Inference and Cover Stories

A common use for cover stories is to provide a plausible explanation for an otherwise sensitive event. For example, a plane might be said to carry food when its actual cargo is weapons. Without a cover story, the fact that the cargo is not identified may lead to increased interest from an uncleared user, something which may not be desirable if a mission is to be successful.

Cover stories may also be used to release shades of information. Here, instead of lying they are releasing only sanitized information. At the confidential level, a user may be told that a plane is carrying equipment, while a top secret user is told that the plane is carrying electronic equipment.

Cover stories may not always have the ability to protect sensitive information. For example, an uncleared user may have enough world knowledge to discover that a given cover story is not plausible. There is a difference, however, between a cover story that cannot protect sensitive information and a cover story which itself causes a breach of security. This paper examines cover stories that indirectly disclose the very information they are attempting to protect.

INTRODUCTION

Cover stories are plausible explanations which replace gaps of information that the low user (one who is uncleared or insufficiently cleared for the information) would normally see, gaps that might otherwise cause a curious user to attempt to piece together information for which he is unauthorized. A primary goal of a cover story is to satisfy the curiosity of an unauthorized user.

In virtually every plausible cover story, however, is some factual information. For example, if Smith is a radar technician and that information is secret, then a plausible cover for Smith would not be "Jones is an engineer." Generally speaking, the object for which a cover story is being developed must be correctly identified.[1]

1. Where the object is not correctly identified, as in someone going "under cover," some attribute(s) of that object must be acknowledged (i.e., height, weight, etc.)

In addition to identifying the object, plausibility often requires other information about the object to be identified. Ensuring that the factual information released is unclassified[2] is not sufficient, because an attacker can now use this factual information to derive new and possibly classified information on the object via inference. To be sure the cover story does not breach security, it must be shown that all factual information released in the cover story and all inferences possible from that factual information are unclassified.

This paper focuses on

- Cover stories,
- How an improper cover story can lead to a breach of security, and
- Recognizing potential inferences caused by cover stories.

COVER STORIES THAT CAN'T PROTECT INFORMATION

Cover stories are used to protect information. Typically, they give a plausible explanation for information that would not otherwise exist at a user's security level. However, a user may have enough data to piece together what the cover story is trying to protect. When this happens, the cover story cannot be relied on for protection, although it may be enough of a deterrent to mislead a portion of the unauthorized users.

In the Mission table (M) shown below, the cover story for flight# C1A2946 is that it is a supply mission, running medical supplies to Europe.

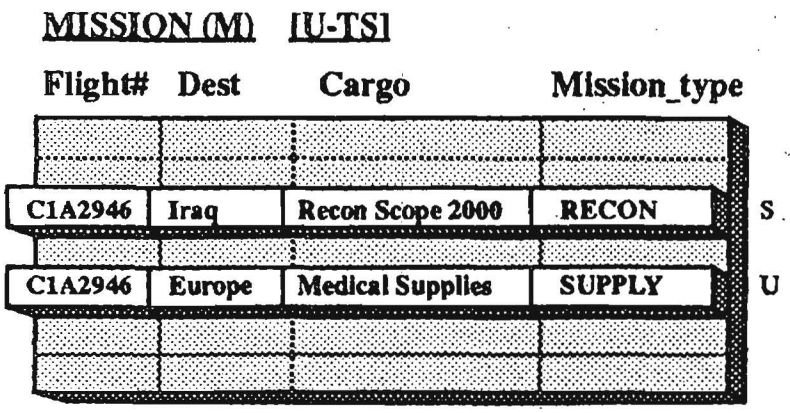


Fig. 1.

This may appear sufficient to convince the novice user that flight# C1A2946 is a supply mission. Figure 2, however, shows that an unclassified user can piece together the fact that flight# C1A2946 is a reconnaissance mission. If this relationship is secret, then the

2. or *properly* classified

cover story in figure 1 alone cannot protect that relationship. This is an example of a cover story that simply cannot protect information (because the information is available elsewhere). For this particular example, a second cover story[3] in either FP, P, or OM is one alternative solution to protecting the secret relationship; a second alternative is to consider redesigning or reclassifying the schema to avoid this type of information flow.

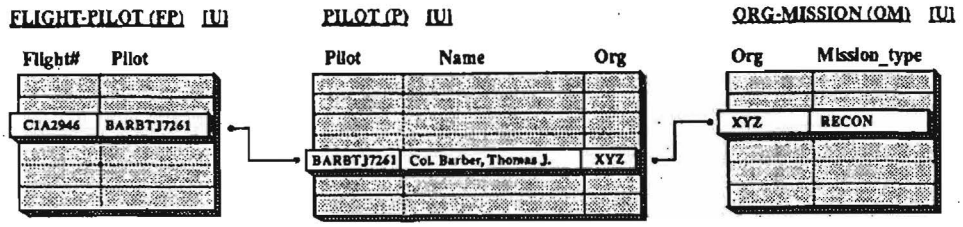


Fig. 2.

COVER STORIES THAT BREACH SYSTEM SECURITY

In cover stories where some factual data are released, there exists the possibility that a low user could exploit these data to infer high (i.e., highly classified information or that which a "low" user cannot have access to) information. The Organization (O) table shown in figure 3 contains the relationship between org "XYZ" and specialty "Russian Language" at the confidential level. At the unclassified level, XYZ's cover story is that its specialty is simply "Language." This cover story is consistent with the user's classification guideline listed in appendix A. In this case, the actual phone number for XYZ was released. Although this is not in direct violation of the classification guideline, it is an indirect breach of security. In conjunction with the database schema shown in figure 4, a low user can use XYZ's phone number to infer its specialty by identifying potential employees of XYZ and *their* specialty.

This is an example of a cover story which itself causes a breach of security, by releasing information necessary to complete an inference path./

3. The original data would have to be removed or reclassified.

ORGANIZATION (O) [U-C]

Org	Specialty	Phone	
ABC	Maintenance	555-1111	U
XYZ	Russian Language	555-1234	C
XYZ	Language	555-1234	U

Fig. 3.

PLAUSIBILITY AND USABILITY

Why release factual information in a cover story? It is required in some circumstances to make the cover story plausible. An incorrect attribute may lead a user to question the validity of other attributes in the record, thereby defeating the purpose of the cover story. Unsatisfied with the information, the user may try to retrieve a different answer using an alternative method (i.e., inference). Where the cover story is really a sanitized version of the truth (fig. 3), factual information is sometimes required to make the cover story usable.

Although it is always possible to force the user to redesign the database to reduce or possibly eliminate poor schema design, it is our goal to allow these designs as long as their inefficiency does not have an adverse effect on security. Forcing users to adhere to strict design principles can have the effect of driving them away from secure systems altogether. The goal here is to impact the users only when security is at risk, allowing them to work without restriction where possible. The user is responsible for proper classification of data within a record, while the database monitors classification consistency among collections of tables.

As stated earlier, virtually every cover story contains factual information. What must be ensured is that this information cannot directly or indirectly disclose sensitive data to an unauthorized user.

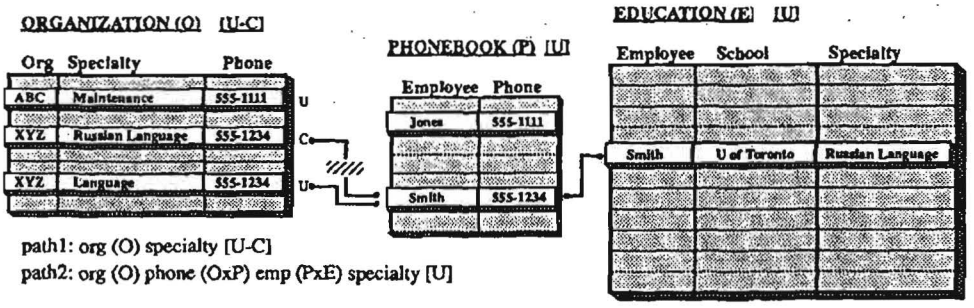


Fig. 4.

RECOGNIZING INFERENCES CAUSED BY COVER STORIES

The design flaw whereby a cover story opens an inference channel was first introduced in [1]. The channel can be characterized by a relation whose attribute leads to an external attribute which coexists (and is classified) in the original relation. The definition uses the notion of a path and level of a path. These are discussed here and are used in the formal definition that follows.

A *path* identifies the set of attributes and relations used to substantiate a relationship between two attributes; it is a road map showing how the attributes are joined. The smallest path is between two attributes in the same relation, and by definition it has a length of one. The Organization (O) table in figure 4 shows the relationship between org and phone. It has a length of one, and the path is written *org (O) phone*. More complex relationships use a recursive definition for path. Each join increases the length by one. The relationship between org and employee in figure 4 is substantiated by joining the Organization and Phonebook relations. The path has a length of two and is written *org (O) phone (OxP) employee*. Cyclical paths are not allowed; neither tables nor attributes can be revisited in a path.

Given that:

A is a set of n + 1 attributes, and

R is a set of n relations

A path of length (n = 1) is defined as:

$P(a_0,a_1,r_1,A,R) = [a_0\,(r_1)\,a_1 \mid$
$a_0,a_1 \in A$
$\wedge\, a_0 \neq a_1$
$\wedge\, r_1 \in R$
$\wedge\, a_0,a_1 \in r_1$
$\wedge$ Cardinality (A) = 2
$\wedge$ Cardinality (R) = 1]

A path of length (n > 1) is defined as:

$P(a_0,a_n,r_n,A,R) = [P(a_0,a_{n-1},r_{n-1},A\text{-}a_n,R\text{-}r_n)\,(r_{n-1} \times r_n)\,a_n \mid$
$a_0,a_{n-1},a_n \in A$
$\wedge\, a_0 \neq a_{n-1} \neq a_n$
$\wedge\, r_{n-1},r_n \in R$
$\wedge\, r_{n-1} \neq r_n$
$\wedge\, a_{n-1},a_n \in r_n$
$\wedge\, a_{n-1} \in r_{n-1}]$

The *level* of a path is defined by the relations used in traversal; it is composed of the path's hierarchical and nonhierarchical security levels unioned together. The hierarchical level is the least upper bound of all the hierarchical levels encountered in the path. The nonhierarchical level is the union of all nonhierarchical levels encountered.

The level of a path *P* at time *t* is defined as:

$L(p,t) = [L_h(R,t) \cup L_c(R,t) \mid \exists\, a_0,a_n,r_n,A\,[p = P(a_0,a_n,r_n,A,R) \wedge t \in \text{time}]]$

Where

$L_h(\text{nil},t) = U$
$L_h(R,t) = [\text{Level}(r,t) \mid r \in R \wedge \text{Level}\,(r,t) \geq L_h\,(R\text{-}r,t)]$
$L_c(\text{nil},t) = \text{nil}$
$L_c(R,t) = [\text{Comp}(r,t) \cup L_c\,(R\text{-}r,t) \mid r \in R]$

Level(r,t) = Hierarchical security level associated with relation *r* at time *t*
Comp(r,t) = Nonhierarchical compartment(s) associated with relation *r* at time *t*

A = set of attributes
R = set of relations

{U, C, S, TS} = Hierarchical security levels, and U < C < S < TS

We say that a cover story is the cause of inference if it releases information that could be used to yield the true state of what the cover story is designed to protect. Formally stated, potential inference through the use of a cover story is defined as[4]

$$I_c(a_i,a_j) = [\text{TRUE} \mid \exists\, p_1,p_2,a_k,r,r_n,A_1,A_2,R_1,R_2,t$$
$$[p_1 = a_i(r)a_j = P(a_i,a_j,r,A_1,R_1)$$
$$\wedge\, p_2 = P(a_i,a_j,r_n,A_2,R_2)$$
$$\wedge\, a_i \neq a_j \neq a_k$$
$$\wedge\, a_i\,(r)\,a_k \in p_2$$
$$\wedge\, \neg(L(p_1,t) \leq L(p_2,t))]]$$

Looking back at our example in figure 4, we see that *specialty* (a_j) is the attribute that is both external and local to Organization. *Phone* (a_k) is the attribute or "hook" that can be used in a path leading to *specialty* outside of Organization. Using our definition, we see that a potential inference does exist. The values used to substantiate this are shown below.

a_i = org, a_j = specialty, a_k = phone

A_1 = {org, specialty}

A_2 = {org, phone, emp, specialty}

r = O

R_1 = {O}

R_2 = {O, E, P}

p1 = org (O) specialty

p2 = org (O) phone (OxE) emp (ExP) specialty

org (O) phone $\in$ p_2

$\neg(L(p_1 = C) \leq L(p_2 = U))$

$\therefore I_c$(org, specialty) = TRUE

Notice that we assign path classifications to suit our needs. The only constraint is that the levels assigned are consistent with the range of possible values. For example, Organization can be assigned either unclassified (U) or confidential (C) security levels; however, Phonebook is strictly unclassified. The fact that there exists a potentially classified path p_1 and a potentially unclassified path p_2 is a necessary ingredient to show the design is inherently flawed and could *potentially* breed inference.

The actual tuple values shown in figure 4 are not used when determining the soundness of the design. They are used here to illustrate how a poor design could lead to an inference path, via specific database instance.

---

4. In this context, $\in$ is used to denote a subpath,. i.e., org (O) phone $\in$ org (O) phone (OxE) emp (ExP) specialty.

A database is said to be free of potential inference through the abuse of a cover story if for all attributes a_i and a_j, $I_c(a_i,a_j)$ is false.

Where it is unfeasible to eliminate potential inference paths, run-time analysis would monitor the contents of the database based on design-time analysis. Run-time analysis would substantiate when a potential inference path becomes an actual inference path.

RELATIONSHIP TO OTHER CLASSES OF INFERENCE

A database where $I_c(a_i,a_j)$ is false for all a_i and a_j is by no means inference free; it means only that a cover story cannot be used against *itself*. The database is still subject to other classes of inference; there is no guarantee that the information released in a cover story could not be used to exploit some *other* type of inference channel, hence the need for additional analysis. The inference definition presented here identifies a specific class of inference. It depends on a set of complementary tools which detect other classes of inference in order to form a comprehensive inference policy. Such a policy is required if we are to trust multilevel databases to truly protect the information it manages.

SUMMARY

Cover stories, whose intention is to protect information, can fail in two ways. They can take either an active or a passive role in the disclosure of information to the unauthorized user.

Garvey [4] recognized that a cover story cannot protect information if the user has enough data to piece together via inference the information the story is trying to protect. In this case, the cover story is taking a passive role in the disclosure of information to the unauthorized user. To counter this he indicates the need to identify such situations and provide additional cover stories to block the inference path(s) which jeopardize the initial cover story.

Plausibility of a cover story may require some factual, noncritical information to be released. Where factual information is released, a hostile user could possibly abuse the information to obtain critical (high) data by using them to complete an inference path. Here, the cover story is taking an active role in the disclosure of information to an unauthorized user. This paper has focused on detecting whether a cover story can be used to disclose information itself is trying to protect. This paper has not addressed the impact a cover story has on other information in the database which the cover story does not directly protect; i.e., this paper has not discussed the impact a cover story has on other classes of inference. Such considerations are being addressed separately, with the long-term goal of developing a policy that *does* address a range of inference classes.

Finally, the definition of $I_c(a_i,a_j)$ would presumably check the entire database to determine if a cover story could exist and could potentially be used to divulge the

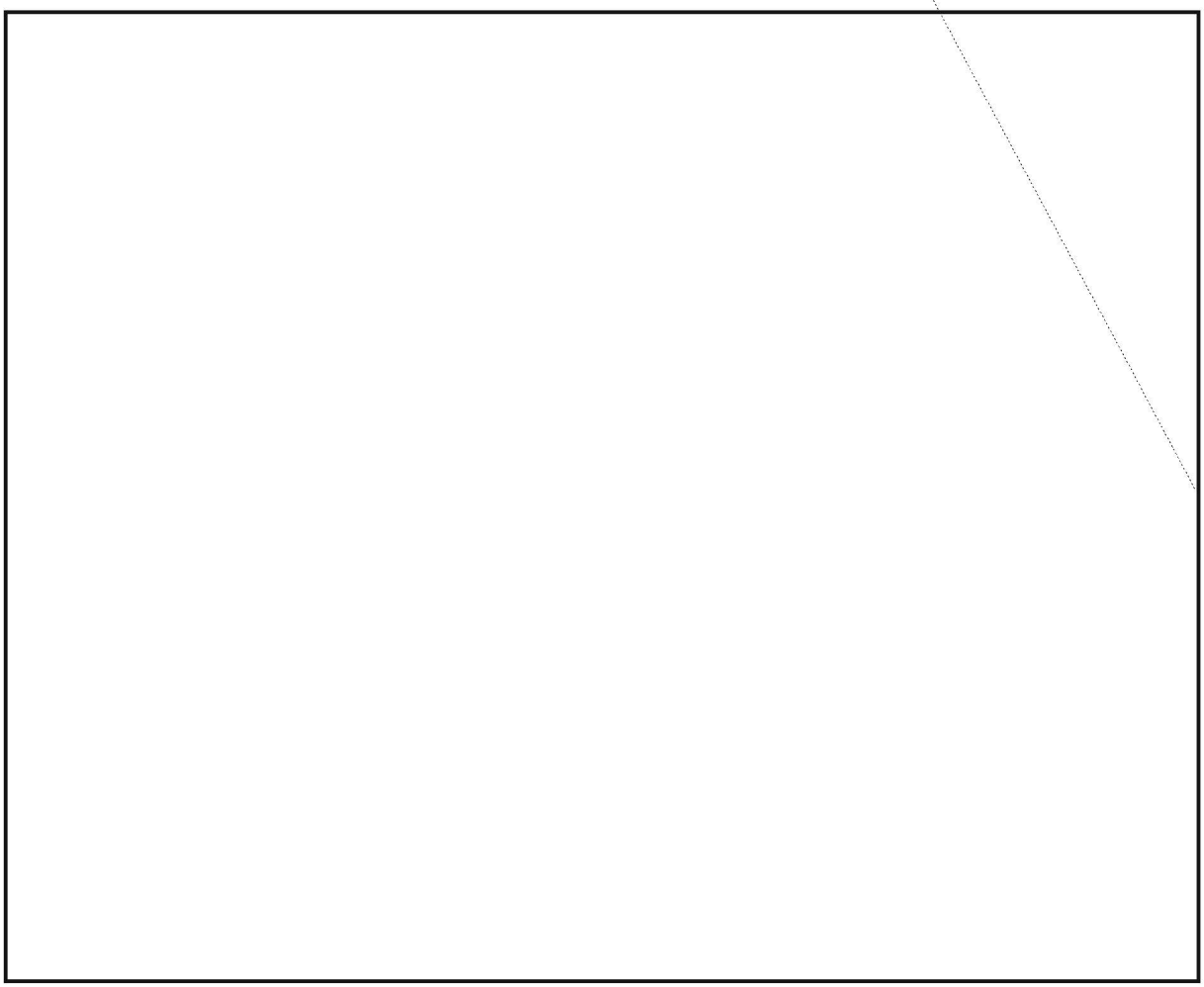
relationship between a_i and a_j. Alternatively, one could modify the definition to use it as a specific tool in developing cover stories. Passing *r* as an argument, rather than testing to see if there exists some *r* that would satisfy the equation, would provide a useful tool to someone creating a cover story in a specific relation. Additionally, instead of returning TRUE, the definition would return the offending piece of information:

$$I_c(a_i,a_j,r) = [a_k | \exists\, p_1,p_2,r_n,A_1,A_2,R_1,R_2,t$$
$$[p_1 = a_i(r)a_j = P(a_i,a_j,r,A_1,R_1)$$
$$\wedge\, p_2 = P(a_i,a_j,r_n,A_2,R_2)$$
$$\wedge\, a_1 \neq a_j\, a_k$$
$$\wedge\, a_i\,(r)\, a_k \in p_2$$
$$\wedge\, \neg(L(p_1,t) \leq L(p_2,t))]]$$

Applying this to the cover story we wish to pose for specialty in the Organization relation of figure 4:

$$I_c(\text{org, specialty, O}) = \text{phone}$$

This indicates that if the actual phone number is supplied in a cover story for Organization, it is possible a hostile user could complete an inference path between *org* and *specialty* using *phone*; to prevent this, a cover story for phone number must be provided.



REFERENCES

[1] Binns, L. J. "Inference Through Polyinstantiation," *Proceedings of the Fourth RADC Database Security Workshop*, April 1991.

[2] Binns, L.J. "Inference Through Secondary Path Analysis," *Proceedings of the Sixth IFIP WG 11.3 Working Conference on Database Security*, August 1992. (Submitted)

[3] Garvey, T.D., T.F. Lunt , and M.E. Stickel. "Abductive and Approximate Reasoning Models for Characterizing Inference Channels," *Proceedings of the Fourth Workshop on the Foundations of Computer Security*, Franconia, NH, June 1991.

[4] Garvey, T.D., and T.F. Lunt. "Cover Stories for Database Security," *Proceedings of the Fifth IFIP WG 11.3 Working Conference on Database Security*, November 1991.

[5] Garvey, T.D., T.F. Lunt, X. Qian, and M.E. Stickel. "Toward a Tool to Detect and Eliminate Inference Problems in the Design of Multilevel Databases," *Proceedings of the Sixth IFIP WG 11.3 Working Conference on Database Security*, August 1992.

[6] Hinke, T.H. "Inference Aggregation Detection in Database Management Systems," *Proceedings of the 1988 IEEE Symposium on Security and Privacy*, Oakland, CA, April 1988.

[7] Hinke, T.H. "Database Inference DEsign Approach," in *Database Security II: Status and Prospects, Results of the IFIP WG 11.3 Workshop on Database Security*, October 1988, C.E. Landwehr, Ed., North-Holland, 1990.

[8] Hinke, T.H., and H.S. Delugach. "AERIE: An Inference Modeling and Detection Approach for Databases," *Proceedings of the Sixth IFIP WG 11.3 Working Conference on Database Security*, August 1992.

[9] Lin, T.Y. "Inference Free Multilevel Databases," *Proceedings of the Fourth RADC Database Security Workshop*, April 1991.

[10] Lin, T.Y. "Inference Secure Multilevel Databases," *Proceedings of the Sixth IFIP WG 11.3 Working Conference on Database Security*, August 1992.

[11] Lunt, T.F. "Aggregation and Inference: Facts and Fallacies," *Proceedings of the 1989 IEEE Symposium on Security and Privacy*, Oakland, CA, April 1989.

[12] Lunt, T.F. "The True Meaning of Polyinstantiation: Proposal for an Operational Semantics for a Multilevel Relational Database System," *Proceedings of the Third RADC Database Security Workshop*, June 1990.

[13] Lunt, T.F. "Polyinstantiation: an Inevitable Part of a Multilevel World," *Proceedings of the Fourth Workshop on the Foundations of Computer Security*, Franconia, NH, June 1991.

[14] Morgenstern, M. "Controlling Logical Inference in Multilevel Database Systems," *Proceedings of the 1988 IEEE Symposium on Security and Privacy*, Oakland, CA, April 1988.

[15] Thuraisingham, B. "Handling Security Constraints during Multilevel Database Design," *Proceedings of the Fourth RADC Database Security Workshop*, April 1991.

[16] Thuraisingham, B. "The Use of Conceptual Structures for Handling the Inference Problem," *Proceedings of the Fifth IFIP WG 11.3 Working Conference on Database Security*, November 1991.

Appendix A
Mock Classification Guideline

- An organization's name alone is not classified.
- An organization's specialty is not classified unless stated otherwise.
- The *relationship* between organization and specialty is confidential if its specialty is one of the following:
    - Russian Language
    - Metalinguistics
    - Optical Fiber Transmission
    - Civil Engineering

If the specialty falls within these classified areas, the following unclassified specialties are to be used when referencing that organization to an unclassified user:

Classified Specialty	Unclassified Specialty Description
Russian Language Metalinguistics	Language
Optical Fiber Transmission Civil Engineering	Engineering

- To provide consistency for the unclassified user, any record which relates organization and specialty at the confidential level must be polyinstantiated at the unclassified level. The record at the unclassified level will reflect the organization's unclassified specialty.