Table of Contents

LCG Firewall System

Introduction

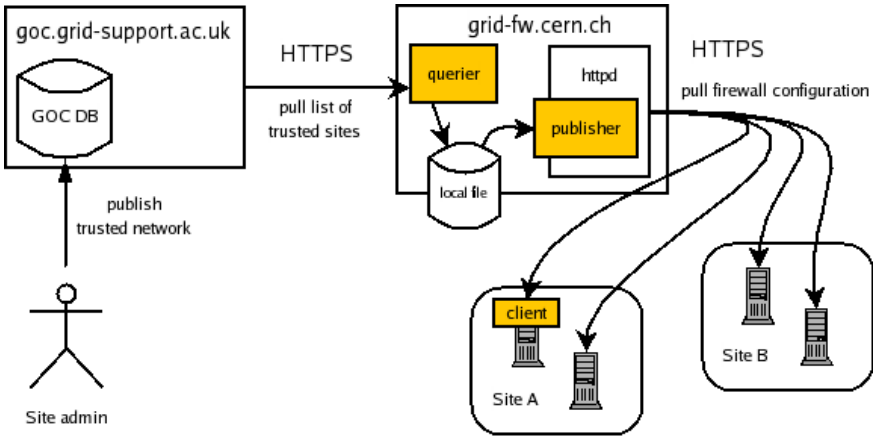
In order to enhance the protection of the Grid from external attacks, a firewall service for the Grid components is desired. This firewall would enable them to isolate the changing list of external hosts that are part of the Grid from the rest of the Internet.

Such a firewall would need to be distributed amongst the different participants, and its rules would need to be built and updated dynamically in order to:

- include the different (sub)networks that may be used by a site;
- dynamically integrate and exclude Grid sites;
- insure that the rules have not been tampered with.

Proposed solution

The GOC database contains a *Firewall* entry, which is intended to store the trusted network for the site as a comma-separated list of IP/MASK addresses. The firewall system would consist of a server host that queries periodically the GOC DB, checks (as possible) that the network addresses are valid, and creates and publishes a iptables configuration file that can be pulled by a client in order to be isolated from outside the grid.



Security considerations

- All the transactions between hosts will be mutually authenticated through exchange of X.509 certificates.
- Download of the published file is only authorized to clients known to belong to the grid.
- The server must validate that the network addresses published in the GOC DB are meaningful, in order to avoid addresses like '0.0.0.0/32' to interfere with the proper configuration of the clients.
- The client must check that the downloaded configuration doesn't leave it without network access.

System components

At the server side:

1. A **querier** process, run periodically as a cron job, that:
   - ♦ queries the GOC DB to get the list of networks that compose the grid,
   - ♦ performs the required sanity checks to the data,

    - creates a iptables configuration file, and
    - stores the file in a location where the publisher can read it.
2. A **publisher** web application, that:
    - handles the authentication and authorization of the clients, and
    - publishes the configuration file.

At the client side:

1. A **client** process, run periodically as a cron job, that:
    - pulls the published configuration,
    - verifies that the new configuration allows it to have network access,
    - installs it in the host, and
    - provides a rollback mechanism when the new configuration fails.

Implementation

- The **querier** has been implemented as a Python script that runs on the server as a cron job.
- For the **publisher**, it was preferred not to have a web application, and rather rely on the web server to serve the published file. The GridSite module handles the authentication of the clients using Grid credentials. Clients that are not authorized to see the rules are excluded using mod_access.
- The **client** has been implemented as a shell script that runs as a cron job.

The implementation can be found in /afs/cern.ch/user/r/rbonvall/public/gridfw/

Proposed development steps

Querier	
Analysis of validation of published addresses	Done
Analysis of rule creation details	Needs review
Coordinate with GOC a more convenient interface	Pending
Implementation	Done
Testing	Pending

Client Analysis of safety measures to avoid breaking local fw Done Implementation Done Testing Pending

Publisher Analysis of client authorization mechanism Done Definition of web server configuration Done Implementation of the application Not needed Testing Pending

Integration Analysis of Querier-GOC authentication issues Done Analysis of Publisher-Client authentication issues Done

---

-- Main.rbonvall - 11 Sep 2006

This topic: LCG > LCGFirewallSystem
Topic revision: r23 - 2007-01-15 - unknown