Provenance for Collaboration: Detecting Suspicious Behaviors and Assessing Trust in Information

M. David Allen, Adriane Chapman, Len Seligman, Barbara Blaustein
The MITRE Corporation
{dmallen, achapman, seligman, bblaustein} @ mitre.org

Abstract–Data collaborations allow users to draw upon diverse resources to solve complex problems. While collaborations enable a greater ability to manipulate data and services, they also create new security vulnerabilities. Collaboration participants need methods to detect suspicious behaviors (potentially caused by malicious insiders) and assess trust in information when it passes through many hands. In this work, we describe these challenges and introduce provenance as a way to solve them. We describe a provenance system, PLUS, and show how it can be used to assist in assessing trust and detecting suspicious behaviors. A preliminary study shows this to be a promising direction for future research.

Index terms—provenance, trust, insider threat, lineage, pedigree

I. INTRODUCTION

There is a growing need for cross-organizational collaboration to address important problems such as improving healthcare outcomes, law enforcement across federal, state, and local agencies, and improving inter-organizational disaster response. Increasingly, complex problems require *data collaboration*, in which disparate partners share data sets to which other partners add value via additional data and analysis steps. As a simple example, a loan broker takes customer information, individual quotes from various banks, and credit rating data to provide a customer a package of different loan options. Such collaborations require the participation and data of multiple banks, the loan broker, a credit rating agency, and a customer.

Other trends, such as increasingly available online services, and government mandates to increase information sharing have driven more such collaborations. One major risk of these collaborations is the potential for bad actors (individuals or organizations) to subvert the overall process. More participating organizations means that more users receive insider privileges, exacerbating the problem of *insider threat [7]*. A recent study by a prominent security firm highlighted the critical nature of supply chain threats [24]; this applies not only to suppliers of physical goods but also to data suppliers in large scale collaborations. Another limitation of current practice is that when data quality problems are detected somewhere in a cross-organizational data collaboration, it is difficult to understand and manage the consequences. Finally, as data comes increasingly from far-flung sources, it becomes more difficult for users to know how much to trust the data and whether it is of sufficient quality to feed their decision making processes.

Consider the following scenario (Figure 1): A user requests a loan quote from a Loan Broker service. Behind the scenes, the Loan Broker uses four different bank services. It also uses a credit agency to check the loan applicant, providing the rating as an input to each of the four bank services. Each lender provides a quote, all of which are returned to the user[1]. Collaboration provides a larger selection of data and services, e.g. more loan quotes, more quickly, but opens up the possibility for problems. If the gateway is not properly protected and an attacker can divert personal information, and if the credit rating is not handled properly, banks may issue incorrect quotes to unqualified loan candidates. Having records of executions of this collaboration would enable the loan broker to look for instances of where the collaboration ran differently than expected. Further, should the loan broker become aware of a specific problem in, for example, the handling of credit scores, the loan broker can trace through the provenance to identify all of the impacted customers and quotes.

Data *provenance*[2] enables critical functionality for these cross-organizational collaborations. First, provenance helps establish a baseline for normal behavior. Participants need to know how the collaboration or workflow normally runs, in order to understand its dynamic and potential exposure points. A baseline of normal behavior is also very useful in detecting possible malicious behavior. In collaborations where many participants must be granted some level of

[1] This example is provided by www.mulesoft.com.

[2] Provenance is "information that helps determine the derivation history of a data product...[It includes] the ancestral data product(s) from which this data product evolved, and the process of transformation of these ancestral data product(s)" [22]

COLLABORATECOM 2011, October 15-18, Orlando, United States

DOI 10.4108/collaboratecom.2011.51

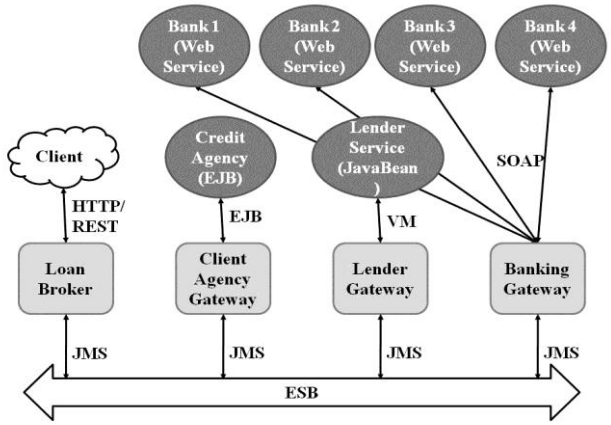

Figure 1. Cross organizational collaboration to produce a loan quote (from mulesource.org)

privileges, there will naturally be many opportunities for bad actors to subvert the process; changing the collaboration's data inputs ultimately may alter its final product. Second, provenance helps users make trust decisions about their data, and know how much they should trust derived data that may come from unfamiliar sources. Third, provenance helps users understand the impact of erroneous data or defective data-generating processes.

In order to provide these services, provenance must contain certain basic information: what, when, who, where, how, why. For instance, in the Loan Broker example, basic information captured for each provenance item would be: *what* the data or service was, a timestamp (*when*), *who* ran it, *where* it was executed, any parameters used (*how*), and where its inputs came from and outputs went to (*why*). Additional information (annotations) can also be stored with the basic provenance information—for example, a given user's assessment of the quality of a particular data source. Importantly, the basic provenance information is a historical record of what actually happened, as opposed to what was supposed to happen and, as such, must be captured as processes are executed and data is modified at each organization. We are agnostic as to whether the information is stored in a centralized provenance manager, or over a distributed set of managers [2].

An example of a provenance graph from one use of the LoanBroker system is shown in Figure 2. The information in this graph helps address the challenges described above. For instance, it is possible to see how the LoanQuote was created: a customer created a request, which was sent through LoanBroker to a credit agency and two banks. Notice that this is distinctly different from the expected behavior of the collaboration. As shown in Figure 1, three banks were expected to contribute to the final loan quote. Thus, provenance can inform a user of the exact occurrences which produced a given piece of

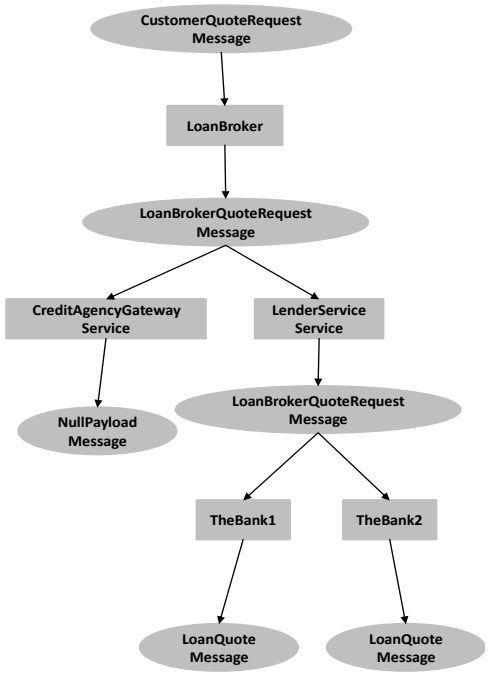

Figure 2. Sample provenance graph captured from an execution of the LoanBroker. Ovals indicate data; rectangles are processes.

data. Additionally, if future investigation shows that Bank1 was hacked and produced bad data, subsequent users of that data can be informed by tracing through the graph. Finally, since we can see what actually occurred over many executions of this workflow, we can establish a baseline of normal behavior, and we can look for deviations from that baseline that may indicate suspicious behavior.

The remainder of this paper is organized as follows. Section II describes the PLUS prototype provenance manager. We then show how PLUS supports the three application needs described above: detection of potential malicious behavior, particularly by authorized insiders (Section III), trust assessment, and taint analysis (both in Section IV). We describe the approach to detecting suspicious behaviors in some detail, and present an initial proof-of-concept evaluation through a user study in Section V. We discuss related work and conclude in Sections VI and VII respectively.

II. PLUS

PLUS [10] is a provenance manager developed at The MITRE Corporation to address these previously unmet requirements shared by most of our U.S. government customers:

- "Open world" collection in distributed, heterogeneous environments, [1]

- Flexible annotation management over provenance, which enables a number of important analysis applications, including the "taint analysis" application in Section IV
- Attribute-based access controls that support flexible sharing of provenance across different classes of users with different privilege levels [23], and
- Techniques to provide more informative provenance when the sensitivity of certain nodes or edges precludes sharing the entire graph [8].

We now describe PLUS capabilities essential for large-scale, distributed collaborations and that enable detection of suspicious behavior and improved tools for assessing trust in information.

A. Distributed Capture Methods

The ability to provide provenance information to assist with collaboration efforts rests on capturingthe provenance information. Similar to [17, 26], we supply an API that any legacy system can call to log provenance information. However, in addition to this basic service, we have focused the system on "coordination" points that are often used in cross-organizational data sharing. For example, an Enterprise Service Bus (ESB) is often used to coordinate applications comprised of data and components from many different organizations. We have modified MULE, a popular open source ESB, to automatically capture and report provenance for all messages passed [1]. Our MULE-based provenance collector is the first provenance capture facility of which we are aware to collect provenance in heterogeneous multi-organizational environments. This capture technique scales effectively and does not noticeably impact the underlying systems [10], but it does enable users of integrated information greater insight into that information's usefulness and flaws. In a nutshell, this method captures provenance by observing the functioning of the ESB as it happens, and reporting appropriate provenance.

The key issue is that the coordination point (in this case the ESB) must capture as much information as possible about the collaboration. Without this information, the ability to provide trust assessments and detect suspicious behaviors is limited. To facilitate collaboration, most collaborative enterprises have a central means of orchestrating information sharing. In our examples, the overall collaboration (such as the loan broker workflow) runs on a common technical infrastructure, such as the MULE ESB. The use of an ESB is not required; however, the use of a central coordination point, whether or not an ESB, provides a simple one-stop place for provenance collection to occur. Alternatively, if no central coordination point is available, this technique can still be used if all collaborators agree on a tool set such that each tool can be provenance enabled (i.e., can report provenance to the PLUS API).

B. PLUS Provenance Storage

Once provenance information is captured, it must be stored for later use. As stated earlier, PLUS can be run as a stand-alone manager, a centralized repository or a set of provenance managers distributed across organizations [2]. PLUS utilizes a MySQL database for provenance storage, and it models provenance similar to OPM [21] as a directed acyclic graph (DAG), $G = (N, E)$, containing a set of nodes, N, and a set of edges, E. Each node has a set of features describing the process or data it represents, e.g., timestamp, description, etc. Edges in the graph denote relationships, such as usedBy, generated, inputTo, etc., between nodes in the graph. A provenance graph may include disconnected subgraphs.

A data node can represent any object the user wishes to register, for example, strings, files, XML messages, relational data items of arbitrary granularity, etc. The data itself is not stored in the PLUS system for security and archiving reasons. However, the provenance capturers can provide any additional "breadcrumbs," such as access method and identifier, to allow users to access the underlying information. Users may annotate anything in a provenance graph with additional metadata.

III. DETECTING SUSPICIOUS BEHAVIORS

Bad actors are a potential problem for all collaborations, especially collaborations among larger groups. In a large, distributed collaboration environment, it is possible for a malicious user to wreak havoc, from stealing data, to disrupting service, to altering results. A malicious user may not even be a part of the collaboration, but merely an opportunistic outsider who has found the weak security link within the set of collaboration services. As a data user today, it is impossible to know for sure that every service provided by other organizations is sufficiently secure.

Consider the problem faced within the intelligence community: sharing data across organizations is being encouraged, but an astute bad actor may be able to quietly redirect data in a workflow to an unauthorized external recipient. In an alternative scenario, instead of merely copying information to an illegal recipient, a bad actor may be able to modify data and therefore decrease the reliability of that data. One of the most effective attacks is to alter just enough data that the entire dataset is untrusted and not used.

We have developed a pilot capability to detect suspicious activities. Given provenance information, we have the ability to describe what *actually* happened, as opposed to what *should have* happened. By using this information, we can discover anomalous and suspicious

behavior. Figure 3 shows modifications to the graphs from Figure 2, illustrating four ways that provenance might be altered in the face of an attack, along with notional explanations for what could cause such modification: (a) disruption of service, (b) data modification, (c) data stealing, and (d) a man-in-the-middle interception of all processing. Even an unknown attack that somehow slips around the provenance capture device would still cause changes to the provenance information that are detectable, as shown in Figure 3e.

In other words, we can use changes in the expected provenance graph as the basis to detect suspicious behavior. We posit that certain attacks will have characteristic signatures in provenance graphs. For instance, consider the provenance graph in Figure 3b, which is identical to the original except for the addition of the black nodes, showing how a man-in-the-middle attack may manifest in the provenance store. Because the point of coordination (an ESB in our earlier example) logs all interactions between the data and processes, if an attacker were to insert an additional step in the workflow, that step would be captured as an additional process and data item.

However, there are many different places a man-in-the-middle might be placed, depending on the security of the underlying services, and thus enumerating all possible attacks in a given system and how they would manifest in the provenance would be impractical. Further, not all attacks will have characteristic signatures, or some classes of attacks may have many possible patterns or manifestations. Instead, we should look for three distinct patterns:

- Truncation: Workflows get shorter or less complex than the expected norm.
- Augmentation: Workflows get longer or more complex than the expected norm.
- Modification: Workflows stay the same size, but the arrangement of internal nodes changes.

These general patterns are easy to identify, so there is no need to enumerate how different types of attacks would affect particular workflows and provenance capture setups. Obviously, with this oversimplification, the false positive rate may be high. For instance, we may flag test runs or user-aborted runs as suspicious behavior. However, at an initial, proof of concept phase, we deem this acceptable.

A. Technique Considerations

We make several assumptions in order to use provenance information to detect suspicious behavior. These include:

- The provenance system itself has not been compromised.
- When an attack happens, it will modify the provenance graph in some way, as the provenance is a record of what executed, and what its result was. In other words, provenance capture is instrumented in a way that it captures actual interactions; and the capture record will change as the interactions change.
- Provenance can typically only detect anomalies at the level of granularity for which it was configured.

We believe that these are valid assumptions. The first assumption conforms to the belief that the goal of any security system should be to raise the bar beyond the reach of most attackers. An attacker must be exceedingly sophisticated to hack both an underlying data or service *and* a separate provenance system. Ultimately there is no such thing as an impenetrable system; additional security layers seek to make it progressively more unlikely that an attacker could cause problems and go undetected.

The second assumption places the onus on the provenance system developer to ensure that provenance capture points are placed at appropriate points in the system. Above we argue that there are high-value capture points which can maximize provenance capture at minimal developer cost. To improve the ability to handle detection of suspicious activities, high-value capture points and obvious exit-points (such as open points in a firewall) should be provenance enabled. Provenance capture must be truly observational and make no assumptions like "if the workflow reached this checkpoint, then these three processes must have just executed successfully."

Finally, provenance granularity can play a large part in whether an attack is detectable or not. Attacks that are substantially below the captured level of granularity cannot necessarily be detected. For example, if a provenance system is monitoring web services, and someone hijacks the operating system running the web service, that may not be detectable through provenance. For this reason, provenance should not be considered a silver bullet; other more traditional layers of security are still required to provide defense in depth.

IV. ASSESSING TRUST IN INFORMATION

Collaboration participants may have a clear picture of the overall set of services and data and how they interplay, but may only truly understand the details of services and data local to themselves. For example, consider the collaboration in Figure 1 to provide loan quotes. The participant responsible for the Banking Gateway understands the minutiae of contacting banks, but only has a vague overall understanding about what occurs to run a credit check. Alternatively, some participants (e.g. the end user) may not have any real idea how data was created or services are expected to behave, and therefore have no basis on which to judge their quality.

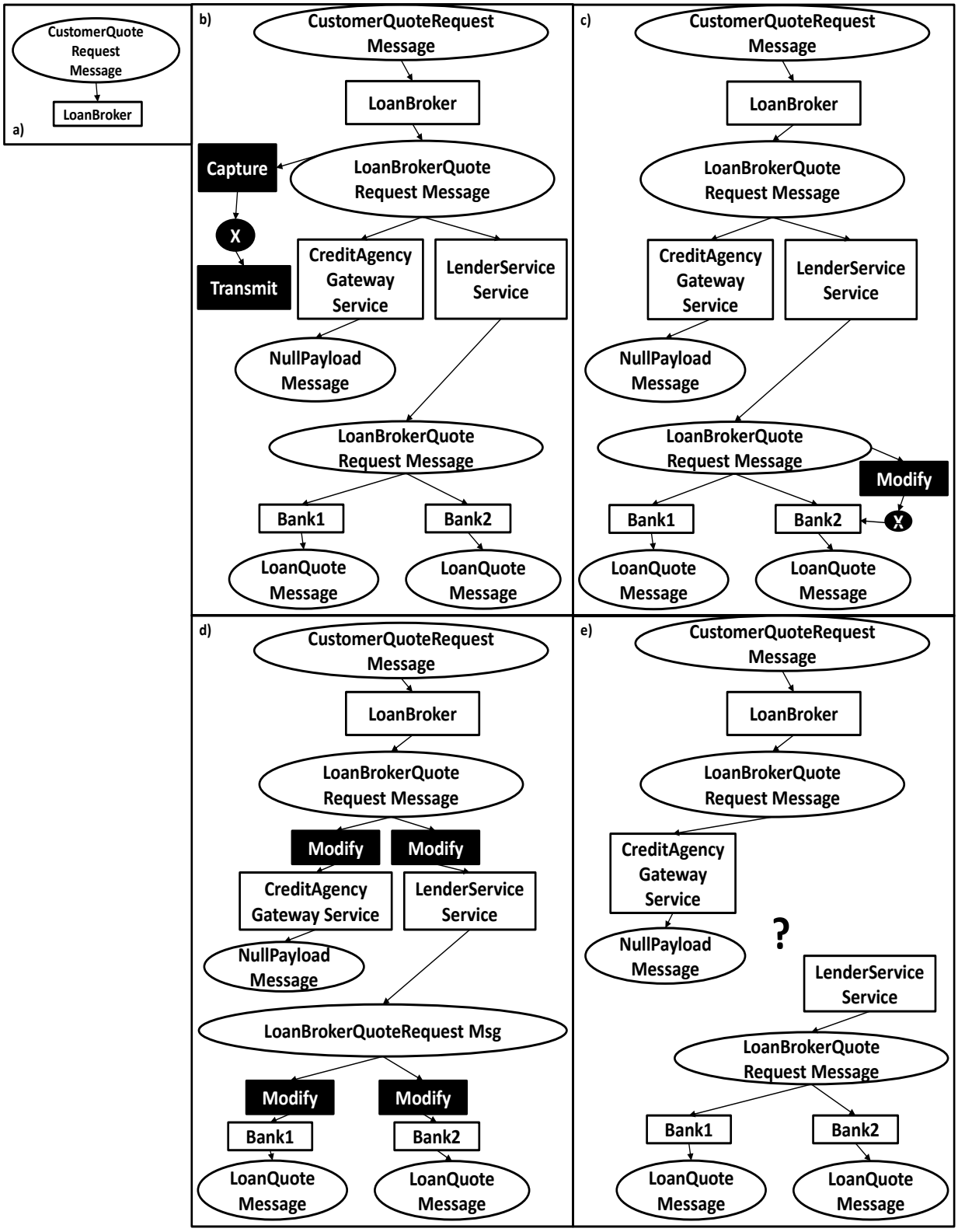

Figure 3: Examples of how different attacks could alter the provenance graph from Figure 2. Suspicious behaviors include: a) disruption of service; b) data stealing; c) data modification; d) man-in-the-middle; e) external/unknown attack.

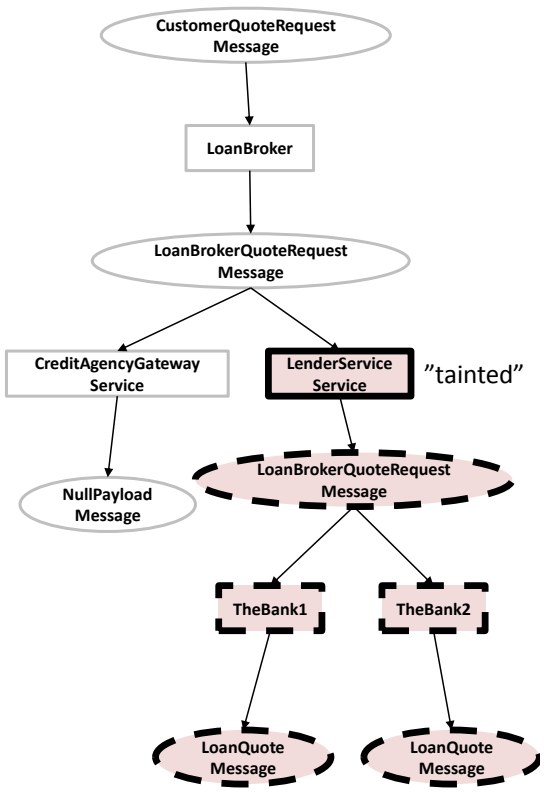

Figure 4: Taint propagation using PLUS

Provenance provides the ability to give participants greater information about actions that occurred outside their purview. Using this information, users can more easily assess whether to use data produced through the data and services utilized. The provenance that PLUS captures is immutable, mirroring the assumption of most prior work. However, in exploring our customers' requirements, it quickly became clear that many of them needed a flexible facility for adding annotations to provenance information. For example, a user may want to enter an opinion about his confidence in a particular piece of information or to note special circumstances that surrounded a certain process execution. These additional annotations (not strictly provenance information) are mutable, since any of these assessments might change. Such annotations are essential in cross-organizational information sharing, in which a user from Organization2 who uses data from Organization1 may have no knowledge of how that data was generated, and whether it can truly be used for her purposes. In such cases, provenance together with user assessments of confidence and social networks of trusted colleagues can do a great deal to increase trust that information is suitable for the intended purpose.

In addition, provenance provides additional functionality that can help alert users to potentially harmful data. In a widespread collaboration, if a problem with a dataset or service is detected, the process of identifying other users of that dataset is often either impossible or at least arduous. Building a "taint analysis" application over provenance can help inform all users of potentially bad information.

Such annotations can help our customers understand the consequences of a data modification cyber-attack. For example, suppose that a credit check agency discovers that an attacker has subverted the LenderService to send incorrect credit scores with all loan requests. In the past, an organization would correct the problem locally, but downstream users of that data, such as Bank1, would remain unaware of the consequences of any actions they had already taken on the basis of bad information. PLUS provides the ability for a user to annotate the suspect data or service invocation as being "tainted" (e.g., the LenderService service in Figure 4) This taint marking is then propagated forward to all data and processes that rely upon it (the nodes with a bold, dashed outline in Figure 4), and can be seen by the data owners in a different organization, thus alerting them to a potentially serious issue. While a bank may have already generated a quote on the basis of the bad credit report, provenance provides the ability to assess the scope and severity of the problem, since the bank and the loan broker can enumerate exactly which customers were affected. Whenever integration is being performed with data generated across multiple organizations, this is an essential capability to support proper data usage and error correction.

V. FEASIBILITY EXPERIMENT

To our knowledge, this work represents the first effort to apply provenance to the problem of identifying suspicious behavior that may indicate insider attacks. Section III presented our initial ideas about attack signatures that could be detectable in provenance. This section describes a small scale feasibility experiment we conducted to see if this approach warranted further development and more rigorous evaluation.

A. Implementation

Provenance can be used to detect suspicious behavior by finding deviations from the norm. If we can compare any provenance graph to a known good set, it will be possible to see suspicious patterns. However, provenance graphs quickly become large and unwieldy with too many nodes and edges for a human to track. Additionally, sometimes there are a number of legitimate patterns to the collaboration, not just one. (For example, some banks might not respond with quotes for low-credit applicants; others might provide multiple quotes for different products). Even further, the population of provenance graphs can grow quite large, and an attacker may only modify a few instances of a collaboration which runs dozens of times per hour.

We have developed an easy way for a human user to "eyeball" the properties of a large population of graphs to highlight information outside of the norm.

Each provenance graph is given a number of summary statistical "features", such as total number of nodes, ratio of processes to data, and so on. All of these statistical features together comprise a "fingerprint" of a particular provenance DAG; Figure 5 shows selected properties from 6 sample graph fingerprints. The interface provides the user with the ability to sort a large number of graphs on the basis of how much one of those features deviates from the norm for the database; for example, the program quickly allows users to display all graphs whose number of nodes is three standard deviations higher than the mean for a given population of known graphs. Using this information, a user can take a population of many thousands of graphs, and quickly determine the small set of graphs where potentially suspicious activity is present.

Once a suspicious graph is located, the software permits searching on other criteria besides just the statistical measures. For example, users can find all nodes in all graphs that share a common name, or particular annotation or item of metadata.

B. *Dataset*

Our limited initial experiment was conducted with more than 17,000 captured provenance graphs from a large scale simulation exercise. These graphs were captured through the Mule ESB, and represent real records of a complex, multi-organizational collaboration (one of the 17,000 is shown in Figure 6). Starting with these graphs, five different target suspicious graphs were inserted into the base (clean) set of provenance graphs. Briefly, the five suspicious graphs represented notional examples of:

- A data modification attack, where data is "stolen", modified, and then re-injected into the normal workflow.
- A data stealing attack, where data is ex-filtrated via an unknown process.
- A disruption attack (or "denial of service"), where a number of services are taken off-line.
- A man-in-the-middle attack, where every application interaction was run through an external (unknown) process.
- An uncategorized attack that severs a single graph into two distinct graphs by breaking or obscuring a single application interaction.

C. *Method*

We recruited an experienced information security analyst with some prior knowledge of provenance but no experience with the PLUS tool. The user was given some background information about what constituted a "normal" pattern in the graph, and what the mission behind the graph was. He then spent three hours over two sessions searching the data set for suspicious graphs. The user was given a significant amount of time to browse graphs chosen at random to build up an intuition of what was considered normal. The user concluded that small graphs that generally looked like a tree with one main branch, splitting to two branches, and comprising five to seven nodes were most prevalent.

Link to Graph	Total Nodes	Total Edges	Max Incoming Edges	Max Outgoing Edges	Node/Edge Ratio	Average Total Edges	Data/ Invocation Ratio	Invocation/ Total Ratio
Go	500	781	2	453	0.64	3.27	7.93	0.11
Go	100	133	2	94	0.75	3.46	1.38	0.42
Go	100	257	2	232	0.39	2.76	24.00	0.04
Go	100	514	2	229	0.19	2.38	1.08	0.48
Go	85	84	2	90	1.01	3.95	3.25	0.24
Go	84	83	2	40	1.01	3.95	2.36	0.30

Figure 5. Example of graph "features"; each row is considered a fingerprint

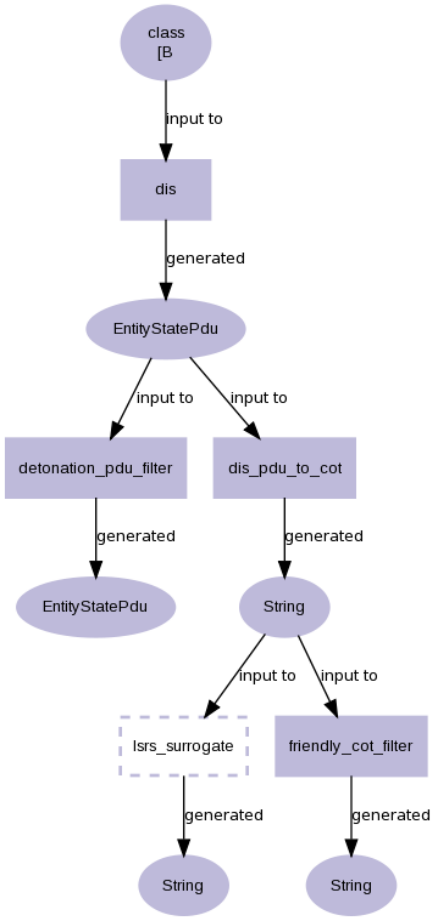

Figure 6: Sample real provenance graph generated via a large scale simulation exercise.

D. *Results and Discussion*

The user reported using the following search heuristics:

- Looking for graphs with large Z scores for an attribute – graphs were filtered based on having a Z score of 3 or more for total nodes, total edges, and max incoming edges. The graphs with the largest Z scores using these measures were then each considered. If a small number of these looked unusual compared to the others, then those were tagged as being suspicious. However, not all such graphs were suspicious.
- Looking for graphs with a node labeled "x" – node labels tended to be identifiers of one or two multi-letter identifiers. Some nodes were simply labeled with an "x" and those stood out as being unusual.
- Looking for graphs with nodes with repeated labels – most graphs seemed to represent a flow from one process to the next. No graphs had cycles represented graphically but all nodes in a graph tended to have distinct labels. So two nodes in the same graph, in the same path, with the same label, appeared anomalous.

Using these heuristics, the user discovered two of the five attacks in the data set, specifically the "uncategorized" attack, and the "man-in-the-middle" attack. In addition, the user discovered two additional anomalous graphs that were not part of the original experimental setup. These graphs corresponded to test data from an unrelated process that was present in the database at the time.

While the test graphs that were unexpectedly found were not part of the experiment at the time, they did serendipitously demonstrate the user's ability to find something out of the ordinary in the data set. Additionally, the two attacks that were found provide initial evidence that provenance can be a useful tool for detecting suspicious behavior. In addition, our expert security analyst was enthusiastic about the potential of this approach, and was especially interested in how it could be combined with other techniques, such as more traditional intrusion detection systems, to provide improved multi-layer defenses. Indeed, looking for anomalous patterns inside of transaction logs is a technique used in many other places within the security community; provenance however represents a fundamentally new data feed that details high-level application and service interactions. Existing security tools have traditionally focused on much lower-level data feeds (such as network protocol interactions) that provide value at a different layer of security.

VI. RELATED WORK

Provenance, or the history of information, has garnered interest in government, commercial and scientific circles. Topics of provenance study include capture [9, 17], storage [11], reasoning [6, 15], security [18, 27], usability [12, 20, 25], etc.

In particular, provenance has been touted as a tool to assist with scientific collaboration. A large number of scientific applications [3, 9, 12, 16, 17, 19, 20, 25, 26] have been built to assist scientists harness the power of provenance using specific applications. For instance, in ES3 [16], the applications used by scientists for data analysis are modified to capture provenance. Other applications that wish to be provenance-aware build this capability directly into the application [9], or require the user to utilize a specific workflow manager which quietly collects provenance information [3, 19, 25]. Each of these methods is limiting in terms of the applications and environments that scientists are constrained to use. Other methods such as: PASS [22], Karma [26], and PreServ [17] allow more general capture by positioning capture points anywhere of interest. Of the techniques described

so far, this technique can capture provenance from the most diverse set of applications, and does not require pre-planning on the part of the user.

Descriptions of provenance usage for scientific collaboration was used during a galaxy cluster finder experiment [4], or for finding appropriate visualization for scientific experimentation [5]. In [14] the needs and requirements for provenance to assist in scientific collaboration are discussed. Further research into what provenance is needed to support collaboration in a given domain is needed.

[13] describes provenance-based techniques for assessing data trustworthiness using data and path similarity. The work does not address provenance collection and could leverage PLUS' "open world" collection techniques. In addition, their trust assessment framework could be implemented on top of the PLUS annotation facility.

VII. CONCLUSIONS AND FUTURE WORK

In this work, we describe how provenance adds value in large-scale collaborations by detecting suspicious behaviors that could result from cyber-attacks, helping users assess trust in information, and managing the downstream consequences of faulty data or process executions. We showcase PLUS, a working provenance system with robust capture, storage and administrative capabilities. Using PLUS, we present an initial user study on the ability of provenance to detect suspicious behavior. This early feasibility experiment provides some encouragement that provenance can improve detection of improper behavior in large-scale collaborations.

In our limited experiment, detection of anomalous behavior was performed by a skilled human analyst using a simple tool that allowed query over provenance graph fingerprints. A promising research direction is to automate some or all of the anomaly detection, using both supervised learning (under the guidance of a security expert) and unsupervised approaches. Better visualization tools may supplement machine learning approaches; for example, analysts might benefit from flipbooks of graph thumbnails (inspired by the iPod's "cover flow"). Finally, research is needed to combine provenance-based approaches with more traditional intrusion detection techniques and to evaluate the performance of the resulting hybrids.

In addition, once we provide better tools to identify suspicious behaviors, further research is needed to attribute them to bad actors. Finally, additional patterns that distinguish suspicious activities should be explored, in particular the pattern of ownership, i.e., the "chain of custody" showing which organization owns which data at which point, may change in the provenance store during an attack.

BIBLIOGRAPHY

[1] M. D. Allen, A. Chapman, B. Blaustein, and L. Seligman, "Provenance Capture in the Wild," *International Provenance and Annotation Workshop (IPAW)*, 2010.

[2] M. D. Allen, A. Chapman, B. Blaustein, and L. Seligman, "Getting It Together: Enabling Multi-organization Provenance Exchange," *TaPP*, 2011.

[3] I. Altintas, O. Barney, and E. Jaeger-Frank, "Provenance Collection Support in the Kepler Scientific Workflow System," Provenance Collection Support in the Kepler Scientific Workflow System, 2006.

[4] J. Annis, Y. Zhao, J.-S. Vöckler, M. Wilde, S. Kent, and I. T. Foster, "Applying Chimera virtual data concepts to cluster finding in the Sloan Sky Survey," *SC*, pp. 1-14, 2002.

[5] L. Bavoil, S. Callahan, P. Crossno, J. Freire, C. Scheidegger, C. Silva, and H. Vo, "VisTrails: Enabling Interactive Multiple-View Visualizations," *IEEE Visualization*, pp. 18-26, 2005.

[6] O. Benjelloun, A. D. Sarma, A. Halevy, and J. Widom, "ULDBs: Databases with Uncertainty and Lineage," *VLDB Seoul, Korea*, pp. 953-964, 2006.

[7] E. Bertino, "Protecting Information Systems from Insider Threats - Concepts and Issues (Keynote)," *IEEE International Conference on Information Reuse and Integration (IRI 2011)*, 2011.

[8] B. Blaustein, A. Chapman, L. Seligman, M. D. Allen, and A. Rosenthal, "Surrogate Parenthood: Protected and Informative Graphs," *PVLDB*, 2010.

[9] P. Buneman, A. Chapman, and J. Cheney, "Provenance Management in Curated Databases," *ACM SIGMOD*, pp. 539-550, 2006.

[10] A. Chapman, M. D. Allen, B. Blaustein, and L. Seligman, "PLUS: A Provenance Manager for Integrated Information," *IEEE International Conference on Information Reuse and Integration (IRI '11)*, 2011.

[11] A. Chapman, H. V. Jagadish, and P. Ramanan, "Efficient Provenance Storage," *SIGMOD*, pp. 993-1006, 2008.

[12] S. Cohen-Boulakia, O. Biton, S. Cohen, and S. Davidson, "Addressing the provenance challenge using ZOOM," *Concurrency and Computation: Practice and Experience*, vol. 20, pp. 497-506, 2008.

[13] C. Dai, D. Lin, E. Bertino, and M. Kantarcioglu, "An Approach to Evaluate Data Trustworthiness Based on Data Provenance," *Secure Data Management Workshop at VLDB*, 2008.

[14] D. Donaldson and K. Fear, "Provenance, End-User Trust and Reuse: An Empirical Investigation," *TaPP*, 2011.

[15] J. N. Foster, T. J. Green, and V. Tannen, "Annotated XML: Queries and Provenance," *PODS*, pp. 271-280, 2008.

[16] J. Frew, D. Metzger, and P. Slaughter, "Automatic capture and reconstruction of computational provenance," *Concurr. Comput. : Pract. Exper.*, vol. 20, pp. 485-496, 2008.

[17] P. Groth, S. Miles, and L. Moreau, "PReServ: Provenance Recording for Services," *Proceedings of the UK OST e-Science second All Hands Meeting 2005 (AHM'05)*, 2005.

[18] R. Hasan, R. Sion, and M. Winslett, "The Case of the Fake Picasso: Preventing History Forgery with Secure Provenance," in *FAST*. San Francisco, 2009, pp. 1-14.

[19] P. Missier, K. Belhajjame, J. Zhao, and C. Goble, "Data lineage model for Taverna workflows with lightweight anotation requirements," Data lineage model for Taverna workflows with lightweight anotation requirements, 2008.

[20] P. Missier, S. M. Embury, M. Greenwood, A. Preece, and B. Jin, "Managing information quality in e-science: the qurator workbench," *SIGMOD*, pp. 1150-1152, 2007.

[21] L. Moreau, J. Freire, J. Futrelle, R. McGrath, J. Myers, and P. Paulson, "The Open Provenance Model," University of Southampton 2007.

[22] K.-K. Muniswamy-Reddy, D. A. Holland, U. Braun, and M. I. Seltzer, "Provenance-Aware Storage Systems," *USENIX Annual Technical Conference*, pp. 43-56, 2006.

[23] A. Rosenthal, L. Seligman, A. Chapman, and B. Blaustein, "Scalable Access Controls for Lineage," *First Workshop on Theory and Practice of Provenance Systems (TaPP)*, 2009.

[24] RSA, "Advanced Persistent Threats Summit: Findings," http://www.rsa.com/innovation/docs/APT_findings.pdf, 2011.

[25] C. E. Scheidegger, H. T. Vo, D. Koop, J. Freire, and C. Silva, "Querying and Re-Using Workflows with VisTrails," *SIGMOD*, 2008.

[26] Y. Simmhan, B. Plale, and D. Gannon, "Karma2: Provenance Management for Data Driven Workflows," *Journal of Web Services Research*, vol. 5, 2008.

[27] J. Zhang, A. Chapman, and K. LeFevre, "Fine-Grained Tamper-Evident Data Pedigree," *University of Michigan Technical Report*, 2009.

ACKNOWLEDGMENTS

The authors thank Aaron Temin for valuable advice and serving as the security expert in our feasibility study