List of Slides

AAA servers need to exchange and parse policies

In some cases, sending out a request to have a policy evaluated at a remote server is not feasable:

- Proxy environment
- Home agent not reachable
- Bandwith/latency issues

Goal of the policy draft

In order to be able to exchange policies between AAA servers and parse them, a standard for policies must be defined. The capabilities that are needed in policies are investigated, and some suggestions about the format in which policies are represented are given.

What are policies?

- Logical expressions
- Actions attached to the result of subexpressions
- Can do computations on variables
- Actions can send commands to ASMs.

Possible representations of policies

- DNF–notation (Disjunctive Normal Form)
- A common expression (like in C, Java, Pascal etc.)
- XML
- Java

Problems left

- Policy conflicts: Must AAA servers resolve conflicts or must the implementors of the policies take care of it? IBM's Courteous Logic Programming suggests a language to handle conflicts, but at the same time it mentions that everything written in CLP can be translated into normal logical expressions.
- How to represent actions: Use AVPs? Some application specific interface?
- Policy language: Do we define our own language in XML or do we use an existing platform independent language like Java? Versatility versus security tradoff?
- Termination of evaluation of distributed policies

XML

Advantages of XML:

- Off-the-shelf parsers available
- Light-weight
- Allows strict definition of the policy language

Disadvantages:

- XML is just a preparser
- Extending the language means extending and spreading new DTDs

Java

Advantages of Java:

- Very versatile
- Byte-compiled: optimizes execution time
- Easily extendable

Disadvantages:

- Hard to check for validity
- Needs a restricted environment

Analogy with smartcards

- Smartcard is a AAA server
- Card applets are policies (written in Java)
- APDU commands are requests
- Uploading a new applet is pushing a policy

This shows that smartcards already provide most of the things we want in AAA:

- End-to-end security
- Restricted environment
- Light weight implementation
- Pushed policies

Termination of distributed policies

What about AAA servers who want confirmation from each other? (Father says: ask your mother, mother says: ask your father.)

- Use a TTL, but how to handle parallel spawning of policies then?
- Use a real timeout, but this might not be helpful in very fast networks.
- Loop detection?