R|R|Z|N

Dynamic Firewall Overview

Gian Luca Volpato, Christian Grimm
D-Grid Integration project

Started in September 2005 as part of the German e-Science program funded by the federal ministry of education and research.

D-Grid consists of:

- **1 integration project**
- **6 community projects**
  - HEP
  - Astrophysics
  - Climate research
  - Medicine & Life sciences
  - Engineering applications
  - Humanities

The integration project aims to develop, build, establish and sustain a general Grid infrastructure for German scientists.

Dynamic Firewalls

R|R|Z|N

Goal: protect a network so that it appears completely inaccessible from external systems but still responds to trusted clients.

Allow external connections on-demand.

Current solutions:

- Dynamically add/remove firewall filtering rules.
- Place a new component outside of the network that relays the communication flows.

Three proposals:

1. Dyna-Fire
2. Cooperative On-Demand Opening
3. Generic Connection Brokering

Developed by University of Buffalo.

Central database storing information about users and resources belonging to a VO.

Daemon running on the same host of the firewall and monitoring connection requests.

It adds/removes filtering rules.

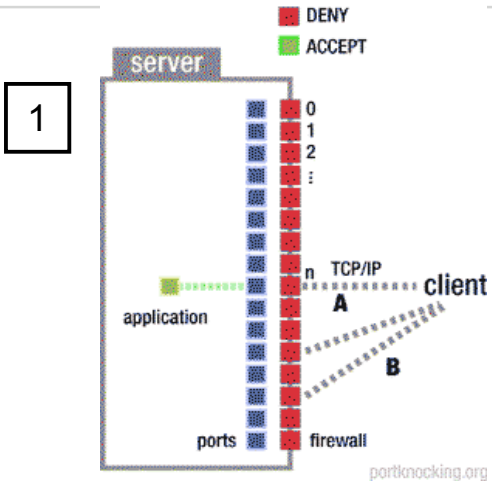
A connection through the firewall is allowed when:

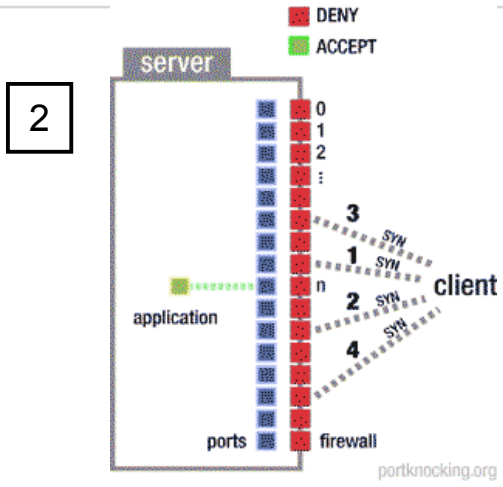
- **the client request is authenticated**
- **the client application is authorized according to the central database**

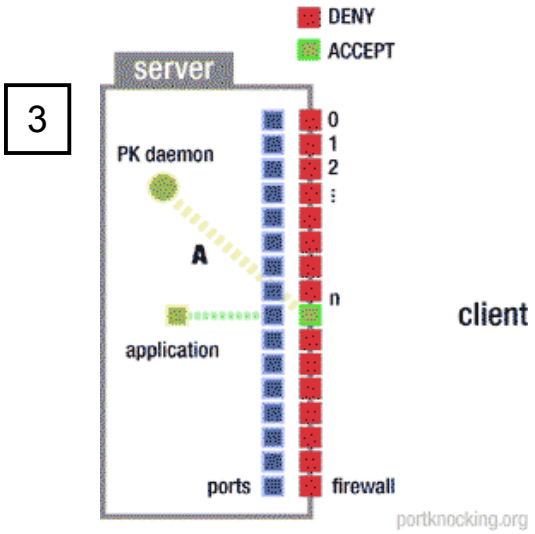
Port knocking is used as the signaling protocol to convey connection requests.

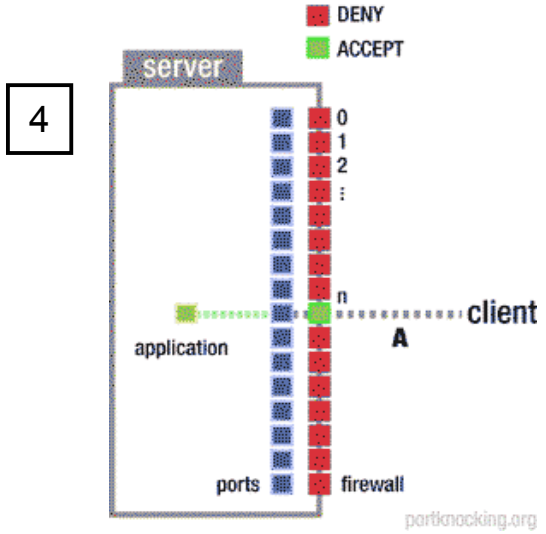
Port knocking

R | R | Z | N









Dyna-Fire evaluation

R|R|Z|N

Pros:

- **No modification of server applications.**
- **Integration with Globus gatekeeper.**

Cons:

- **Computing overhead for monitoring connection attempts.**
- **Reservation of a port range for port knocking communication.**
- **Port knocking is unidirectional: no protection against message replay attacks.**
- **Distribution of knocking sequences to clients.**

Developed by University of Wisconsin and Argonne National Laboratory.

Agent running in the same host of the firewall and listening on the only one open port.

It adds/removes filtering rules.

It maintains a list of applications that can traverse the firewall.

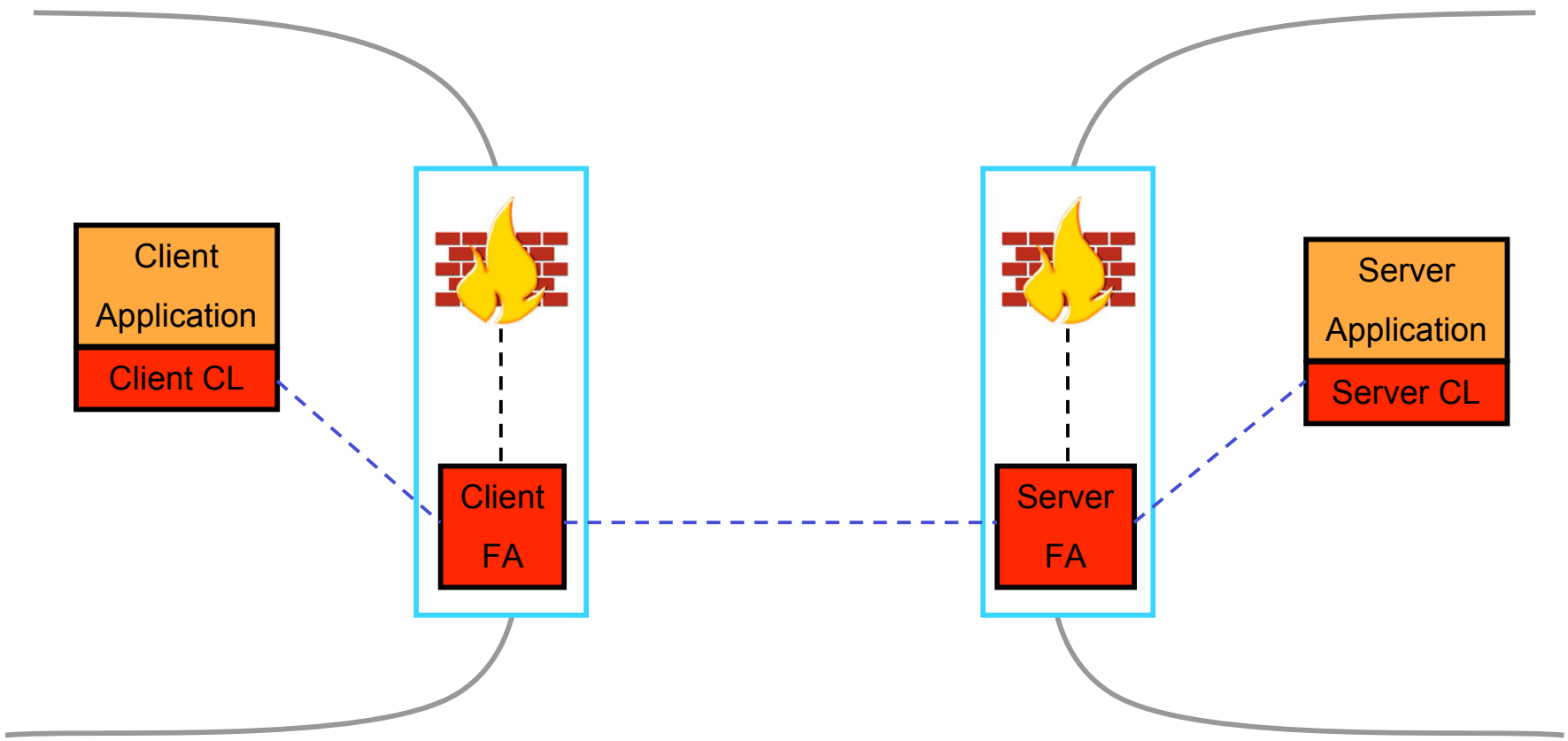
Communication with the agent occurs over a mutually authenticated TCP channel. Optionally, commands can be encrypted.

A connection through the firewall is allowed when:

- **the client request is authenticated**
- **the client application is authorized according to agent's list**

CODO architecture

R|R|Z|N



CODO

Son, Allcock, Livny

CODO evaluation

R|R|Z|N

Pros:

- **Strong authentication of all communicating parties.**
- **Control of both inbound and outbound connections.**
- **Cooperation with stateful firewall minimizes the amount of filtering rules to be added/removed.**

Cons:

- **Server and client applications must be modified.**
- **Manually update routing table to decide when CODO commands must be used.**

Developed by University of Wisconsin.

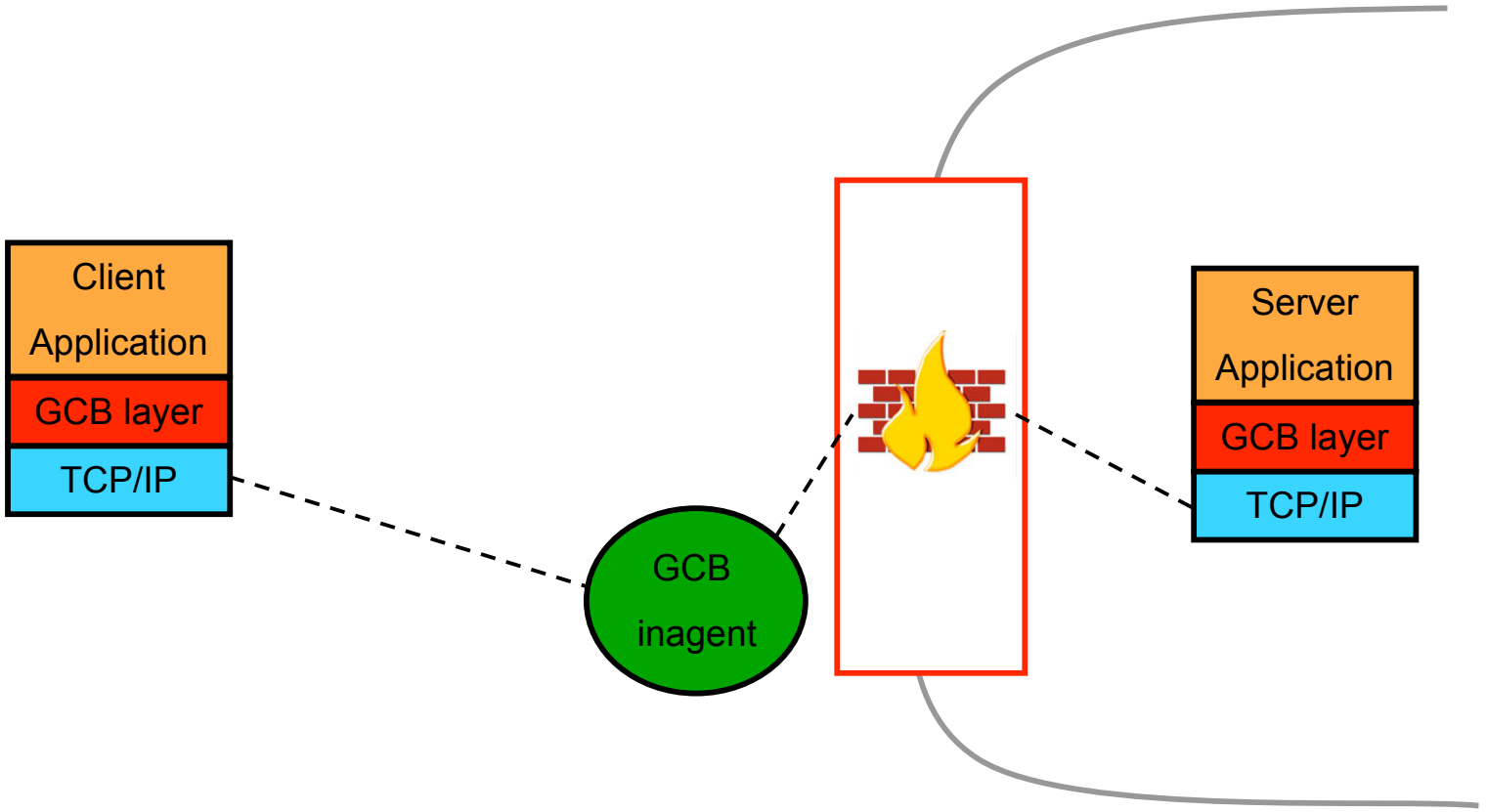
Brokering system (GCB inagent) outside of the protected network.

This system

1. **reverses the direction of the connection according to the network configuration of server and client;**
2. **may act as a relay point;**
3. **maintains a list of applications that can traverse the firewall.**

No filtering rules are ever added/removed to the firewall.

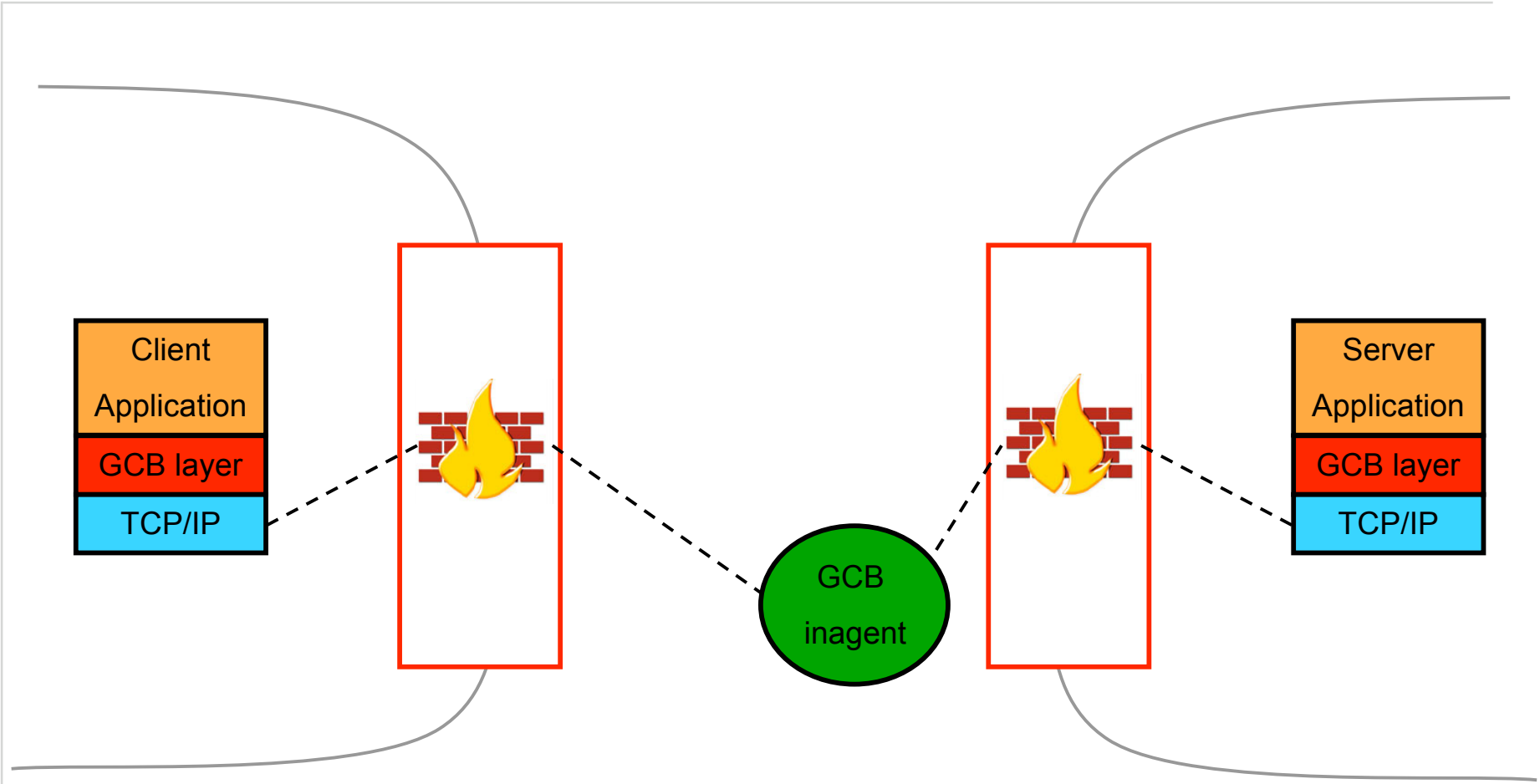
GCB architecture (1)



GCB:
Son, Livny

GCB architecture (2)

R|R|Z|N



GCB:
Son, Livny

GCB evaluation

R|R|Z|N

Pros:

- **Support of legacy client.**
- **Independent from firewall implementation.**

Cons:

- **Outbound connection must always be allowed.**
- **Neither authentication nor authorization of connection requests.**
- **Only one instance of a well-known service per GCB inagent.**

Conclusions

R|R|Z|N

Dyna-Fire and CODO implement a signaling protocol between the applications and the firewalls. Signaling is used to notify running applications and to request connections.

Signaling messages must be always authenticated and requests must be always authorized.

GCB introduces a new element in the network (inagent).

It is interesting for legacy support but it is weak in terms of security.

Dyna-Fire and CODO should be further investigated.