# IMPLEMENTING A PGP-LIKE WEB OF TRUST IN A X.509 HIERARCHY THROUGH A TRUST SCORING SYSTEM

**Marco Antônio Carnut (kiko@tempest.com.br)**
Tempest Security Technologies
Centro de Estudos e Sistemas Avançados do Recife - CESAR
Universidade Federal de Pernambuco – CIn/UFPE

**Evandro Curvelo Hora (evandro@tempest.com.br)**
Tempest Security Technologies
Centro de Estudos e Sistemas Avançados do Recife - CESAR
Universidade Federal de Pernambuco – CIn/UFPE
Universidade Federal de Sergipe – DCCE/UFS

**Cristiano Lincoln Mattos (lincoln@tempest.com.br)**
Tempest Security Technologies
Centro de Estudos e Sistemas Avançados do Recife - CESAR
Universidade Federal de Pernambuco – CIn/UFPE

**Fábio Silva (fabio@cesar.org.br)**
Centro de Estudos e Sistemas Avançados do Recife - CESAR
Universidade Federal de Pernambuco – CIn/UFPE

## ABSTRACT

*This paper describes a simple way to merge PGP's web-of-trust model with a X.509 hierarchy to provide low-cost but trustable digital certificates, through a CA hierarchy that provides two families of certificates: "entry-level" certificates focusing on easy initial enrollment in the system, whose CAs are simple enough to be able to replace the conventional, less secure, web application registration systems based only on email address and unencrypted traffic; and "verified idenity" certificates that provide identity guarantees based on a trust scoring web application that performs both automated identity checks on public databases and the traditional user-to-user introduction method. The system is also capable of detecting and mitigating certains kinds of identity spoofing attempts.*

## 1 INTRODUCTION

The X.509 public-key infrastructure was conceived in a hierarchical fashion where a root CA would certify possibly several layers of intermediate CAs, forming a tree on which the leaf CAs would then certify the end entities (users, institutions, internet hosts or servers). The CAs would act as trusted third parties that would hide the complexity of verifying the identities from the final users; all they would be required to do is to choose a CA to trust. Its adoption in the real world, however, has been slow: the CAs' high setup and operational costs makes the certificates expensive, often as much as competing technologies such as hardware tokens; implementing the overly complicated X.509/PKIX/PKCS family of standards (ITU-T, 1997; Housley et al, 1999; Kaliski Jr., 1993) has proven difficult and prone to interoperability problems (Gutmann, 2000); and dislike from user communities suspicious with too much centralized control.

The PGP PKI (Zimmermann, 1995; Garfinkel, 1994; Callas et al., 1998; Stallings, 1998), in contrast, puts the cross-certification power directly in the user's hands: since each user is free to certify whoever he/she chooses and to decide who to trust. The trust relationships can be represented as the edges of a directed graph (McBrunnet, 1997) called "web-of-trust" where the users are the nodes. While more flexible and providing zero-cost certification (in the sense that the user performs the identity validation himself, not having to pay for someone else to do it), this approach puts the burden of deciding other parties' trustworthiness in the user – often a nontrivial security decision. Besides, different users would be more or less rigorous in their validation of other users' identities. This unevenness makes the credibility of a particular certificate chain much more uncertain that would be desirable.

So far, these two approaches have always been presented as inherently antagonic (Branchaud, 1997; Gerck, 1998) and extensive discussion has been presented about its unsuitability for global e-commerce (Winn, 2001; Schneier & Ellison, 2000; Guida, 2000; Goodenough, 2000). This paper proposes a way to take the best of both worlds, showing one possible way to endow a X.509 hierarchy with a collaborative trust system somewhat like the PGP's web of trust model, but with considerable advantages.

The rest of this paper is organized as follows: section 2 describes the overall system components: the Entry-Level and Verified Identity family of Certificate Authorities, the Trust Manager and the trust scores. Section 3 details the combination of automatic and human-assisted identity validation procedures used by the Trust Manager to ascertain the users and hosts identities. Particular attention is given to the resilience to misbehavior with a description of the identity contention management scheme. It is also argued that these metrics adhere to good design principles proposed in the literature. Section 4 presents conclusions and future work directions.

## 2 SYSTEM ARCHITECTURE

### 2.1 CA and Key Hierarchy

We start by proposing a fairly standard CA hierarchy: a root CA which certifies two families of intermediate CAs:

- **The Entry-Level (EL) CA family:** these CA applications generate certificates online to any user that requires it, with just minimal validation, such as complying to a simple naming policy, avoiding duplicates, checking the validity of the email address by replying to it. Its sole purpose is to put a valid, working, fully functional digital certificate into the user's applications – most likely, his web browser – immediately and for free. This certificate would have short validity period compared with the more trusted ones – two or three months seem

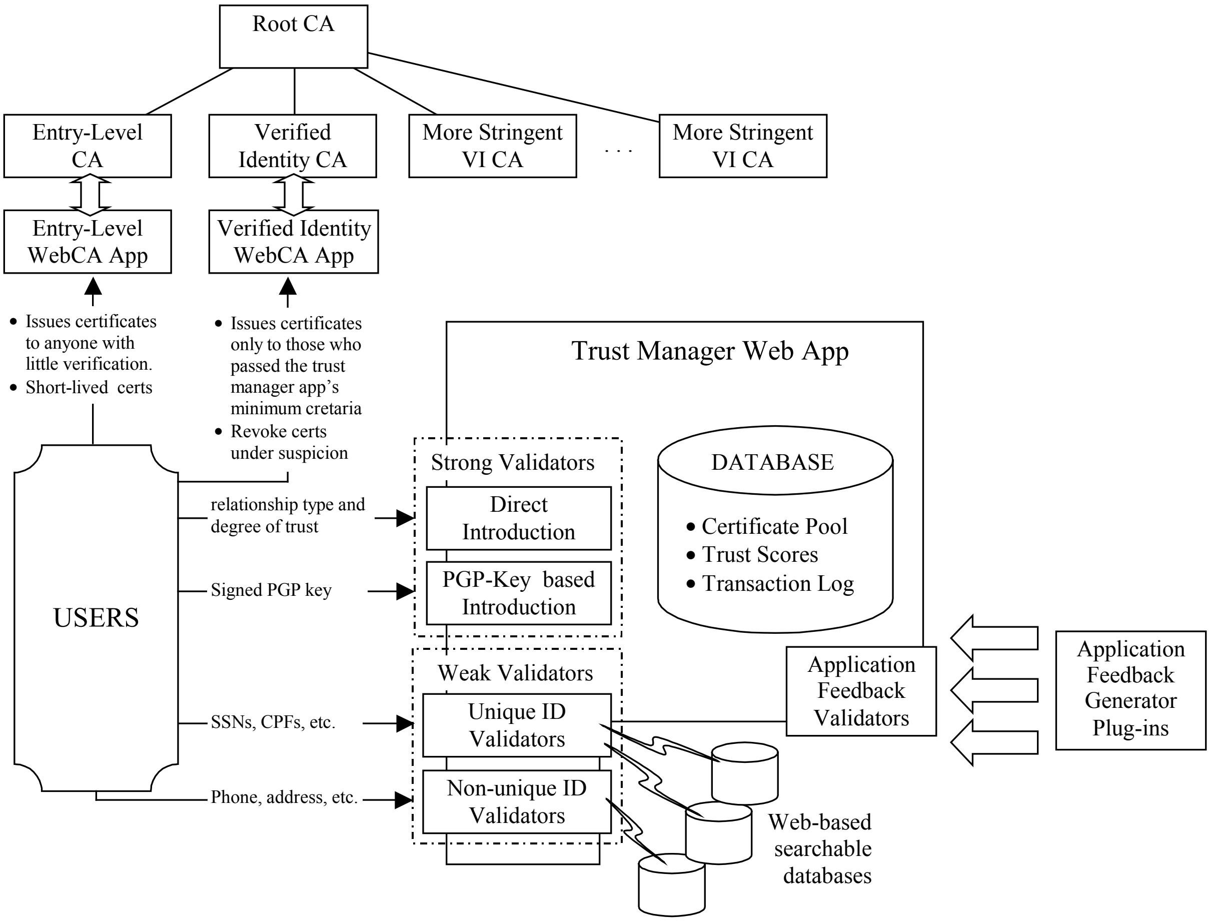
Root CA
Entry-Level CA
Verified Identity CA
More Stringent VI CA
. . .
More Stringent VI CA
Entry-Level WebCA App
Verified Identity WebCA App
• Issues certificates to anyone with little verification.
• Short-lived certs
• Issues certificates only to those who passed the trust manager app's minimum cretaria
• Revoke certs under suspicion
Trust Manager Web App
USERS
relationship type and degree of trust
Signed PGP key
SSNs, CPFs, etc.
Phone, address, etc.
Strong Validators
Direct Introduction
PGP-Key based Introduction
Weak Validators
Unique ID Validators
Non-unique ID Validators
DATABASE
• Certificate Pool
• Trust Scores
• Transaction Log
Application Feedback Validators
Application Feedback Generator Plug-ins
Web-based searchable databases

**Figure 1: Overall system architecture:** above, the quite traditional root CA → intermediate CAs key heirarchy. The two main CAs are the Entry-Level and the Verified Identity, associated to their respective web sites. The first issues certificates to nearly any user, just to allow them to log on the trust manager web application. Users are allowed to issue certificates under the Verified Identity CA only when they meet a minimum set of scores. The way to increase them is by passing through several kinds of validators: the weak validators check their personal information on web databases; and the strong validators are based on user-to-user introduction. Additional validation may be provided by actuarial data from external applications, which is normalized to fit the scoring scale. The VI CA also has a PGP keypair and adds his signature to users with PGP keys; and accepts signed PGP keys as strong-validation introductions. This leverages the userbases of both PKIs, helping them to reinforce each other.

sensible. It should be accepted only for testing or initial enrollment in a single application.

- **The Verified Identity (VI) CA:** this CA would issue users digital certificates when they met some specific credibility and trustability scoring. These certificates would have a larger validity period, something like six to twelve months. Actually, there could be several such CAs, each with successively more stringent scoring requirements. Large-scale production applications should require these certificates for the bulk of their functionality; Entry-Level certificates should be accepted only for testing, initial enrollment or minimal funcionality; the application should "insist" that the user get him/herself a VI certificate.

The CAs would have both X.509 certificates/private keys and PGP keypairs, so they could act as cross-certifiers. The idea is to leverage each PKI's user base to reinforce each other and foster wider adoption.

### *2.2 The Entry-Level Certification Authorities*

The Entry-Level CAs are designed around the notion that the users should be able to get their first digital certificate quickly. All the user is required to inform is a valid email address for contact and, if he wishes, his real name. Even taking into account the private key generation, it's not difficult do make it no more complicated or time-consuming than enrolling in a web application.

In fact, our prototype implementation adheres to the following UI design principle: the user should be presented just one single form field. This allows the entire CA to be included as a small visual element (a box or sidebar) in a larger web application.

The certificate is approved immediately and is installed in the very next screen. After getting his certificate installed, the user is sent a registration confirmation e-mail that clearly informs:

- The website name and the exact URL he accessed to perform the enrollment;
- A brief description of the certificate's purpose – often, its sole purpose is to access the website application; and the URL to complete his enrollment *in the web application*; that means, if the user doesn't click this URL, the web application won't grant him access.

Alternatively, a URL to a directory of compatible applications may be presented.

- A URL to revoke this certificate in a single click – for instance, in case the user feels the certificate was issued by someone other than himself;
- The somewhat brief validity of the certificate and the fact that he can apply for a Verified Identity Certificate which grants him greater validity and, possibly, more access privileges; an URL where the user can learn more about the certificate classes, CPSs, etc., is usually given.

The loose identity tie, based mostly on the email address, makes the EL certificate somewhat like the "Verisign Class 1" certificate (Verisign). This simple process, however, introduces are several differences worth commenting:

- The registration process is a lot simpler. Experience with our prototype implementation shows that users feel it as a kind of "odd registration process", but, since it's so quick and simple, they go along. This makes a lot of sense in light of current web usability studies, which teaches us that the user tends to quit using a web-based service when presented with overlong registration forms.
- The user's email is implicitly trusted, i.e., the user's email is accepted without validation. However, a notification is sent, presenting an opportunity for the user to revoke the certificate if he suspects it was issued by a spoofer. Another advantage of this process is that it awakens the user for the possibility of revocation from the very beginning.

  This particular choice may be considered controversial. It does make the identity guarantee of the certificate even weaker. It is justified by our design principle that the EL certificates are not meant to be "secure"; they are meant to be hassle-free to get.
- The application won't let the user though if he doesn't read the validation email and finishes the enrollment process by clicking in the appropriate URL to activate his newly created account. This is similar to what most website enrollment processes require. In other words, the certificate issuance is decoupled from application-level authorization. Notice that since the user's digital certificate is already issued and properly installed in his web browser; he can activate his account now or later. He can also enroll automatically on any application that accepts certificates from the same EL CA.

It is conceivable that the user might lose the email with the revocation URL. An easy way to deal with this situation, in keeping with the usability simplicity requirement is to direct the user to request a certificate again. The EL Web CA application would then detect that the supplied email address already has a valid certificate associated with it and offer the user three choices:

- **Revocation**: the EL Web CA application resends the email with the revocation URL; if and when the user wants to revoke the certificate, he accesses the URL.
- **Reissuing:** the EL Web CA sends a email with a special URL that revokes the previous certificate and issues a new one in a single step.
- **Do nothing:** leave things as they are.

There is considerable debate about whether revocation is a good idea or even needed at all in PKI systems (Rivest, 1998; McDaniel et al, 2000). In light of this "revoke if it wasn't you who requested it" philosophy, along with the need to reissue certificates often due to the short certificate validity, revocation seems well suited, even though most applications neither correctly process CRLs nor support OCSP or the like.

Another important point is the naming policies. It follows the following principles:

- **Globally Unique DNs**: The certificate holder's Distinguished Name in the Subject field should identify only his email address (with the Email OID), his name (in the Common Name OID) and the name of the Entry-Level CA who issued the certificate in a OU field. This makes DNs globally unique, preventing name clashes in case some user tries to issue certificates under more than one EL CA. Thus, the EL CA does not need to check elsewhere to see if this DN has already been taken. This also simplifies building associated directory services, like a global LDAP database.
- **Only one certificate per email address:** otherwise, the identity guarantee would be even slacker and the reissuing/revocation detection wouldn't work.
- **Server Certificates:** If a user supplies a valid DNS name as his name, the EL CA may issue a server certificate instead of a client. It does need to do any kind of checks to see if the address exists. Several server certificates may be issued for the same contact email address (presumably, the servers' administrator). The EL CA has the option, according to its own policies, of *not* issuing server certificates at all.
- **Identity privacy:** nickname and email offer little to correlate the user with his real world persona. This is in stark contrast with several other certification services, which require lots of personal data *in advance* to perform identity validation – an extreme example being the Brazilian PKI, which not only demands the user's ID in the four most proeminent national registries (Comitê Gestor da ICP-BR, 2002), but includes them in the certificate, making them easy prey for spammers and identity thieves. In

our system, personal data is required only when the user wants to get his identity validated, as shown in section 3.1 .

The deliberate bias towards user friendliness instead of "security" (as represented by identity guarantees) may be regarded as distateful by PKI purists. In fact, the scheme proposed above provides only slightly more features than the PGP PKI (because of the much clearer revocation process) and the same level of identity validation – nearly none at all. The almost single-step key generation and certificate installation procedure eliminates many little chances for user error; for instance, in traditional CAs, most users tend to try to pick up the certificate using a different client (web browser, typically) or even a diferent computer from where the private key was generated.

We argue, however, that it all these usability trade-offs are fundamental to get user acceptance. To the best of our knowledge, there are no comprehensive studies on how usability problems affect the X.509 PKI – despite profuse folkloric horror user support stories within CA managers and PKI practitioners communities. However, (Whitten & Tygar, 1999) explain why PGP, widely thought as being "user friendly" because its Windows versions have a decent GUI, is much more non-intuitive and less usable than many of its enthusiasts would like to admit. Many of its results are very well applicable to the X.509 arena and have inspired our design for extreme simplicity.

Admittedly, even this simplicity cannot solve many *compliance defects* (Davis, 1996) inherent in PKI systems, like the impossibility to enforce good passphrases to protect the private key (since its generated by the client software; Internet Explorer, in particular, makes it upsettingly easy to have a private key with no passphrase at all; both Netscape and IE don't provide a way to require a minimum passphrase complexity) or to securely distribute the root CA's certificate (all of our EL CA's pages invite the user to reinstall the root CA certificate and check their fingerprints). However, yet again we are trusting the client software and user to do the "Right Thing".

Notwithstanding, the EL CA's Certificate Practice Statement must make it very clear what "Entry-Level certificate" means: no identity guarantees, good for testing, learning and initial entry in the trust system; and that the user's ultimate goal should be to upgrade the Entry-Level certificate to a Verified Identity one.

### *2.3 The Trust Manager Application and the Verified Identity CAs*

Along with the VI familiy of CAs there would be the *trust manager* web application (TMWA, for short). It would require SSL client certificate authentication, accepting any user whose certificate was issued by both the EL and VI CAs. For each of them, the application would store their certificates, personal and contact data that the user voluntarily made available for purposes of identity checking and three *trust scores*:

- **Credibility score**: measures how certain we are that this individual is who he claims he is. It will be calculated as a weighted average of several *validators*, described below.
- **Introducer score:** indicates how trustable this user is when attesting or repudiating other users' identities. EL-certified users cannot have introducer points; only VI-class users may introduce other users.
- **Suspicion score:** keeps track of how much this user is involved in identity contention with someone else. Users whose suspicion points exceed their credibility cannot have certificates issued or reissued under the VI CAs; besides, their introducer power is suspended. Notwithstanding, they can accumulate credibility points normally. If his credibility score exceeds his suspicion points, his privileges will be granted back.

It is instructive to compare this scheme with other proposals like Thawte's Freemail Web-of-Trust program (Thawte): there is only one score, instead of the three above, that handles both the user's credibility and its experience/reliability as an introducer (which are called "notaries"). There is no suspicion management, since each notary is required to meet in person with any individual he introduces.

Each VI CA would have an "eligibility criteria", based on the trust scores, metrics from the trust graph and, possibly, other criteria (e.g., requiring a specialized client, more secure than the mainstream web browsers). When some EL certificate user meets or exceeds these criteria, the VI CA would send an email inviting him to issue a VI certificate (this most likely requires generating a new private key, since most client software require a one-to-one mapping between a certificate and a private key).

## 3 VALIDATORS

Validators are procedures executed by the TMWA for verifying the identity of the certificate holder. An important design principle is that they should not, insofar as possible, require on-site CA operators; they should be performed automatically, either by querying an online public Web database or being driven by remote users' input. They are to be triggered by the client users themselves, by accessing the proper web pages in the TMWA. Their main function is to allow users who already possess an entry-level certificate to increase their scores, up to the point for qualifying to get a VI certificate issued – without having to go in person or send paper credentials over snail mail to the CA. We propose three main kinds of validators:

- **Weak validators:** verify some of the users personal data through automated queries on public websites. For instance, checking names and addresses in whitepage services such as knowx.com or public government services (section 3.1 presents more specific examples). Users passing on these validators would receive a small amount of credibility points. It has to be small because, since it is based on public data, it's rather easily spoofed. However, the weak validators fulfill an important role: tying the certificate holder with a verifiable identity in the real world, as verified by other independent sources. If anyone wants to spoof anyone else, they would spoof someone who probably exists and may eventually expose the spoof and/or dispute with the spoofer.

- **Strong validators:** the traditional way of cross-certification through trusted introducers – the user gets someone else *already with a high introducer trust rating* to vouch for his identity. The introducer would access his personal account in the Web CA and fill in a form saying that he has *x* percent certainty that the newcomer is who he says he is. This number would be multiplied by his introducer trust score and added to the newcomer's credibility score.
- **Traditional (operator-assisted) validators:** Since we are interested in building a network of CAs, it is expected that some of them will find the EL CA weak guarantees unsuitable for their particular application or user community and opt for a traditional CA system where the user has to present himself in person to get his certificate or be approved by some kind of enrollment process and real world credential validation. There's nothing inherently wrong with this approach; in some situations, it makes perfect sense. So, our PKI has space for them; and, in fact, they may act as a special kind of strong validator, as further described in section 3.2 .

All that means that the web-of-trust would be built on the Web CA application's database as a set of trust scores; a graph of introducer-introductee relationships; and a log of validation procedures followed by each user. This last item is specially interesting for debugging and auditing, for it allows us to reconstruct the user's history and justify why the system has given him the score he has.

Any given user should be capable of, at any time, check his scores and be informed of what steps to take in order to increase them. When the scores of a particular user grows beyond a specified threshold, he should be issued a certificate under the Verified Identity CA. That would mean that the user passed enough challenges and validations for the VI CA to be sure enough of his identity to issue him a certificate.

### *3.1 Weak Validators*

Weak validators provide new users a way to gain a small but significant initial credibility quickly, online, without having to ask other people to vouch for them, as is the case with the other validators.

It works like this: a new user would log on the TMWA using his EL certificate/private key pair and supply certain kinds of online-verifiable personal data, such as postal address, phone numbers, IDs in public services – Social Security Numbers for US residents, for example; or CPF numbers for Brazilians (CPF is the Brazilian Internal Revenue Service nine-digit numbers plus two check digits that uniquely identifies taxpayers).

The user would not be *required* to enroll his personal data in the TMWA; however, as he earns credibility points for each successful validation, he has an incentive to voluntarily do so. Obviously, the Web CA/TMWA should have a strict and clearly published privacy policy about keeping this data.

Also notice that the newcomer's personal data will be seen only by introducers (and possibly external auditors), which are expected to be much less than all Verified Identity users, and even less that the public at large. The TWMA may also offer to show the newcomer's personal data only to introducers he explicitly allows or invites, such as close friends, business associates, etc.

Groupings of the user's personal data could be validated by performing a HTTP web query on widely known and respected services. (This query would be performed by an automated script; no CA operator or human assistance should be necessary.) For instance:

- Addresses and phone numbers could be validated by checking them on whitepages directories such as knowx.com, whitepages.com and the like. It is considered valid only if the phone/address is registered with the user's name. Other people living in the same address would not pass this validation, but they have other alternatives.
- Country-specific identifiers could also be verified. Unique identifiers would be especially desired. For instance, Brazilian IRS IDs (called "CPF numbers") could be verified against their public query site, and so on.
- PGP Keys: if the newcomer has a PGP key, he could post it to the TWMA. If it is signed by some trusted introducer, then it will be regarded as a strong validation, as described in section 3.2 . Otherwise, it's regarded as weak validation and the user earns just a small fixed amount of credibility points.
- Certain personal data, such as headshot photographs, could also be accepted. Since they cannot be validated automatically, they would just "sit there" waiting for a human introducer to validate (as a means of saying "I attest that this individual looks like this photo") or repudiate ("This is the picture of a slug and this certificate holder is fooling around with the system"). More about that in section 3.2 .

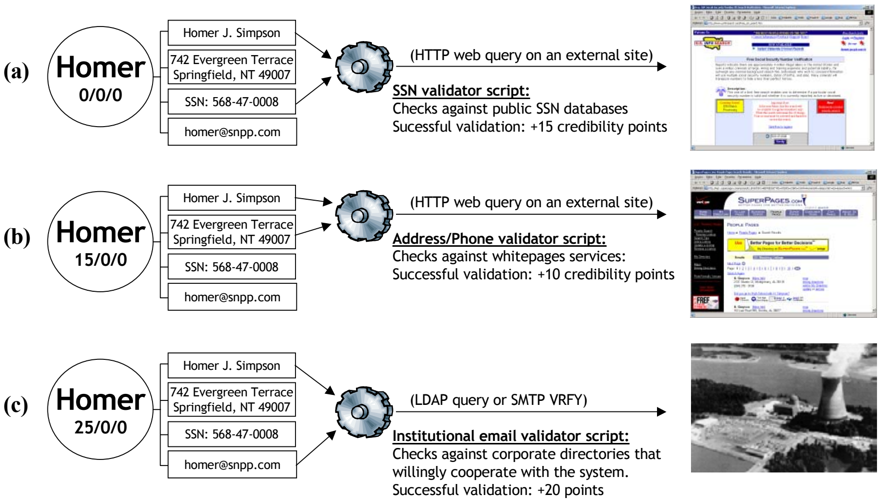
(a)
Homer
0/0/0
Homer J. Simpson
742 Evergreen Terrace
Springfield, NT 49007
SSN: 568-47-0008
homer@snpp.com
(HTTP web query on an external site)
SSN validator script:
Checks against public SSN databases
Sucessful validation: +15 credibility points
(b)
Homer
15/0/0
Homer J. Simpson
742 Evergreen Terrace
Springfield, NT 49007
SSN: 568-47-0008
homer@snpp.com
(HTTP web query on an external site)
Address/Phone validator script:
Checks against whitepages services:
Successful validation: +10 credibility points
(c)
Homer
25/0/0
Homer J. Simpson
742 Evergreen Terrace
Springfield, NT 49007
SSN: 568-47-0008
homer@snpp.com
(LDAP query or SMTP VRFY)
Institutional email validator script:
Checks against corporate directories that
willingly cooperate with the system.
Successful validation: +20 points

**Figure 2: Weak validator process:** Homer Simpson enrolls in the TWMA and starts with no credibility, introduction or suspicion points. After having posted some verifiable personal information in the TWMA database, the system runs several scripts to confirm his claims: in (a), the TWMA queries an external web site (say, usinfosearch.com) to validate his name and SSN, earning him 15 points. In (b), the TWMA queries another website (in the example, superpages.com) to verify his address, earning 10 more points. In (c), it queries his empolyer's LDAP database and/or mail server. Homer gets out of the weak validator process with 45 points (not shown in the picture), with shouldn't be enough to grant him a VI certificate.

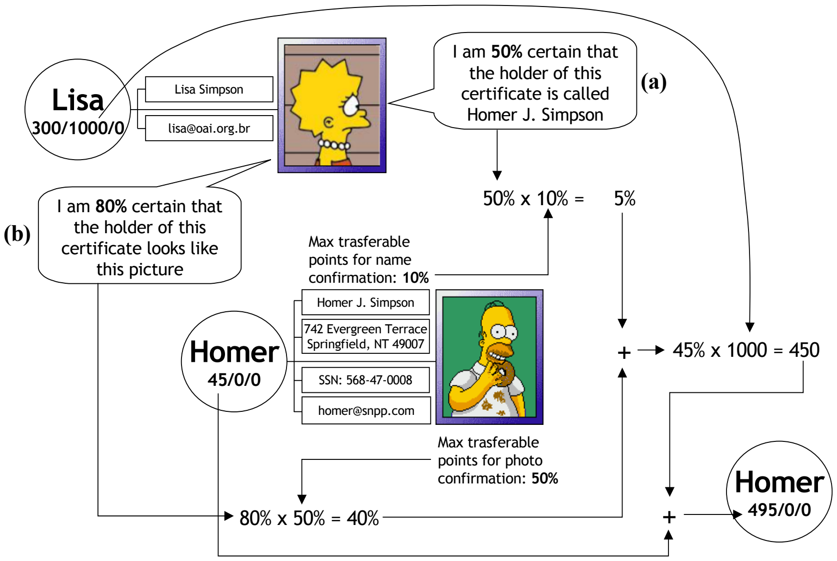
Lisa
300/1000/0
Lisa Simpson
lisa@oai.org.br
I am 50% certain that the holder of this certificate is called Homer J. Simpson
(a)
50% x 10% =
5%
I am 80% certain that the holder of this certificate looks like this picture
(b)
Max trasferable points for name confirmation: 10%
Homer J. Simpson
742 Evergreen Terrace
Springfield, NT 49007
SSN: 568-47-0008
homer@snpp.com
Homer
45/0/0
+
45% x 1000 = 450
Max trasferable points for photo confirmation: 50%
80% x 50% = 40%
+
Homer
495/0/0

**Figure 3: Strong validator process:** A case of strong validation through direct introduction: Lisa is a highly trusted introducer, with 1000 introducer points. In (a), she transfers 50% of certaintiy that the certificate's owner name is Homer J. Simpson. The TWMA presets this validaton as yielding at most 10% of the introducer's trust score, so it transfers only 5% of Lisa's 1000 points to Homer. In (b), Lisa also attests with 80% certainty that this is Homer's picture, which, multiplied by the TWMA's limit of 50% for photo validations, grant him 40% of Lisa's 1000 points. He finishes this accredidation session with 45 points from the weak validators and 450 points from the strong validators. If this exceeds some VI CA minimum thresholds, it will grant him a VI certificate.

It is important to remind that all this personal data is to be kept in the TWMA database only. It should not be included in the digital certificate when it is finally granted to the user. The VI certificate's DN should be the same of the EL certificate.

As each successful validation is achieved, the user's credibility points should be increased by the validator's trust weight multiplied by a measure of the success of the validation and its difficulty to be spoofed.

These kinds of validations are said to be "weak" because they are based on public data. They don't really prove the user is who he says he is. Thus, the amount of credibility points a user receive by these validations should be small compared with other validators, given that anyone can get personal data from some random individual in the very same services the TWMA uses to validate them and claim to be someone else.

The primary security function of the weak validators is to make it harder for a spoofer to get a certificate issued to an entirely fictious individual whose existence is unlikely to be challenged. By having to assign a verifiable identity to the certificate, a spoofer incurs the risk of being challenged by the spoofed individual some point in the future, as detailed in section 3.3 .

The weak validator concept can be extended for Internet hosts (say, for IPSec using IKE) or SSL servers: the checking software would "ping" the service to see if it is up and running. In the case of SSL, it could also check if it is returning a proper set of certificates, etc. It should be possible to validate many kinds of services: HTTPS (HTTP over SSL), POP3 and SMTP over SSL, and possibly other less popular services, such as TELNET, FTP, VNC or Jabber over SSL.

### *3.2 Strong Validators*

The fastest way for a user to gain credibility points in the trust scoring system is by having other participants, especially highly trusted ones, to *voluntarily* verify his identity. This is particularly easy if the newcomer has a friend, supervisor, business associate or anyone within his acquaintance that holds a sizeable amount of introducer points.

The process is envisaged in the following ways:

- **TWMA introduction:** suppose Newton the newcomer asked (by email or though the

TWMA community service) Ingus the introducer to vouch for him. Ingus logs on the TWMA, searches Newton in the database and fills a form specifying the amount of certainty he has that the individual he is introducing is who he says he is. This number, multiplied by his introducer score and an attenuation factor, is added to the Newton's credibility score. The attenuation is to prevent a single individual from being able to escalate someone else's credibility too fast.

In order to encourage Ingus to perform the confidence level evaluation with the greatest care, the system informs him that if Newton is later determined to be a fraud, Ingus will have his introducer points reduced by the same percentual amount of confidence he deposited in Newton; and will receive as many suspicion points – which might put him directly in suspicion mode if it turns out to exceed his credibility. In other words, Ingus' evaluation is interpreted to be like an *insurance*: the amount of his own trust he would be willing to lose if Newton is found not to be who he says he is.

- **PGP Key Introduction:** imagine that both Newton and Ingus have PGP keys. If Newton provides the TWMA with a PGP key signed by Ingus, this is verifiable proof that Ingus introduced Newton, albeit on a different PKI. But since one of our objectives is to promote multiple-PKI interoperability, the TWMA, upon verifying the key signatures, transfers a fixed fraction of Ingus' introducer score to Newton's credibility points. This assumes, of course, that Ingus had previously posted is PGP key to the TWMA, so it can "establish the links" between his various identities and verify the signatures on Newton's PGP key; and established this fixed fraction in this personal profile in the TWMA application. He also gets notified that this "client-side" introduction took place.
- **Cross-Certification**: A natural generalization of the PGP-based introduction is to accept certificates from other CAs or key hierarchies whose validation processes are known and that can be easly assigned a credibility rating. For instance, Verisign certificates could be accepted as another level of validation – Class 1 certificates, which validate only the email address, would add little extra credibility, while Class 2 and 3 certificates, which rely on institutional credentials and in-person enrollment, respectively, would grant much more points. Certificates from the CAs within our own hierarchy that employed traditional validation processes, as described in section 3 , could be likewise accepted.

It is worth reminding again the importance that all these operations be carefully logged, both for debugging and auditing purposes, so it becomes possible to reconstruct exactly why any particular user has got his scores.

Internet hosts could be introduced in a similar way, except that their administrators would act in their behalf, inviting introducers to vouch for the identity of their SSL servers or IPSec-enabled hosts.

### *3.3 Contentions*

If a user Charlie the challenger supplies an identifier (say, his name, e-mail address, SSN, etc.) already claimed by someone else, he is to be put in *suspicious mode*: he earns as many suspicion points as the sum of the credibility scores of each user he contends IDs with.

If Charlie's credibility reaches a certain fraction (say, half) of the credibility of some user he is contending with, the challenged user gets notified of this fact by email. This warning should give him time to take precautions against *takeover*: if the Charlie's credibility exceeds the challenged user's, Charlie is awarded possession of the contended IDs. The challenged user then is put into suspicious mode: it's now his problem to prove his identity beyond Charlie's credibility.

These rules attempt to foil some avenue of identity theft attacks outlined below:

- **Post-takeover:** Suppose a legitimate user has already got his VI certificate issued without incident. Then, a persistant attacker issues several EL certificates with his name and uses them to log in the TWMA and generate contentions, supplying the legitimate user's public personal data to pass through many weak validators and gain a modest amount of credibility points. As long as the legitimate user keeps his own credibility points high, the contenders won't be able to steal his identity. He is probably in the best position to do so, since he can convince introducers to attest his identity and strong validators yeild so much more credibility points. The legitimate user gets early notification about contenders and their chances to take over his identity.
- **Pre-population:** The attacker enrolls in the TWMA before the legitimate user, supplying some of the spoofed user's public personal data to pass some of the weak validators. If the eligibility criteria for getting a VI certificate is set to near or more than the sum of what all possible weak validators could give, or requires a minimum number of introducers regardless of the credibility points, the attacker won't be able to assume the (unsuspecting) legitimate user's identity.

All that relies, of course, in the trustworthiness of the introducers and the rigor with which they perform a identity check. A rogue introducer can help attackers to bootstrap themselves through the credibility ranks or even create whole cliques of self-certifying fake communities (as long as the real users being spoofed don't enroll in the system and start contending with the fakes). The population base of our prototype implementation has not reached enough critical mass to allow these phenomena to be empirically observed, measured and characterized, but it's natural to expect these issues will manifest themselves as the population base grows.

The fact that credibility points given from the introducer to the introductee act as insurance creates an incentive for caution: the introducer should know that if an identity validation error from his part is discovered (say, by other more graduated introducers or external audits), it will revert against himself, almost certainly quelling his privileges – a phenomenon we call "introducer demise".

In our prototype implementation, we didn't make introducer demises propagate through all of his introductees – this forces the whole trust scores to be recalculated, and, if not carefully calibrated, may make the entire trust web collapse. It was felt as undersirable in our small web, making it too fragile; but may be considered a minor local event in a large scale (say, millions of nodes) web, adding a self-correcting nature against introducer-aided fraud.

At any rate, it is expected that contentions require much more human intervention than identity validations that go about without incidents. On the other hand, it should be possible to calibrate the system so that the former happens much more rarely than the latter.

### *3.4 VI certificate eligibility criteria*

Our prototype implementation has only one VI CA with a very simple acceptability criteria: if the user exceeds 100 credibility points given from at least two introducers, he is granted a "VI level 1" certificate. This simplistic approach was chosen because it's easy to explain, simple for users to know what to do and makes the process of getting the VI1 certificate very quick: the user enters as much personal verifiable data as he wants, gathering a small amount of credibility due to the weak validators; then he consults the public list of introducers in the TWMA community page, asking the one he knows to vouch for him.

Typically a few hours later, when the introducers check their emails (the TWMA informs them that someone asked to be introduced), the newcomer is validated and he is invited to the VI certificate generation page (it is worth noting that all this is made with SSL client authentication, thus requiring his EL certificate). His VI certificate is then issued in the same single-step manner adopted by the EL CAs and, finally, the user is informed of the applications that accept/require his newly issued certificate, along with instructions about how to register with them.

We plan to have "level 2", "level 3", VI certificates with stricter validation requirements, such as requiring several introducers, allowing the introducers to specify the validity of their trust grant and allowing the newcomer to attain VI status only if at least one introducer vouches for him for at least one year (the suggested validity period of the VI certificates), etc. Another idea is to have a VI CA that requires the introduces to be members of stricter PKIs, such as ICP-BR; if the ICP-BR would accept these certificates, this VI CA could act as a "multiplier" that would help to spread its adoption.

This makes a good moment to remind that each certificate from each CA has a life cycle of its own; they are not necessarily coupled or associated in any way. There's no need, for instance, to revoke an EL or a lower level VI certificate because the user has been issued a higher-level VI certificate. The only tying association is that they're kept in the TWMA.

It is interesting to compare this authentication metric with others, such as the ones studied by (Reiter & Stubblebine, 1999). It is worth repeating the eight authentication principles they laid out and comment how our system adheres to or deviates from them.

- *Principle 1: The model, to which a metric is applied, should not require the user to infer bindings between keys and their owners. In particular, when representing certificates in a model: entities don't sign certificates, keys do.*

  In our system, the TWMA clearly identifies the several identities associated with a particular keypair/certificate, leaving no room for guesswork.

- *Principle 2: The meaning of the model's parameters should be unambiguous. This especially applies to the meaning of probabilities and trust values in the models that use them.*

  The numeric trust scores provide quantitative estimates of each trust quality (credibility, introducer, suspicion, etc). The scale and calibration may be somewhat arbitrary, but, within itself, it's self-consistent.

- *Principle 3: A metric should take into account as much information as possible that is relevant to the authorization decision that the user is trying to make.*

  The user (or application) doesn't make much more authorization decisions than choosing what EL or VI CAs to trust. But their acceptability criteria can be very well specified. We have tree different scores, which seem already a great deal of relevant authorization information – our system even has suspicion detection and management, a feature not found in many other metrics. We feel that more than that would overcomplicate the system.

- *Principle 4: A metric should consult the user for any authentication relevant decisions that cannot be accurately automated. A decision that could affect authentication should be hidden from the user only if it can be reached using* *unambiguous, well-documented, and intuitive rules.*

That's precisely what strong validators are for. Since it was felt that automated validations could be rather easily spoofed, we made them the weak validators.

On the other hand, our concept of "trust insurance" doesn't mean "monetary insurance" that would be paid in case of system failure (although it may be conceivable that it may provided as a add-on commercial service); instead, it means only a guarantee that introducers will be penalized for errors or misbehavior.

- *Principle 5: The output of a metric should be intuitive. It should be possible to write down a straightforward natural language sentence describing what the output means.*

  It is easy to explain what the metrics measured: "you got *n* points from one introducer, *m* points from another one, *i* points from posting your SSN, *j* points from posting your email, *k* points from posting your Brazilian CPF number, which add up more than the *t* threshold needed to get you a VI certificate."

  This opens up an interesting possibility: the page containing the certificate's CPS could add, within the bulk of the CPS text, an automatically generated, natural language explanation of these metrics and the guarantees (technical and legal) they provide – much like the "Unabridged Certificate" proposed in (Gerck & Bohm).

- *Principle 6: A metric should be designed to be resilient to manipulations of its model by misbehaving entities, and its sensitivity to various forms of misbehavior should be made explicit.*

  Section 3.3 detailed some of the contention management and their resistance to misbehavior. More field experience is needed, however, to ascertain their efficiency in practice.

- *Principle 7: A metric should be able to be computed efficiently.*

  Since the TWMA enforces only direct introductions, there is no need to construct the entire introduction graph to compute the trust scores nor run graph-theoretic algorithms with superlinear time complexities (it may be useful to build the graph for other purposes, though). The calculations can be done incrementally and even reconstructed from the transaction log in linear time.

- *Principle 8: A metric's output on partial information should be meaningful.*

  Any user registered in the TWMA has trust scores, even if they have passed no validators. So, the metric is meaninful even in the absence of information.

## 4 CONCLUSIONS AND FUTURE WORK

We proposed two CA families to implement a PKI mixing the PGP and X.509 models based on the realization that the process of aggregating strong identity guarantees to a certain key/certificate should not be tied to its issuance; it should be done at a later moment, if and when convenient to the certificate holder. In fact, there are many instances when it's simply not worth the hassle to go through an extremely strict identity validation procedure when a not-so-trusted certificate would do just fine.

In our system, the entry-level family of CAs provide this focus on user and administrative simplicity. We've argued that it provides roughly the same kinds of protections that the PGP infrastructure: confidentiality through encryption but with little certainty of who the keys onwers are in respect to other identification systems. The proposed scheme allows the certificate to be granted immediately, becoming well suited for replacing website registration systems and similar end-user applications. The short lived certificates, when combined with application demand, creates an incentive for the user to "upgrade" his entry-level certificate to the longer lived, more widely trusted, Verified Identity ones.

Space constraints prevented us from being able to report the many interoperability pitfalls we ran into, the nontrivial solutions we were often forced to adopt and several other interesting implementation details. These may make material for a future paper; meanwhile, the reader is invited to visit our implementation site: www.freeicp.org.

The proposed Verified Identity family of CAs provide the higher identity assurance levels. It can be seen as a framework to unify several identification services and strictness criteria. It encompasses both the human-operator-based identity check systems now common on commercial or institutional CAs and a novel idea of a trust scoring web application that allows borrows the PGP's web-of-trust model but implemented over a centralized database to provide online-only, semi-automated identity validation – vaguely resembling the credit scoring systems now common in financial institutions. We argue that its collaborative nature may be exploited to make near-zero-cost certificates possible and thus allowing the "commoditization" of trustable digital certificates.

A trust managing system was described that allows the users to tie their certificates to automatically verifiable real world identities and accumulate credibility by having these identities verified by veteran users that act as trusted introducers. The proposed model uses a much more precise system based on numeric scores that evaluate the user's identity credibility, trustworthiness as an introducer, and the amount of dispute that the user is having to gain control of other user's identities. In fact, contention detection and control is another area that this system proposes and both PGP and X.509 lack. Precisely because of its novely, it deserves deeper study.

We've shown that Verified Identities CAs can use these trust metrics to decide, according to their own acceptability criteria, if a particular user or internet host is eligible to one of his certificates. A simple threshold criterion was proposed that subjectively adheres to all the authentication metric design principles posed by Reiter and Stubblebine. An interesting point is that the metric allows for an easy description of itself in natural language that could be added directly to an automatically generated Certificate Practice Statement.

Other interesting avenue being pursued is the use graph-theoretic algorithms to monitor the growth of the certification network – as made for a fraction of the PGP PKI in (McBrunnet, 1997) – and provide feedback to help calibrate the system parameters to achieve specific security guarantee goals. Their use as authentication metrics may be also considered.

The field of automated identity verification has been blossoming with interesting new proposals. For instance, in (Authentify) it is described a system in which an automated voice system dials to the telephone number the user supplied in the enrollment process and requests the user to confirm a challenge number and record his name and affiliation, for audit purposes. A whole different idea, much more sophisticated, would be to accept digitized fingerprints to be matched against law enforcement's databases. The inclusion of these kind of automated identity verification systems within an implementation of the framework proposed in this paper may become a worthwhile research avenue.

## 5 REFERENCES

AUTHENTIFY, Inc. *Authentify|Register™ and RSA Keon® OneStep – Assuring User Identities in the Registration Process*, http://www.authentify.com/images/pdf/AR_RSA_Keon.pdf

BRANCHAUD, Marc. *A Survey of Public-Key Infrastructures*. MSc. Thesis, Department of Computer Science, McGill University, 1997.

CALLAS, J.; DONNERHACKE, L.; FINNEY, H.; THAYER, R., *RFC 2440: OpenPGP Message Format*, 1998, www.ietf.org/rfc/rfc2440.txt

Comitê Gestor da ICP-BR, *Resolução nº 11 de 14 de fev de 2002*, www.icpbrasil.gov.br/RES_ICP11.htm

DAVIS, Don. *Compliance Defects in Public-Key Cryptography*, Sixth Usenix Security Symposium Proceedings, July 1996, pp 171-178.

GARFINKEL, Simon. *PGP: Pretty Good Privacy. O'Reilly & Associates*, 1994, ISBN 1565920988

GERCK, Ed, BOHM, N. *X.509 Certificates: A Readable Unabridged Inside View*, www.mcg.org.br/x509cert.htm

GERCK, Ed. *Overview of Certification Systems: X.509, CA, PGP and SKIP*, MCG Group, 1998, www.mcg.org.br/cert.htm

GOODENOUGH, David. *A Heretic's view of Certificates*, www.dga.co.uk/customer/publicdo.nsf/public/WP-HERESY

GUIDA, Richard. *Rebuttal to "Ten Risks of PKI"*, Computer Security Institute Alert, n 204, 2000, www.gocsi.com/pdfs/expert.pdf

GUTMANN, *Peter. X.509 Style Guide*, 2000, www.cs.auckland.ac.nz/~pgut001/pubs/x509guide.txt

HOUSLEY, Russel; FORD, Warwick; POLK, Tim; SOLO, David. *RFC 2459: Internet X.509 Public Key Infrastructure Certificate and CRL Profile*, 1999

ITU-T, *Recommendation X.509/ISO/IEC 9594-8: Information Technology – Open Systems Interconnection – The Directory: Authentication Framework*, Internation Telecommunication Union, 1997.

KALISKI Jr., Burton S. *An Overview of the PKCS Standards*, RSA Laboratories, 1993

McBRUNNET, Neal. *PGP Web of Trust Statistics*, 1997, bcn.boulder.co.us/~neal/pgpstat

McDANIEL, Patrick; RUBIN, Aviel. *A Response to "Can We Eliminate Certificate Revocation Lists?"*, Proc. Financial Cryptography 2000, February 2000.

REITER, Michael K.; STUBBLEBINE, Stuart G. *Authentication Metric Analysis and Design,* ACM Transactions on Information and System Security, Vol. 2, No. 2, May 1999, pp 138-158.

RIVEST, Ronald. *Can We Eliminate Certificate Revocation Lists?*, Proceedings of Financial Cryptography 98, LNCS 1465, Springer-Verlag, pp. 178-183, Anguilla, BWI, February 1998

SCHNEIER, Bruce; ELLISON, Carl. *Ten Risks of PKI: What You're Not Being Told About Public Key Infrastructure*, Computer Security Journal, v 16, n 1, 2000, pp. 1-7, www.counterpane.com/pki-risks.pdf

STALLINGS, William. *Cryptography & Network Security: Principles & Practice*, 2$^{nd}$ Edition, Prentice-Hall, 1998, ISBN 0138690170

THAWTE Inc., *Certifying your Credentials through the Thawte Web of Trust*, www.thawte.com/whitepapers/guides/pdfversion/wotguide.pdf

VERISIGN, Inc., *VeriSign PKI Disclosure Statement*, www.verisign.com/repository/disclosure.html

WHITTEN, Alma; TYGAR, J. D. *Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0*, Carnegie Mellon University, Proceedings of the 8th USENIX Security Symposium, August 1999, www.cs.cmu.edu/~alma/johnny.pdf

WINN, Jane K., *The Emperor's New Clothes: The Shocking Truth About Digital Signatures and Internet Commerce*, 2001, faculty.smu.edu/jwinn/shocking-truth.htm

ZIMMERMANN, Phillip R. *The Official PGP User's Guide*, MIT Press, 1995.