Attribution in the Future Internet: The Second Summer of the Sisterhood

Matt Bishop[1], Mina Doroud[1], Carrie Gates[2] and Jeffrey Hunker[3]
[1]Dept. of Computer Science, University of California at Davis, USA
[2]CA Labs, New York, USA
[3]Jeffrey Hunker Associates, Pittsburgh, USA
bishop@cs.ucdavis.edu
sdoroud@ucdavis.edu
carrie.gates@ca.com
hunker@jeffreyhunker.com

Abstract: Attribution is the binding of data to an entity. An attribution framework is an infrastructure for managing attributes and their values. It consists of four components: a set of entities (actors) having an interest in attribution with respect to a transaction; a set of data to be attributed; the level of assurance with which values of attributes can be determined, and with which they can be associated with an entity; and a policy negotiation engine that actors use to negotiate an acceptable set of attributes and levels of assurance for their values in order to conduct a transaction (the "policy"). The actors include the sender and recipient, the sender's and recipient's organizations, ISPs, backbones, and political entities. This paper assumes that such a general attribution framework has been implemented. It examines the implications of such a framework upon the Internet, and upon transactions (specifically, the sending and receiving of packets) among actors. The embedding of attribution requirements in policies controlling communications between parties raises the question of who can communicate with whom. Specifically, how does the use and enforcement of policies based upon attributes affect users of the Internet? We examine this question in two contexts: that of the societal revolution known as "Arab Spring", and that of elections in the United States. We present requirements and the attributes that must be supplied to meet those requirements. We then examine some of the implications of supplying the attributes from the point of view of servers, clients, and intermediaries (such a ISPs and governments). We conclude with a discussion of when attribution is desirable, and when the inability to attribute actions is desirable.

Keywords: attribution, attribution framework, policy, enforcement, security

1. Introduction

Attribution is the binding of data to an entity. It usually arises in the context of identity. For example, much discussion has focused on attribution as a tool for identifying attackers (Burch and Cheswick, 2000; Pyun and Reeves, 2007). Other uses of attribution abound. In a distributed system, the location of nodes controlling a particular resource is an attribute of the resource; in an ordinary computer system, the role (job function) of each user is an attribute of that user. Indeed, the login name is an attribute of the entity that name represents.

The management of attributes depends on the nature of the attributes and the scope of their effect. On a Linux system, the attribute of "login name" with value "bishop" may be associated with the first author. On a file server, that same attribute-value pair may be associated with the user Michael Bishop. The scope of each pair is limited to the system on which the pair is defined, so the inconsistency is irrelevant; there is no conflict. But if the Linux system is trusted to authenticate users for the file server, then the inconsistency of attribute values creates a conflict, and the first author will have access to Michael Bishop's files.

The management of attributes requires an *attribution infrastructure* with four components. A set of *actors* specifies those entities with an interest in attribution for a particular transaction or set of transactions. Each actor has an associated policy describing which attributes it requires, and which attributes it will provide, to interact with other actors. An *attribution vector* lists the attributes for which values are desired, or a set of (attribute, value) pairs. Associated with each attribute value is a *level of (attribute) assurance* that describes how certain the reported value of the associated attribute is. Finally, the actors use a *policy negotiation system* to negotiate an acceptable set of attributes, values, and levels of assurance, or to conclude that what their policy allows is not acceptable to one or more parties. The details of this model are discussed in Bishop, Gates, and Hunker (2009).

As an example, consider the Linux system mentioned above. The attribution framework is implemented using a cryptographic hash function to bind the value "bishop" to the (external) entity using

the computer. This binding occurs at login time, because the password that the entity types is hashed, and the hash compared to that associated with the login name “bishop”. If correct, the system performs the binding, and a kernel table maintains the binding. (More precisely, the binding is done using an integer that is associated with the login name; we omit the details for simplicity.) Here the actors are the external entity and the system. The attribute is “login name”. The attribute value after assignment is “bishop”. The level of assurance is that associated with the correct entity using the password. The policy associated with the entity is that the entity will supply the attribute-value pair “login name”-“bishop”, and the policy associated with the system is that it requires the provision of the value associated with the entity’s attribute “login name”, plus assurance that the value supplied is indeed associated with the entity. The entity supplying the password associated with the login name provides an acceptable level of assurance for this. (In some cases, the system may require two-factor authentication if the privilege level of the user is significantly high The policy negotiation system is constant; the system requires the correct attribute-value pair, or access is denied.

To illustrate the framework further, add to the above example the “trusted hosts” mechanism enabled by *rlogin* or *ssh* (Barrett ad Silverman, 2001). Recall that these two mechanisms have the system check the remote host from which the user is trying to log in. In this case, there are three actors: the external entity (Matt Bishop), the remote host (call it “nobhill”), and the Linux system being accessed. The attributes of the Linux system and of Matt Bishop are the same, but the system “nobhill” has associated with it the attribute “trusted by the Linux system” and the value “true”. The Linux system requires two attribute-value pairs, the first being “login name”-“bishop” and the second being “trusted by the Linux system”-“true”. The policy negotiation system now requires the second attribute-value pair be provided with some level of assurance (for *rlogin*, the assurance is obtained by looking the IP address up in a table of trusted hosts; for *ssh*, the assurance is based on cryptography). If the level is sufficient, the value of the attribute “login name” has sufficient assurance that the system accepts it without further checking. Again, the policy negotiation system is constant.

This paper assumes that a general attribution framework as described above has been implemented for the Internet. It, and the supporting infrastructure is built using existing Internet infrastructure, augmented as needed for the attribution framework and the integrity of both the framework and the information (attributes and values) that it contains. The attribution framework uses a distributed database similar to the Domain Name System. We also assume that the framework is correct; that is, when asked for an attribute-value pair for a particular entity, it supplies the information as required by the policy of that entity, and the information is conveyed correctly to the requester.

This paper examines the implications of this framework upon the use of the Internet, and upon transactions (specifically, the sending and receiving of packets) among actors. The embedding of attribution requirements in policies controlling communications between parties raises the question of who can communicate with whom. We are interested in situations where the attribution framework supports detection of violations of security policies. These may constitute attacks; they may result from mistakes; they may simply be due to carelessness or non-malicious factors.

We briefly review the framework structure. Next, we present two cases in which attribution is critical to the successful resolution of a specific problem. We then present some general thoughts on the role of attribution and how it affects both organizations and individuals.

2. Background on the framework

The attribution framework provides a basis for all the entities involved in a transaction to determine how to act (or to decline to act). The framework distinguishes between 4 classes of actors:

- An entity;
- The organization associated with that entity;
- The system(s) associated with that entity; and
- The government(s) associated with that entity.

Note this includes intermediate entities. So, if a message is sent from an entity in the United States to an entity in France via a hub in the Netherlands, the actors are the sender, all network providers in the U.S., the Netherlands, and France that the message transits, and the receiver; all the systems involved; all the organizations that run those systems and networks; and the governments of the U.S., the Netherlands, and France, including any political subdivisions (such as the state in the U.S. and

the region in France). The governments apply their policies through laws; other entities apply them more directly, for example by policy-based routing or access control (as in the Linux example above).

Finally, the framework must support many types of attribution. The type sometimes is tied to the level of assurance. Consider entities interacting, each with a policy. They may require:

- *Perfect attribution*, in which the entities know each others' relevant attribute-value;
- *Perfect selective attribution*, in which one entity wants certain attribute-value pairs known to some entities but not to others;
- *Perfect non-attribution*, in which the entities do not want any other entity to know the attribute-value pairs;
- *Entity non-attribution*, in which an entity wants attribute-value pairs known but not that they are bound to the entity;
- *False attribution*, in which an entity will determine an attribute-value pair consistently—but the value will be wrong;
- *Randomized false attribution*, which is false attribution but the values determined are inconsistent;
- *Imprecise attribution*, in which the value of an attribute can eventually be determined to the level of assurance needed, but doing so takes so long the attribution is useless or determining the attribution costs more than the value of knowing the attribution; and
- *Unconcern*, in which the entities do not care about attribution.

In all cases, the level of assurance of the attribute-value pair must meet, or fail to meet, the entity's requirements.

3. Use cases

Each case begins with the goal—what is the problem? This drives the requirements, which in turn define the attributes and associated levels of assurance necessary to meet those requirements. Multiple actors are involved—-those who are dealing with the problem, and the intermediaries who pass the packets along, possibly augmenting them or taking other actions as needed. These may have differing, possibly mutually contradictory, requirements.

We do not specify implementation details. The programs and protocols used are incidental to this analysis. We note that they may require more attributes of the parties in order to succeed, but those attributes are products of the protocols and implementations and not of the requirements of a solution to the problem—and this section focuses on those solutions. Nor do we worry about the policy negotiation protocols; we simply note when one must exist, and any relevant properties.

3.1 Arab spring

The Internet enables communication across geographic and political boundaries. This provides a natural way for people in one part of the world to communicate with others. Indeed, the ubiquity of Internet service providers such as Google, and of social networking services such as Facebook, Twitter, and YouTube, mean that videos and text can "go viral" throughout the world.

Social activists have learned to exploit this availability very effectively. The "Arab Spring", a term for the uprisings in the Middle East, is a good example of this. A March, 2011 survey (Salem and Mourtada, 2011) showed that almost 9 out of 10 Egyptians and Tunisians used Facebook and other social networking sites to organize and spread awareness of the protests in those countries. As one activist put it (Kiss and Rosa-Garcia, 2011), they used Facebook to schedule the protest, Twitter to coordinate it, and YouTube to tell the world. The role that the social networks played was critical not only within the countries, but also in communicating events with the rest of the world.

During the Egyptian Arab Spring uprisings, the government attempted to block the peoples' access to the Internet. In response, Google and Twitter provided a "telephone-to-tweet" service. Google established 3 telephone numbers that people could call and record a message. Google then posted it to Twitter. Associated with each message was a hash tag indicating the geographic origin of the message, when that could be determined; otherwise, the message was posted without a hash tag.

Because the governments of these countries controlled the infrastructure, and attempted to crack down on the protests, the intermediaries are critical. The end actors (individuals and social media sites) have one set of requirements. The managers of the intermediate hosts and networks have another.

The requirements for the end actors were:

Anonymity: the tweets, and the recordings, must be anonymous.

Accuracy of origin: A hash tag indicating geographic origin, if present, must be accurate.

The requirements for the intermediate actors (governments or networks under the jurisdiction of the government) were:

Identity: the identity of the individual who sent the message.

Consider the transactions involved in the use of these social networks. First, the user must register with the social media services. Facebook requires that the user supply a real name and date of birth. The level of assurance is minimal; essentially, mere assertion provides sufficient assurance unless Facebook questions it. Facebook also checks users' IP addresses. If the user is using a very different IP address, Facebook will request additional assurance evidence (such as correctly identifying a friend's photo). Twitter and YouTube, on the other hand, simply require a name that the user will post under; unlike Facebook, the user need not identify herself at all, and indeed most users of those services use pseudonyms. So, the initial transaction requires the user to provide the attribute "name" and "date of birth", but the values of both typically have a low level of assurance. Similarly, the transaction requires that the server provide its identity to a level of assurance high enough to convince the user that it is the genuine social media site.

Associated with each message are the "from account" attribute indicating the account posting the message, and the "post to" attribute indicating where the message is to be posted. The user supplies these values; the server requires the second and may require the first. The assurance level for these attributes are both typically very low, because errors can be corrected, or the errors are deemed "harmless" (of course, this may not be how the poster feels).

The "telephone-to-tweet" service had different requirements. The only attribute the server requested was the country from which the message originated. In most cases, the telephone system would supply this information to an acceptable level of assurance. If the information were not available, no associated hash tag would be generated. In other words, the server would request the "country of origin" attribute, but if the client could not supply it, the policy negotiation mechanism rolled back to accepting the (unattributed) message.

The "telephone-to-tweet" service is an example of perfect non-attribution: none of Google, Twitter, nor any listener is to be able to identify the speaker from any metadata.

Now consider intermediaries. The intermediary ISP is an entity over which the government exercises *de facto* control. One set of requirements involve the ISP interacting with the server (actually, other entities that communicate with the server; in this case, those other entities have no requirements), and the other set involves the ISP interacting with the user client. The former having no special requirements, we examine the latter.

The goal of the ISP is to prevent communication with a set of servers that the government deems undesirable. The origin of the message is not relevant; where it is going, is. Thus, the requirement for this intermediate actor is:

Identity: the (Internet address of the) destination of the message

The attribute therefore is "destination address". The level of assurance required by the ISP is high, to ensure proper delivery of the message; as a side effect, the ISP can block the desired IP addresses.

The need for this level of attribution is instructive. One may circumvent this control by using a proxy. In this case, the messages transit the intermediate network with the destination being the proxy, so the intermediate network allows the messages through. The use of VPNs, encrypted communications channels, and mix routers such as Tor (Dingledine, 2011) encipher the information about the sender's identity and the ultimate destination. Thus, the intermediary cannot accurately attribute the origin of the message to an individual, or the destination identity to a site. Thus, it must block all such messages, allow all such messages through, or require use of another mechanism to identify the sender and destination.

3.2 Elections

Elections are the foundation of democratic and republican societies. Recently, many jurisdictions began exploring people voting over the Internet. We look at the use of the attribution framework in the context of elections within the United States. We focus on a key subset of the requirements:

- *Secrecy of the ballot*: a third party cannot associate a ballot with a voter.
- *Anonymity of the ballot*: the voter cannot prove that she cast a particular ballot to a third party.
- *Accuracy of the count*: the votes are counted correctly.
- *Authorization of the voter*: the voter submitting the ballot is authorized to do so.
- *Integrity of transmission*: the ballot is received and counted as cast.

A number of other requirements exist, but our exposition focuses on these.

United States residents live under at least 3 political jurisdictions (federal, state, and county) and in most cases many more (city, school board, and so forth). Each of the 50 states is responsible for holding its own elections; most delegate this responsibility to the counties, with the state having the ultimate authority to certify the results. For purposes of elections, the counties are divided into precincts, each of which matches one set of overlapping political jurisdictions. Each precinct conforms to a single set of candidates, so all voters in a precinct will vote using the same ballot. But two precincts may have different sets candidates, and thus two different *ballot types*. In Yolo County, California, for example, one election required over 100 different ballot types. In jurisdictions where more than 5% of the voters have a language other than English as their primary tongue, the ballots must also be printed in that language. For example, one election in San Francisco, California, required some ballot types to be printed in 7 languages. The location where the county counts the results is called *Election Central*.

This multiplicity of ballot types and languages means that three transactions are involved in elections. First, the voter registers to vote. Second, the voter receives the correct ballot from the ballot generator (most jurisdictions have strict formatting requirements, and require ballots to come from the county). Third, the voter marks his ballot and then transmits the marked ballot to Election Central, where the votes are counted.

Consider each transaction separately. The first transaction speaks to authorization of the voter. By the voter supplying attributes that uniquely identify her, the registration authorities can determine which ballot type she should receive, and set the "authorization to vote" attribute to a value that will enable her to obtain the correct ballot. (In practice, this is usually an address, because political jurisdictions in the United States are geographic.) The voter requires that the registration authority have the attribute "authorized to register voters" with value "true". The registration authority specifies the unique attributes it requires and the assurance evidence that the value of that attribute (in practice, the address)is correct (in practice, evidence that the prospective voter lives there). It in turn must possess the attribute "authorized to register voters" with value "true". The intermediate actors must allow this information to transit their systems and networks. Thus, the attributes that they require also must be present.

The second transaction occurs when the voter acquires the ballot. The voter first verifies that the ballot generation system is authorized to generate the ballots by checking the value of the attribute "authorized to generate ballots". The voter presents the value of the "authorization to vote" attribute, which the ballot server validates as having an acceptable level of assurance. That authorization is used to determine the ballot type that the voter requires. A ballot of that type, with the attribute "issued to authorized voter" and value a nonce is generated and sent to the user. Note that no attribute ties the ballot to the actual voter, which meets the requirement of secrecy of the ballot. But an obvious

attack is for the voter to mark the ballot and copy it. To prevent this, attributes “access control” with value “originator-control”, “originator” with value “election-official”, “write access” with value “voter-issued-to”, and “read access” with value “election central, voter issued to” indicate that the ballot is not to be copied or marked by any entity other than the original voter.

The third transaction is the casting of the vote. The important artifact here is the ballot; it must come from the ballot server, and be voted by an authorized voter. The voter first contacts Election Central, and checks that the attribute “Election Central” is “true”. The voter then transmits her ballot to Election Central, which checks that the attribute “issued to authorized voter” has a nonce that is unused so far. It then processes the ballot and tallies the votes.

Thus, the attributes of interest here are:

- *Voter*. provides “identity”, “authorization to vote”; requires “authorized to register voters” from registration authority, “authorized to generate ballots” from the ballot generator, “Election Central” from Election Central
- *Registration authority*: provides “authorized to register voters”, “obtain ballot”; requires “identity”, “authorization to vote” from voter
- *Ballot generator*: provides “authorization to generate ballots” to voter; provides “issued to authorized voter”, “access control”, “originator”, “read access”, “write access” to ballot; requires “authorization to vote”
- *Election Central*: provides “Election Central”; requires “issued to authorized voter” from ballot
- *Ballot*: “issued to authorized voter”, “access control”, “originator”, “read access”, “write access”

The intermediate services and actors must simply pass messages on unchanged and without delay, regardless of the values of any attributes. (The lack of delay is required to ensure ballots arrive in time to be counted.)

4. Discussion

We have discussed in detail two different case studies where attribution, or non-attribution, would be desirable, and the effects of such attribution. What needs to be considered is the desirability of attribution by each of the different actors: end users, organizations, intermediary nodes and governments.

The governments have strong desires for attribution in the case studies presented. In the case of elections, the government as an entity desires a fair and lawful election. We note that this particular example assumes benevolence on the part of the government. In those cases where it does not want a fair election, that desire results from specific individuals within the government, not from “government” as a separate entity. Thus the government, in the case of elections, has a desire for proper attribution that matches the laws of that particular government (e.g., a citizen can only vote once; it must not be possible to match a vote to a voter, and so forth). Perhaps more importantly, the government desires that the electorate perceives the election to be fair, and the use of appropriate attribution technology can assist in creating such a perception. In the case of Arab Spring, the government also desired attribution, specifically of the individuals responsible for organizing the revolutionary activities. This is similar to the original push for attribution on the Internet, which was done by the government with regard to the ultimate identification of the actor(s) responsible for launching distributed denial-of-service attacks. In this case, the government is not interested in attribution to identify itself to others, but rather attribution to identify other actors. In general, the government as an entity is interested in the attribution of individuals performing particular actions (either against the government, or on behalf of the government). However, in some circumstances, a government might only need to know what (other) government to attribute an action to. For example, a government of a country under cyberattack might only care that the attack is sanctioned by a particular nation state, and not necessarily about knowing the individuals who performed the attack.

In the case of intermediary nodes, such as ISPs, attribution is likely not desired. In the same vein as ISPs not wanting to have the responsibility for recording traffic traversing their networks and maintaining that information for some amount of time as determined by government regulation, ISPs will also not likely wish to have the responsibility for providing attribution information. We note that such desire might change in the face of increased regulatory pressure or if sufficient profit is available; however, this seems unlikely given the current environment.

Organizations may wish to provide attribution to end users for whom they are providing services. For example, the use of an attribution framework will likely considerably reduce instances of successful phishing attacks. Thus, organizations such as financial institutions will likely be interested in providing such services. This could shift liability from financial institutions back to end users. Thus attribution might provide a benefit to those organizations. Similarly, organizations might desire attribution during negotiations with other organizations (such as when one organization desires to acquire another). However, we note that in this case the attribution is actually at the end user level, because the organizations will want to know the individual who is performing the actions in order to determine that this person has the appropriate authorization to perform the associated actions on behalf of their organization. On the opposite end of the spectrum, due to liability issues, organizations might not want to know the identity of the end user, and thus not desire that level of attribution. For example, during financial transactions, an organization (such as a store) might only want to know that the credit card information is valid, and not the identity of the user providing the credit card information. This ensures that liability remains with the end user and does not shift to the organization.

The case of the end user is much more complex. In an election, the end user may wish a vote attributed to him (to sell his vote, or to prove that he voted for a particular candidate or party). But it is against the government's interest (and may, in fact, be illegal) for such attribution to be possible. In the case of Arab Spring, the end users explicitly wanted to remain anonymous for fear of reprisals against themselves or their families. Thus the desires of an end user might be complex, and go against their organization's or government's desires. Often such conflicts are philosophical in nature. There isno "right" answer. In general, especially outside the political realm, an end user might want to provide attribution information to receive credit for some action (such as writing a book). End users may also desire anonymity in order to preserve privacy. This often elicits a claim that "if you have nothing to hide, you do not need privacy. In fact, many users do not want even benign activity tracked by any third parties (e.g., web browsing habits or search terms). Privacy is an extremely complex issue, and the reader is directed to the numerous papers written on the subject (for example, see Warren and Brandeis (1890) and Dwork (2008)).

5. Conclusion

The widespread deployment of an attribution framework, or set of frameworks, provides both benefit and risk. Its use simplifies some actions and relationships; it also makes more difficult the protection of privacy under some circumstances. The point of this paper is to argue that the implications of such a framework, even if perfectly implemented and supported, are unclear, and need to be considered as such frameworks develop and are deployed. Especially critical are the types of attribution (or non-attribution) the frameworks will support, and the level of assurance associated with the values of those attributes and the binding of the attributes to the entity. We need to consider the broad implications for social policy, and indeed for societies themselves.

References

Bishop, M., Gates, C. and Hunker, J. (2009) "The Sisterhood of the Traveling Packets", In *Proceedings of the New Security Paradigms Workshop,* PP 59–70.

Burch, H. and Cheswick, B. (2000) "Tracing Anonymous Packets to Their Approximate Source", In *LISA '00: Proceedings of the 14th USENIX Conference on System Administration*, Berkeley, CA, USA. USENIX Association, PP 319–328.

Dingledine, R. (2011) "Tor and Circumvention: Lessons Learned", *Advances in Cryptology—CRYPTO 2011,* Lecture Notes in Computer Science, 2011, Vol 6841/2011, Publisher: Springer Berlin / Heidelberg, PP 485-486.

Dwork, C. (2008) "Differential Privacy: A Survey of Results", In *Proceedings of the 5th International Conference on Theory and Applications of Models of Computation*, Lecture Notes in Computer Science, 2008, Vol 4978/2008, Publisher: Springer Berlin/Heidelberg PP 1–19.

Kiss, H., Rosa-Garcia, A. (2011). "Why do Facebook and Twitter facilitate revolutions more than TV and radio?" MPRA Paper No 33496.

Pyun, Y. and Reeves, D. (2007) "Strategic Deployment of Network Monitors for Attack Attribution" In *BROADNETS '07: Proceedings of the Fourth International Conference on Broadband Communications, Networks and Systems*, PP 525-534.

Salem, F. and Mourtada, R. (2011) "Arab Social Media Report: Civil Movements: The Impact of Facebook and Twitter", *Dubai School of Government*, Vol. 1, No. 2, May.

Warren, S. and Brandeis, L. (1890) "The Right to Privacy", *Harvard Law Review*, Vol. 4, No. 5 PP 193–220.