Ken Hines, Ph.D

GraniteEdge Networks

BLACK HAT BRIEFINGS

Using Causal Analysis to Establish Meaningful Connections between Anomalous Behaviors in a Networking Environment

Fueled by business needs such as supply chain integration and outsourcing, modern enterprises must open up portions of their networks to potentially untrusted outsiders. Combined with the troubling aspects of malicious insiders, ever more sophisticated attacks, increasing network complexity, and strong pressure from regulatory bodies to rapidly identify breaches and assess damages, there is a rapidly growing concern over internal network security. IT departments must work harder than ever to prevent insiders and outsiders from gaining unauthorized access to critical assets deep in the network, and if such access ever occurs, identify and report on, the impact of such a security breach.

In order to gain real insight into the dynamic behavior of their networks, IT departments must monitor huge quantities of data, where individual elements of a sophisticated attack may be spread out over long periods of time, and vast numbers of logs. Many tools are available to identify individual phases of an attack, such as IDSs, network based anomaly detection tools, host based monitoring tools, and even firewalls. However, this data is presented to the security analyst as a series of unrelated suspicious events. Because of the complexity of modern networks there are always isolated and seemingly suspicious things occurring on the network. To find a sophisticated breach the individual pieces of an attack need to be tied together for successful analysis.

One approach to determining relationships between events is by defining rules, such as: if some set of events happens around the same time, they are probably related, and should be presented as a correlated event. Unfortunately this places the burden on the security analyst of predefining attack scenarios for their particular network. Unlike virus detection which can leverage the entire anti-virus community to identify and write appropriate signature files, internal network security has no such analogy. Every enterprise network has unique characteristics requiring company specific rules. While rules are good for identifying problems with well defined signatures, they aren't capable of relating attack elements separated by large time intervals, and obscured by benign activity on the surrounding hosts. The missing piece is causal analysis, which can automatically link together suspicious events independent of the normal network activity that occurs between the various phases of a security breach. The benefit of the causal analysis approach is that chains of related and suspicious activity provide a strategic overview of network behavior allowing a security analyst to focus their efforts on attacks in progress. When they have a readable chain of anomalous behavior, the security team can trace the attack vector back to the entry point, and find the so-called "patient zero."

This presentation demonstrates the value of causal analysis using a simple example that involves social networks rather than computer networks, how this example is really a metaphor for a very common form of computer network attack, and how causal analysis is equally appropriate in finding this sort of attack in enterprise networks. It then presents some of the factors that compound the difficulty of this analysis in real networks, and describes approaches that simplify this complexity. Using the techniques described, two real "stepping stone" attacks are outlined and diagrammed to illustrate the power of causal analysis. Finally, it demonstrates how this analysis can be combined with other forms of security analytic and mitigation techniques to provide a formidable barrier against network attacks.

Ken Hines *earned his Ph.D. in computer science at the University of Washington in 2000, by successfully defending his dissertation, which applied causal analysis to debugging heterogeneous distributed embedded systems. Since then, he has founded two venture funded companies, and actively developed commercial products that apply causal analysis to solving complex problems related to distributed embedded systems, network processor based network infrastructure, and finally networks as a whole.*

While a graduate student, Ken was one of the primary researchers on the Chinook Hardware/Software Co-synthesis project, and published a number of papers on distributed debugging, distributed hardware/software co-simulation, and co-synthesis for heterogeneous distributed embedded systems.

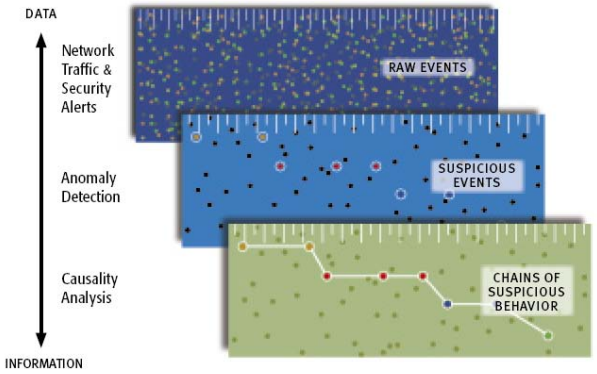
Background

- Network Security Challenge
  - Increased bandwidth == increased number of events
  - Attacker sophistication increasing
  - Perimeter dissolving as network becomes more complex
- Traditional Approaches attempt Point Solutions
  - IDS & IPS
  - Firewall and VPN
  - Network-based anomaly detection

GRANITEEDGE™

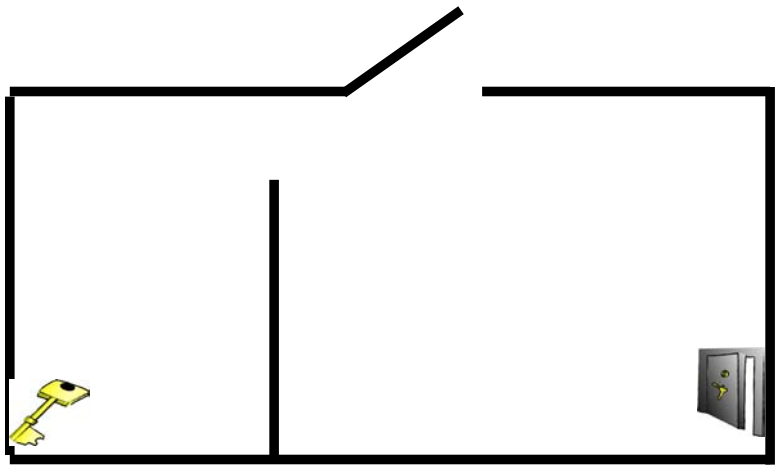
Causality analysis

- Connecting elements of a network based attack



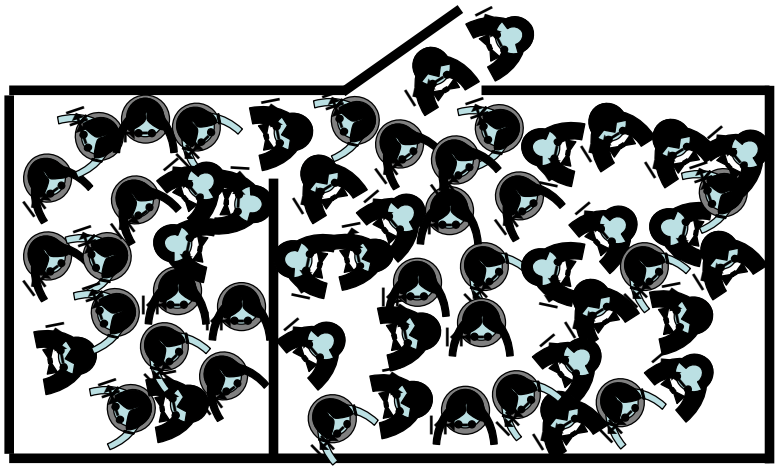
GRANITEEDGE™

Causality Analysis Example



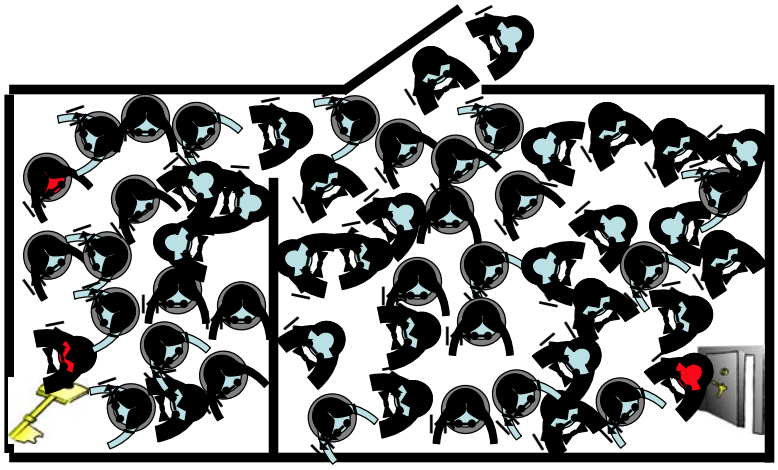
GRANITEEDGE™

Causality Analysis Example



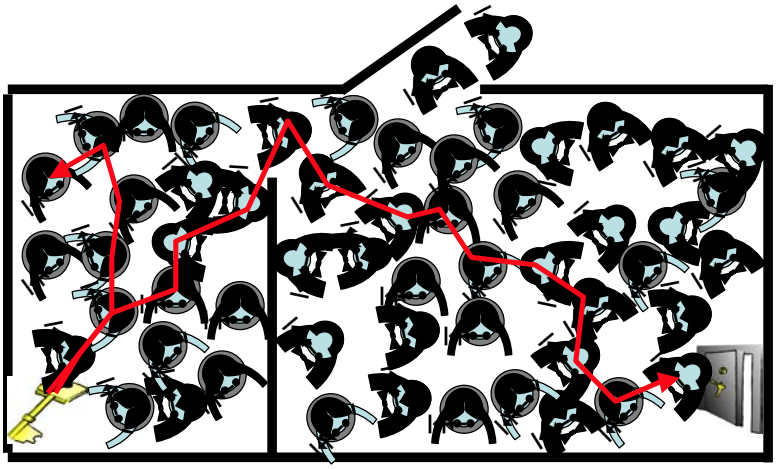
GRANITEEDGE™

Causality Analysis Example



GRANITEEDGE™

Causality Analysis Example



GRANITEEDGE™

Important Relationships

- Causal:
  - Stolen key to missing documents
  - Stolen key to light off
- Non-causal:
  - Light off and missing documents.

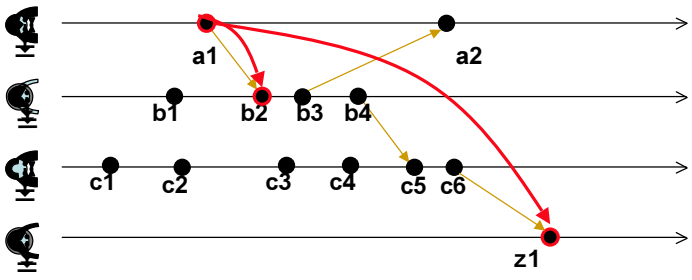
GRANITEEDGE™

Causal Relationship

a1 ~> b2 (a1 is causally related to b1)

a1~>z1

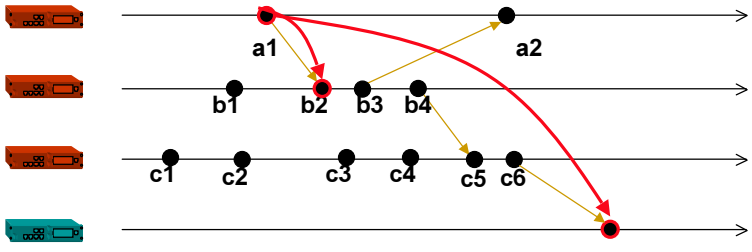
a1 ~/> c1 (a1 is not causally related to c1)



GRANITEEDGE™

Metaphor for Network

- Guests == network nodes
- Handshakes == communication events
- Safe == some high value asset



GRANITEEDGE™

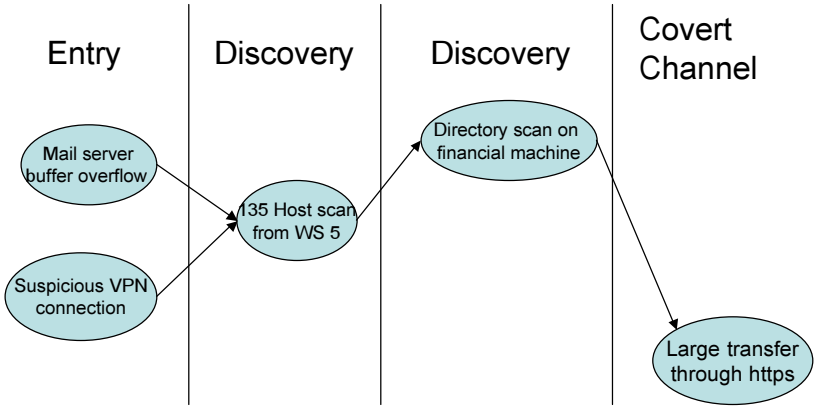
Basic Attack Form

- Gain access to private network (entry)
- Surreptitiously discover and interact with existing resources (discovery, stepping stones)
- Gain access to key information (discovery, entry)
- Escape with key information (covert channel)

GRANITEEDGE

Causality analysis

- Ties disparate attack elements together:
  E.g.



GRANITEEDGE

Causality Analysis Benefits

- Trace attack by connecting anomalous events
- Proactive
  - Determine that an attack may be in progress
  - Assess vulnerabilities and potential damage
  - Perform mitigation
- Reactive
  - Determine which machines may have been affected
  - Determine which assets may have been compromised
  - Track attack source

GRANITEEDGE™

Logistics

- Millions of normal events surrounding suspicious activity
  - Suspicious relationships not visible to the human eye
  - No time to walk all possible event sequences to determine causal relationship
  - The slower the attack, the harder it is to find
  - Multiple breaches may be in progress at the same time
- What is needed
  - Adequate network visibility
  - Reasonably fast approach to computing transitive causality
  - Deliver rapid “yes/no” on relationship between two events

GRANITEEDGE™

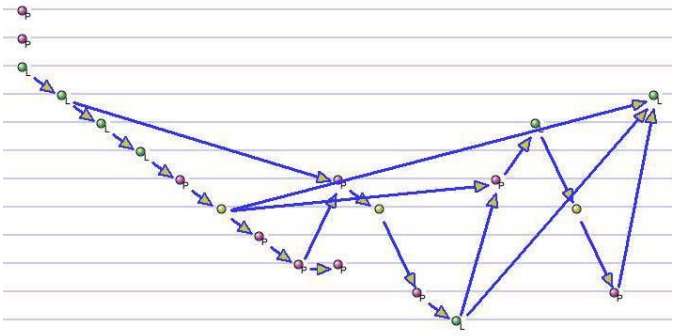
Overall Approach

- Monitor all network traffic at key points
- Also look at security alerts from firewall, IDS etc
- Store events in a high performance data repository
- Compute & store symbolic representation of transitive causal relationship
  - Provides real-time answers for causal queries
  - One month on a typical enterprise network = 4 terabytes of data

GRANITEEDGE™

Real Results

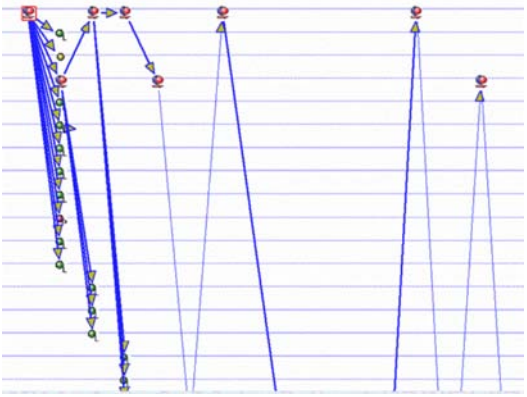
- Slammer worm propagation from laptop



GRANITEEDGE™

Real Results

- NetBIOS stepping stone/discovery by insider



GRANITEEDGE™

Conclusion

- Causality analysis is a new approach to understanding network behavior and security
- Causality analysis can provide insight into non-causal relationships as well
- Need symbolic approach to provide rapid “yes/no” causality answers
- Not quite a silver bullet
  - Sometimes highlights ambiguous relationships
  - Additional analysis sometimes required to refine root cause

GRANITEEDGE™