implementation of functions by mechanisms can be made independently of how the function is to be used or how the supporting mechanisms are provided.

As defined in our definition of firewall technology, this model operates on communication traffic entering or leaving a network policy domain. It therefore does not address the security problems associated with communication traffic that does not cross a domain's perimeter, as is the case when insiders of an organization launch network-based attacks against the own organization. Some mechanisms, however, are capable of protecting against such threats. Nevertheless, this scenario is not addressed as part of firewall technology.

It is necessary to be precise about the perimeter of the network policy domain because a circumvention of a firewall defeats its purpose. For example, it is not obvious if a company-owned laptop, used by an employee on a business trip, is to be considered part of the company's protected network policy domain. For the purpose of our work we assume that the network perimeter is specified, so that it is clear what equipment is "inside" and what equipment is "outside" of the network policy domain, and therefore which communication traffic crosses the perimeter and becomes subject to firewall controls.

As mentioned in section 1 the security services provided by a firewall system are only a subset of those required to make a system "secure." Firewalls need to interact with other security aware systems and components. For example, the firewall may allow an anonymous connection to be established to an information server and delegate the file access control decision to that server.

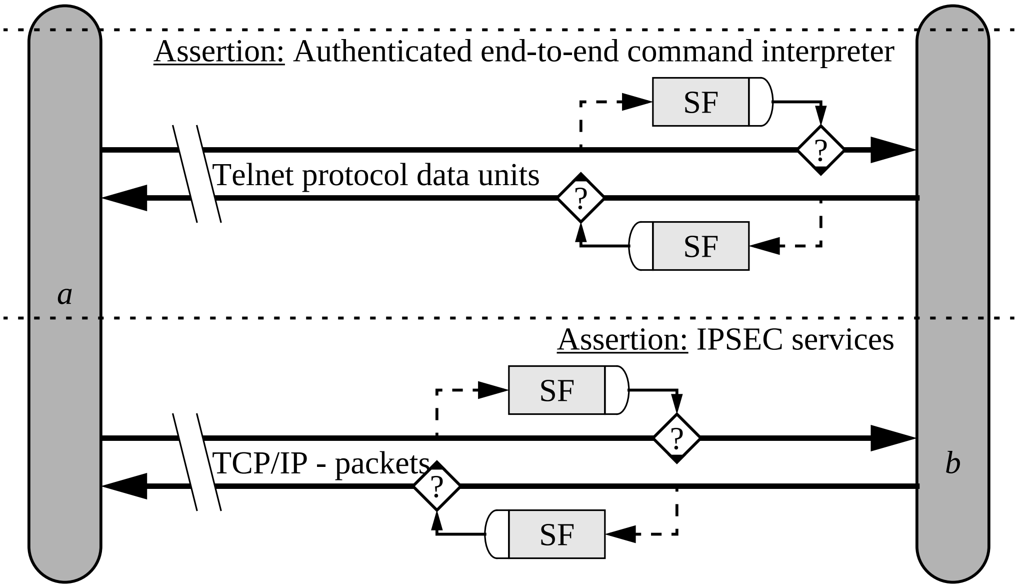
Assertion: Authenticated end-to-end command interpreter
SF
?
Telnet protocol data units
?
SF
a
Assertion: IPSEC services
SF
?
TCP/IP - packets
?
SF
b

Figure 7. Example of the application of the reference model at several layers.

7 Repeated Application of Reference Model

Security services as covered by the reference model at a given protocol layer or range of layers can be expressed by assertions. For example, authenticated signaling in the *asynchronous transfer mode* (ATM), offers endpoint authentication and integrity assurance of signaling messages. Thus, an assertion of the reference model applied to ATM connection establishment traffic is of the form: “For all established ATM connections, participating principals are authenticated and the authenticity and integrity of all signaling messages is verified.” This assertion can then be used as a basis for further application of the reference model to communication traffic at other protocol layers.

In designs that include the repeated application of the reference model such assertions are used to determine if the assumptions of security mechanisms at higher layers are met. Through the matching of assertions and assumptions, designers can combine the repeated application of the reference model into a multilayered firewall security architecture.

This approach is flexible and an improvement over the previous state of the technology that favored monolithic designs because it allows the composition of network security mechanisms in layered communication systems. The assertions provide an understanding of which security services are provided by lower layers and can be relied upon. If mechanisms in lower layers are changed, their assertions are likely to change, and designers can determine if the assumptions of higher layer mechanisms are still fulfilled by the lower layers' assertions.

Figure 7 illustrates an example of the repeated application of the reference model at several protocol layers: at the ATM layer the application of the model asserts the authenticity and integrity of ATM signaling. These authenticated connections are then used for the transmission of TCP/IP packets. If the IPSEC (*IP security working group*) security enhancements for IP are used, the model asserts at this layer that the integrity and authenticity of each IP packet within the authenticated ATM connections are protected by the IPSEC *authentication header* (AH). Finally, at the highest layer in this example, application layer services (e.g., telnet) can provide additional security services, such as the authentication of the remote user through a password exchange.

8 Conclusions

This paper presented the reference model for firewall technology. The functional model is a guide to structure firewall security services at a single layer or a range of layers in a layered model of computer networking. It identifies a fundamental set of functions to providing network access control: authentication function, integrity function, access control function, audit function, and access enforcement function. A characterization of each function is given without the provision of details of how the functionality is to be achieved. Furthermore, this paper explains how the distribution of the functional components can be used to decrease the performance overhead introduced by firewalls in classical firewall architectures.

Likely future trends in computer networking are addressed by the model. It can be applied to networking technologies, such as those required by wireless computing (the difficulty here is the definition of the network perimeter) and high performance networking as described above. We expect the reference model to have an educational and a guiding influence on the design of future firewalls.

Acknowledgments

We thank Katherine Price for valuable technical discussions. We thank J. Bryan Lyles for contributions to a previous version of the paper. Suggestions from the anonymous referees helped improve the presentation. This work was funded in part by Sprint Corporation, Xerox Corporation, and Sun Microsystems. We gratefully acknowledge the financial support.

References

[1] M. Abadi, M. Burrows, B. W. Lampson, and G. Plotkin. A Calculus for Access Control in Distributed Systems. Technical Report DEC/SRC-070, Digital Equipment Corporation (DEC), Feb. 1991.

[2] R. Atkinson. *RFC-1825 Security Architecture for the Internet Protocol.* Network Working Group, Aug. 1995.

[3] F. M. Avolio and M. J. Ranum. A Network Perimeter with Secure External Access. In *2^{nd} Symposium on Network and Distributed System Security (NDSS)*, San Diego, California, Feb. 1994. Internet Society (ISOC).

[4] F. M. Avolio and M. J. Ranum. A Toolkit and Methods for Internet Firewalls. In *Technical Summer Conference*, pages 37–44, Boston, Massachusetts, June 1994. USENIX.

[5] S. M. Bellovin and W. R. Cheswick. *Firewalls and Internet Security.* Addison-Wesley Publishing Company, Inc., 1994.

[6] B. Braden (editor), D. Clark, S. Crocker, and C. Huitema. *RFC-1636 Report of IAB Workshop on Security in the Internet Architecture.* Network Working Group, June 1994.

[7] M. Burrows, M. Abadi, and R. Needham. A Logic of Authentication. Technical Report SRC-039, Digital Equipment Corporation (DEC), Feb. 1989.

[8] M. Burrows, M. Abadi, and R. Needham. Rejoinder to Nessett. *Operating Systems Review*, 24(2):39–40, Apr. 1990.

[9] CERT. *Active Internet tftp Attacks, CA-91:18.* Computer Emergency Response Team, Carnegie Mellon University, Pittsburgh, Pennsylvania, Sept. 1991.

[10] CERT. *IP Spoofing Attacks and Hijacked Terminal Connections, CA-95:01.* Computer Emergency Response Team, Carnegie Mellon University, Pittsburgh, Pennsylvania, Jan. 1995.

[11] CERT. *TCP SYN Flooding and IP Spoofing Attacks, CA-96:21.* Computer Emergency Response Team, Carnegie Mellon University, Pittsburgh, Pennsylvania, Sept. 1996.

[12] D. B. Chapman and E. D. Zwicky. *Building Internet Firewalls.* O'Reilley & Associates, Inc., Sebastopol, California, Sept. 1995.

[13] R. Chelminski. The Maginot Line. *Smithsonian Magazine*, pages 90–100, June 1997.

[14] J. D. Day and H. Zimmermann. The OSI Reference Model. In *Proceedings of the IEEE*, volume 71, pages 1334–1340. IEEE, Dec. 1983.

[15] D. E. Denning. *Cryptography and Data Security.* Addison-Wesley Publishing Company, Inc., 1982.

[16] S. Garfinkel and G. Spafford. *Practical UNIX & Internet Security.* O'Reilley & Associates, Inc., Sebastopol, California, second edition, 1996.

[17] C. Ghezzi, M. Jazayeri, and D. Mandrioli. *Fundamentals of Software Engineering.* Prentice-Hall, Englewood Cliffs, New Jersey, 1991.

[18] N. M. Haller and R. Atkinson. *RFC-1704 On Internet Authentication.* Network Working Group, Oct. 1994.

[19] J. D. Howard. *An Analysis Of Security Incidents On The Internet 1989-1995.* PhD thesis, Carnegie Mellon University, Apr. 1997.

[20] D. Icove, K. Seger, and W. VonStorch. *Computer Crime.* O'Reilley & Associates, Inc., Sebastopol, California, 1995.

[21] K. Jensen. *Coloured Petri Nets: Basic Concepts, Analysis Methods, and Practical Use*, volume 1. Springer-Verlag, New York Inc., second edition, 1996.

[22] L. Joncheray. A Simple Active Attack Against TCP. In *Proceedings of the 5^{th} UNIX Security Symposium*, pages 7–19, Salt Lake City, Utah, June 1995. USENIX.

[23] B. Lampson, M. Abadi, M. Burrows, and E. Wobber. Authentication in Distributed Systems: Theory and Practice. Technical Report SRC-083, Digital Equipment Corporation (DEC), 1992. Reprinted in ACM Transactions on Computer Systems, volume 10, number 4, November 1992, 265–310.

[24] C. E. Landwehr. Formal Models for Computer Security. *ACM Computing Surveys*, 13(3):247–278, Sept. 1981.

[25] A. Liebl. Authentication in Distributed Systems: A Bibliography. *ACM Operating Systems Review*, pages 31–41, Oct. 1993.

[26] T. F. Lunt. Access Control Policies: Some Unanswered Questions. *Computers & Security*, 8(1):43–54, Feb. 1989.

[27] B. Lyles. Requirement for Authenticated Signaling, Apr. 1994. ANSI Committee T1S1.5/94-118.

[28] J. B. Lyles and C. L. Schuba. A Reference Model for Firewall Technology and its Implications for Connection Signaling. In *Open Signaling Workshop*, Columbia University, New York, New York, Oct. 1996.

[29] J. B. Lyles and C. L. Schuba. A Reference Model for Firewall Technology and its Implications for Connection Signaling. Technical Report CSD-TR-96-073, Department of Computer Sciences, Purdue University, West Lafayette, Indiana, Dec. 1996.

[30] D. M. Martin, Jr., R. Sivaramakrishnan, and A. D. Rubin. Blocking Java Applets at the Firewall. In *5th Symposium on Network and Distributed System Security (SNDSS)*, San Diego, California, Feb. 1997. Internet Society (ISOC).

[31] P. Metzger and W. A. Simpson. *RFC-1828 IP Authentication using Keyed MD5*. Network Working Group, Aug. 1995.

[32] N. C. S. A. NCSA. Firewall User Profile. An NCSA Focus Report. Carlisle, Pennsylvania, 1997.

[33] D. M. Nessett. A Critique of the Burrows, Abadi and Needham Logic. *Operating Systems Review*, 24(2):35–38, Apr. 1990.

[34] J. Picciotto. The Design of an Effective Auditing Subsystem. In *Symposium on Research in Security and Privacy*, Oakland, California, Apr. 1987. IEEE.

[35] J. Postel, editor. *RFC-791 Internet Protocol*. Information Science Institute, University of Southern California, Sept. 1981.

[36] J. Postel, editor. *RFC-792 Internet Control Message Protocol*. Information Sciences Institute, University of Southern California, Sept. 1981.

[37] K. T. Randall, editor. *Recommendation X-811 Information Technology - Open Systems Interconnection - Security Frameworks for Open Systems: Authentication Framework*. International Telecommunications Union, 1993.

[38] K. T. Randall, editor. *Recommendation X-812 Information Technology - Open Systems Interconnection - Security Frameworks for Open Systems: Access Control*. International Telecommunications Union, 1995.

[39] M. J. Ranum. A Network Firewall. In *Proceedings of the 1st International Conference on Systems and Network Security and Management (SANS-I)*, June 1992.

[40] M. J. Ranum. Internet Firewalls — An Overview, Oct. 1993. (unpublished).

[41] M. J. Ranum. Thinking About Firewalls. In *Proceedings of the 2nd International Conference on Systems and Network Security and Management (SANS-II)*, Apr. 1993.

[42] M. J. Ranum, A. Leibowitz, B. Chapman, and B. Boyle. Firewalls-FAQ, 1994.

[43] B. Schneier. *Applied Cryptography*. John Wiley & Sons, Inc., second edition, 1995.

[44] C. L. Schuba, I. V. Krsul, M. G. Kuhn, E. H. Spafford, A. Sundaram, and D. Zamboni. Analysis of a Denial of Service Attack on TCP. In *Proceedings of the Symposium on Security and Privacy*, pages 208–223, Oakland, California, May 1997. IEEE.

[45] C. L. Schuba, E. H. Spafford, and B. Kercheval. Prototyping Experiences with Classical IP and ARP over Signaled ATM Connections. *Journal of Systems and Software*, III 1998. (to appear).

[46] T. Shimomura. IP Spoofing and Connection Hijacking. 3rd Annual Workshop on Computer Misuse and Anomaly Detection (CMAD), Jan. 1995. (presentation).

[47] K. Siyan and C. Hare. *Internet firewalls and network security*. New Riders Pub., Indianapolis, Indiana, 1995.

[48] E. H. Spafford. The Internet Worm Program: An Analysis. Technical Report CSD-TR-823, Department of Computer Sciences, Purdue University, West Lafayette, Indiana, 1988.

[49] J. G. Steiner, C. Neuman, and J. I. Schiller. Kerberos: An Authentication Service for Open Network Systems. In *Proceedings, Winter USENIX*, Dallas, Texas, 1988.

[50] D. R. Stinson. *Cryptography — Theory and Practice*. CRC Press Inc., 1995.

[51] A. Tallberg. The Property of Audit Trail. Technical Report C:252, Swedish School of Economics and Business Administration, 1992. `http://www.nan.shh.fi/NAN/Papers/AUTR92/autrtoc.htm`.

[52] A. B. Tucker, Jr. *The Computer Science and Engineering Handbook*. CRC Press Inc., 1997.

[53] T. Y. C. Woo and S. S. Lam. A Framework for Distributed Authorization (Extended Abstract only). In *Proceedings of the Conference on Computer and Communications Security*, Fairfax, Virginia, Nov. 1993. Association for Computing Machinery (ACM).

[54] T. Y. C. Woo and S. S. Lam. A Semantic Model For Authentication Protocols. In *Symposium on Research in Security and Privacy*, Oakland, California, May 1993. IEEE.

[55] T. Y. C. Woo and S. S. Lam. Authorization in Distributed Systems: A New Approach. *Journal of Computer Security*, 1993.

[56] N. Yialelis and M. Sloman. A Security Framework Supporting Domain-Based Access Control in Distributed Systems. In *4th Symposium on Network and Distributed System Security (SNDSS)*, San Diego, California, Feb. 1996. Internet Society (ISOC).