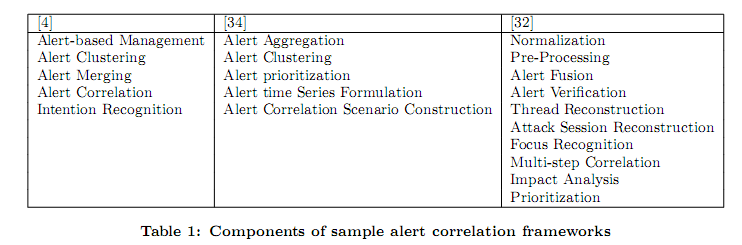
Alert correlation framework:

* Input: raw alerts
* Output: high level attack scenarios

Các framework trước đây: 

Các thành phần của Framework chính:

* Normalization
* Aggregation
* Correlation
* False alert reduction
* Attack strategy analysis
* Prioritization

Các hướng tiếp cận của các thành phần trong framework:

* **Normalization**:
  + Các alert từ các nguồn thu thập được sẽ được chuyển đổi sang IDMEF (để diễn tả relationship giữa các alert).
  + Vấn đề là các IDS khác nhau sẽ có cách điền các thông tin cho IDMEF khác nhau, nên cần có một taxonomy để phân loại các attack. Có 1 hướng dùng database để giải quyết vấn đề này ( CVE [33] – common vulnerability exposure và Bugtraq-ID[2])
* **Aggregation**: (mục tiêu để nhóm các alert giống nhau lại dựa trên các thuộc tính của nó)
  + Dùng xác suất[31]: tính toán sự giống nhau của các alert dựa trên sự giống nhau của từng thuộc tính, mỗi thuộc tính sẽ có hệ số khác nhau trong khi tính toán.
  + Clustering [12]: nhóm các alert chung root lại với nhau ( dựa vào generalization hierarchies ). Sự khác nhau của 2 thuộc tính sẽ là longest part giữa chúng, và sự khác nhau của 2 alert là tổng các thuộc tính khác nhau. Hạn chế: phương pháp này cần định nghĩa sẵn là min\_size của alert (số type của cluster trả về), hạn chế thứ 2 của[12] là không phân biệt các thuộc tính khác nhau trong việc chọn để so sánh ( weight của từng attribute).
  + Crim correlation framework[4]: similarity của các field trong alerts được tính toán dựa trên domain-specific rule, clustering dựa vào predicate logic (bằng các expert rules)
  + Data mining [8]: dùng data mining để nhóm dựa vào các attribute.
  + Machine learning [38]: tính toán xác suất giống nhau giữa alerts (dựa vào causal relationship để unify, hoặc dùng Static matrix [31] sự giống nhau được chuyên gia định nghĩa trước và sẽ được thay đổi dần trong quá trình learning)
* **Correlation**: ( mục tiêu tìm causal relationship của alerts để tạo scenario, giảm số lượng alert)
  + *Scenarios base Correlation:* (Lambda[6], Statl[10], Adele[29] là các ngôn ngữ được dùng )
    - Consequence rules[9]: tạo các quan hệ giữa các attack class. ????
    - Temporal logic[17]: dùng chronicle để correlate, chronicle là một tập các pattern với ràng buộc về thời gian và ngữ cảnh. Khi một alert mới đến, nó sẽ được matching với chronicles và trạng thái của chronicles sẽ thay đổi nếu matching true.
    - Machine learning[8]: để learning attack scenarios từ các label alert (manually by author). Các label alert xác định alert thuộc scenario nào. Phương pháp này giúp xây dựng tự động model cho alert correlation. Hạn chế: phải biết trước các attack scenarios, cần có nhiều trainning, kết quả có thể không đúng đối với các tập ngoài trainning data set.
  + *Rule base Correlation [28,5,19,3]*: (relationship được xác định bởi các rule precondition- postcondition).
    - JIGSAW[28]: yêu cầu tất cả tiền điều kiện phải thỏa mãn.
    - MIRADOR[5]: cải tiến được vấn đề trên, dùng ngôn ngữ LAMBDA để xác định attacks gồm các field: attack pre-condition, attack post-condition, description scenario.
      * Cơ chế : các alert ( IDMEF) được chuyển sang các logical fact. Để 2 alert được correlate thì ít nhất phải có 1 predicate trong post-condition của 1 alert có thể unify với 1 predicate trong pre-condition của alert còn lại. (phải có giai đoạn build các rule offline).
  + *Statistical Correlation[24]* : (tìm được unknown attack)
    - Mạng Bayesian được dùng để mô hình causality relationship giữa alerts. Alert được diễn tả như các node và mối quan hệ là các cạnh nối giữa các node này. Alert được chia ra theo time slot và có gán binary value cho các node này ???? .
  + *Temporal Correlation* [34]: (tìm được unknown attack)
    - Granger Causality Test: xác định causality relationship dựa vào time series-based statistical analysis. Hệ thống sẽ test giá trị time series của X (hyper-alert) và Y (hyper-alert) dựa trên statistical hypothesis. Các hyper-alert được tạo từ việc aggregate các đặc điểm giống nhau của alerts chỉ trừ giá trị time.
    - Hạn chế: đối với các random time.
* **False alert reduction:**
  + Data mining [16]: dùng frequent episode rules để mining những alert bình thường, nếu các alert match trùng với các rule này thì được xem là false positive.
  + Adaptive alert classification [21]: tính toán độ tin cậy chung của alert nhận được từ network topology. Độ tin cậy của alert có thể tính toán bằng cách sử dụng security state[37] (trong IDS sensors). Causal relationship và security state của hệ thống tạo nên alert attribute graph. Từ đó dùng mạng Bayes để suy luận
  + Subjective logic[27]: dùng consesus hoặc multiplication operation để cung cấp chỉ số confident cho việc correlate alert.
  + Hidden colored Petri Net [35]: phân biệt các alert có thuộc attacker action hay không. Để làm được điều này thì họ xây dựng 1 giải thuật suy luận các attacker action từ các observable alert.
  + Weighted Dempster-Shafer[36]: tiến hành xác định chỉ số confidence cho các nguồn cung cấp alert (alert từ IDS dùng signature-base thì có weight cao hơn…)
  + Fuzzy cognitive map[26]: phương pháp này cũng chia alert thành 2 loại : cause alert và effect alert (trong có có chứa các fuzzy valule) ?????
* **Attack strategy analysis** (suy luận mục đích của user từ low level alert)
  + Complementary method[20]: suy luận các attack bị bỏ lở bởi IDS, ý tưởng cơ bản là dùng tiền, hậu điều kiện và correlate dựa trên sự giống nhau của các thuộc tính (clustering). Từ đó có thể thêm vào các alert (bị miss) nếu có 2 alert giống nhau và alert đầu thỏa mãn tiền đk của alert sau.
  + Abductive rules[5]: dùng để suy miss alert. Nếu 1 alert là 1 consequence của 1 event, nhưng event này chưa được hệ thống tiếp nhận thì sẽ có thể suy được miss alert.
  + Hierarchical approach[25]: dựa trên attack tree, nó được chia thành nhiều subgoal, sau đó alert được đem so với attack tree để phát hiện hoạt động của user. Để dự đoán được thì mạng bayes được sử dụng (phải có knowledge base trước)
* **Prioritization**: (để classify alert)
  + M-corelator[22]: tính toán và sắp xếp thứ bậc của alert, sau đó clustering dựa vào các thứ bậc này. Hệ thống có một database để lưu các thông tin liên quan đến alert, ngoài ra nó còn quản lý topo của network bằng Nmap[11]. Dựa vào database và các thông tin từ Nmap hệ thống sẽ tính toán độ ưu tiên cho alert.