General Idea

pledge is an interesting tool from the openbsd world, it allows a program to pledge

to operate in a restricted fashion. If those restrictions are violated in any ways, the underlying operating system will SIGABORT

the violating process. This from a security point of view is quite nice, and having a similar tool to interact with a distributed operating system would be handy.

Terminology

Before I propose my ideas, I will define some terms:

1. Operator :: A person or collective who control some application or shared knowledge base.
2. Application :: A set of public resources, or database, or normal program that is under the operator's control.
3. gas :: Some computational payment to run public code.
4. NOTE: Not all public applications will charge users gas.
5. NOTE: Not all public applications will charge users gas.

Concrete Ideas and Vision

I believe for the Anoma OS, we should be able to pledge to a variety of things

1. The code executed only touches some restrict knowledge/resource set
2. Certain pathways, namespaces, primitives, etc, can be barred from running.

If the code that pledged to the operation violates the pledge, then any gas consumed during the execution is forfeit, and the operation is rolled back, as if it never occurred.

For the Feature set:

1. Operators decide what the application's pledge environment is like, and can change it whenever they so wish
2. They can do this by vote if it's a staked committee, or arbitrary if it's under a single entities control
3. They can do this by vote if it's a staked committee, or arbitrary if it's under a single entities control
4. Code sent in by users, can pledge to be in an even bigger restriction set inside some Application
5. If the application is some web operating environment, where users can submit extra functionality and applications, then a user calculator program may disable time calls or other such functions.
6. If the application is some web operating environment, where users can submit extra functionality and applications, then a user calculator program may disable time calls or other such functions.

The API for pledges should be quite simple, and there should be tools built to visually see what applications have pledged to what, and tools made to see what any particular symbol/function may violate (for functions like eval, this would be potentially everything, but users can tag specific instances to directly tell the system rather than it infering it.)