Duke Systems

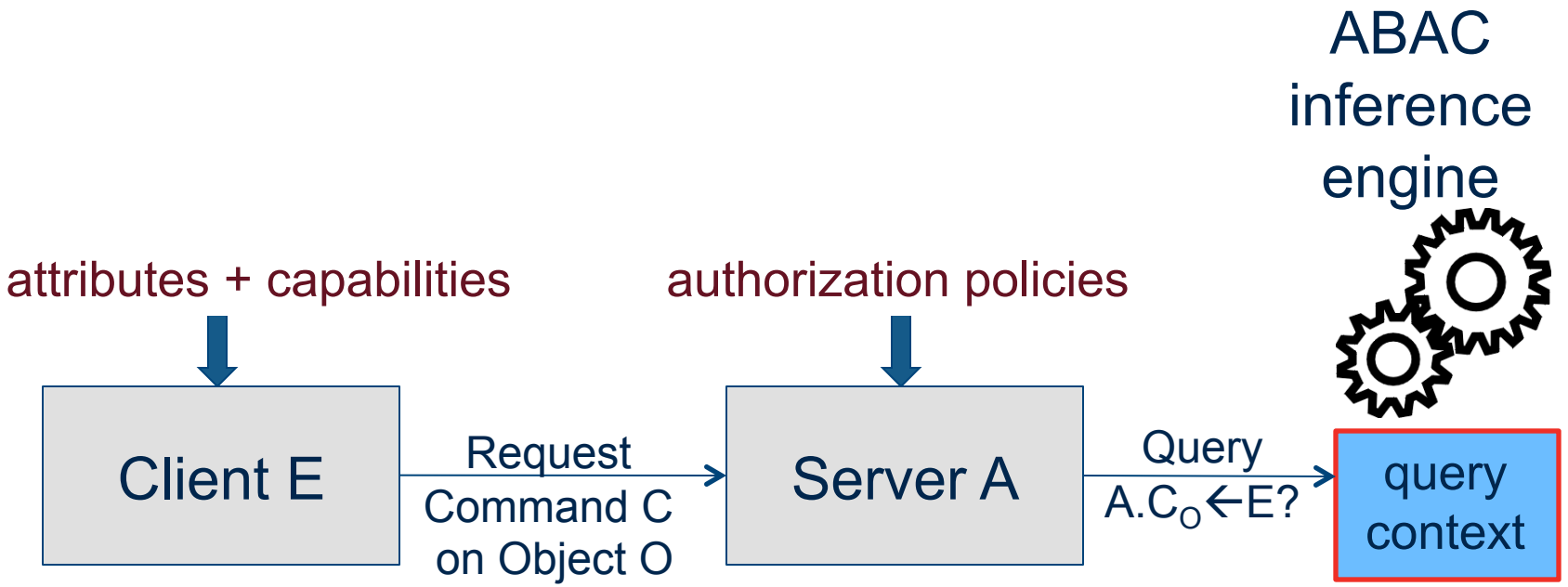
Accountability and Authorization
GEC 12

Jeff Chase

Duke University

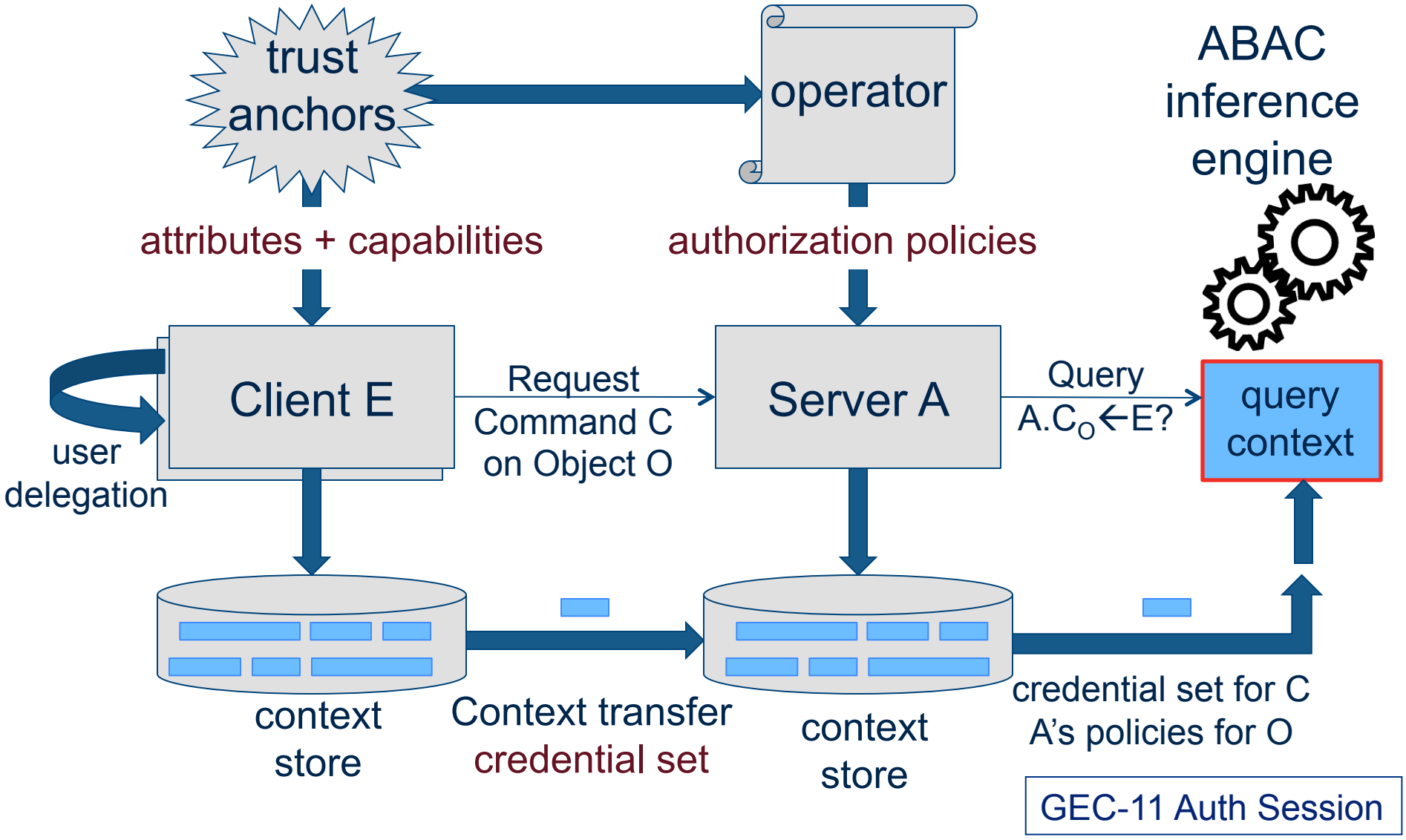
Thanks: NSF TC CNS-0910653

Authorization with ABAC



A.B←A.C

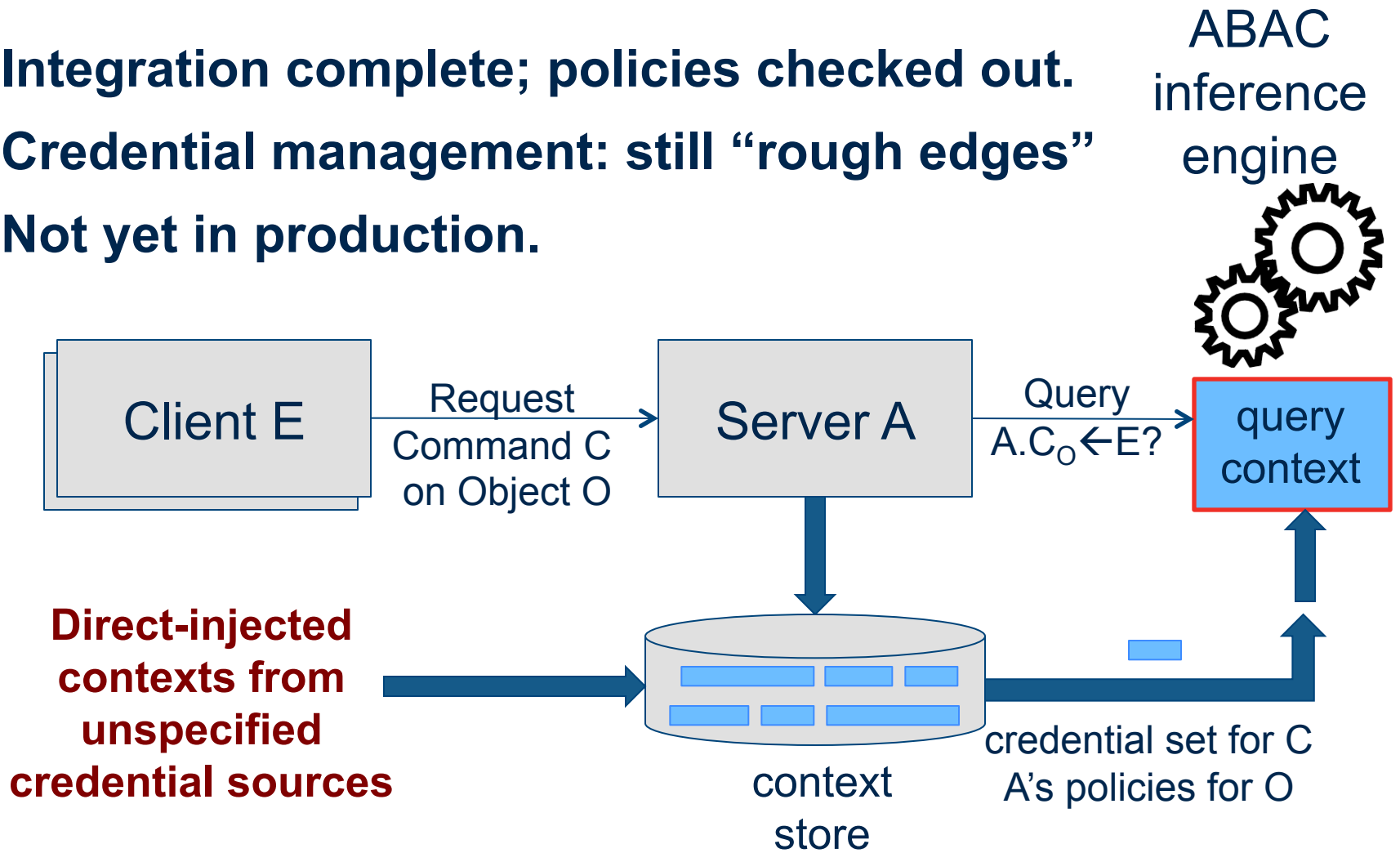
ABAC in Context



GEC-11 Auth Session

ABAC in ORCA

- **Integration complete; policies checked out.**
- **Credential management: still "rough edges"**
- **Not yet in production.**



ABAC in GENI

- ABAC is a powerful declarative representation that can capture the GENI authorization/trust model.
- It saves a lot of code, provides a rigorous foundation, and preserves flexibility for future innovation.
- It should be easy for users, although we need some better tools there. (E.g., to delegate rights.)
- Libabac "works off the shelf".
- In progress: policies for safe operational deployment.

ABAC policies

- The basic mechanisms are in place:
  - Simple user certs issued by identity portal
  - Slice capabilities with delegation
  - Groups (projects) with flexible membership
  - Delegation of capabilities to groups
  - Trust structure: AM endorsements, etc.
- Some details to resolve:
  - Specific user/group attributes
  - Their use in resource allocation policy
  - Slice credentials in ABAC
- Open question: CH role

All is not sweetness and light

- **But** it's based on signed credentials (certs).
  - And on X.509….
- That presents challenges for which there is no perfect solution.
- And so there is:
  - Fear
  - Uncertainty
  - Doubt

Image used without permission or right from 'Stories of the Gods and Heroes' by Sally Benson, 1940, Dial Press. Reprinted in Colliers Junior Classics, 'Legends of Long Ago', 1962.

Credential management

- Each principal possesses many certs.
  - Which ones are relevant to a given request? Where are they?
- Some of those certs are delegated.
  - Server needs even more certs to validate delegation chain.
  - Those certs belong to someone else. Server gets them…how?
- Credentials expire.
  - How to automate renewal?
- People change…and people lose their keys.
  - Revocation: how to do it fast and make it stick?
  - How to rebuild credentials with new keys?
  - How to keep the system safe in the real world?


PERIGO
DANGER
FALÉSIA INSTÁVEL
Não se aproxime
UNSTABLE CLIFF
Keep away
Sesimbra
câmara municipal
www.cm-sesimbra.pt

Summary: what's on the table

1. Policies for safe operational deployment
2. "Clearinghouse" (CH) role
   - Synchronous intermediary?
   - Credentialing authority?
   - How much does it know about:
     - Users and groups?
     - Powers of users and groups?
3. Credential management
   - Revocation, renewal, key rotation
   - Principal names vs. public keys

A1. Every action that allocates a resource is taken with the public key of a registered GENI experimenter (E). Some GENI-authorized identity provider (I) knows the binding to an actual human (H) who can be punished. Given E, G*OC can determine H. Or at least GOC can determine I, which can determine H.

A2. Every action that allocates or uses a resource is taken in the context of a slice (S). Given S, GOC can determine a human project leader who is accountable for S.

A3. Every conforming AM logs all resource-related actions together with the public key E that took the action, and the slice S that was the context for the action. These logs are available to GOC.

A4. Each GENI service publishes to the GOC all credentials that have been used by any E to take any action within GENI. From these credentials GOC can determine how and why E was authorized to take the action.

A5. Various monitoring facilities record interesting events at various levels, and associate them with a slice S. These records are available to GOC.

A1. Experimenter accountability

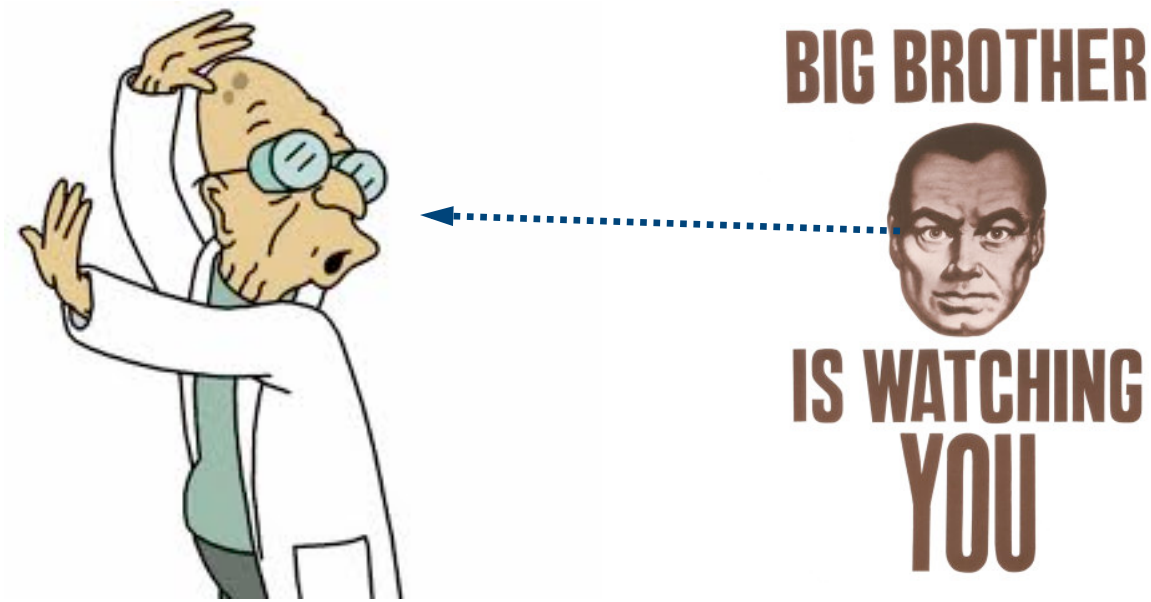
- Every action that allocates a resource is taken with the public key of a registered GENI experimenter (E).
- Some GENI-authorized identity provider (I) knows the binding to an actual human (H).
- Given E, G*OC can determine H. Or at least GOC can determine I, which can determine H.

GOC.registeredPrincipal
E→H

A2. Group accountability

- Every action that allocates or uses a resource is taken in the context of a slice (S).
- Given S, GOC can determine a human project leader L who is accountable for S.

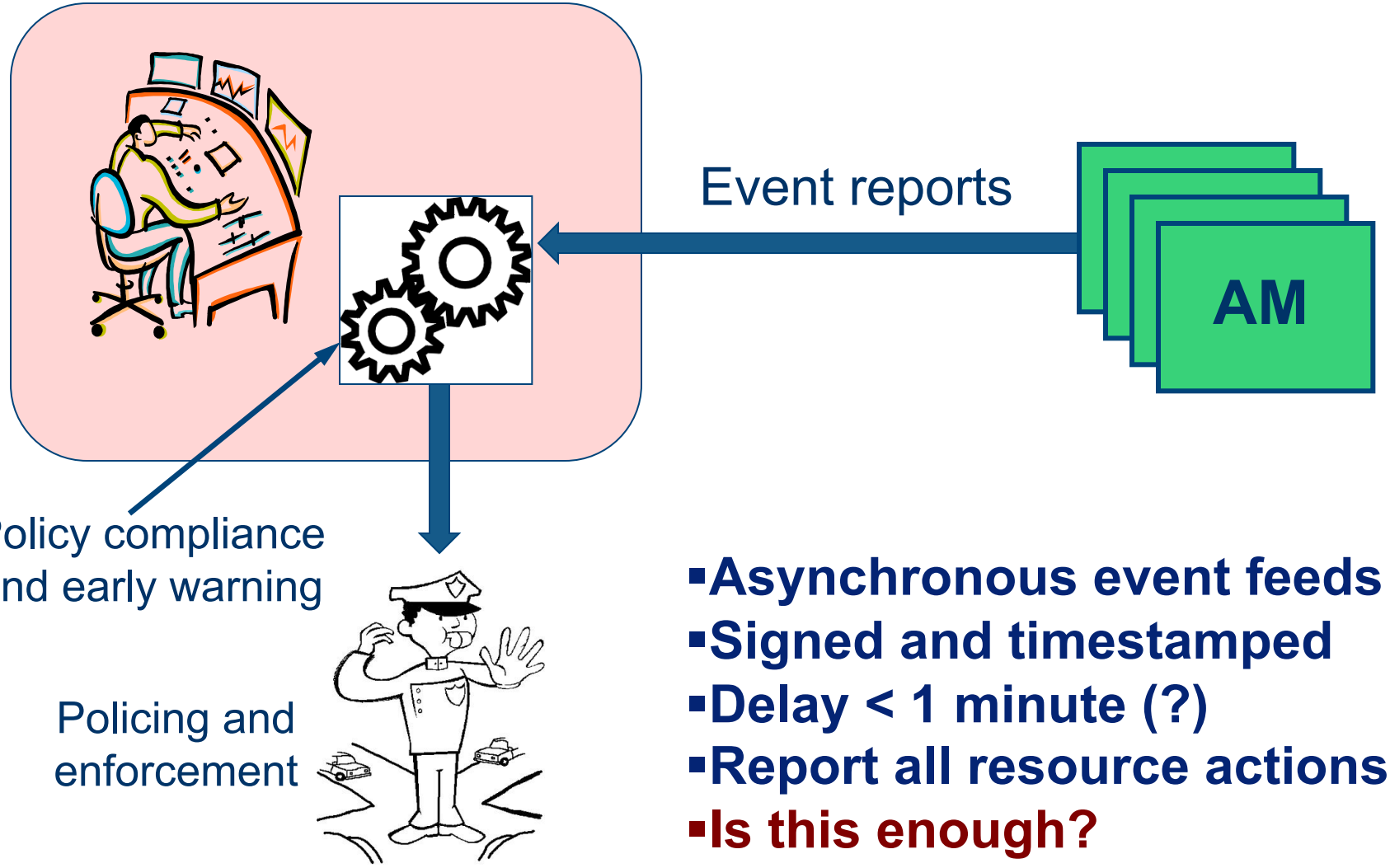
GOC.PI(G)
S→G



A3. GOC learns E and S

- **Every conforming AM logs all resource-related actions together with the public key E that took the action, and the slice S that was the context for the action. These logs are available to GOC.**

CH: Auditing and Accountability



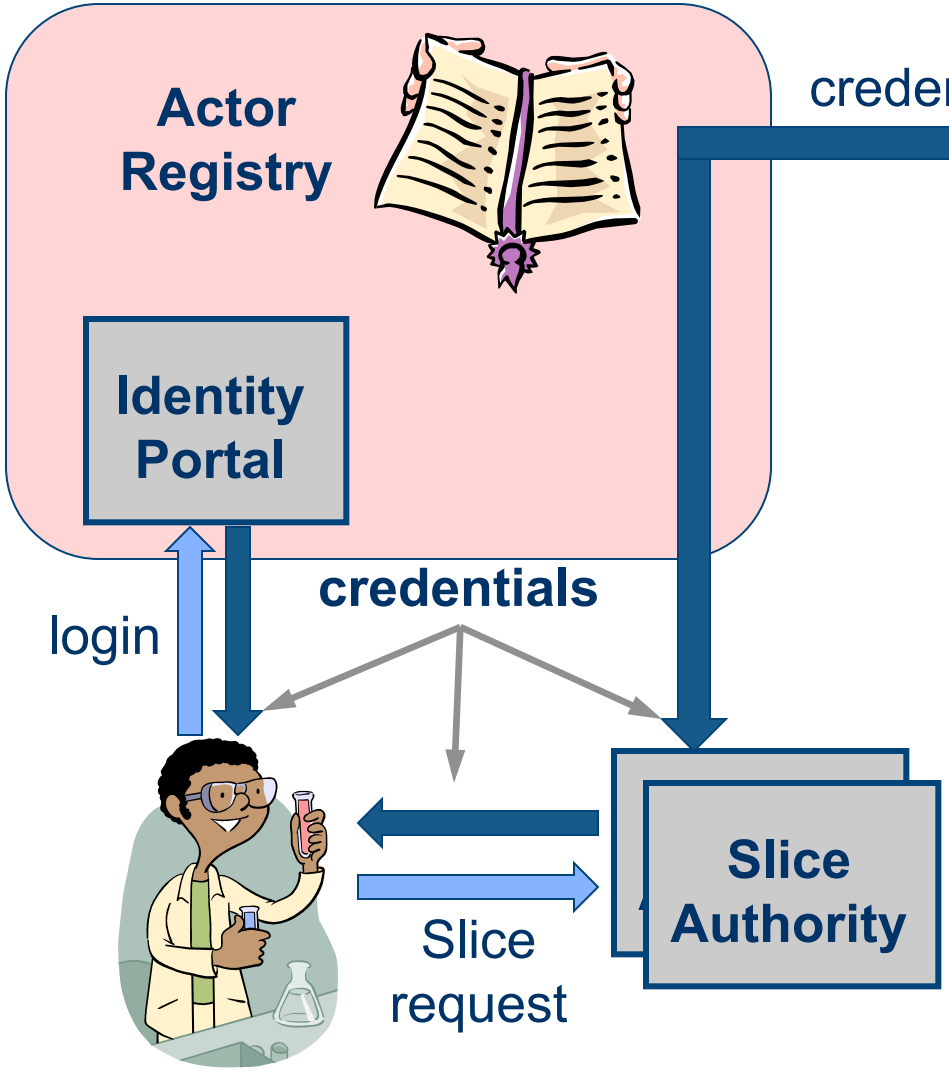
- **Asynchronous event feeds**
- **Signed and timestamped**
- **Delay < 1 minute (?)**
- **Report all resource actions**
- **Is this enough?**

A4. GOC learns all delegations

- **Each GENI service publishes to the GOC all credentials that have been used by any E to take any action within GENI. From these credentials GOC can determine how and why E was authorized to take the action.**

CH: Credentialing

A.B←A.C



- Endorse AMs, SAs, IdPs
- Issue user credentials
- Groups (“projects)
- Execute agreements etc.
- Is this enough?

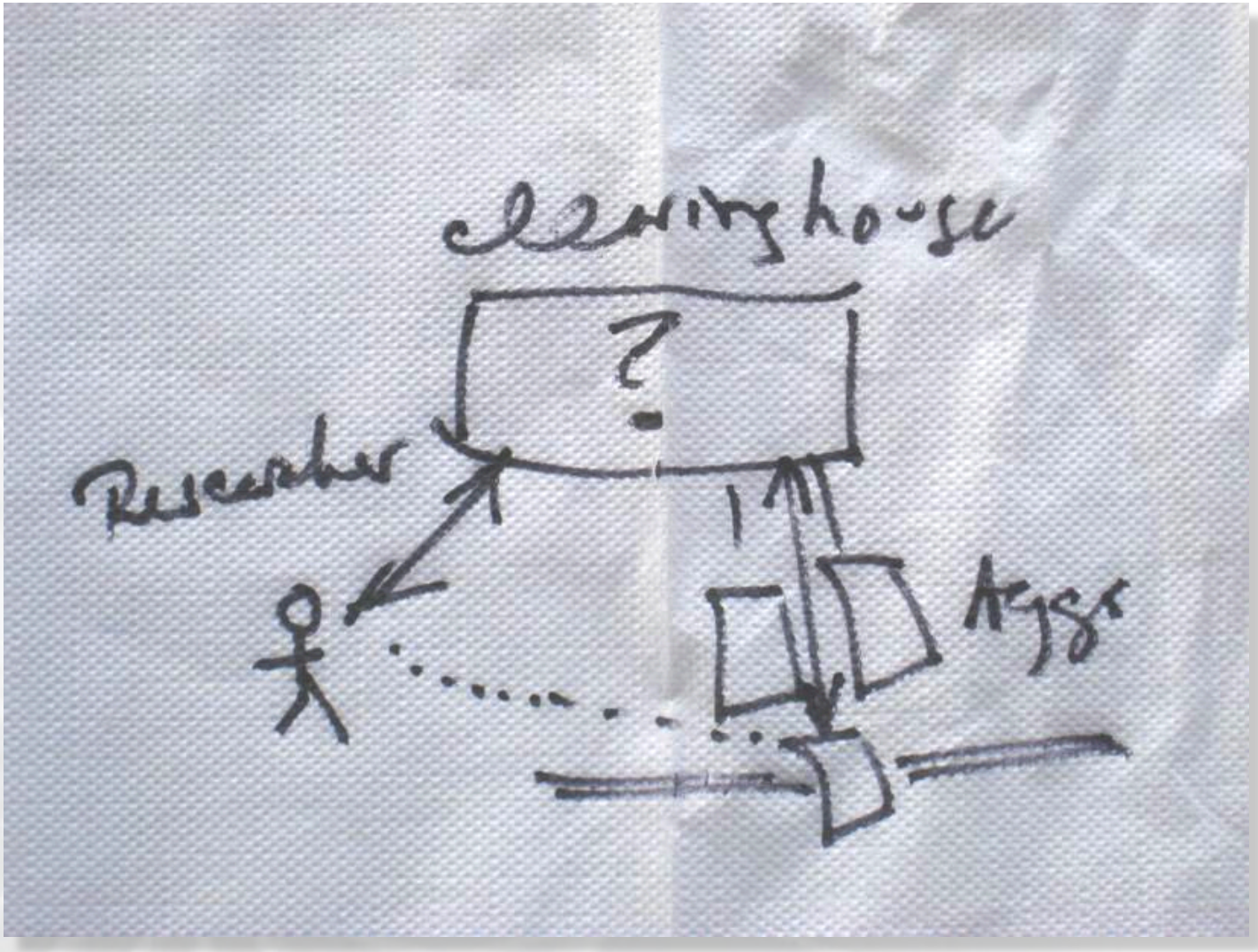
Summary and a look ahead

- Signed security assertions enable decentralization
  - Essential CH functions distill down to credentialing.
- **Problem**: we need Big Brother, at least for now.
  - **Solution**: event logs and registeredPrincipal
  - But Big Brother needs the certs to identify other accountable parties.
  - And Big Brother is nervous about PKI…
- **Proposal**: public always-on credential store
  - Cert query → context
  - Short-term caching, configurable TTL
  - Refresh for renewal
  - “Poisoning” for revocation

Clearinghouse (CH): Position summary

- GPO requires strong central control over GENI in the near term.
- Even so, the architecture should enable a transition to decentralized deployments in the future.
- Consider CH functions separately. Focus on safety.
- Resource management is wide open → see ORCA.
- Other essential functions are “easy” given a strong core for identity and trust (off-the-shelf).
- Operational concerns for credential management (e.g., revoke/renew) are the crucial focus.

“Clearinghouse” has always been a shorthand for “that which manages federation”.



Chip Elliott @ GEC4

Standard issue BBN napkin

Clearinghouse Functions

A. Auditing and accountability

GOC receives event logs (audit trails) distributed by pub/sub.
Avoid central authorization services where we can.

B. Brokering requests and allocations

Resource quotas/caps, sharing policies: rarely discussed in GENI. ORCA uses ticket-granting brokers. Central authorization services are useful here!

C. Credentialing users and services

Federated identity (e.g., Shib) + ABAC credentials

D. Discovery/Directory of resources/services

Dissemination: non-essential, cannot subvert system → replaceable and "easy" to build scalable implementations