3GPP TSG-SA 3 Meeting #28
Berlin, Germany, 6 - 9 May 2003

Tdoc ⌘S3-030233

CR-Form-v7

CHANGE REQUEST

⌘ **SpecNumber** **CR** **CRNum** ⌘**rev** **-** ⌘ Current version: **x.y.z** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: UICC apps⌘ ☐ ME ☐ Radio Access Network ☐ Core Network ☐

Title:	⌘	NDS/AF Trust Model
Source:	⌘	Nokia, Siemens, SSH, T-Mobile, Verisign
Work item code:	⌘	NDS/AF
Date:	⌘	30/04/2003
Category:	⌘	
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction in an earlier release)*
B *(addition of feature),*
C *(functional modification of feature)*
D *(editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	Document the decision for simple trust model made at SA3#27
Summary of change:	⌘	- Included decision in the main body of the TS - Added an informative annex as background for decision
Consequences if not approved:	⌘	

Clauses affected:	⌘	5, Annex A

Other specs affected:	⌘	Y	N		⌘
		☐	☐	Other core specifications	
		☐	☐	Test specifications	
		☐	☐	O&M Specifications	

Other comments:	⌘	

*** First modified section ***

5 Use cases and profiling of the NDS/AF

[Editor's note: This section shall list the security requirements emerging from identified use cases.]

The roaming CA certificate of the owning operator shall be stored securely in the SEG. It defines who is the authority that the device trusts when connecting to the other devices. It is assumed that each operator domain could include 2 to 10 SEGs.

The NDS/AF is initially based on a simple trust model (See Annex A) that avoids introduction of transitive trust and/or additional authorisation information. The simple trust model implies manual cross-certification.

*** Next modified section ***

Annex A (informative): Decision for the simple trust model

A.1 Introduction

In order to document the decision for the "simple trust model", which requires manual cross-certification, this section discusses technical advantages and disadvantages of two basic approaches to providing inter-operator trust for purposes of roaming traffic protection, namely **cross-certification** and a **Bridge CA**. The Bridge CA is an extension of the cross-certification approach, and identified as one of the recommendable solutions for providing inter-operator trust in NDS/AF feasibility study (TR33.810). Taking into account the current state of PKI software and the general need for simple solutions when there is a choice, there is pressure to make the cross-certification without a Bridge CA as the working assumption for the NDS/AF TS. This document discusses the background motivation for such direction.

The direct cross-certification without Bridge CA model is associated strongly with the current practice in the Internet IPs~~S~~ec world, where each IPs~~S~~ec connection is configured with a list of trusted CAs, and anyone with a certificate that has a trust path that can be followed up to such trusted CA (trust anchor) is allowed access. In this model, cross-certification is done at the time the roaming agreement is made. We call this the "**simple trust model**."

The Bridge CA model assumes that all operators wishing to establish a roaming agreement with other operators will first get certified by the Bridge CA for purposes of identification by other operators. This is a necessary preliminary step. Next, when the roaming agreement is done, the operators will configure their IPs~~S~~ec tunnels, with information about which one of the identifiable operators (who have a certificate issued by the Bridge CA) can use that tunnel. This is called the "**extended trust model**", or "separated trust and access control."

This Annex~~document~~ does not discuss the benefits of certificates vs. Pre-Shared Keys. The benefit of cross-certification vs. the explicit listing of roaming peer CAs includes the easier evolution path to a possible eventual Bridge CA model.

A.2 Requirements for trust model in NDS/AF

The following is a list of requirements for the trust model for NDS/AF:

A. *Simplicity and ease of deployment*. PKI brings many benefits when a large number of operators need to tunnel traffic in a mesh configuration, but its adoption should not be hindered by an unnecessarily

complex technical solution. The required technical and legal operations necessary for exchanging traffic with another operator should be as easy and straightforward as possible.

B. *Compatibility with existing software and hardware products.* Unless there are explicit requirements why existing PKI products should be extended to accommodate 3GPP environment, the 3GPP specifications should be accommodated to the existing products. This allows best and cheapest choice of equipment for operators and allows interoperability with non-3GPP environments.

C. *Usable by both GRX and non-GRX operators.* Both operators making use of GRX providers and those without (using leased lines or even the public Internet), should be able to make use of NDS/AF measures to exchange traffic securely.

A.3 Cross-certification approaches

Cross-certification is a process that establishes a trust relationship between two authorities. When an authority A is cross-certified with authority B, the authority A has chosen to trust certificates issued by the authority B. Cross-certification process enables the users under both authorities to trust the other authority's certificates. Trust in this context equals to being able to authenticate.

A.3.1 Manual Cross-certification

Mutual cross certifications are done directly between the authorities and this approach is often called manual cross-certification. In this approach the authority does the decisions about the trust locally. When an authority A chooses to trust an authority B, the authority A signs the certificate of the authority B and distributes the new certificate (B's certificate signed by A) locally.

The down side of this approach is that it often results into scenarios where there needs to be lot of certificates available for the entities doing the trust decisions: There need to be a certificate signed by the local authority for each security domain the local authority wishes to trust.

However, all the certificates can be configured locally and are locally signed, so the management of them is often flexible.

The trust model of manual cross certification is characterized by the clause: "Trust nobody unless explicitly allowed". Issuing a certificate for the authority we wish to trust creates the allowances. The manual cross certification is easy to understand. Also the security of this depends only on the decisions done locally.

A.3.2 Cross-certification with a Bridge CA

The Bridge CA is a concept that reduces the amount of certificates that needs to be configured for the entity that does the certificate checking. The name "bridge" is descriptive; when two authorities are mutually cross-certified with the bridge, the authorities do not need to know about each other, however they can still trust each other because the trust in this model is transitive. (A trusts bridge, bridge trusts B, so A trusts B and vice versa.) The Bridge CA acts like a bridge between the authorities. However, the two authorities shall~~must~~ also trust that the bridge does the right thing for them. All the decisions about the trust can be offloaded to the bridge, which is desirable in some use cases. If the bridge decides to cross certificate with an authority M, the previously cross-certified authorities start to trust the M automatically.

The bridge-CA style cross-certifications are useful in scenarios where all entities share a common authority that everybody believes to work correctly for them. If an authority needs to restrict the trust or access control derived from the bridge-CA, it needs to implement those restrictions separately.

The trust model of bridge-CA can be characterized by the clauses:

"Trust everybody that the Bridge-CA trusts unless explicitly denied". Explicit denials are handled by writing the restrictions (in the form of name constraints) to the certificate issued to the bridge.

"Trust everybody listed in the certificate which I issued to the bridge". Explicit allowances are listed in the certificate issued to the bridge (in the form of name constraints).

Name constraint is a rarely used extension for X.509 certificates. In essence it is a clause that says who to trust or who not to trust based on names on certificates. The fact that they are relative rarely used and the fact that there is so little official documentation about them is a risk. Name constraints also require that there is some organization doing registration of names in order to avoid name collisions.

A.4 Issues with the Bridge CA approach

A.4.1 Need for nameConstraint support in certificates or strong legal bindings and auditing

If no precautions are taken, it is possible that an operator (M) whose Roaming CA has been signed by the Bridge CA (= certified by the Bridge), creates certificates that resemble another operator's (A) certificates, letting M access to operator (B)'s network, even without authorization.

Let's say operator B has the following configuration for access to her subnetwork reserved for handling roaming traffic:

Local-Subnetwork = some ipv6 subnetwork address

TrustedCA's = BridgeCA

AllowedCertificateSubject = O=Operator A or O=Operator C or O=Operator D

Note: The IP addresses of the remote SEGs are not limited, as authentication is done based on certificates, and all trusted operators are allowed similar access. If different foreign operators would require to access different subnetworks, there would be several configuration blocks like the above, with the IP addresses appropriately specified.

Such "AllowedCertificateSubject" feature (the term name is imaginary) is widely supported by PKI-capable IPSec devices.

If Operator M was ethical and used certificates of the following form for her certificates, she would not be allowed in:

Subject: CN=SEG 1, O=Operator M

Signer: CN=Roaming CA, O=Operator M

However, she can fabricate certificates of the following form:

Subject: CN=SEG 1, O=Operator A

Signer: CN=Roaming CA, O=Operator M

Using such certificates would allow full but illegitimate access to Operator B's network revealed for use by Operator A.

Now, there are the following possibilities to circumvent the problem:

1. Checking also the Signer name when authenticating foreign operators, either by a) a proprietary "AllowedCertificateSigner" property or b) support for nameConstraints in the Bridge CA certificate issued to operator M.

2. Establishing strong legal bindings and auditing that would discourage Operator M from such illegitimate fabrication of Operator A certificates.

The problem with solution 1.a is that such "AllowedCertificateSigner" is not commonly supported by current PKI end-entity products, being in conflict with requirement B.

The problem with solution 1.b is that such "nameConstraints" attribute in certificates is not commonly supported by current PKI CA or end-entity products, being in conflict with requirement B.

The problem with solution 2 is that first of all, an organization willing to run a Bridge CA has to be found before any pair of operators can exchange roaming traffic with NDS/AF mechanisms. Next, there ~~must~~shall be established paperwork and auditing procedures to make sure that the exploit described here can be detected. This is in conflict with requirement A. Also, the illegitimate act described could not be technically prevented beforehand.

If name constraints are used, every time a new roaming agreement is made, each operator shall~~must~~ update the certificate they issue for the Bridge, adding the new roaming partner's name into the certificate. From the point of view of one operator, the number of new certificate signing operations is the same whether a Bridge CA or a direct cross certification model is in use.

A.4.2 Preventing name collisions

If name constraints are used to prevent the additional "bureaucracy" involved with the Bridge CA, the names written into the certificate need to be registered with a third party to prevent two operators accidentally or on purpose using the same name in their certificates. This is in conflict with requirement B.

A.4.3 Two redundant steps required for establishing trust

As described in the introduction, with the "extended trust model", each operator shall~~must~~ first be certified by the bridge (authentication), and then as the second step, enumerate the trusted operators when configuring the IPSec tunnel (access control).

For the Bridge CA model to work, there is a need for organization that all the other parties involved can trust - and the trust shall~~must~~ be transitive! If you trust the bridge, you shall~~must~~ also trust the other organizations joining to the bridge via the cross certification. If Operator A and the Bridge CA cross certify with each other, Operator A will automatically trust every other certified operator to obey the rules. And this trust is not related to the roaming traffic tunnel; the tunnel has to be configured independently of the PKI.

So even if we avoid configuring new certificates in the SEG's when we use cross certification, we shall~~must~~ configure and maintain the roaming information in the SEG some other way. And the hard part: How do we combine the trust provided by the PKI and the roaming agreements, because clearly in this case PKI provided trust is not the same as roaming agreements.

We would need two steps:

1. building "trust" through Bridge CA => authenticating the peer SEG
2. specify in the tunnel configuration which peering SEGs we can trust

If the cross-certification is done without a Bridge CA, the steps can be combined into one. What is the additional value of the PKI provided trust (step 1), if the peering SEGs have to be restricted in any case?

A.4.4 Long certificate chains connected with IKE implementation issues

If Bridge CA is used, a Roaming CA certificate has to be sent in the certificate payload in addition to the local end entity (SEG) certificate. This leads in Ethernet environments to the fragmentation of the IKE packet, which some current IKE implementations do not support. It is a problem in the implementation, not the protocol. Even in IPv6, the IKE UDP packets need to be fragmented, posing a potential interoperability problem. Clearly it is not a solution to use a

different protocol, but instead the current implementations should be fixed. Still, taking into account requirement B, it is safer to avoid the problem altogether by not forcing the fragmentation of IKE packets by not using a Bridge CA.

A.4.5 Lack of existing relevant Bridge CA experiences

The Federal PKI in the USA is an example deployment where a Bridge CA is used to connect together CAs of the various federal agencies. It seems to be however the only documented one of its kind, and is connected with very heavy policy documentation and obviously heavy auditing practices, even within one organization, the federal government. The bridge approach is warranted in the case, because they want to automatically check whether some entity has legal rights to sign some document. The number of entities doing cross-domain PKI validation can be several millions, and it is impossible for one validating entity to keep count of individual signers.

In 3G roaming, the situation is in many ways different. When a new operator is born, the other ones do not automatically want to exchange roaming traffic with the new one, but a legal agreement with that operator and a technical tunnel establishment shall~~must~~ be done. In Federal PKI, the situation is the opposite: nothing should need to be done and still be able to trust the other.

In the Federal PKI, the paperwork and processes make name constraints in certificates unnecessary, and IKE is supposedly not used together with the Bridge CA.

A.5 Feasibility of the direct cross certification approach

This chapter discusses the direct cross certification, i.e. manual cross certification approach, where operators are doing the cross certification operation only when agreeing to set up a tunnel with another operator. This tunnel setup is a legal and technical operation in any case, so it is feasible to do also the cross-certification at this time, removing the need for the initial step to cross-certify with the Bridge CA.

There is no technical difference regarding the feasibility of direct cross certification or Bridge CA in the context of GRX or non-GRX environment. GRX might be one possible choice for providing the Bridge CA services.

A.5.1 Benefits of direct cross certification

The benefits of the direct cross certification is that as a mechanism it is well known, supported widely by current PKI products and there even exists an evolution path to a Bridge CA solution if the products come to support it adequately, a Bridge CA is established, and the number of operators becomes so large to warrant the use of the Bridge CA technology. Bridge CA uses the cross certification mechanisms in any case.

The tunnel configuration would look like the following:

Local-Subnetwork = some ipv6 subnetwork address

TrustedCA's = LocalCA

The information of which operator is allowed access is implicit in the direct cross certifications that have been done by the LocalCA, thus authentication and access control are tightly connected. If different foreign operators need to access different subnetworks, there would be separate tunnel configurations with SEG IP address for each, including an "AllowedCertificateSubject" limitation. The "AllowedCertificateSigner" limitation is not needed as necessary in this model (compared to the bridge CA model), since the set of operators who we are able to authenticate are only the ones, we have previously agreed to trust when doing the direct cross certification. In the bridge CA case, the set of operators we are able to authenticate includes all operators who have joined to the bridge.

A.5.2 Memory and processing power requirements

In case of direct cross certification, each operator shall~~must~~ store the certificates issued for the other operators locally. They could be stored in the SEG devices, or then in a common repository.

If an operator makes roaming agreements with 500 other operators, this would require roughly 1000 kilobytes of memory, if the operator signs the certificates herself, and one certificate takes 1 kilobyte of memory. This should be quite feasible taken into account the high-end nature of SEG hardware.

Processing power benchmark for validating certificates:

Hardware: 800 MHz Pentium III, 256 MB of memory.

200 x 1024-bit RSA certificates, 1 Root CA (operator's own CA), 200 Sub CAs (other operator CAs) and 200 end entity (SEG) certificates. Also CRLs were verified. Both certificates and CRLs were loaded from disk during the test. The whole test took 3.5 seconds, with probably disk I/O taking most of the time.

In this test 200 certificate chains were validate up to the trusted root.

A.5.3 Shortcomings

As discussed in the previous section, the Bridge CA approach saves memory or storage space in SEGs, because all the other operators Roaming CA certificates do not need to be stored with other operators. Just the Bridge CA certificate would be stored, and other certificates retrieved during IKE negotiation.

A.5.4 Possible evolution path to a Bridge CA

If needed, it is possible to take the Bridge CA into use gradually, given that the support by PKI products becomes reality. From one operator's point of view, the bridge CA would be like any other operator so far, and a cross-certification would be made, but additionally the name constraints in the certificate issued for the Bridge CA should be updated every time a new roaming agreement is made.

*** End of modified section ***