Cauldron

Mission-Centric Cyber Situational Awareness with Defense in Depth

Sushil Jajodia, Steven Noel, Pramod Kalapa, Massimiliano Albanese
Center for Secure Information Systems
George Mason University
Fairfax, VA 22193 USA
[jajodia,snoel,pkalapa,malbanes]@gmu.edu

John Williams
CyVision Technologies, Inc.
Bethesda, MD 20817 USA
johnrw@cyvisiontechnologies.com

***Abstract*—Cyber situational awareness determines timeliness, cost effectiveness, and success in responding to attacks. Mission success depends on complex computer networks, which are vulnerable to various types of attacks. Today, situational awareness capabilities are limited in many ways, such as inaccurate and incomplete vulnerability analysis, failure to adapt to evolving networks and attacks, inability to transform raw data into cyber intelligence, and inability for handling anomalous data. We describe advanced capabilities for mission-centric cyber situational awareness, based on defense in depth, provided by the Cauldron tool. Cauldron maps all paths of vulnerability through networks, by correlating, aggregating, normalizing, and fusing data from a variety of sources. It provides sophisticated visualization of attack paths, with automatically generated mitigation recommendations. Flexible modeling supports multi-step analysis of firewall rules as well as host-to-host vulnerability, with attack vectors inside the network as well as from the outside. We describe alert correlation based on Cauldron attack graphs, along with analysis of mission impact from attacks.**

Keywords - cyber situational awareness, attack graphs, cauldron security tool, vulnerability analysis, intrusion detection

I. INTRODUCTION

Network components and their configurations evolve over time, changing the vulnerability landscape and the interdependencies among vulnerabilities. Network hosts are added or removed, new vulnerabilities are discovered, existing vulnerabilities are patched, and communication links are continually opened or closed. Attacker capabilities and their strategies also evolve and improve. Traditional approaches with siloed monitoring tools and isolated static analyses are ineffective for rapidly evolving attack/defense situations.

Network monitoring tools generate data and alerts with little context. These data are uncertain, often ambiguous, and may be of little consequence or even incorrect. Given large numbers of alerts from intrusion detection systems, they need to be filtered to identify the most informative ones for analysis. Defenders need to recognize real threats, understand their potential impact on missions, and respond quickly and accurately for minimizing the impact. Recently published cyber attacks have been based on multi-step attacks using combined vulnerabilities. Defenders need to be prepared against such multi-step attacks through complex network vulnerability landscapes. Cyber attacks may begin with an initial alert, but then morph while traversing the network. A single defense of IDS is just one level of defense in depth that must be coordinated to be effective.

For advanced analytics to enable cyber situational awareness, we describe an integrated framework for automated attack modeling, alert correlation, and mission impact analysis. This framework includes correlating data from a variety of disparate sources; these include vulnerability scan reports, firewall/router configurations, vulnerability databases, and intrusion detection alerts. With this data, we build a comprehensive network attack model, to map all possible multi-step, combined network vulnerability paths. We associate intrusion alerts with these vulnerability paths, and provide alert correlation, mission impact analysis, and attack mitigation based on mission workflow.

In a complex network, each machine's susceptibility to attacks depends on the vulnerabilities of the other machines in the network. Attackers are now combining midlevel legacy vulnerabilities in many ways, giving numerous options for incrementally penetrating a network and compromising mission-critical systems. The traditional approach treats network elements and their associated data and events in isolation, without the context provided by multi-step vulnerability analysis. To adequately protect critical network infrastructures and missions, we must understand not only the vulnerabilities of each individual system, but also their interdependencies and how they support missions.

Prioritized remediation planning is achieved by calculating the impact of attacks by knowing the possible vulnerability paths through our networks. We transform raw security data into situational awareness, producing cyber security roadmaps that let us proactively prepare for attacks, manage vulnerability risks, and mitigate the impact of attacks. We deliver a capability for automated analysis of vulnerability paths so analysts can understand overall security posture. This provides the necessary context for cyber situational awareness over the full security life cycle, including alert correlation and optimal mitigation strategies and remediation planning.

II. TOPOLOGICAL VULNERABILITY ANALYSIS

Topological vulnerability analysis (TVA) [1][2] models multi-step attack vulnerability, then analyzes and visualizes the resulting attack graph (set of all vulnerability paths). The TVA tool *Cauldron* [1] provides unique capabilities, transforming raw security data into a roadmap that lets one proactively prepare for attacks and manage vulnerability risks. Cauldron attack graphs provide a common operating picture and a concrete understanding of how individual and combined vulnerabilities impact overall network security.

Cauldron integrates a variety of data sources, correlating and normalizing to a common operational model. The data sources include information about the monitored network environment (vulnerability scans, firewall settings, intrusion alerts, etc.), and reported cyber vulnerabilities from a number of vulnerability databases.

Figure 1 shows the architecture of Cauldron. Network capture builds a model of the network, in terms of relevant security attributes. The vulnerability database is a comprehensive repository of reported vulnerabilities listing the affected software and hardware. Cauldron integrates with vulnerability scanners Nessus [3], Retina [4], FoundScan [5] (and others not shown in the figure) for populating its network model. Cauldron imports firewall configuration data to capture network connectivity to vulnerable host services.

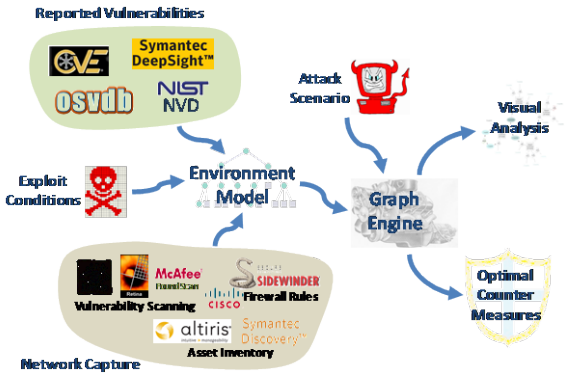

Figure 1. Cauldron topological vulnerability analysis.

The exploit conditions encode how vulnerabilities can be exploited (pre-conditions) and the results of their exploitation (post-conditions). Together, all these inputs are used to build an environment model for multi-step attack graph simulation. The graph engine uses the environment model to simulate multi-step attacks through the network, matching exploit pre-conditions and post-conditions. The result is all possible paths through the network for a given attack scenario.

To manage complex attack graphs, Cauldron provides sophisticated capabilities for interactive visual analysis with high-level overviews and detail drilldown [6][7], as shown in Figure 2. It aggregates portions of the network into managed zones (e.g., subnets, mission units, etc.) according to its network model. The analyst can begin with a high-level overview of vulnerability paths across zones, and then drill down on demand for interactions among individual hosts, vulnerabilities, etc. Analysts can interactively specify which parts of the network should be hardened (patched, blocked via firewall, etc.). Cauldron also provides recommendations for optimal network hardening [8][9].

Supported in part by Army Research Office MURI W911NF-09-1-0525.

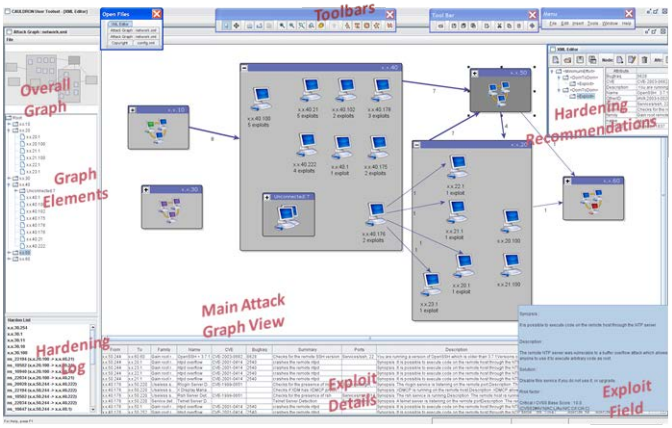

Figure 2. Cauldron interactive attack graph visualization.

The Cauldron network model includes detailed data about reported vulnerabilities. There are a number of such vulnerability databases maintained by the government, commercial companies, and the security community. Examples include NIST's National Vulnerability Database (NVD) [10], the Bugtraq security database [11], Symantec DeepSight [12], the Open Source Vulnerability Database (OSVDB) [13], and the Common Vulnerabilities and Exposure (CVE) referencing standard [14]. Cauldron leverages a storehouse of knowledge gathered by security researchers around the world, rather than being limited to vulnerabilities detected by a single tool like Nessus.

As shown in Figure 3, Cauldron performs data fusion, bringing together vulnerability data, access policies, and metadata specific to an enterprise environment. Access policy data is fused across multiple devices and vendor platforms. Access rules are correlated with the services (port/protocol) on each network host, and cross referenced to known vulnerabilities reported by a variety of popular network scanning tools. Enterprise knowledge is incorporated such as assumed threat sources, critical assets to protect, software patches, and known errors in vulnerability detection. The open architecture of Cauldron allows for expansive inclusion of data types to expand overall understanding of the security posture.

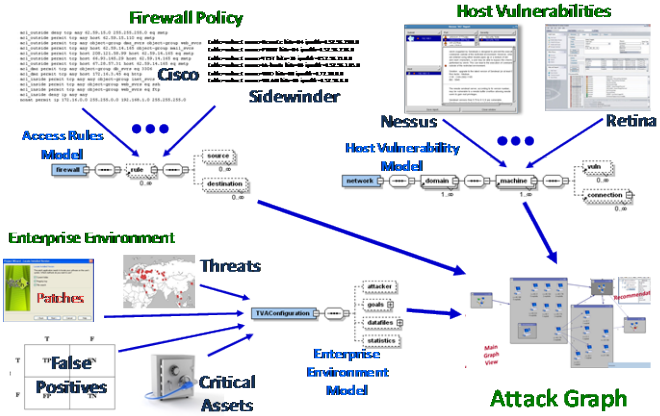

Figure 3. Cauldron data fusion.

III. MODEL VARIATIONS

Cauldron imports data from a variety of sources, and then correlates, aggregates, and normalizes the data into a common network attack vulnerability model. Cauldron's modular architecture offers a variety of ways that network attack models can be built. Figure 4 shows various modeling options in Cauldron. The *Access Rule Interpreter* converts native firewall/router configuration data (Cisco, Juniper, etc.) and into a vendor-neutral access rule model, i.e., the access control rules that each device enforces.

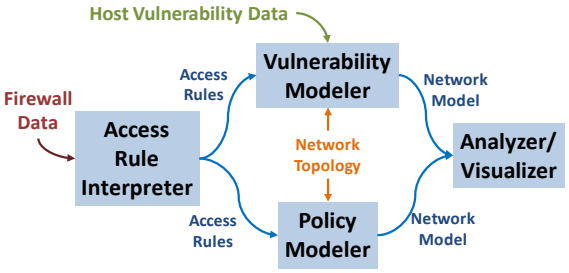

Figure 4. Modeling options within Cauldron.

The *Vulnerability Modeler* applies the access rules to host vulnerability data. This represents how hosts can connect across the network to other host vulnerabilities, according to the access rules. It aggregates hosts into protection domains (e.g., IP subnets), which are sets of hosts with unconstrained access to one another's vulnerabilities. Protection domains are part of the given network topology definition, or are generated from IP addresses in the host vulnerability data. The network topology also defines how devices (with their access rules) are inter-connected, and which domains they protect. The attack graph in Figure 2 is built with such a vulnerability model, with protection domains containing hosts (IP addresses). A graph edge indicates that an attacker on the source host can reach the destination host as an attack victim.

The *Policy Modeler* in Figure 4 builds a network model from access rules alone, independent of host vulnerabilities. In this case, the graph vertices are sources and destinations of access rules. Figure 5 shows such an access rule graph.

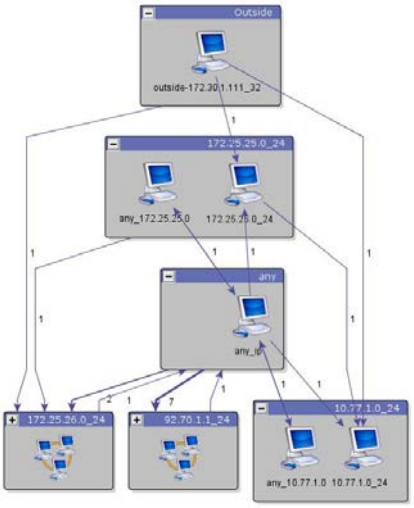

Figure 5. Access rule graph.

Here, each edge represents a single access rule, pointing from source to destination. Drilldown shows all relevant details for a rule, including source/destination IP address/mask and protocol/port. A distinguished "any IP" node indicates a rule that allows any IP address (with all protocols and ports) as a source and/or destination. Again, the endpoints are aggregated according to network protection domains.

In Figure 5, the domain labeled "outside" contains rule sources that lie outside the defined network topology – the "back door" into the network. These vectors indicate potential outside attack sources. Figure 6 shows another example of this outside source modeling, this time for a host vulnerability graph. This example shows that there are two access rules that each allow sources from the outside. By only considering scanned hosts within the defined network, such attack vectors from outside the network would be missed.

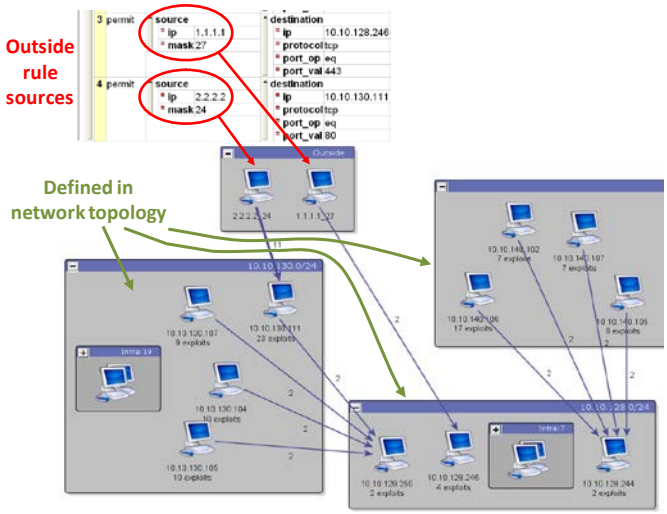

Figure 6. Modeling outside attack sources.

As shown in Figure 7, Cauldron attack graphs can be constrained by attack start and/or goal. The left side shows an unconstrained attack graph. The middle of the figure shows the same graph, constrained by attack start. One subnet is unreachable from the start, and is removed in the constrained graph. The right side of the figure is constrained by start and goal, with outgoing edges from the goal removed.

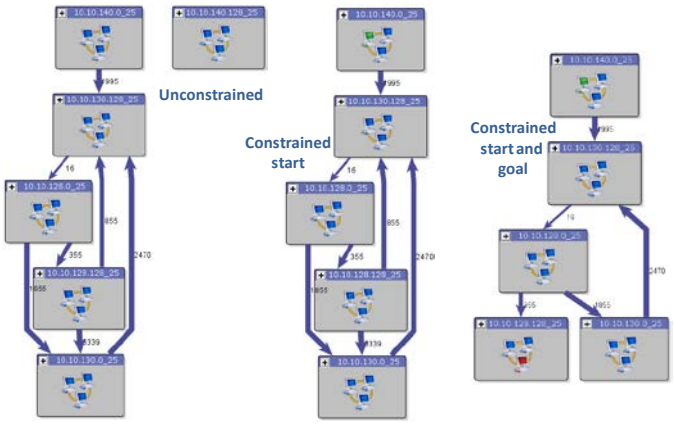

Figure 7. Constraining by attack start and goal.

IV. Defense in Depth

A comprehensive approach to network security relies on multiple layers of defense to prevent espionage and direct attacks against critical systems. This not only helps prevent security breaches, it also gives more time to detect and respond to attacks, and reduces the consequences of a breach.

For this kind of defense in depth, Cauldron needs to model the policy rules that each traffic-filtering device (firewall, router, etc.) enforces. It also needs to consider the topology of how devices are connected, and which portions of the network each device protects. As in Figure 8, a network device can connect to other devices for routing traffic, and connects to its protected domains (e.g., subnets). Each device filters traffic between routes and domains according to its access rules.

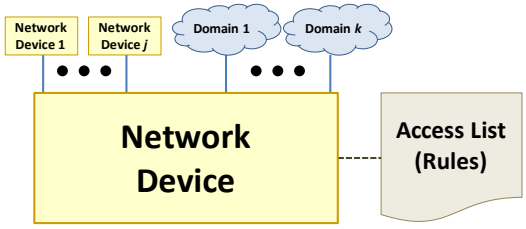

Figure 8. Model for network device that filters traffic.

As shown in Figure 9, inter-connected network devices and their subnets form a network topology graph. From this graph, we determine vulnerable connectivity among hosts. From a source host to a destination host, we traverse the topology graph. During traversal, when a network device is encountered, its access rules are applied. If the rules allow the source and destination, traversal continues to the next device.

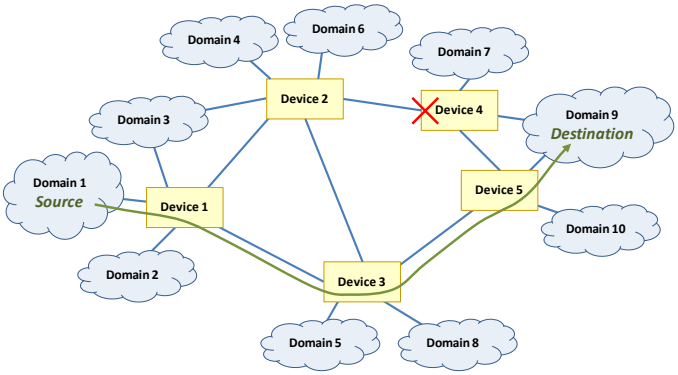

Figure 9. Modeling multiple firewalls for defense in depth.

If traversal successfully reaches the destination (IP address, protocol, and port), then a connection is added to the Cauldron network model. Domains themselves are not traversed, only network devices. Domains are a set attribute of a network device, to be tested for source or destination only. Defense in depth requires understanding each firewall, as well as the combined effects of multiple firewalls, in arbitrary serial and/or parallel configurations.

V. Optimal Mitigation Strategies

Cyber defenders have limited resources (time and/or qualified workforce) for comprehensive remediation. There is an expanding pool of legacy vulnerabilities. Simply put, not all vulnerabilities can be addressed. Defenders are faced with an expanding set of threats of varied sophistication and impact, with an increasing availability of mid-level vulnerabilities to achieve the goal. Cauldron combines multiple data sets and models vulnerability mitigation strategies to provide maximum return on investment.

Figure 10 is an attack graph generated by Cauldron, by correlating Nessus vulnerability scans with firewall policy rules. This graph includes all reported vulnerabilities, showing all the paths that attackers can penetrate through the network, vectoring step-by-step from one subnet to another. There are a total of 46,000 vulnerable connections among 16 subnets, with 1,200 endpoint hosts. This visualization represents a substantial remediation effort.

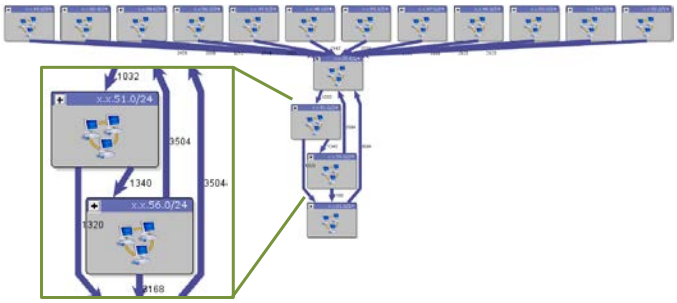

Figure 10. Attack graph before remediation.

The management challenge is to direct remediation efforts to achieve the most effective overall results. Cauldron can indentify individual vulnerabilities (or selected groups of vulnerabilities), the number of hosts affected, and the number of vulnerable connections that attackers can exploit. Sorted and grouped vulnerabilities enable modeling for the greatest impact for remediation planning.

We first prioritize by CVSS score [15], a traditional POA&M methodology a team might take given limited resources for remediation. We include in the attack graph those vulnerabilities with CVSS score above 7 (CVSS ranges from zero to 10). This addresses vulnerabilities rated as high, ignoring their context within our network. For vulnerabilities with CVSS > 7, we get the attack graph in Figure 11.

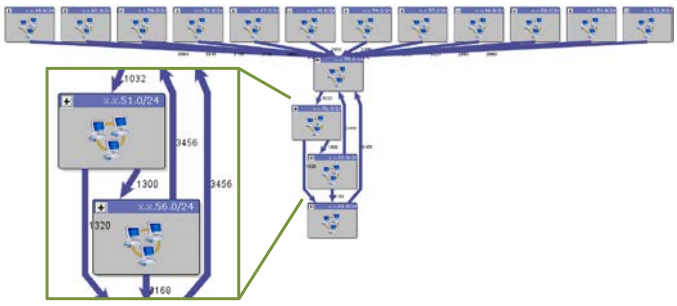

Figure 11. Remediation based on CVSS.

There are 15 distinct vulnerabilities with CVSS > 7 that need to be remediated across the network. While these are the 15 most "critical" vulnerabilities (in terms of attack

complexity, authentication required, etc., as defined by CVSS), they do not take into account the organization access rules. The attack graph still allows nearly the same exploitation from subnet to subnet. While the remaining vulnerabilities are less critical, vulnerability to attack across subnets is nearly unchanged.

To better address vulnerabilities in the context of the network topology and access rules, we now select by number of hosts with a given vulnerability. If we address only the 3 most common vulnerabilities by host, we get the attack graph in Figure 12. This remediation plan has a significant overall improvement. After remediating only 3 vulnerabilities across the network (using a configuration management technology such as Baseline [16]), the number of vulnerability vectors is dramatically reduced.

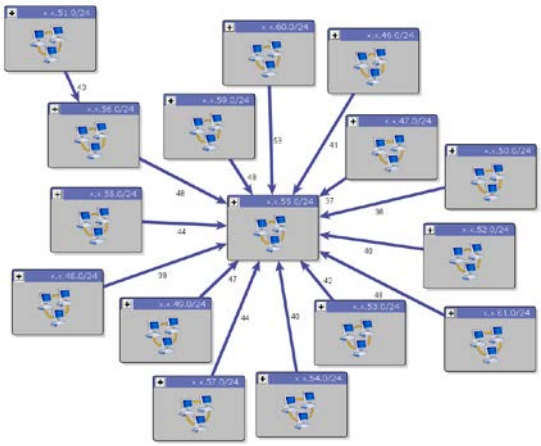

Figure 12. Remediation based on host vulnearabilities.

To remediate directly in the context of the attack vectors, we select by number of connections (subnet-to-subnet vulnerabilities). If we remediate only the top 3 vulnerabilities by vulnerable connection we get the attack graph in Figure 13. This remediation plan has the greatest overall improvement. For the same effort (3 vulnerabilities) as prioritizing by host, there are only a very small number of remaining paths of vulnerability through the network.

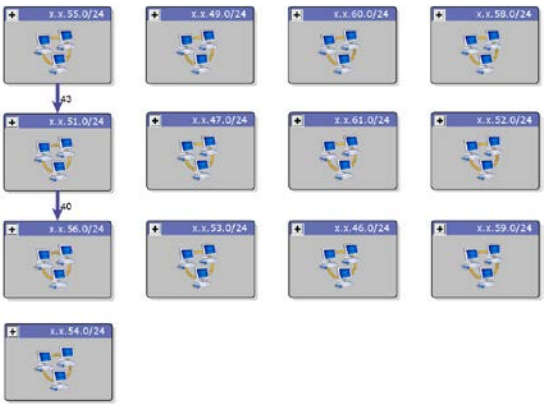

Figure 13. Remediation based on attack vectors across subnets.

Cauldron could even constrain the analysis to a given critical network asset, which may not even be reachable from these remaining paths. In comparison, by prioritizing by CVSS, remediation of nearly half the seeming critical vulnerabilities has almost no impact on securing the network. Given the same raw data, Cauldron has a unique ability to pinpoint the most effective use of mitigation resources. Cauldron quickly finds the critical problems and improves security posture through proactive remediation.

VI. Alert Mapping and Correlation

For alert mapping and correlation, we identify criteria to preprocess, filter, and prioritize received alerts, to reduce their volume and focus on the most informative or relevant alerts in subsequent analysis [17][18]. As an example, consider the attack graph of Figure 14. If an alert is received at time t_1, and maps to a vulnerability on host h_C, then an attack described by the graph has started. From the graph, we know that the attacker can now exploit vulnerabilities on either host h_D or host h_F. On the other hand, at time t_1, if the alert is mapped against host h_B (instead of host h_C), we know that only host h_J is vulnerable to attack at this point.

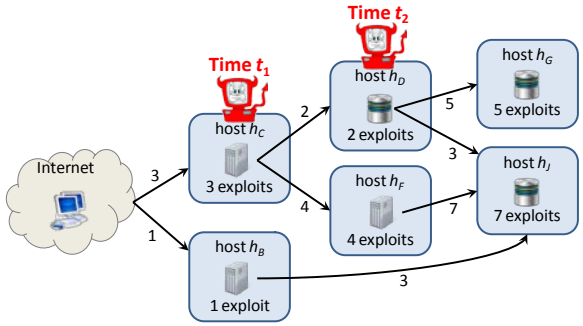

Figure 14. Mapping alerts to attack graph.

Later, at time t_2, say we observe another alert mapped to host h_D. We can correlate these two alerts (at t_1 and t_2) and know exactly which path the attacker has penetrated through the network. Receiving two such correlated alerts raises their priority. We know at this point (from host h_D), the attacker can exploit vulnerabilities on host h_G or host h_J. If the alert at t_2 had been against host h_F (versus h_D), only host h_J would have been vulnerable to this attacker (versus both h_G and h_J).

In a scenario like this, there may be missing alerts, i.e., false negatives reported by the intrusion detection system. For the example in Figure 14, once we receive an alert against h_C, we need to consider the possibility that the attack will progress without us receiving alerts against h_D, i.e., we might observe an alert against h_G only. Looking at the problem in the other direction, if we observe an alert against h_G after an alert for h_C (without observing an alert for h_D) then we can hypothesize that the attacker actually compromised h_D but the attack was missed by the intrusion detection system. We extend the above reasoning to the general case where multiple alerts from arbitrary attack graphs are missing.

VII. Mission Impact Analysis

Cauldron provides the foundation for a range of attack responses and mitigation strategies. We can generate good (or even optimal) mitigation strategies based on reduction of projected attack impact.

For impact analysis, we assume that mission workflows are given. A workflow describes the tasks required to complete a mission, and tells what resources are needed to complete each task. An attack can impact a mission by compromising network resources that are needed to complete one or more tasks. Failure to complete a task could reduce mission effectiveness or cause the mission to fail. In an organization assigns a value to each mission and its tasks, that provides a basis for measuring mission impact of an attack.

For attack impact analysis, we combine information in the attack graph, information about ongoing attacks gained by analyzing sequences of alerts, and mission workflows [19]. For exploited vulnerabilities, we look at which network assets are compromised. Leveraging mission workflows, we know which tasks are jeopardized because their services are compromised, and how this impacts the overall mission.

We use attack graphs to understand vulnerabilities an attacker can exploit next, and based on the current state of the mission, anticipate the impact of attack steps. We then suggest the best courses of actions to minimize future impact. In other words, we assess the future impact of ongoing attacks assuming that certain corrective actions are taken.

As a simple example, consider Figure 15. There are two missions, M_1 and M_2. Mission M_1 relies strictly on the availability of hosts h_C and h_D (with no other redundant services available), while mission M_2 relies on host h_F (as its only available service). Let us say that an attacker has compromised host h_C. Although the attacker can now exploit either host h_D or h_F, exploiting h_D will have no further impact on the mission availability. Mission M_1 is already compromised because of the compromise of h_C. Mission M_2 (so far uncompromised) does not rely on h_D. We should therefore focus on protecting h_F to protect the surviving mission M_2.

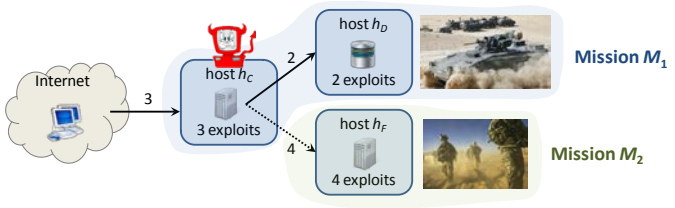

Figure 15. Mission dependency on network focuses attack responses.

Multiple missions can depend on multiple overlapping network resources. For example, two missions may depend on a shared network server; then compromise of that server means that both missions are compromised. Also, there may be built-in redundancy of services so that a mission does not critically depend on a single server for a particular service. In such a case, the compromise of one instance of the redundant server does not compromise the mission. Our algorithms handle such overlapping and/or redundant mission service dependencies.

Acknowledgments

The work presented in this paper is supported in part by the Army Research Office MURI award number W911NF-09-1-0525.

References

[1] S. Jajodia, S. Noel, "Topological Vulnerability Analysis," in *Cyber Situational Awareness (Advances in Information Security 46)*, S. Jajodia, P. Liu, V. Swarup, C. Wang, (eds.), Springer, 2010.

[2] S. Jajodia, S. Noel, P. Kalapa, B. O'Berry, M. Jacobs, E. Robertson, R. Weierbach, "Network Attack Modeling, Analysis, and Response," U. S. Patent 7,904,962, awarded March 8, 2011.

[3] Tenable Network Security, *Nessus®*, http://www.nessus.org, last retrieved June 2011.

[4] eEye Digital Security, *Retina Network Security Scanner*, http://www.eeye.com/Products/Retina/Network-Security-Scanner.aspx, last retrieved June 2011.

[5] Foundstone, *FoundScan*, http://www.foundstone.com/, last retrieved June 2011.

[6] S. Noel, S. Jajodia, "Managing Attack Graph Complexity through Visual Hierarchical Aggregation," *Proc. ACM Workshop on Visualization and Data Mining for Computer Security (VizSec)*, 2004.

[7] S. Noel, S. Jajodia, "Attack Graph Aggregation," U. S. Patent 7,627,900, awarded December 1, 2009.

[8] L. Wang, S. Noel, S. Jajodia, "Minimum-Cost Network Hardening Using Attack Graphs," *Computer Communications*, 29(18), 3812-3824, November 2006.

[9] S. Noel, S. Jajodia, B. O'Berry, M. Jacobs, "Minimum-Cost Network Hardening," U. S. Patent 7,555,778, awarded June 30, 2009.

[10] NIST, *National Vulnerability Database (NVD)*, http://nvd.nist.gov/, last retrieved June 2011.

[11] Security Focus, *Bugtraq*, http://www.securityfocus.com/vulnerabilities, last retrieved June 2011.

[12] Symantec Corporation, *Symantec DeepSight™ Threat Management System*, https://tms.symantec.com/Default.aspx, last retrieved June 2011.

[13] *Open Source Vulnerability Database*, http://osvdb.org/, last retrieved June 2011.

[14] MITRE, *CVE – Common Vulnerabilities and Exposures*, http://cve.mitre.org/, last retrieved June 2011.

[15] First, *A Complete Guide to the Common Vulnerability Scoring System*, http://www.first.org/cvss/cvss-guide.html, last retrieved June 2011.

[16] *Baseline Network Vulnerability Remediation*, http://www.baseline.com/, last retrieved June 2011.

[17] S. Noel, E. Robertson, S. Jajodia, "Correlating Intrusion Events and Building Attack Scenarios through Attack Graph Distances," *Proc. 20th Annual Computer Security Applications Conference (ACSAC)*, 2004.

[18] S. Noel, E. Robertson, S. Jajodia, "Intrusion Event Correlator," U. S. Patent 7,735,141, awarded June 8, 2010.

[19] M. Albanese, S. Jajodia, A. Pugliese, V. Subrahmanian, "Scalable Analysis of Attack Scenarios," *Proc. 16th European Symposium on Research in Computer Security (ESORICS)*, to appear, September 2011.