1 Attack trees :

En 1999, Bruce Schneier publie un article [4] qui modélise l'attaque d'un coffre sous forme d'un arbre nommé ***attack tree***.La figure ci-dessous représente l'attack tree :

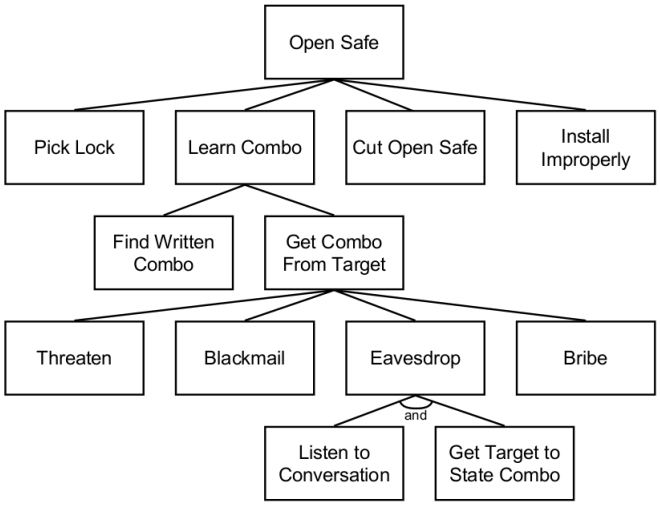

FIGURE 1 – Exemple d'une attack tree

Avec l'attack tree on peut représenter les attaques et les contre-mesures sachant que le noeud racine représente le but de l'attaque et les feuilles représentent les différentes façons pour atteindre cet objectif, et aussi on peut associer à chaque noeud une varibale boolean(or,and) où une variable qui exprime le coût d'une attaque...etc.

2 MulVal :

En 2005 Ou et al. présentent MulVAL[5] comme un framework open source qui permet de générer les graphes d'attaques pour analyser la sécurité des réseaux informatique. MulVal utilise Datalog comme un langage de modélisation et Prolog comme un moteur de raisonnement. Le temps d'exécution de *MulVal scanner* est 236 secondes et *MulVal engine* est capable de traiter un exemple de 2 000 machines en 16 secondes. La figure ci-dessous représente l'architecture de framework :

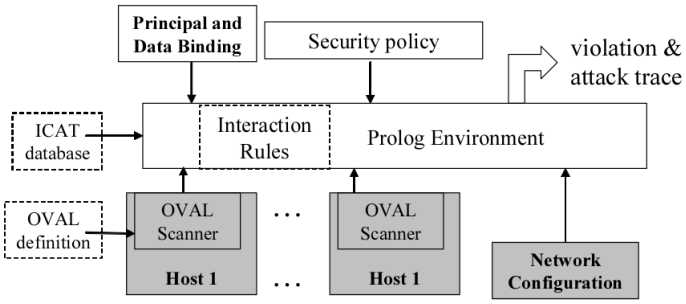

FIGURE 2 – Architecture de MulVal framwork

Ce framework possède deux logiciels. D'une part *MulVal scanner* , qui permet d'analyser les configurations des machines et aussi déterminer les vulnérabilités qui existent et d'autre par *MulVal engine* qui raisonne sur la configuration.

En 2006, Ou et al. proposent une évolution de MulVal nommée *logical attack graph*[6]. Ils ont démontré comment utiliser les traces d'attaques de MulVAl pour générer un *logical attack graph* en temps quadratique.

3 TVA :

En 2005 . Sushil Jajodia et al[3]. présentent *Topological Vulnerability Analysis (TVA)* comme un framework de génération des graphes d'attaques. La figure ci-dessous représente l'approche de TVA :

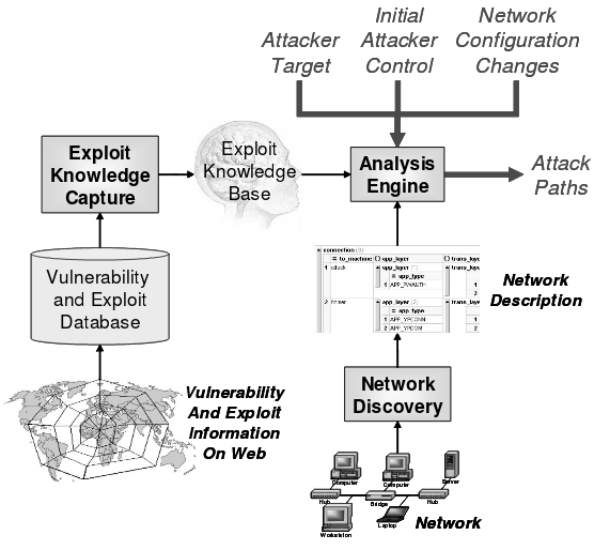

FIGURE 3 – Architecture de TVA

Ce framework possède trois composants : *(1) une base de connaissances* regroupe l'ensemble de vulnérabilités et les règles de l'éxploitation, *(2) la description de réseaux*, et *(3) une spécification de scénario d'attaque*. Le moteur d'analyse TVA fusionne ces trois composantes et découvre les chemins d'attaques.

4 NetSPA :

En 2002. le Lincoln Laboratory du MIT démarre le projet *Network Security Planning Architecture (NetSpa)*[1] qui permet de générer automatiquement les graphes d'attaques à partir des actions modélisée avec REM un simple langage de description d'attaque . L'outil NetSpa est réalisé en C++ et dans sa première version, est capable de générer un graphe d'attaque à partir 17 machines en 1.5 minutes. La figure ci-dessous représente l'architecture NetSPA :

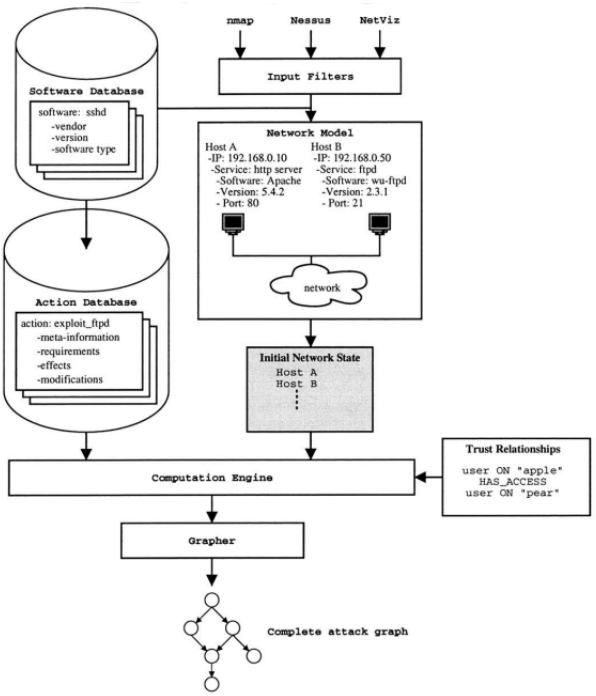

FIGURE 4 – Architecture de NetSPA

En 2006, une nouvelle version de NetSPA est présentée[2], basée sur *predictive graphs* qui sont des graphes d'attaques simplifiés. cette version est capable d'analyser des réseaux ayant jusqu'à 50 000 machines en moins de 4 minutes.

Références

[1] Michael Lyle Artz. Netspa : A network security planning architecture. Master's thesis, Massachusetts Institute of Technology. Dept. of Electrical Engineering and Computer Science, 2002.

[2] Richard Lippmann Kyle Ingols and Keith Piwowarski. Practical attack graph generation for network defense. In *Computer Security Applications Conference, 2006. ACSAC '06. 22nd Annual*, pages 121–130. IEEE Xplore, 2006.

[3] S. Noel S. Jajodia and B. O'Berry. Topological analysis of network attack vulnerability. In *Managing Cyber Threats : Issues, Approaches and Challenges*, pages 248–266. Springer US, 2005.

[4] B. Schneier. Attack trees. *Dr. Dobb's Journal*, 1999.

[5] S. Govindavajhala X. Ou and Andrew W. Appel. Mulval : A logic-based network security analyzer. In *4th USENIX Security Symposium, Baltimore, Maryland, U.S.A*, August 2005.

[6] Wayne F. Boyer X. Ou and Miles A. McQueen. A scalable approach to attack graph generation. In *CCS '06 : Proceedings of the 13th ACM conference on Computer and communications security*, pages 336–345. ACM Press, 2006.