

INTERACTIVE ANALYSIS OF ATTACK GRAPHS USING RELATIONAL QUERIES

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit of U.S. Provisional Application No. 60/821,052, filed Aug. 1, 2006, entitled "Interactive Analysis of Attack Graphs Using Relational Queries," which is hereby incorporated by reference in its entirety.

STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR DEVELOPMENT

[0002] This invention was made with government support under: FA8750-05-C-0212 awarded by Air Force Research Laboratory/Rome; contracts nos.: DAAD19-03-1-0257 and W91 INF-05-1-0374FA8750-05-C-0212 awarded by Army Research Office; contract no. DTFAWA-04-P-00278/0001 awarded by the Federal Aviation Administration; and contract nos. IIS-0242237 and IIS-0430402 awarded by the National Science Foundation. The government has certain rights in the invention.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

[0003] FIG. 1A depicts a running example of an attack graph with the exploits shown as ovals as per an aspect of an embodiment of the present invention.

[0004] FIG. 1B illustrates an example of a simplified version of the attack graph with the exploits shown as triplets as per an aspect of an embodiment of the present invention.

[0005] FIG. 2 shows an example of a network configuration and domain knowledge used in generating an attack graph as per an aspect of an embodiment of the present invention.

[0006] FIG. 3 shows a table that describes a relational model composed of four relations as per an aspect of an embodiment of the present invention.

[0007] FIG. 4 shows a table with an example of one iteration in deriving an attack graph as per an aspect of an embodiment of the present invention.

[0008] FIG. 5 shows a table used to illustrate an example of analyzing attack graphs for alert correlation and prediction as per an aspect of an embodiment of the present invention.

[0009] FIG. 6 shows a table used to illustrate an example of enumerating relevant exploits and network hardening as per an aspect of an embodiment of the present invention.

[0010] FIG. 7 shows a table that illustrates an example of incremental updates as per an aspect of an embodiment of the present invention.

[0011] FIG. 8A is a graph showing the performance of generating attack graphs as per an aspect of an embodiment of the present invention.

[0012] FIG. 8B is a graph showing the performance of analysis execution as per an aspect of an embodiment of the present invention.

[0013] FIG. 9 is a block diagram of an aspect of an embodiment of the present invention.

[0014] FIG. 10 is a flow diagram of an aspect of an embodiment of the present invention.

DETAILED DESCRIPTION OF EMBODIMENTS

[0015] Embodiments of the present invention enable interactive analysis of attack graphs. Attack graphs depict ways in which an adversary exploits system vulnerabilities in a network such as a computer network. Attack graphs may be important in defending against well-orchestrated network intrusions. However, the current analysis of attack graphs may require an algorithm to be developed and implemented, causing a delay in the availability of analysis. Such a delay is usually unacceptable because the needs for analyzing attack graphs may change rapidly in defending against network intrusions. An administrator may want to revise an analysis upon observing its outcome. Such an interactive analysis, similar to that in decision support systems, is difficult, if at all possible with current approaches based on proprietary algorithms. Embodiments of the present invention enable interactive analysis of attack graphs.

[0016] Embodiments of the present invention include a relational model for representing necessary inputs including network configuration and domain knowledge. An attack graph may be generated from those inputs as relational views. Analyses of the attack graph may be realized as relational queries against the views. These embodiments should eliminate the need for developing a proprietary algorithm for each different analysis, because an analysis is now simply a relational query. The interactive analysis of attack graphs should now be possible, because relational queries may be dynamically constructed and revised at run time. Moreover, the mature optimization techniques in relational databases may also be used to improve the performance of the analysis.

[0017] As the result of topological vulnerability analysis, an attack graph may describe all possible sequences of exploits an attacker can follow to advance an intrusion [16, 18, 1] into a network. Attack graphs have been explored for different purposes in defending against network intrusions. First, an attack graph may more clearly reveal the weakness of a network than individual vulnerabilities do by providing the context of attacks. Second, attack graphs may indicate available options in removing identified weaknesses and help administrators to preferably choose an optimal solution. Third, the knowledge encoded in attack graphs may also be used to correlate isolated alerts into probable attack scenarios. However, many current approaches to the analysis of attack graphs share a common limitation. That is, a proprietary algorithm may need to be developed and implemented before the corresponding analysis becomes possible. Standard graph related algorithms usually do not apply here due to unique characteristics of attack graphs. However, the delay in the analysis of attack graphs is usually unacceptable for defending against network intrusions. The needs for analyzing an attack graph usually changes rapidly due to constantly changing threats and network configurations. An administrator may need to modify an analysis after the results of that analysis are observed. Such an interactive analysis, similar to that in decision support systems, is difficult if at all possible with current approaches based on proprietary algorithms.