Preventing coordinated attacks via alert correlation

J. Garcia, F. Autrel, J. Borrell, Y. Bouzida

S. Castillo, F. Cuppens, G. Navarro

{jgarcia,jborrell,scastillo,gnavarro}@ccd.uab.es,

{fabien.autrel,yacine.bouzida,frederic.cuppens}@enst-bretagne.fr

Main Points

- Introduction
- Classical architectures
- Prevention framework
- Current Development
- Conclusions

Coordinated Attacks

- "*Combination of actions performed by a malicious adversary to violate the security policy of a target computer system.*"
- Networks resources can become an active part of a coordinated attack
- E.g. An attack might start with an intrusion

  ⇒ Nodes have to be monitored
- A global view of the whole system is needed for detection

  ⇒ Collection and combination of events from different nodes

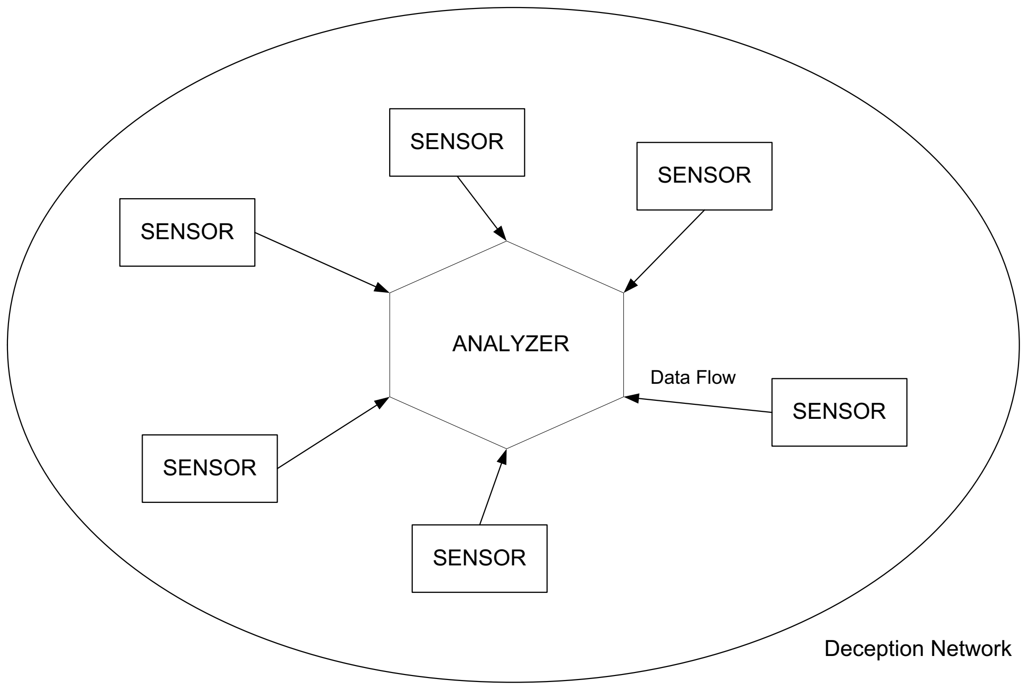
Components needed to prevent coordinated attacks

- Sensors (host, application or network based)
- Analyzers (misuse or anomaly based)
- Managers (data consolidation and alert correlation)
- Response units (active or passive reaction)

- Intrusion Detection Systems use these same components to prevent a node getting compromised by an attacker

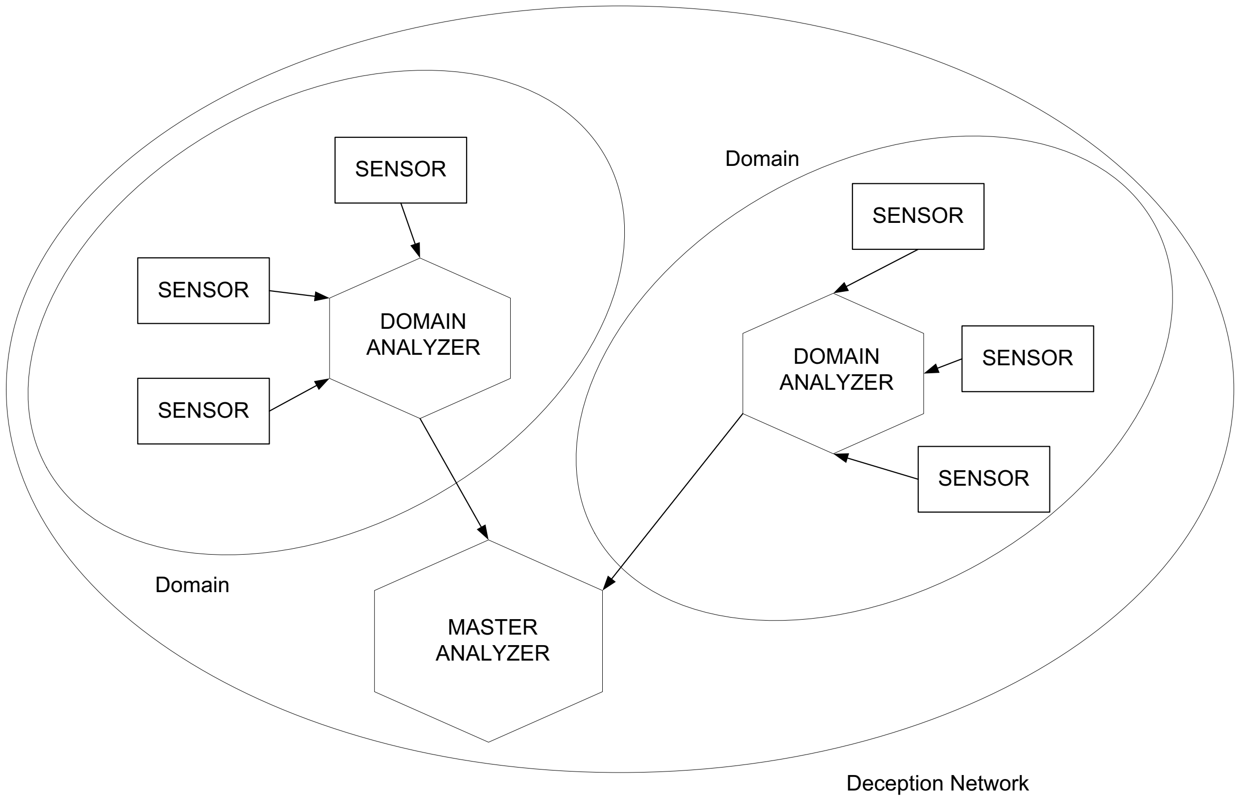
$\Rightarrow$ We use these components to prevent a compromised node becoming an active part of a coordinated attack.

Centralized event correlation



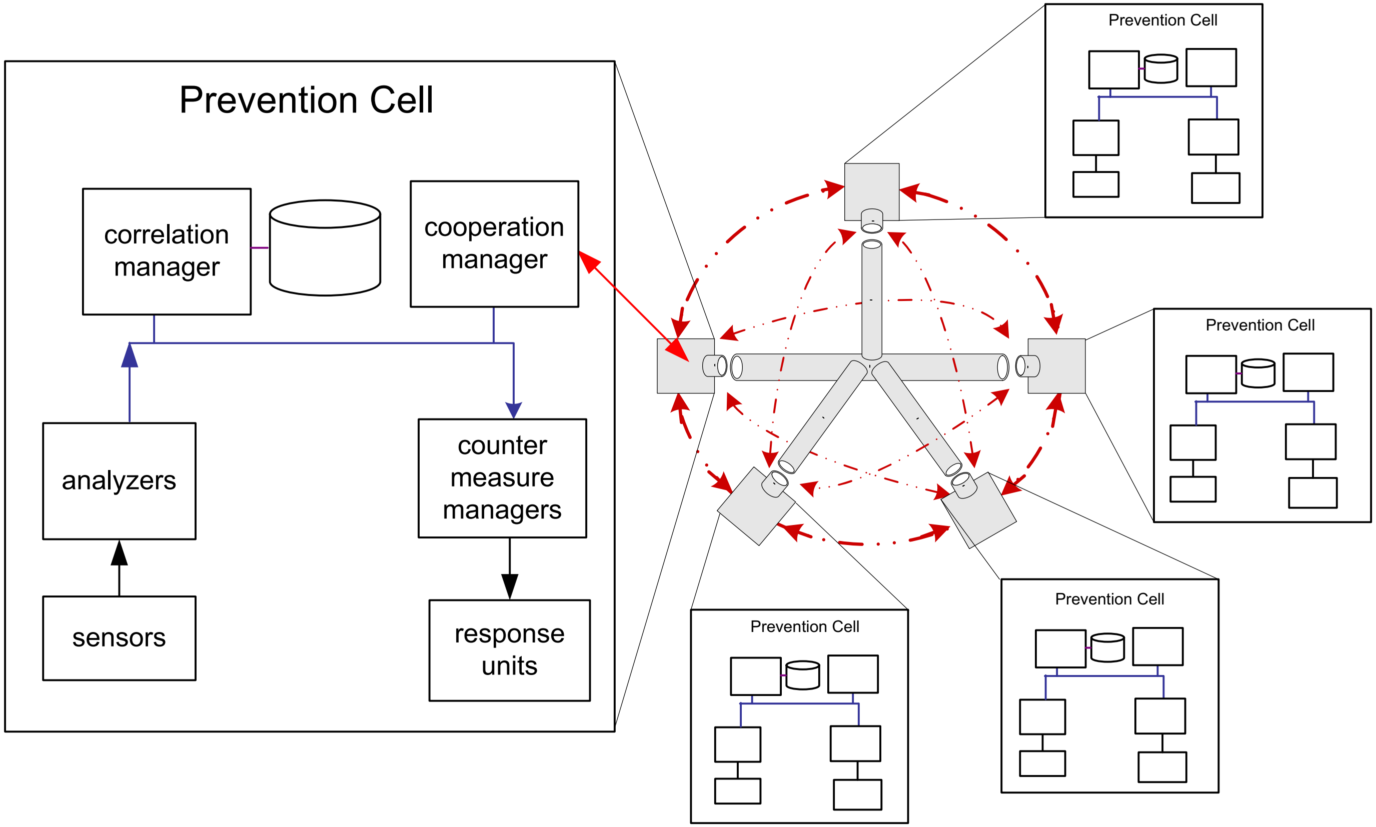
- DIDS - University of California, Davis (1991)
- STAT - University of California, Santa Barbara (1992)

Hierarchical event correlation



- EMERALD - SRI International, California (1997)
- AAFID - CERIAS, Purdue University (1998)

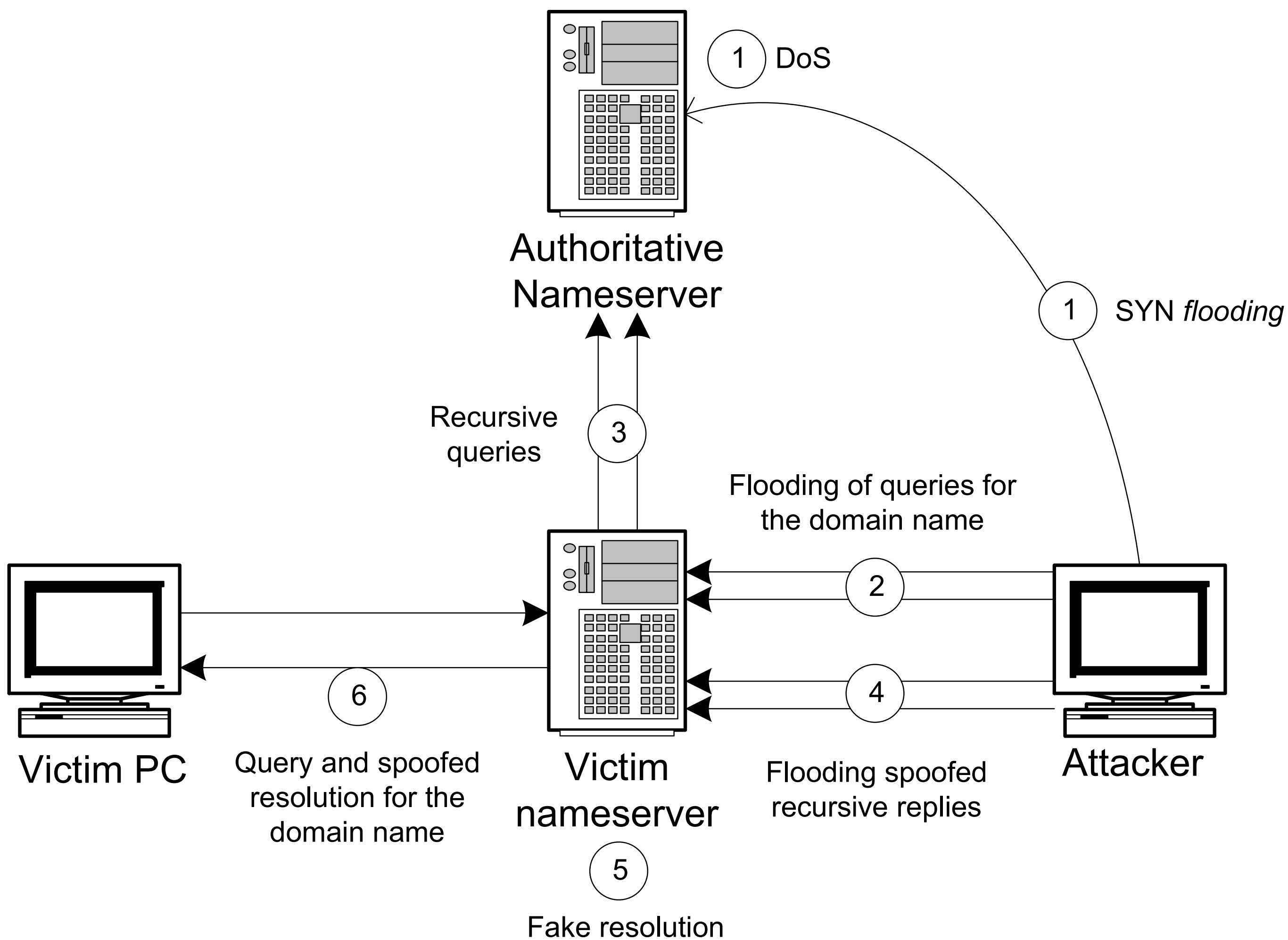
3. - Prevention Cells System



- Message passing architecture

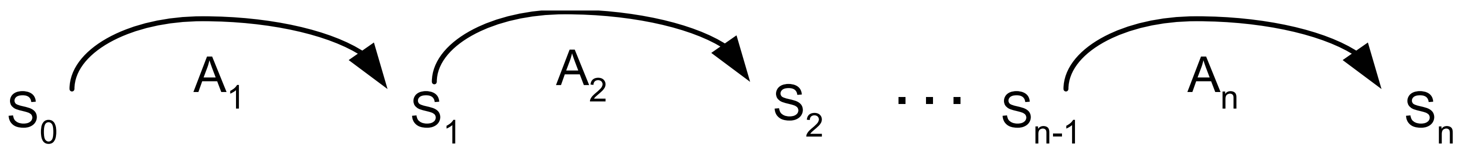
  ⇒ The detection process can be completely distributed

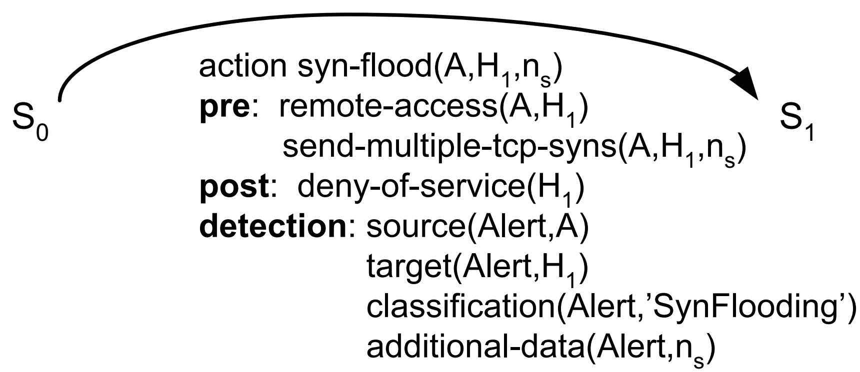
Sample scenario



Detection Process

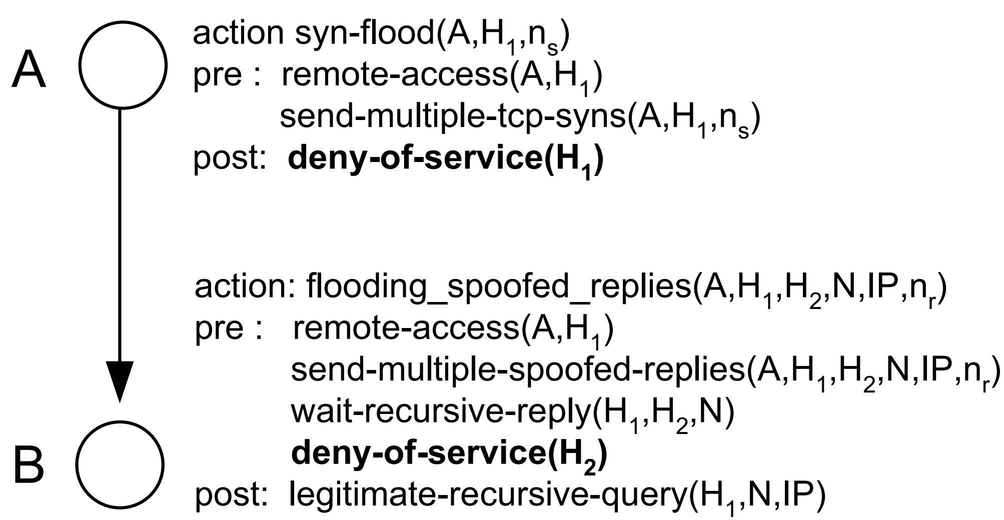
- Find the set of actions which transforms the system from an initial state S_0 to a final state S_n.





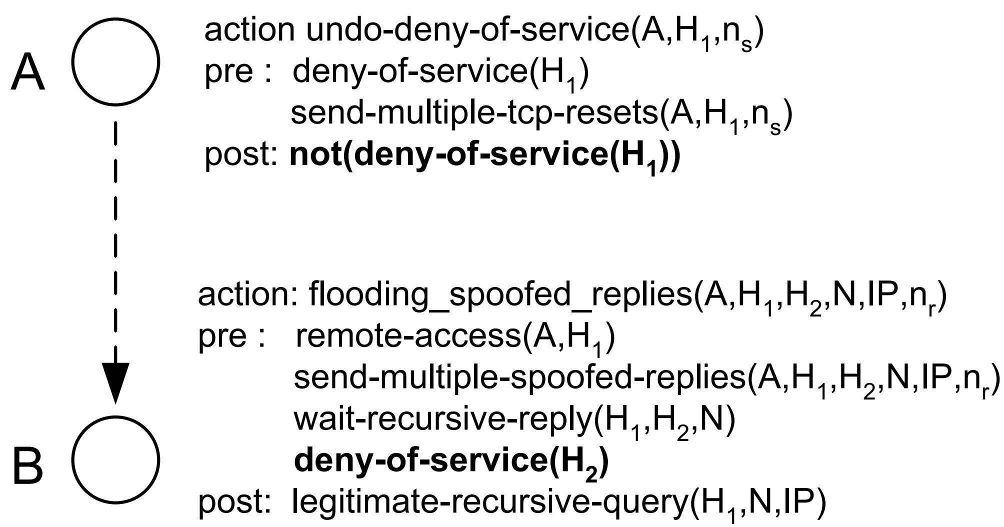
Detection process via alert correlation

- Two actions A and B can be correlated when the realization of A has a **positive influence** over the realization of B (given that A occurred before B):
  - $(E_a \in post(A) \wedge E_b \in pre(B)) \vee (not(E_a) \in post(A) \wedge not(E_b) \in pre(B))$
  - E_a and E_b are unifiable through a unifier θ

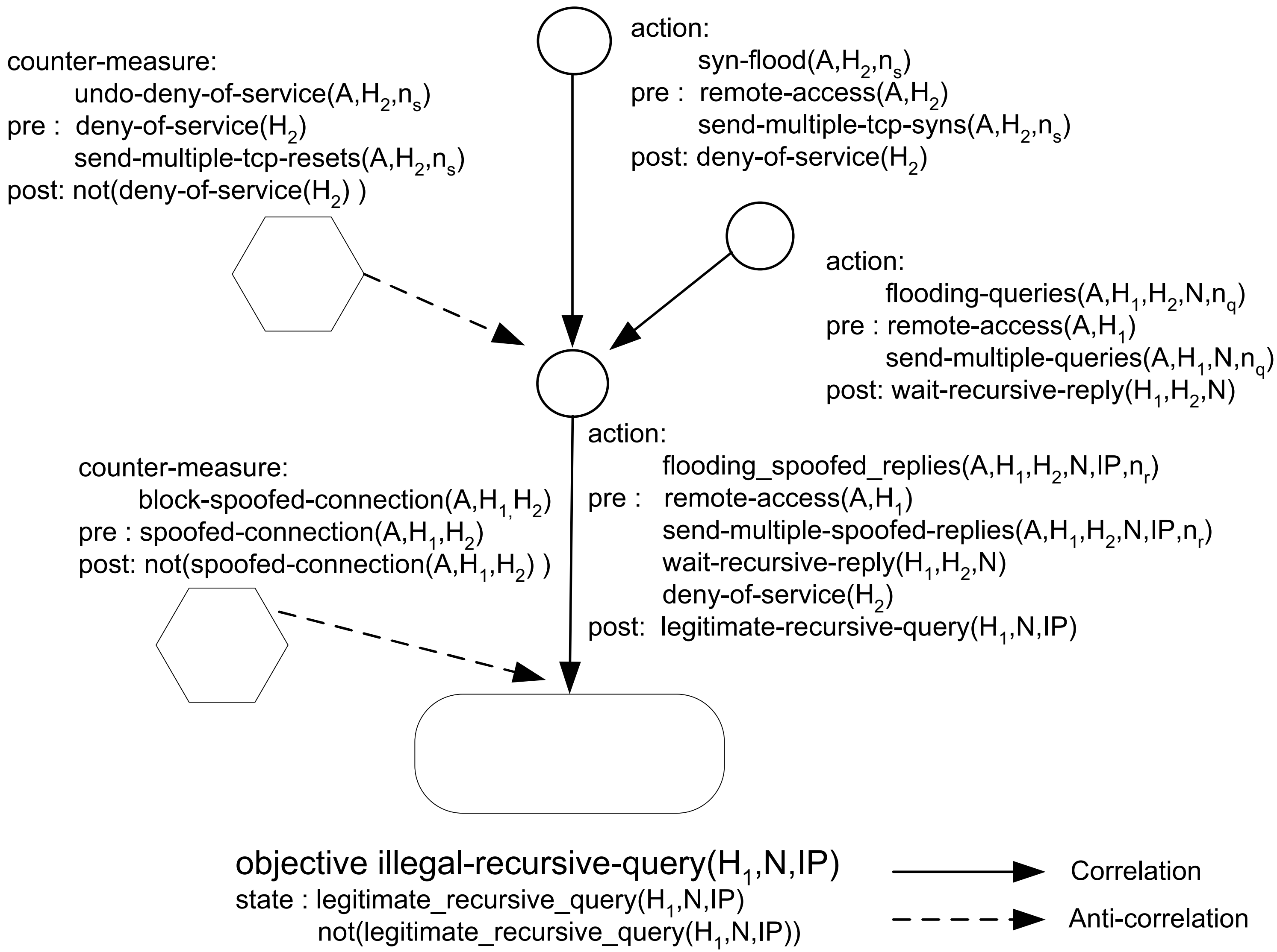


Reaction process via anti-correlation

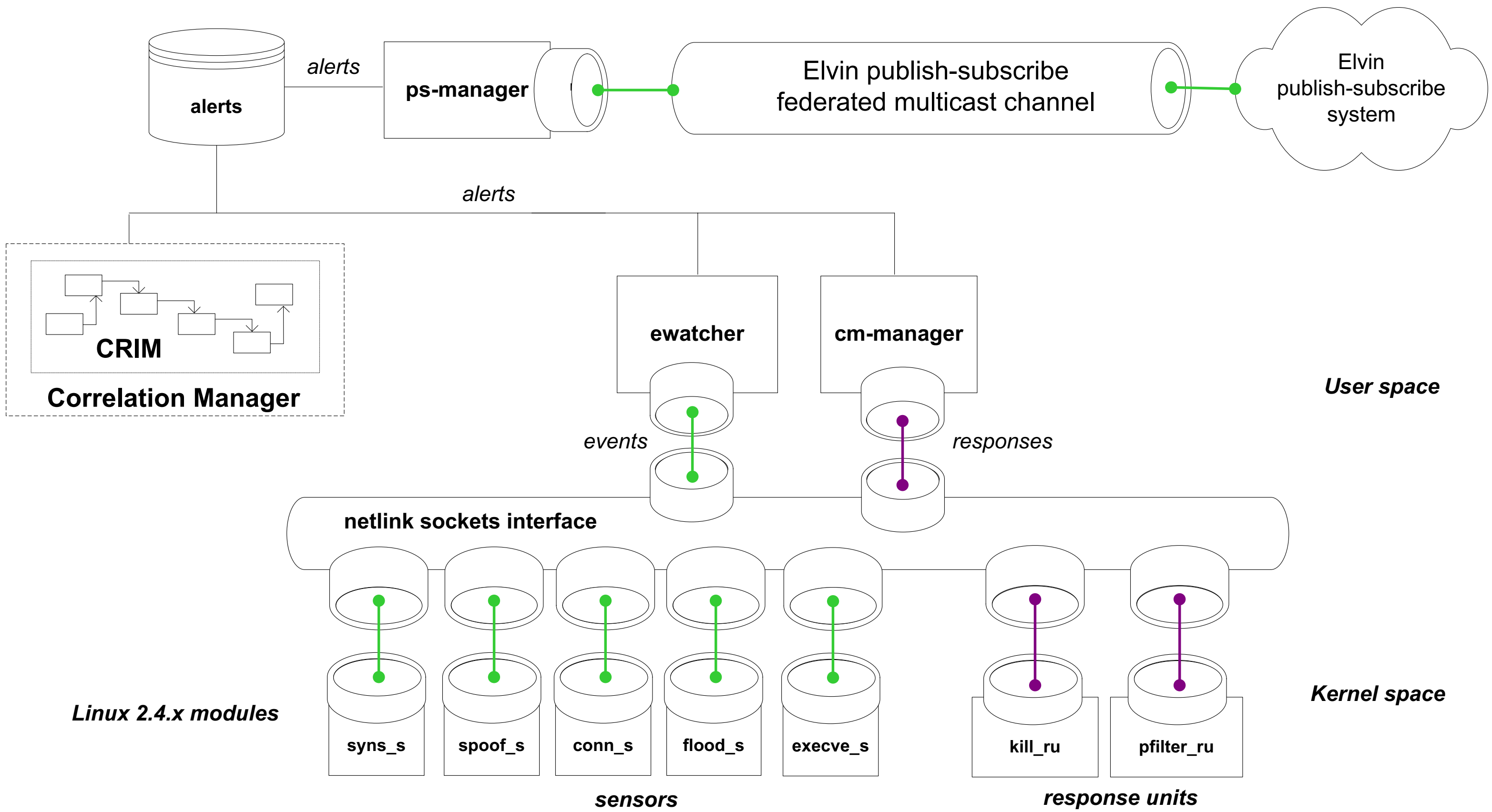
- Two actions A and B are anti-correlated when the realization of A has a **negative influence** over the realization of B (given that A occurred before B):
  - $(not(E_a) \in post(A) \wedge E_b \in pre(B)) \vee (E_a \in post(A) \wedge not(E_b) \in pre(B))$
  - E_a and E_b are unifiable through a unifier θ



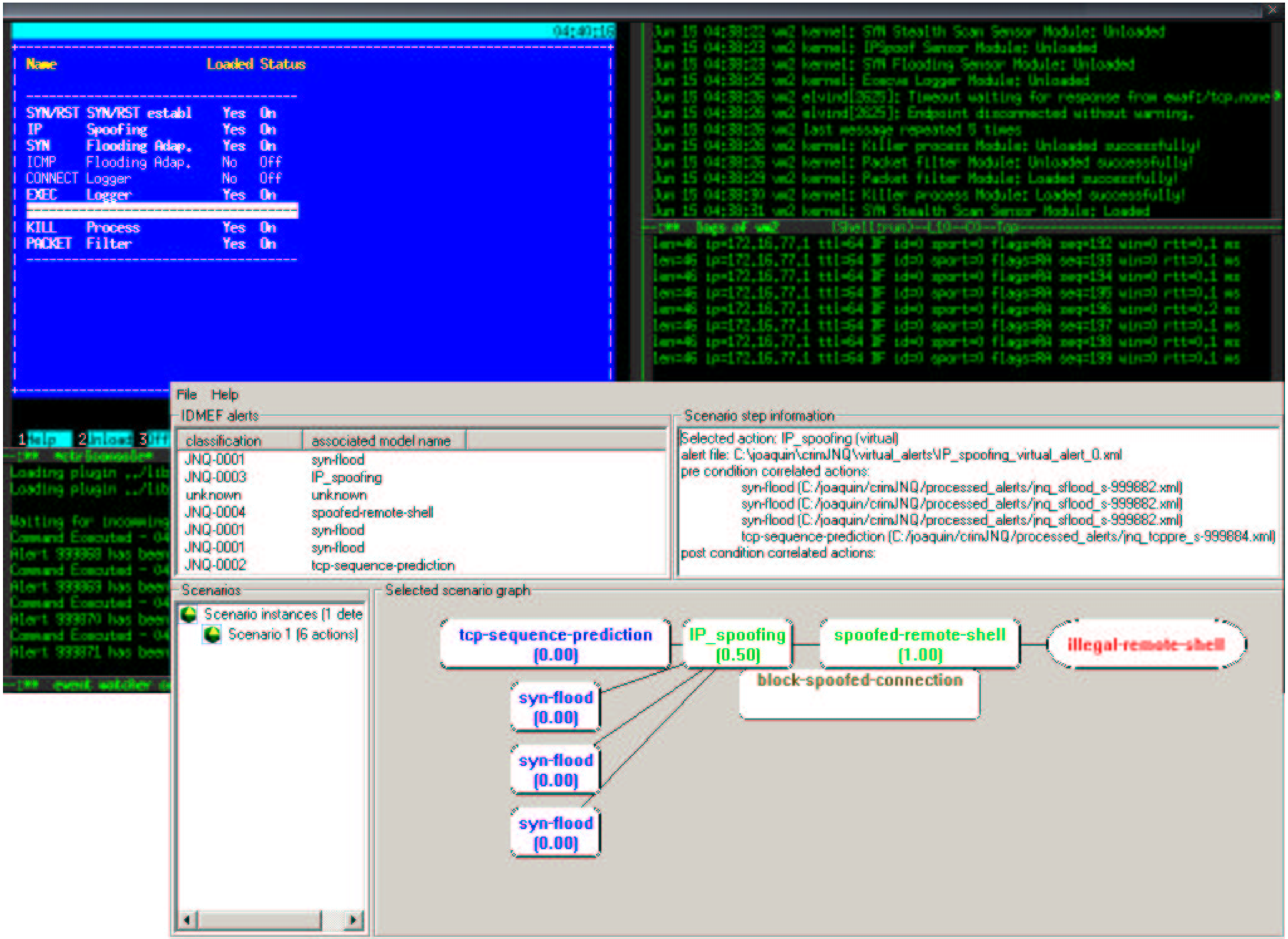
Detection and reaction graph for the sample scenario



Current Development



4. - Current Development



Results of our work

- State of the art about coordinated attack prevention
- Study about alert correlation mechanisms
- Development of a generic framework avoiding bottleneck of centralized architectures using a distributed approach
- Both detection and reaction are performed by using the same formalism

Future work

- Incorporate fault tolerant mechanisms
- Make a more in-depth study of the format used for alerts
- Incorporate other information about the environment

Thank you! Questions?

Preventing coordinated attacks via alert correlation

J. Garcia, F. Autrel, J. Borrell, Y. Bouzida

S. Castillo, F. Cuppens, G. Navarro

{jgarcia,jborrell,scastillo,gnavarro}@ccd.uab.es,

{fabien.autrel,yacine.bouzida,frederic.cuppens}@enst-bretagne.fr