The Pervasive Trust Foundation for Security in Next Generation Networks
(A Position Paper)

Leszek Lilien
Department of Computer Science
Western Michigan University
Kalamazoo, MI
01-269-276-3116
leszek.lilien@wmich.edu

Adawia Al-Alawneh
Department of Computer Science
Western Michigan University
Kalamazoo, MI
01-269-276-3101
adawia.alalawneh@wmich.edu

Lotfi Ben Othmane
Department of Computer Science
Western Michigan University
Kalamazoo, MI
01-269-276-3101
lotfi.benothmane@wmich.edu

ABSTRACT

We propose a new paradigm—named the *Pervasive Trust Foundation* (*PTF*)—for computer security in Next Generation Networks, including the Future Internet. We start with a review of basic trust-related terms and concepts. We present motivation for using PTF as the basis for security in ISO OSI networks. The paper includes our five contributions. First, we define *trust in the small* (*TIS*) and *trust in the large* (*TIL*), where TIL is equivalent to PTF. Second, we list and contrast required and prohibited features of PTF-based systems. Third, we enumerate claims of benefits derived from using PTF. Fourth, we identify two major obstacles to PTF realization, and discuss multiple approaches to overcoming these obstacles. The more important of the two obstacles can be eliminated by showing an efficient implementation of PTF-based security. Fifth, we present an outline for the Basic Reference Model for PTF for Next Generation Networks. Summary and discussion of future work concludes the paper.

Categories and Subject Descriptors

K.6.5 [**Management of Computing and Information Systems**]: Security and Protection.

General Terms

Design, Economics, Performance, Security, Standardization.

Keywords

Distributed Systems, Future Internet, ISO 7498-2, Next Generation Networks, Pervasive Trust Foundation, Privacy, Security, Security Mechanisms, Security Services, Trust, Trust in the Large, Trust in the Small.

1. INTRODUCTION

We present a new paradigm—named the *Pervasive Trust Foundation* (*PTF*)—for using trust as the basis for security in Next Generation Networks, including the Future Internet. *Networks* considered here are as defined by the ISO OSI Reference Model, which among its 7 layers contains also the application layer; hence any distributed system is a network (an ISO OSI network).

This is a position paper rather than a research paper since most of the claims that we make are based on our opinions rather than hard theoretical or experimental results.

We limit our considerations to use of trust only for security purposes (such as authentication, authorization, access control, malicious node detection). In general, trust can also be used, e.g., for routing, data aggregation, time synchronization, and even stimulating cooperation in autonomous wireless networks [13]. Trust and reputation systems have also played an important role in arbitrary decision making in the Internet world.

The paper is organized as follows. Section 2 presents basic trust-related terms and concepts. Section 3 presents motivation for using PTF as the basis for security in ISO OSI networks, and then introduces and discusses the notions of *trust in the small* (*TIS*) and *trust in the large* (*TIL*), where TIL is equivalent to Pervasive Trust Foundation. This section also lists and contrasts required and prohibited features of PTF-based systems. Section 4 enumerates our claims of benefits derived from using PTF. Section 5 identifies two major obstacles to PTF realization, and discusses multiple approaches to overcoming these obstacles. Section 6 presents an outline for the Basic Reference Model for Pervasive Trust Foundation for Next Generation Networks. Section 7 summarizes the paper and discusses plans for future work.

2. BASIC TRUST-RELATED TERMS AND CONCEPTS

A prominent dictionary [1] defines trust as "reliance on the integrity, ability, or character of a person or thing." A more "operational" definition of trust describes it as *"the extent to which one party is willing to participate in a given action with a given partner in a given situation, considering the risks and incentives involved."* [11].

2.1 Trustor and Trustee

We define a *trustor* as an entity (human or artificial) that must decide whether to trust or not other (human or artificial) entities, which are called *trustees*.

Trust decision may involve two basic types of decisions (cf. [9]). First, in a traditional trust dilemma, a trustor decides if it can trust the trustees enough to allow them use its (trustor's) resources. For example, an online service *S* (a trustor) makes a trust decision

NSPW'10, September 21–23, 2010, Concord, Massachusetts, USA.
Copyright 2010 ACM 978-1-4503-0415-3/10/09...$10.00

allowing or not a user *U* (a trustee) to log in, which precedes using service *S* by *U*. In a reciprocal fashion, the user *U* (a trustor) using the online service *S* might need to decide if it trusts *S* enough to provide it with its login name or password; in other words, *U* must decide whether to allow *S* use *U*'s login/password "resources." (This reciprocity shows that Entity A can be a trustor and Entity B—a trustee, and at the same time B can be a trustor and A—a trustee.)

Second, in a newer version of the trust dilemma, a trustor decides if it can trust the trustees enough to use the trustees' resources. For example, a user *U* (a trustor) using an online service *S* (a trustee) decides if it trusts *S* enough to rely on *S* (that is, to use *S*'s "service resources").

Reciprocally, the online service *S* (a trustor) might need to decide if it trusts *U* (a trustee) enough to use *U*'s login/password (*U* might be mounting an impersonation attack).

2.2 Trustworthiness, Misplaced Trust and Misplaced Distrust

Solhaug *et al.* [16] define *trustworthiness* as "the *objective* probability that a trustee performs a particular action on which the interests of the trustor depend." Trustworthiness is thus contrasted with *subjective* trust.

Trust decision processes may result in a *misplaced trust* when the produced subjective trust values exceed the objective trustworthiness level. The radical case of misplaced trust is *trust naïveté*, resulting in exorbitant risks to the trustor (which, in turn, may lead to insurmountable trustor's losses). Symmetrically, trust decision processes may result in a *misplaced distrust* when the produced subjective trust values are lower than the objective trustworthiness level. The radical case of misplaced distrust is *trust paranoia*, which may result in a loss of opportunities to cooperate with highly trustworthy partners [10].

We believe that considering a totally *subjective* trust in the context of computing systems (with the possible exception of human-computer interaction and related subareas) is rather useless. We also think that the adjective "objective" used in the definition of trustworthiness by Solhaug *et al.* [16] does not mean absolute or perfect objectivity but the best objectivity that can be achieved at the current state of knowledge and technology. With this reservation, the notion of trust as used by us is a synonym of trustworthiness. In other words, we focus on using trust defined and determined as objectively as possible.

2.3 Trust in Social and Computing Systems

Trust is pervasive in social systems [2]. We constantly apply it in interactions between people, organizations, animals, and even artifacts ("Can I trust my car on this vacation trip?"). We use it instinctively and implicitly in *closed and static* systems, or consciously and explicitly in *open or dynamic* systems. An epitome for the former case is a small village, where everybody knows everybody, and the villagers instinctively use their knowledge or stereotypes to trust or distrust their neighbors. A big city exemplifies the latter case, where people use explicit rules of behavior in diverse trust relationships. A city dweller builds up trust, for instance, by asking friends or recommendation services for a dependable plumber.

If trust is so pervasive and beneficial in complex social systems, why not exploit *pervasive trust* as a paradigm in computing environments [2]? (Using pervasive trust even in non-pervasive computing is not a contradiction!) We already use trust in computing systems extensively, although usually subconsciously. Examples are users' trust-based decisions to search for reputable ISPs or e-banking sites, or to ignore emails from "Nigerians" asking for help transferring millions of dollars out of their country. The challenge for exploiting trust in computing lies in extending the use of trust-based solutions, first to artificial entities such as software agents or subsystems, then to human users' subconscious choices.

In future networks and pervasive computing environments, people will be surrounded by zillions of computing devices of all kinds, sizes, and aptitudes [2]. Most of them will have limited or even rudimentary capabilities and will be quite small, such as radio frequency identification tags and smart dust. Most will be embedded in artifacts for everyday use, or even human bodies (with possibilities for both beneficial and apocalyptic consequences).

Radically changed reality demands new approaches to computer security (and privacy). We believe that socially based paradigms, such as trust-based approaches, will play a big role in future networks as well as pervasive computing. As in social settings, solutions will vary from heavyweight ones for entities of high intelligence and capabilities (such as humans and intelligent systems) interacting in complex and important matters, to lightweight ones for less intelligent and capable entities interacting in simpler matters of lesser consequence.

2.4 Selected Trust Characteristics

The pervasive trust can be used in one of the few ways. First, it can be used *consciously*—either *explicitly* (i.e., considering trust level) or *implicitly* (i.e., *assuming* a sufficient trust level). Second, it can be used *unconsciously* (i.e., be ignored); in this case this can result in either *no* adverse effects (due to pure luck) or in adverse effects (i.e., bearing the costs of ignorance).

When considering trust, for example in the context of asking an entity for a service, the *two basic dimensions* have to be taken into account: (a) *competence*—is the entity *able* to perform the service adequately?; and (b) *intention*–is the entity *willing* to perform the service adequately?

Trust is *not binary*, not all-or-nothing. There are degrees of trust.[1]

One can*not trust everybody* but one has to *trust somebody*. Trusting everybody is naïve and can result in severe security consequences. Not trusting anybody is a paranoid behavior, with extreme costs (feeling unsecure all the time under all circumstances, always looking over one's shoulder). A completely distrusting system (even just implicitly) would be paranoid, and highly inefficient.

Trust is *bidirectional*. In a two-party interaction, say an interaction between Bob and Alice, it is not enough to consider Alice's trust in Bob; Bob's trust in Alice must be considered as well. However, in some situations we might consider on party's trust explicitly, and another party's trust implicitly. For example, a computer system's trust in a user should be considered explicitly (e.g., using access controls) while the user's trust in the computer system can be

[1] Non-binary trust is commonly used to make *binary yes-or-no decisions* [10] whether to trust (*yes*) a trustee or not (*no*). Optionally, ternary decisions can be made: *yes-no-unknown* or *yes-no-conditional(<condition>)* (cf. [17]).

implicit (the very fact that the user uses the system shows that user's trust in the system is sufficiently high).

Trust is *asymmetric*. I trust the airline pilot on my flight, but he would not trust that I can fly the airliner safely.

We can *trust artifacts* as well, not only people. A person can trust a car, a cell phone, a PDA, RFID tags in store, a smart refrigerator.

2.5 Types of Trust

A few different classifications of trust help understanding its meaning and scope. In the first classification, two types of trust are distinguished: subjective trust[2] and decision trust [12]:

1) *Subjective trust* is "the subjective probability by which an individual, A, expects that another individual, B, performs a given action on which its welfare depends."
2) *Decision trust* is "the extent to which one party is willing to depend on something or somebody in a given situation with a feeling of relative security, even though negative consequences are possible."

In another classification, we can distinguish trust on three levels (cf. [17]): (1) *infrastructure-level trust*, which is a part of the infrastructure; (early trust research concentrated on infrastructure trust [17]); (2) *service-level trust*, which underlies services available in computing systems (esp. the open services); and (3) (*user*) *community-level trust*.

Citing Jøsang *et al.* [12] (based on Grandison and Sloman's classification [9]), we can indicate the following five trust classes (with an emphasis on trust for online services):

1) *Provision trust* describes the relying party's trust in a service or resource provider. It is relevant when the relying party is a user seeking protection from malicious or unreliable service providers.
2) *Access trust* describes trust in principals for the purpose of accessing resources owned by or under the responsibility of the relying party. This relates to the access control paradigm which is a central element in computer security.
3) *Delegation trust* describes trust in an agent (the delegate) that acts and makes decision on behalf of the relying party. Acting on one's behalf can be considered a special form of service provision [9].
4) *Identity trust* describes the belief that an agent identity is as claimed. Trust systems that derive identity trust are typically authentication schemes such as X.509 and PGP.
5) *Context trust* describes the extent to which the relying party believes that the necessary systems and institutions are in place in order to support the transaction and provide a safety net in case something should go wrong. Factors for this type of trust can, for example, be critical infrastructures, insurance, legal system, law enforcement and stability of society in general.

Yet another classification [13] creates a dichotomy of direct and indirect trust. *Direct trust* is established through observations on whether the previous interactions between the subject and the agent are successful [13]. *Recommendation trust*—often determined by checking consistency between one's observations and received recommendations, or among multiple received recommendations—is a subset of direct trust [13].

Indirect trust is due to the fact that trust can be *transitive* through third parties [13]. For example, if Alice has established a recommendation trust relationship with Bob, and Bob has established a trust relationship with Yolanda, Alice can trust Yolanda to a certain degree if Bob shares with her his trust opinion (i.e., recommendation) for Yolanda. This phenomenon is called *trust propagation*. Indirect trust is established through trust propagation [13]. So is a *chain of trust*.

2.6 Trust as a Basis for Soft Security Mechanisms

Rasmusson and Janssen [14] identified two approaches to security: hard security and soft security. *Hard security* is used in relation to traditional security mechanisms (e.g., authentication, authorization, access control). It typically protects resources from malicious users by preventing access by unauthorized users.

Hard security does not work when we are not protecting resources from users, but protecting users from resources. For example, we need to protect users from overwhelming information (such as in denial-of-service attacks) or false information (such as in phishing). In such situations, *soft security* is required. It relies on trust management systems, reputation systems, and other systems using elements of social control (including "society" of artifacts, not humans).

Zheng [15] proposes using the analogous notions of hard and soft trust, as the two basic approaches to trust relationships. *Hard trust* solutions build up trust through structural and objective regulations, standards, as well as widely accepted rules, mechanisms and sound technologies (e.g., PKI and trusted computing platform) [15]. In contrast, *soft trust* solutions provide trust based on trust evaluation according to subjective trust standards, facts from previous experiences and history [15].

The two approaches can be integrated together in a given system, cooperating with and supporting each other [15]. In particular, hard trust can verify functionalities of soft trust solutions, and soft trust solutions can help in selecting suitable and complementary hard trust solutions (and determine when they should be applied).

2.7 Trust Management

A problem in using credentials is that they might be subject to diverse, uncoordinated trust decisions whether a given credential is true or not [3]. If the problem were moved to an everyday scene, we might see a situation that our driver license is accepted by some police officers and rejected by others. It is much better that all police officers accept each driver license issued by a proper state authority. Similarly in the cyberspace, we want a single authority responsible for deciding whether a given credential is true or not, which determines who and when is trusted. This problem is broadly described as trust management.

Pioneering work on trust management [18] had as its goal separation of security and trust [3]. The benefit of the separation is allowing individual systems to have different trust policies, separate from the common, global authentication and security system.

[2] The original name "reliability trust" has been changed by us since, in our opinion, it abuses the term "reliability" that has a very precise technical meaning, also in computer science.

Ruohomaa and Kutvonen [17] indicate that early forms of trust management systems, such as PolicyMaker [18], KeyNote [23], REFEREE [22] and Trust-Builder [21], began by automating authentication and authorization decisions with the help of varying sets of credentials.

Blaze *et al*. [18] defined trust management as "a unified approach to specifying and interpreting security policies, credentials, relationships which allow direct authorization of security-critical actions."

Grandison and Sloman [9] say that trust management is concerned with collecting the information required to make a trust relationship decision, evaluating the criteria related to the trust relationship as well as monitoring and re-evaluating existing trust relationships.

Jøsang *et al*. [19] state that trust management is the activity of creating systems and methods that allow relying parties to make assessments and decisions regarding the dependability of potential transactions involving risk, and that also allow players and system owners to increase and correctly represent the "reliability" (we'd say "trustworthiness") of themselves and their systems.

Cho and Swami [10] explain that trust management includes *trust establishment* (i.e., collecting appropriate trust evidences, trust generation, trust distribution, trust discovery, and evaluation of trust evidence), *trust updates*, and *trust revocation*.

Cho and Swami [10] provide an 2-page tabular summary of existing trust management schemes (in Appendix A of their paper), listing the schemes by name, methodology of collecting trust evidence, attacks targeted, performance metrics used, and other notable characteristics.

Finally, let us heed a warning of Jøsang *et al*. [19], who state: "In order to avoid misunderstanding it is always good to be as specific as possible when using the term trust management, by providing additional information about its meaning in a given context."

2.8 Reputation and Reputation Management

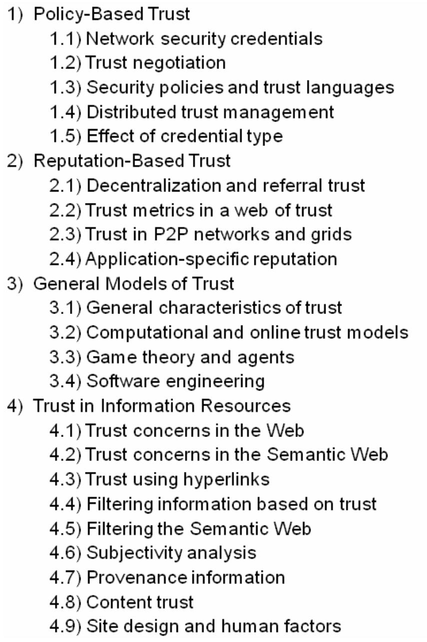
Jøsang et al. [12] define *reputation* as a collective measure of trustworthiness based on the referrals or ratings from members in a community. They nicely differentiate trust from reputation with the following "perfectly normal and plausible statements" [12]:

1) "I trust you because of your good reputation."

2) "I trust you despite your bad reputation."

They claim that Statement 2 reflects that the relying party has some private knowledge about the trustee, e.g., through direct experience or intimate relationship, and that these factors overrule any reputation that a person might have. We believe that no additional or private knowledge is necessary for Statement 2: the trustor might just be willing to take a risk, possibly expecting (if the risk pays off) large benefits. As a physical world example, a politician might decide to walk without a bodyguard through bad neighborhoods to gain popularity.

Reputation and reputation management can be components of trust and trust management, respectively (most advanced trust management systems use reputation; vide a reputation-based trust management framework presented by Conner *et al*. [20]).



1) Policy-Based Trust
    1.1) Network security credentials
    1.2) Trust negotiation
    1.3) Security policies and trust languages
    1.4) Distributed trust management
    1.5) Effect of credential type
2) Reputation-Based Trust
    2.1) Decentralization and referral trust
    2.2) Trust metrics in a web of trust
    2.3) Trust in P2P networks and grids
    2.4) Application-specific reputation
3) General Models of Trust
    3.1) General characteristics of trust
    3.2) Computational and online trust models
    3.3) Game theory and agents
    3.4) Software engineering
4) Trust in Information Resources
    4.1) Trust concerns in the Web
    4.2) Trust concerns in the Semantic Web
    4.3) Trust using hyperlinks
    4.4) Filtering information based on trust
    4.5) Filtering the Semantic Web
    4.6) Subjectivity analysis
    4.7) Provenance information
    4.8) Content trust
    4.9) Site design and human factors

Figure 1. Selected trust research issues [3].

2.9 Selected Trust Research Issues

Research on trust is very broad and multifaceted, as evidenced by the list of major trust research subareas shown in Figure 1 [3].

The report from a National Science Foundation Information and Data Management workshop session on trust, privacy, and security raises many other issues in the trust-related research area [4].

3. TRUST IN THE SMALL AND TRUST IN THE LARGE

3.1 Motivation for Pervasive Trust Foundation for Security

Table 1 compares some trust-related characteristics of the early Internet and the Future Internet;[3] the latter is an example of the Next Generation Networks.

Current networks, including Internet 2.0, already use trust. For example, users routinely use and trust certificates issued by certification and registration authorities (CAs and RAs). As another example, many systems include trusted central control systems to provide services.

However, use of trust in current networks, including Internet 2.0, is far from the proposed *Pervasive Trust Foundation* (*PTF*) approach. Trust is used implicitly (in strong trust-related assumptions) and in a piecemeal fashion (with separate security services supported by separate trust solutions).

Next Generation Networks will need to consider trust in a comprehensive, integrated, system-wide manner. We propose that

[3] The following Internet classification has been proposed [5]: (1) Internet 1.0—from 1969 till 1989; (2) Internet 2.0—from 1989 till now; and (3) Internet 3.0—the Future Internet (an example of the Next Generation Networks).

Next Generation Networks do it to exploit many benefits of the PTF approach (as listed in Section 4).

Table 1. Motivation for Pervasive Trust Foundation

	Internet 1.0	Internet 3.0
Designed for	**Closed communities and systems**	**Opened communities and systems**
Prevalent trust level	**High** (Low-trust situations very unusual—low risk)	**Low** (Low-trust situations very common—high risk)
Can use no/weak security services (SSs)	**Yes** (High trust => low risks => null/weak SSs suffice)	**No** (Low trust => high risks => strong SSs needed)
Dealing with low-trust situations	**"Patching" low-trust subsystems** (Efficiency / trustworthiness still acceptable)	**System-wide trust foundation** (Designed-in solutions necessary for efficiency / trustworthiness)

3.2 Security Services

The following set of *security services* (*SSs*) is defined by the standard ISO 7498-2[4] (and presented here in our preferred order):

1) *Confidentiality*—the property that information is not made available or disclosed to unauthorized individuals, entities, or processes.
2) *Integrity*—the property that data has *not* been altered or destroyed in an *un*authorized manner.
3) *Availability*—the property of being accessible and useable upon demand by an authorized entity.
4) *Authentication*—the corroboration that an entity is the one claimed, and the source of data received is as claimed.
5) *Access Control*—the prevention of unauthorized use of a resource. It includes the prevention of use of a resource [by an authorized entity] in an unauthorized manner.
6) *Non-repudiation*[2]—the prevention of entities' denial to be involved in all or part of a communication.
7) *Notarization*—the registration of *data* with a trusted third party that allows [to assure] the accuracy of its characteristics such as content, origin, time, and delivery.

The first 3 SSs, with the acronym *CIA*, constitute the set of *classical security services* [7]. The next 3 SSs, with the acronym *AAN*, are the first extension of the classical set of SSs. The last SS, with the acronym *N*, was added as the second extension of the set of SSs. The full set of SSs can be referred to by its combined acronym as *CIA-AAN-N*. (It is quite possible that additional SSs will appear in the Next Generation Networks.)

3.3 Trust in the Small and Trust in the Large

We need to introduce a fundamental distinction in approaching trust in computing systems, namely, the distinction between trust in the small and trust in the large.

Trust in the small (*TIS*) supports *small* subsets of SSs, *individual* SSs or, in the worst case, only *portions* of individual SSs. In contrast, *trust in the large* (*TIL*) supports either *all* SSs, or—at least—*large* sets of SSs.

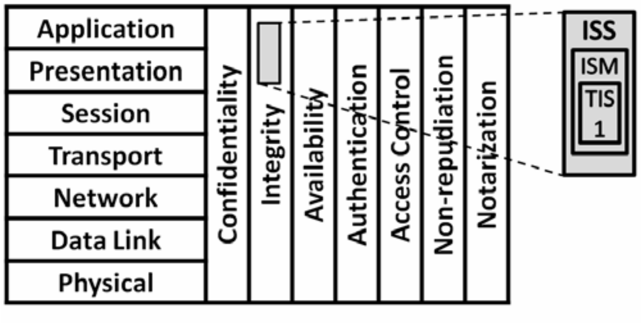


Figure 2. Block diagram illustrating use of a specific trust-in-the-small support (TIS1) by Integrity Security Service.

(TIS1 is used directly by Integrity Security Mechanism or ISM underlying Integrity Security Service or ISS.)

3.4 Sample TIS and TIL Implementations for Security Services

We show here a block diagram for an implementation of a sample SS, an Integrity Security Service (represented by "I" in the set CIA-AAN-N).

Figure 2 shows the *Integrity Security Service* (*ISS*), implemented on top of an *Integrity Security Mechanism* (*ISM*). In turn, ISM is supported by TIS1 (trust in the small 1). TIS1 is implemented within ISS and not used by any other SS.

We believe that ISM should be *fully* supported by TIL (trust in the large). By "fully supported" we mean that no trust add-on's, patches, etc., should be used to support ISM via TIL1, an interface to TIL (cf. Figure 3).

TIL is proposed here as the trust foundation for the Next Generation Networks. The terms TIL and PTF (Pervasive Trust Foundation) are synonymous.

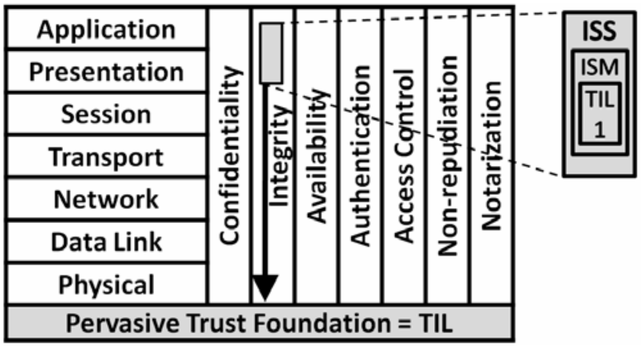


Figure 3. Block diagram illustrating use of the trust-in-the-large support (TIL1) by Integrity Security Service.

(TIL1 is an interface to TIL/PTF, as indicated by the solid vertical arrow. TIL1 is used directly by Integrity Security Mechanism or ISM underlying Integrity Security Service or ISS.)

3.5 Required and Prohibited Features of PTF-based Systems

Table 2 lists and contrasts required and prohibited features of PTF-based systems.

[4] All definitions and descriptions are *exactly* as they appear in the ISO 7498-2 [6], unless noted otherwise.

Table 2. Required and prohibited features of PTF-based systems

Features that PTF-based system *must* exhibit	Features that PTF-based system must *not* exhibit
Trust is pervasive (This means that each interaction in the system is based on trust; most "local" interactions may rely on implicit trust.)	Trust is *not* pervasive (This means that some interactions ignore trust altogether; they either trust or distrust blindly.)
Trust support is system-wide ("Trust in the Large" or TIL. See Note 1.)	Trust support is at the *subsystem or lower* level ("Trust in the Small" or TIS. See Note 2.)
Trust is designed-in into (new or completely redesigned) networks	Trust is an add-on for (new or legacy networks) networks
Risk is an independent factor, not a component of trust	Risk is a component of trust, not an independent factor
Trust is viewed in a generic way. (Trust information is available to any entity that wants to use it for any security purposes, or for any other purposes outside of the scope of this paper).	Trust is viewed in a specific way. (Trust information is available only to entities that want to use it for specific needs, such as authentication, or ability to pay for purchases; cf. [9].)
Dynamic form of trust is considered. (This requires monitoring and re-evaluation of trust, adapting trust establishment and maintenance to the changing conditions of the environment in which trust decisions are made; cf. [9].)	Only static form of trust is considered.
Trust can be negotiated. (An entity can negotiate with the system to ask for trust-based permissions or capabilities other than the predefined ones; cf. [9]. See Note 3.)	Trust cannot be negotiated.
Trust is always verified, never assumed blindly. (Implicit trust is allowed, but only after verification that assuming it is justifiable. See Note 4.)	Trust is sometimes assumed blindly. (Esp. assumed trust is never or not always verified.)
Trusted "anchors" (if used) are subject to continuous verification. (These anchors can be composed of hardware, firmware and software. Continuous verification starts with an extensive initial verification of trustworthiness at deployment, and is performed continuously to assure that the anchors were not subject to any tampering.)	Trusted "anchors" (if used) are not subject to continuous verification. (They are verified only at deployment, or at intervals too long to assure lack of tampering with them.)

Notes for Table 2

1) System-wide trust support is a "subfeature" of the pervasive trust feature (from the preceding row). It has been added as a distinct row of the table due to its importance.
2) TIS, though inferior to TIL, is an improvement over the past alternatives where trust decisions were hard-coded into applications. Hard coding increases application complexity, hampers the ability to adapt to dynamic trust changes, and reduces flexibility when setting up new trust relationships [9].
   TIS allows for limited separation of the application's purpose and its trust management framework, thus offering a more scalable and flexible solution, especially for distributed or pervasive/ubiquitous environments [9].
3) Typical examples of predefined permissions or capabilities are resource access permissions (incl. authentication and authorization).
4) In particular, trusting oneself implicitly can be assumed only in the intention dimension of trust. In contrast, trusting oneself implicitly in the competence dimension must be very carefully verified; too often entities tend to overestimate their own capabilities.

None of the trust management systems (TMSs) known to us possesses all features that PTF-based system must exhibit, while simultaneously avoiding all features that PTF-based system must be free of. For example, PolicyMaker [18], KeyNote [23], REFEREE [22] and Trust-Builder (a negotiation architecture for sensitive credential exchange) [21] do not re-evaluate trust dynamically based on available information.

4. BENEFITS OF PTF/TIL

We claim that PTF will result in the following benefits or avoiding the following disadvantages.

Claim 1: Security *without* trust (not based on trust) is more difficult to achieve than security *with* trust (based on trust).

Ignoring trust issues in security introduces problems that considering trust would prevent.

Claim 2: Confusing *trust in the small (TIS)* with *trust in the large (TIL)* leads to denying the need for a pervasive trust foundation.

Some perceive trust in only one of two fragmentary ways. First, they might see "TIS/no TIL," i.e., see only TIS and don't see TIL at all (which makes their trust perception non-comprehensive, local, and limited). Second, they might incorrectly perceive "TIS=TIL," i.e., perceive TIs and TIL as identical.

Both fragmentary perceptions lead to denying the broader role of trust provided by TIL, which, is the true foundation for security (that is, "PTF = TIL").

Since Trust in the Large (TIL) is the foundation for security, it is the focus of this position paper. However, an attention is paid to Trust in the Small (TIS) wherever necessary.

Claim 3: Ignoring trust (both TIL and TIS) leads to high risks.

Claim 4: Trust can be used implicitly only after making a *conscious* decision that there is a sufficient trust level.

Claim 5: Using TIL is necessary for comprehensive and consistent consideration of trust by any Security Service (SS) serving any network layer.

Claim 6: Context-dependent trust level that a user has for a given environment should decide how *strong* SS is needed to satisfy a given SS request.

It should be clear that the required strength of a security service (SS) is a function of a user's perceived trust level, that is:

$$strength\ (SS) = f\ (userTrustLevel, \dots)$$

If a user highly trusts the environment, she will accept a weak SS. If her trust level is low, she will demand a strong SS.

Similarly, the required strength of a security mechanism (SM)—supporting a given SS—is also a function of a user's perceived trust level, that is:

strength (SS) = f (userTrustLevel, …)

Claim 7: PTF provides a trust value for any new human and artificial system component.

The system will have to provide trust ratings for any human/artificial system component, including a new one, never seen by the system before. This means that the system has no first-hand experience (history) recorded for the component. The system will have to rely on foreign credentials (digital signatures, certificates, etc.) and second-hand recommendations or reputation ratings.

Claim 8: PTF improves consistency and fairness of trust values and, hence, trust decisions.

Claim 9: PTF facilitates higher trust service availability and overcoming resource limitations.

This is due to the distributed nature of trust management services.

Claim 10: PTF increases network efficiency.

Extended trust relationships facilitate collaboration among network components, leading to increased efficiency and utilization of network resource and capabilities.

5. OBSTACLES TO PTF REALIZATION AND LOWERING THEM

This section first presents obstacles to realization of the Pervasive Trust Foundation approach, and then discusses ways to eliminate or at least reduce the obstacles.

5.1 Obstacles to PTF Realization

The two major obstacles to PTF realization, that are strongly intertwined, are:

1) Getting a wide acceptance for the PTF principle;
2) Finding an efficient PTF implementation.

Overcoming the first obstacle is more difficult since both the "for PTF" and "against PTF" positions are based on belief, not proofs (neither positive nor negative "proofs" exist). However, overcoming the second obstacle seems more critical. Fortunately, it is more tangible so dealing with it should be easier. It requires "only" demonstrating an efficient implementation of the PTF (i.e., of the TIL). It should also be more fruitful since overcoming it should at least to lower, if not eliminate, the first obstacle.

5.2 Finding an Efficient PTF Implementation

As we have indicated, the second obstacle, claiming that PDF is (and always will be) too expensive to be practical, is the major argument against PDF.

To counter this argument, we identify two categories of approaches to reducing the costs of PTF implementations:

1) Inherent cost-saving PTF properties;
2) Additional cost-saving approaches and techniques for PTF-based security subsystems.

The categories are discussed in the first two sub-subsections.

Comparing the proposed PTF approach with the traditional "no-PTF" approach requires considering the costs imposed by the latter. They are shown in the third sub-subsection.

5.2.1 Inherent Cost-saving PTF Properties

The inherent cost-saving PTF properties result in the following benefits of using PTFs:

1) *Avoiding excessively strong ("heavy") SSs*: Weaker SSs are less expensive then stronger SS. We reduce PTF costs by using SSs no stronger than required for a given *userTrustLevel*, which is facilitated by the *strength* (SS_i) = *f* $(userTrustLevel_i$ *, …)* dependencies.

   In other words, we can avoid the *strongest-fits-all* (the one-size-fits-all) approach. Instead of using expensive strong SSs in all situations, weaker and less-expensive SSs can be used in high-trust situations. Strong SSs are necessary only for low-trust situations.
2) *Reducing inconsistent trust views*: Comprehensive trust view provided by PTF reduces penalties (be they material or not) due to *inconsistent* (partial) trust views. Without PTF, different SSs/SMs rely on partial and potentially inconsistent trust views provided by their TIS modules.
3) *Reducing inaccurate trust views*: Comprehensive trust view provided by PTF reduces penalties due to *inaccurate* trust views. Without PTF, different SSs/SMs rely on partial, hence less accurate, trust views provided by their TIS modules.
4) *Exploiting economies of scale in trust management*: Integrated trust management in PTF reduces overall costs thanks to the economies of scale; they are obtained by avoiding replication of trust management efforts in different SSs (or even different SMs within a given SS). Without PTF, TIS modules underlying different SSs/SMs replicate each others' trust management efforts.
5) *Utilizing comprehensive PTF feedback*: PTF knows trust levels provided by all system entities. Hence, PTF can provide valuable comprehensive feedback to users. In contrast, feedback from TISs is *not* comprehensive (only partial).
6) *Gaining from PTF-based service filtering*: PTF can facilitate service discovery that includes user-provided trust threshold as one of the (QoS) search criteria. This can simplify and speed up the search (by filtering only services with trust levels above the threshold).

5.2.2 Additional Cost-saving Approaches for PTF-based Security Subsystems

The additional cost-saving approaches for PTF-based security subsystems include the following:

1) *Selecting all and only useful trust aspects needed for the system:* Using the trust paradigm requires that one carefully selects all and only those useful trust aspects needed for the PTF-based system being designed. Otherwise, either flexibility or performance will suffer.
2) *Avoiding excessive or insufficient demands for evidence or credentials:* Unwarranted demands for evidence or credentials render trust-based interactions laborious and

uncomfortable, while insufficient requirements brand them too lax. (In the latter case, who wants to be friends with someone who befriends crooks and thieves?)

Thus, we need to optimize demands for evidence or credentials, avoiding asking for too much as well as being careful not to ask for too little.

3) *Avoiding excessive reliance on explicit trust:* Excessive reliance on explicit trust relationships hurts performance. For example, modules in a well-integrated system should rely on implicit trust (just as villagers in their "closed system" do). In a crowd of entities, only some communicate directly, so only they are *candidates* for using explicit trust; some of them might turn out to need only implicit use of trust.

   As mentioned, we must avoid paranoia of asking for an exceedingly high level of trust.

4) *Leveraging enclaves that rely on implicit trust:* We define an *enclave* as an area under the same Security Manager. Both cost-saving techniques proposed for this approach start with identifying high-trust enclaves (such as intranets in a user's company).

   Technique 1 calls for using *implicit* trust in high-trust enclaves wherever and whenever possible.

   Technique 2 calls for using *weak SSs and SMs* in high-trust enclaves, which reduces costs when compared to regular/generic SSs and SMs.

   Let us consider an analogy, looking at a city of law (with a very capable sheriff) in the Wild West. The sheriff requires depositing guns at a checkpoint before entering the city. The city of law is a high-trust enclave. People in the city feel secure even without guns: their perception of a high trust level means feeling secure even without a gun. Once outside of the city limits, people perceive a low trust level and to feel secure despite it, they carry a gun. The city corresponds to a high-trust enclave, and leaving the city corresponds to entering a low-trust enclave.

   In our terms, we can summarize the above example by saying that security in a high-trust enclave (the city) is assured even with weak SSs/SMs (without a gun), while security in a low-trust enclave (outside of the city) can be provided only with strong SSs/SMs (having a gun).

5) *Representing a lower-trust entity by a high-trust local representative*: This approach allows for representing (vouching for) a lower-trust (possibly global) entity *E* by its local high-trust *representative* R_E.

   We see at least two classes of representatives. First, a special local *insurer* entity could be a representative. The insurer takes responsibility for any potential losses caused by *E*, and compensates them when they occur. (The forms of compensation for a user could be monetary, additional free services, purchase credits, etc.).

   Second, a special local *lawyer* entity could be a representative. The lawyer assures that any potential losses caused by *E* will be eventually compensated by *E*.

   A question arises if insurers and lawyer entities are not simply traditional trusted third parties (TTPs). The answer is that they could be viewed as a subcategory of TTPs: we propose that they be local entities, i.e., the entities from the user's enclave. Using a *local* representative (vs. a *remote* TTP) will result in cost savings.

   There is one more difference between TTPs and our representatives. The latter do more than just assure *E*'s identity (as TTPs typically do). As we said, an insurer takes over covering potential losses, and a lawyer assures (or at least increases the chances) that losses will be covered by the perpetrator.

6) *Vouching for a lower-trust entity by a subset of entities from its enclave*: This approach permits vouching for a lower-trust entity by a subset of entities from its enclave.

   Depending on the required trust level, vouching for an entity from an enclave might require from just one to all other entities from the enclave. Other variants might call for vouching by some high-trust entities, by all high-trust entities, etc.

7) *Using off-line trust to inform PTF*: This approach allows for using off-line trust information (e.g., community trust information) for determining entity's level of trust. An example is utilizing a *trust graph* generated by and available from a *trusted* social networking site (not just any social networking site). For computer professionals this might mean using a trust graph available from a social networking site for the members of the ACM.

8) *Raising trust barrier for user admission*: In this approach, users (either human or artificial) need to satisfy admission conditions, posses and admission tickets, etc., before being permitted to use a network.

   Possible mechanisms used here could be, for example, based on authentication: either an *individual* authentication (ID, certificate, etc.), or a *group* authentication (group membership indicated by certificates, "insurance" tokens, etc.).

5.2.3 Indicating Costs of the "No-PTF" Approach

Costs accrue not only on the side of the PTF approach. They do also on the side of the traditional "no-PTF" approach. (The "no-PTF" approach includes ignoring trust altogether as well as considering TIS only.)

If PTF (TIL) is not implemented, costs might increase due to replacing a more efficient TIL-based approach with collections of TISes. The costs are higher for less-coordinated and smaller (more fragmented) collections of TIS-based implementations.

If all costs for TIS-based as well as TIL-based implementations are properly calculated, we believe that the latter will turn out to be considerably higher. In other words, we believe that costs of all multiple partial and separate TISes will significantly exceed the costs of a single comprehensive TIL-based solution.

6. BASIC REFERENCE MODEL FOR PTF AND ITS ISO OSI STRUCTURE

6.1 Reference Model for Trust Foundation for Next Generation Networks

Figure 4 shows the Basic Reference Model for PTF for Next Generation Networks (to be referred to as *PTF4NGN* BRM). It shows PTF as the basis (the foundation) underlying the ISO OSI Layers (from Physical to Application Layer) as well as the SSs.

Figure 4 also shows relationships between the OSI Layers and SSs (showing the *potential* for use of any SS by any layer).

Figure 5 shows the alternative *PTF4NGN* BRM. It shows PTF as pervading, this time, the system layers—Hardware, OS, Network, Application, and User—as well as the same set of SS Groups as before (CIA-AAN-N).

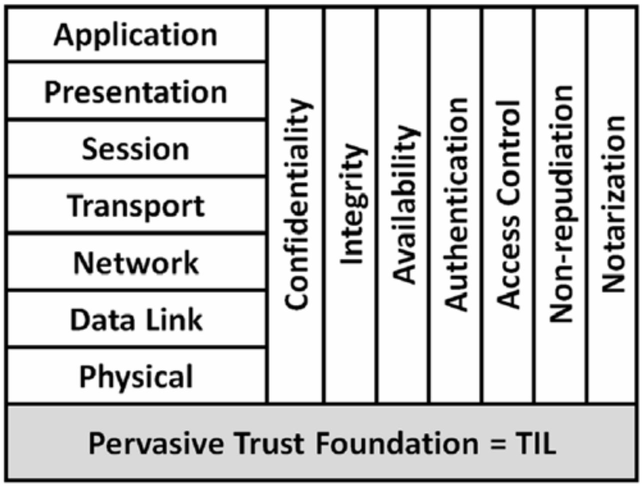


Figure 4. The Basic Reference Model of Pervasive Trust Foundation for Next Generation Networks (PTF4NGN BRM).

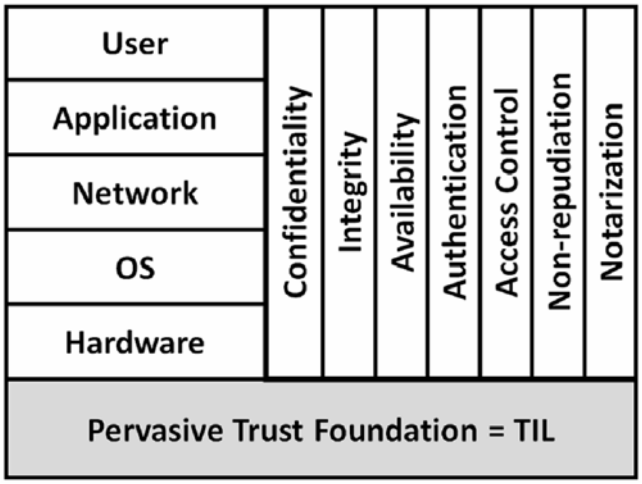


Figure 5.The alternative PTF4NGN BRM.

6.2 Placement of SSs in ISO OSI Layers

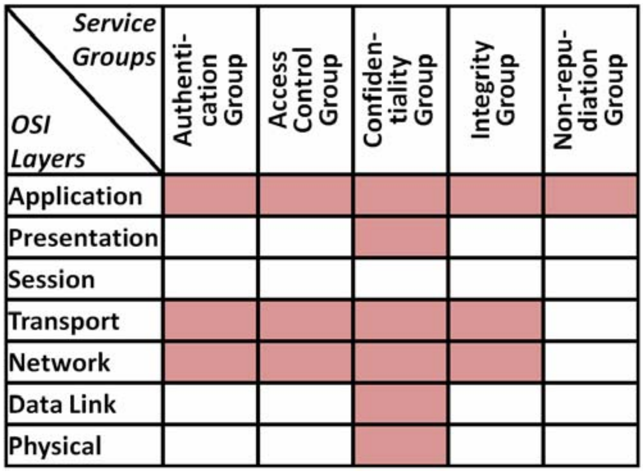
We have presented so far security services (SSs) defined by ISO. Actually, we can define *SS groups*, with each group including one or more related SSs—as defined by us earlier. They are considered *groups* now since, in general, they contain more than a single SS.

Figure 6 shows the proposed placement of SS groups within the ISO OSI network layers.

Figure 7 extends Figure 6 by showing the proposed placement of not just SSs but *SS groups* within ISO OSI network layers.

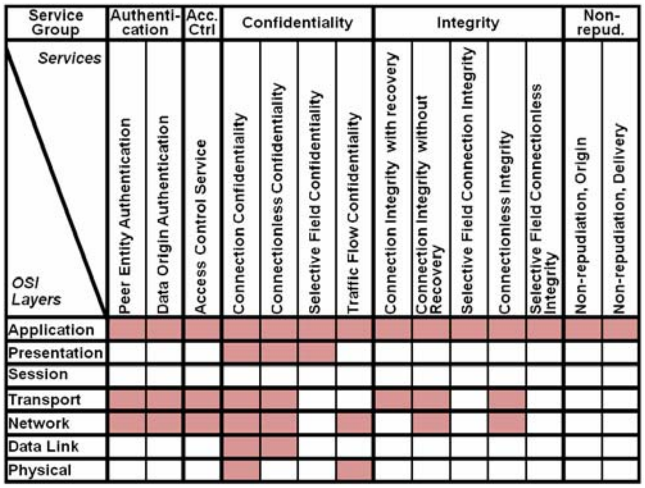
The SS groups and their components (in the order in which they are shown in Figure 7) are:

1) The *authentication service group* (defined earlier). This group includes two SSs:
   a) *(Communication) peer entity authentication*—the corroboration that a peer entity in an association is the one claimed;
   b) *Data origin authentication*—the corroboration that the source of data received is as claimed.
2) The *access control service group* (defined earlier).



Service Groups / OSI Layers	Authenti-cation Group	Access Control Group	Confiden-tiality Group	Integrity Group	Non-repu-diation Group
Application	🟥	🟥	🟥	🟥	🟥
Presentation			🟥		
Session					
Transport	🟥	🟥	🟥	🟥	
Network	🟥	🟥	🟥	🟥	
Data Link			🟥		
Physical			🟥		

Figure 6. Proposed placement of SS groups in ISO OSI layers.



Service Group	Authenti-cation		Acc. Ctrl	Confidentiality				Integrity					Non-repud.	
Services / OSI Layers	Peer Entity Authentication	Data Origin Authentication	Access Control Service	Connection Confidentiality	Connectionless Confidentiality	Selective Field Confidentiality	Traffic Flow Confidentiality	Connection Integrity with recovery	Connection Integrity without Recovery	Selective Field Connection Integrity	Connectionless Integrity	Selective Field Connectionless Integrity	Non-repudiation, Origin	Non-repudiation, Delivery
Application	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥	🟥
Presentation				🟥	🟥	🟥								
Session														
Transport	🟥	🟥	🟥	🟥	🟥			🟥	🟥		🟥			
Network	🟥	🟥	🟥	🟥	🟥		🟥		🟥		🟥			
Data Link				🟥	🟥									
Physical				🟥			🟥							

Notes:

1) SSs at the session layer add no benefits over SSs provided at higher or lower layers.
2) SSs at the presentation layer are not shown in Table 2 in ISO 7498-2 [6], p.16 (probably an error).
3) Availability and notarization that belong to CIA-AAN-N are not considered by ISO-7498-2.

Figure 7. Proposed placement of SSs in ISO OSI network layers (the SS-per-layer view).

(A slightly modified Table 2 from [6].)

3) The *confidentiality service group* (defined earlier). This group includes four SSs:

   a) *Connection confidentiality*—provides for the confidentiality of all *n*-user-data on an *n*-connection;

   b) *Connectionless confidentiality*—provides for the confidentiality of all *n*-user-data in a single connectionless *n*-SDU (Service Data Unit);

   c) *Selective field confidentiality*—provides for the confidentiality of selected fields within the *n*-user-data on an *n*-connection or in a single connectionless *n*-SDU;

   d) *Traffic flow confidentiality*—provides for the protection of the information which might be derived from observation of traffic flows.

4) The *integrity service group* (defined earlier). This group includes five SSs:

   a) *Connection integrity with recovery*— provides for the integrity of all *n*-user-data on an *n*-connection and detects any modification, insertion, deletion, or replay of any data within an entire SDU sequence (with recovery attempted);

   b) *Connection integrity without recovery*— the same as above but with no recovery attempted;

   c) *Selective field connection integrity*— provides for the integrity of selected fields within the *n*-user-data on an *n* -SDU transferred over a connection and takes the form of determination of whether the selected fields have been modified, inserted, deleted or replayed;

   d) *Connectionless integrity*—provides for the integrity of a single connectionless SDU and may take the form of determination of whether a received SDU has been modified *(*additionally, a limited form of detection of replay may be provided);

   e) *Selective field connectionless integrity*—provides for the integrity of selected fields within a single connectionless SDU and takes the form of determination of whether the selected fields have been modified;

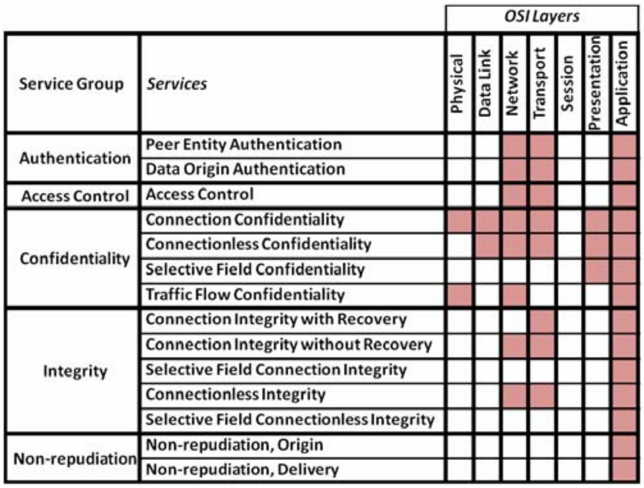
5) The *non-repudiation service group* (defined earlier).

   This group includes two SSs:

   a) *Non-repudiation, origin*—the recipient of data is provided with proof of the origin of data (this will protect against any attempt by the *sender* to falsely deny sending the data or its contents);

   b) *Non-repudiation, delivery*—the sender of data is provided with proof of delivery of data (this will protect against any subsequent attempt by the *recipient* to falsely deny receiving the data or its contents).

Figure 7 is complemented by Figure 8: while the former shows the SS-per-layer view of the proposed placement of SSs in ISO OSI network layers, the latter shows the layer-per-SS view of this placement.



		OSI Layers						
Service Group	*Services*	Physical	Data Link	Network	Transport	Session	Presentation	Application
Authentication	Peer Entity Authentication			■	■			■
	Data Origin Authentication			■	■			■
Access Control	Access Control			■	■			■
Confidentiality	Connection Confidentiality	■	■	■	■		■	■
	Connectionless Confidentiality		■	■	■		■	■
	Selective Field Confidentiality						■	■
	Traffic Flow Confidentiality	■		■				■
Integrity	Connection Integrity with Recovery				■			■
	Connection Integrity without Recovery			■	■			■
	Selective Field Connection Integrity							■
	Connectionless Integrity			■	■			■
	Selective Field Connectionless Integrity							■
Non-repudiation	Non-repudiation, Origin							■
	Non-repudiation, Delivery							■

Notes:

1) SSs at the session layer add no benefits over SSs provided at higher or lower layers.

2) SSs at the presentation layer are not shown in Table 2 in ISO 7498-2, p.16 (probably an error).

3) Availability and notarization that belong to CIA-AAN-N are not considered by ISO-7498-2.

Figure 8. Proposed placement of SSs in the ISO OSI network layers (the layer-per-SS view).

6.3 Security Mechanisms (SMs) for SSs

Security mechanisms (*SMs*) are the means of implementing SSs.

The list and definitions of SMs given by the ISO-7498-2 standard [6] is as follows:

1) *Encipherment*—the cryptographic transformation of data to produce ciphertext (the semantic content of the resulting data is not available).

2) *Digital signature*—data appended to, or a cryptographic transformation of, a data unit that allows a recipient of the data unit to prove the source and integrity of the data unit and protect against forgery, e.g., by the recipient.

3) *Access control*—these mechanisms use the authenticated identity of an entity, or information about the entity, or capabilities of the entity in order to determine and enforce the *access rights* of the entity.

4) *Data integrity*—these mechanisms are used to provide the integrity of a single data unit or field, and the integrity of a stream of data units or fields.

5) *Authentication exchange*—mechanism intended to ensure the identity of an entity by means of information exchange.

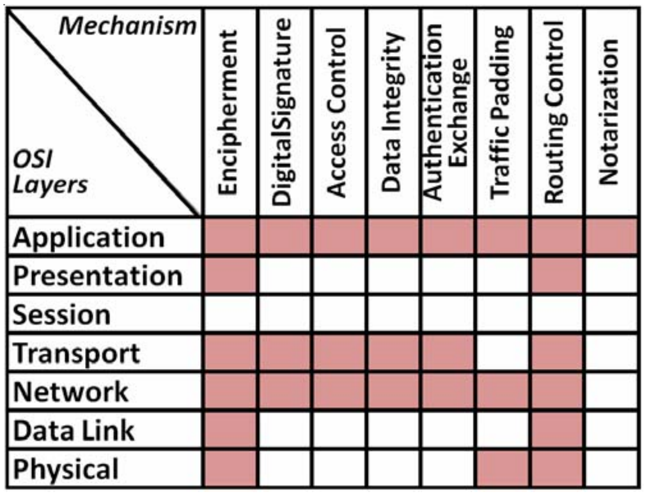
6) *Traffic padding*—the generation of spurious instances of communication, spurious data units and/or spurious data within data units.

7) *Routing control*—the application of rules during the process of routing so as to chose or avoid specific networks, links or relays.

8) *Notarization*—the registration of *data* with a trusted third party that allows the later assurance of the accuracy of its characteristics such as content, origin, time, and delivery.

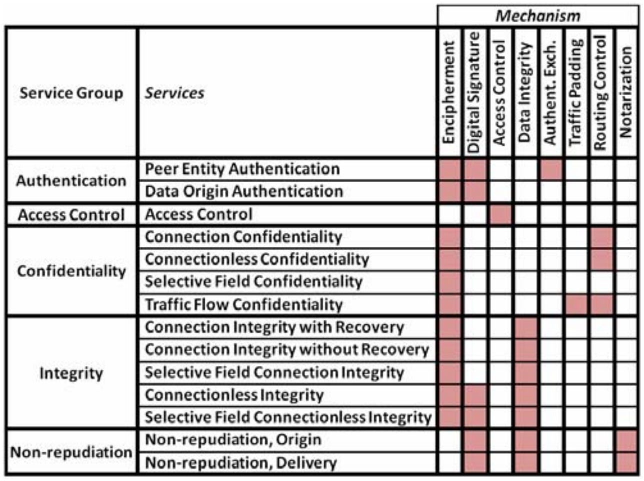
An alternative list of SMs is as follows [8]:

1) Cryptography
2) Cryptanalysis
3) Message authentication code and hash-functions
4) Authentication and passwords
5) Public key cryptography and digital signatures
6) IPSec
7) TLS
8) Firewalls
9) Digital Certificates
10) Intrusion Detection Systems



Mechanism / OSI Layers	Encipherment	DigitalSignature	Access Control	Data Integrity	Authentication Exchange	Traffic Padding	Routing Control	Notarization
Application	■	■	■	■	■	■	■	■
Presentation	■						■	
Session								
Transport	■	■	■	■	■		■	
Network	■	■	■	■	■	■	■	
Data Link	■						■	
Physical	■					■	■	

Figure 9. Proposed placement of SMs in the ISO OSI network layers (the SM-per-layer view).



		Mechanism							
Service Group	Services	Encipherment	Digital Signature	Access Control	Data Integrity	Authent. Exch.	Traffic Padding	Routing Control	Notarization
Authentication	Peer Entity Authentication	■	■			■			
	Data Origin Authentication	■	■						
Access Control	Access Control			■					
Confidentiality	Connection Confidentiality	■						■	
	Connectionless Confidentiality	■						■	
	Selective Field Confidentiality	■							
	Traffic Flow Confidentiality	■					■	■	
Integrity	Connection Integrity with Recovery	■			■				
	Connection Integrity without Recovery	■			■				
	Selective Field Connection Integrity	■			■				
	Connectionless Integrity	■	■		■				
	Selective Field Connectionless Integrity	■	■		■				
Non-repudiation	Non-repudiation, Origin		■		■				■
	Non-repudiation, Delivery		■		■				■

Figure 10. Use of SMs by SSs.

6.4 Placement of SMs in ISO OSI Layers

Figure 9 shows the proposed placement of SMs (as given by the ISO-7498-2 standard [6]) within the ISO OSI network layers.

6.5 Use of SMs by SSs

Figure 10 shows use of SMs by SSs.

7. SUMMARY AND FUTURE WORK

We propose a new paradigm, the *Pervasive Trust Foundation* (*PTF*), for computer security in Next Generation Networks, including the Future Internet.

After an overview of the needed background for the notion of trust in social and computing systems, we present motivation for using PTF as the basis for security in ISO OSI networks. Next, the paper presents our five contributions. First, we define *trust in the small* (*TIS*) and *trust in the large* (*TIL*), where TIL is equivalent to PTF. We also present block diagrams illustrating sample TIS and TIL implementations of a security service (defined by the ISO 7498-2 standard).

Second, we list and contrast features that PTF-based systems must and must not possess. None of the trust management systems known to us possesses all features that PTF-based system must exhibit, while simultaneously avoiding all features that PTF-based system must be free of.

Third, we enumerate benefits that, in our opinion, are derived from using the PTF paradigm. They include improvement of trust decisions, higher trust service availability and overcoming resource limitations, as well as increased network efficiency.

Fourth, we identify two major obstacles to PTF realization, and discuss multiple approaches to lowering them. The more critical of the two obstacles can be eliminated by showing an efficient implementation of PTF-based security. We identify two categories of approaches to reducing the costs of PTF implementations: (1) inherent cost-saving PTF properties; and (2) additional cost-saving approaches and techniques for PTF-based security subsystems. We also show that the traditional "no-PTF" approach imposes its own costs that must be considered in comparing the PTF and no-PTF approaches. We expect that if all costs for both TIS-based and TIL-based implementations are properly calculated, the latter will be more expensive. This will be a topic of our future investigations (starting with simulations).

Fifth, we present an outline for the Basic Reference Model for PTF for Next Generation Networks. We show how security service groups and individual security services should be placed within the ISO OSI Reference Model. Security services are implemented by *security mechanisms* (defined by ISO 7498-2), so we also show how security mechanisms should be placed within the ISO OSI Reference Model, and which security mechanisms are needed to implement each of the five security services.

Our other future work will include the following problems. First, we need to identify and provide proper incentives or penalties that will foster trust relationships. This includes: avoiding perverse incentives (e.g., Smith pays for security but Jones reaps the benefits [4]); and taking responsibility for trust in interactions (the "seller" is ultimately responsible for deciding on the degree of trust required to *offer* a service, and the "buyer" is ultimately responsible for deciding on the degree of trust required to *accept* a service).

Second, we have to investigate whether we can build trusted systems from untrustworthy components, and—if we can—how well we can do it. (We succeeded with analogous achievement in the field of computer reliability.)

Third, we need to consider in detail trust for each security mechanism.

Fourth, we believe that adding PTF-based privacy (to PTF-based security) should be attempted, with Privacy Services and Privacy Mechanisms (PSs and PMs) analogous to Security Services and Security Mechanisms (SSs and SMs).

8. ACKNOWLEDGMENTS

NSPW is a rarity: a true workshop, in which an accepted workshop paper is shepherded to produce its improved pre-workshop version available for the internal use by the workshop participants; then, the pre-workshop version is improved again to produce this final, version. We are grateful to many researchers for their often anonymous help at these two stages of paper improvement. We apologize if we misunderstood or missed any of the comments.

In producing the pre-workshop version, we were helped by anonymous NSPW 2010 reviewers; and by Dr. Michael Locasto, who shepherded updating the submission version of this paper to produce its improved pre-workshop version.

The final version has benefitted from numerous comments and suggestions we received during our presentation at the workshop. The excellent scribe notes from Dr. Matt Bishop and Dr. Michael Locasto assured that we could review all remarks expressed by workshop participants. In addition, Dr. Steven Greenwald provided us with his important remarks in writing.

This research was supported in part by the NSF grants IIS-0242840 and IIS-0209059, and Western Michigan University Faculty Research and Creative Activities Award (FRACAA).

9. REFERENCES

[1] 2010. Trust. *American Heritage Dictionary of the English Language*, Houghton Mifflin. Online at: http://education.yahoo.com/reference/dictionary/entry/trust

[2] Bhargava, B., Lilien, L., Rosenthal, A., and Winslett, M. 2004. Pervasive Trust. *IEEE Intelligent Systems* 19, 5 (Sept.-Oct. 2004), 74-77.

[3] Artz, D., and Gil, Y. 2007. A Survey of Trust in Computer Science and the Semantic Web. *Web Semantics: Science, Services and Agents on the World Wide Web* 5, 2 (Jun. 2007), 58-71.

[4] Bhargava, B., Farkas, C., Lilien, L., and Makedon, F. 2003. Trust, Privacy, and Security: Summary of a Workshop Breakout Session, the National Science Foundation Information and Data Management (IDM) Workshop held in Seattle, Washington. Sep. 14–16, 2003. Technical Report 2003-34. Center for Education and Research in Information Assurance and Security (CERIAS), Purdue University. Online at: http://www.cerias.purdue.edu/tools_and_resources/bibtex_archive/archive/2003-34.pdf

[5] 2009. Verbal communication with participants. *NSF Future Internet Architecture Summit* (Washington, D.C., Oct. 2009).

[6] ISO/IEC, 1991. ISO/IEC DIS 10181-2, May 1991, Information Technology - Open Systems Interconnection - Security Frameworks in Open Systems - Part 2: Authentication Framework. ISO. Used to be online at: http// www.iso.org/iso/catalogue_detail.htm?csnumber=14256

[7] Pfleeger, C.P., and Pfleeger, S.L. 2007. *Security in Computing.* Fourth Edition, Prentice Hall. Upper Saddle River, NJ.

[8] Fischer-Hübner, S., and Hedbom, H. 2008. Benefits of Privacy-Enhancing Identity Management. *Asia-Pacific Business Review* IV, 4 (Oct.-Dec. 2008), 36-52.

[9] Grandison, T., and Sloman, M. 2000. A Survey of Trust in Internet Applications. *IEEE Communications Surveys and Tutorials* 3, 4 (Fourth quarter 2000), 2-16.

[10] Cho, J.-H., and Swami, A. 2009. Towards Trust-based Cognitive Networks: A Survey of Trust Management for Mobile Ad Hoc Networks. In *Proceedings of 14th International Command and Control Research and Technology Symposium* (*ICCRTS*) (Washington, DC, June 2009). Online at: http://www.dodccrp.org/events/papers/191.pdf

[11] Ruohomaa, S., Viljanen, L., and Kutvonen, L. 2006. Guarding Enterprise Collaborations with Trust Decisions – the TuBE Approach. In *Proceedings of the Workshops and the Doctoral Symposium of the Second IFAC/IFIP I-ESA International Conference: EI2N, WSI, IS-TSPQ* (Bordeaux, France, Mar. 2006), 237-248.

[12] Jøsang, A., Ismail, R., and Boyd, C. 2006. A Survey of Trust and Reputation Systems for Online Service Provision. *Decision Support Systems* 43, 2 (Mar. 2007), 618-644. DOI= http://doi.acm.org/10.1016/j.dss.2005.05.019

[13] Sun, Y.(L.), Han, Z., and Liu, K.J.R. 2008. Defense of Trust Management Vulnerabilities in Distributed Networks. *IEEE Communications* 46, 2 (Feb. 2008), 112-119. DOI= http://doi.acm.org/10.1109/MCOM.2008.4473092.

[14] Rasmusson, L., and Janssen, S. 1996. Simulated Social Control for Secure Internet Commerce. In *Proceedings of New Security Paradigms Workshop* (Lake Arrowhead, CA, Sep. 1996), 18-25. DOI= http://doi.acm.org/10.1145/304851.304860

[15] Yan, Z. 2007. *Trust Management for Mobile Computing Platforms*. Doctoral Thesis, Helsinki University of Technology, Helsinki, Finland.

[16] Solhaug, B., Elgesem, D., and Stolen, K. 2007. Why Trust is not Proportional to Risk? In *Proceedings of 2nd International Conference on Availability, Reliability, and Security* (Vienna, Austria, Apr. 2007), 11-18.

[17] Ruohomaa, S., and Kutvonen, L. 2005. Trust Management Survey. In *Proceedings of Third International Conference on Trust Management* (Paris, France, May 2005). LNCS 3477, Springer-Verlag, 2005. 77 – 92.

[18] Blaze, M., Feigenbaum, J., and Lacy, J. 1996. Decentralized Trust Management. In *Proceedings of IEEE Symposium on Security and Privacy*, (Oakland, CA, May 1996) Online at: http://www.crypto.com/papers/policymaker.pdf.

[19] Jøsang, A., Keser, C., and Dimitrakos, T. 2005. Can We Manage Trust?" In *Proceedings of the Third International Conference on Trust Management (iTrust)* **(**Versailles, France, May 2005), 93-107.

[20] Conner, W., Iyengar, A., Mikalsen, T., Rouvellou, I., and Nahrstedt, K. 2009. A Trust Management Framework for Service-Oriented Environments. In *Proceedings of World Wide*

Web Conference (Madrid, Spain, Apr. 2009), 891-900. DOI= http://doi.acm.org/10.1145/1526709.1526829

[21] Winsborough, W.H., Seamons, K.E., and Jones, V.E. 2000. Automated trust negotiation. In *Proceedings of DARPA Information Survivability Conference and Exposition* (Hilton Head, SC, Jan. 2000), 88–102. DOI= http://doi.acm.org/ 10.1109/DISCEX.2000.824965

[22] Chu, Y.H., Feigenbaum, J., LaMacchia, B., Resnick, P., and Strauss, M. 1997. REFEREE: Trust Management for Web Applications. *Computer Networks and ISDN Systems* 29, 8-13 (Sep. 1997), 953–964. DOI= http://doi.acm.org/10.1016/ S0169-7552(97)00009-3

[23] Blaze, M., Feigenbaum, J. and Keromytis, A.D. 1998. KeyNote: Trust management for public-key infrastructures (position paper). In *Proceedings of 6th International Workshop on Security Protocols* (Cambridge, UK, Apr. 15-17, 1998). LNCS 1550, Springer-Verlag, 1998. 59–63.