Honeypots: Catching the Insider Threat

Lance Spitzner
Honeypot Technologies Inc
lance@honeypots.com

Abstract

In the past several years there has been extensive research into honeypot technologies, primarily for detection and information gathering against external threats. However, little research has been done for one of the most dangerous threats, the advance insider, the trusted individual who knows your internal organization. These individuals are not after your systems, they are after your information. This presentation discusses how honeypot technologies can be used to detect, identify, and gather information on these specific threats.

1. Introduction

Honeypots are a powerful, new technology with incredible potential. They can do everything from detecting new attacks never seen in the wild before, to tracking automated credit card fraud and identity theft. In the past several years we have seen the technology rapidly develop, with new concepts such as honeypot farms, commercial and open source solutions, and documented findings released. However, a great deal of research has been focused on identifying, capturing, and researching external threats. While malicious and dangerous, these attacks are often random with attackers more interested in how many systems they can break into then which systems they break into. To date, limited research has been done on how honeypots can apply to a far more dangerous and devastating threat, the advanced insider. This trusted individual knows your networks and organization. Often, these individuals are not after computers, but specific information. This is a risk that has proven far more dangerous, and far more difficult to mitigate.

This paper attempts to discuss how honeypots, an emerging technology, can apply to this threat. The strategy and tactics of how honeypots are used against insider threats, especially advanced insider threats, are vastly different then those of an external threat. We will address some of the new ways that they can apply to the insider. Many of the ideas discussed here are the result of the ARDA Cyber Indications and Warning Workshop, led by the NRRC[1] hosted at MITRE. This paper does not cover proven solutions. Instead it introduces novel applications of a developing technology. It is hoped this paper promotes discussion and research into new fields.

2. Honeypots

A honeypot is a unique security resource. It is something you want the bad guys to interact with. The definition of a honeypot as, defined by the honeypot maillist[2], a public forum of over 5,000 security professionals, is:

A honeypot is an information system resource whose value lies in unauthorized or illicit use of that resource.

What this definition means is honeypots derive their value from threats using them. If the enemy does not interact or use the honeypot, then it has little value. This is very different from most security mechanisms. For example, the last thing you want an attacker to do is interact with your firewall, IDS sensor, or PKI certificate authority. Honeypots are very different, and it is this difference that makes them such a powerful tool in your arsenal.

First, honeypots do not solve a specific problem. Instead, they are a highly flexible tool that has many applications to security. They can be used everything from slowing down or stopping automated attacks, capturing new exploits to gathering intelligence on emerging threats or early warning and prediction. Second, honeypots come in many different shapes and sizes. They can be everything from a Windows program that emulates common services, such as the Windows honeypot KFSensor[3], to entire networks of real computers to be attacked, such as Honeynets. In fact, as we will discuss later, honeypots don't even have to be a computer, instead they can be a credit card number, Excel spread

---

[1] The Northeast Regional Research Center (NRRC) is sponsored by the Advanced Research and Development Activity in Information Technology (ARDA), a U.S. government entity which sponsors and promotes research of import to the Intelligence Community which includes but is not limited to the CIA, DIA, NSA, NIMA, and NRO.

[2] http://www.securityfocus.com/popups/forums/honeypots/faq.shtml

[3] http://www.keyfocus.net/kfsensor/download/

sheet, or login and password (commonly called honeytokens).

All honeypots share the concept; they are a resource or entity with no production value. By definition, your honeypot should not see any activity. Anything or anyone interacting with the honeypot is an anomaly, it should not be happening. Most likely, it implies you have unauthorized or malicious activity. For example, a honeypot could be nothing more then a webserver deployed in your DMZ network. The webserver is not used for production purposes, its does not even have an entry in DNS, its merely physically located with other webservers. Any interaction with the honeypot is assumed unauthorized and most likely malicious. If the webserver honeypot is probed by external systems from the Internet, you have identified a probe or attack, most likely the same one your other production webservers are facing. If your honeypot is probed by one of the production webservers on the DMZ, that can imply that the production webserver has been compromised by an attacker, and is now being used as a launching pad to compromise other systems on the DMZ. Its is because of this very simple concept that honeypots share immense advantages, including

- Small Data Sets: Honeypots only collect data when someone or something is interacting with them. Organizations that may log thousands of alerts a day with traditional technologies will only log a hundred alerts with honeypots. This makes the data honeypots collect much higher value, easier to manage and simpler to analyze.
- Reduced False Positives: One of the greatest challenges with most detection technologies is the generation of false positives or false alerts. It's similar to the story of the 'boy who cried wolf'. The larger the probability that a security technology produces a false positive the less likely the technology will be deployed. Honeypots dramatically reduce false positives. Any activity with honeypots is by definition unauthorized, making it extremely efficient at detecting attacks.
- Catching False Negatives: Another challenge of traditional technologies is failing to detect unknown attacks. This is a critical difference between honeypots and traditional computer security technologies which rely upon known signatures or upon statistical detection. Signature-based security technologies by definition imply that "someone is going to get hurt" before the new attack is discovered and a signature is distributed. Statistical detection also suffers from probabilistic failures – there is some non-zero probability that a new kind of attack is going to go undetected. Honeypots on the other hand can easily identify and capture new attacks against them. Any activity with the honeypot is an anomaly, making new or unseen attacks easily stand out.
- Encryption: It does not matter if an attack or malicious activity is encrypted, the honeypot will capture the activity. As more and more organizations adopt encryption within their environments (such as SSH, IPsec, and SSL) this becomes a major issue. Honeypots can do this because the encrypted probes and attacks interact with the honeypot as an end point, where the activity is decrypted by the honeypot.
- IPv6: Honeypots work in any IP environment, regardless of the IP protocol, including IPv6. IPv6 is the new IP standard that many organizations, such as the Department of Defense, and many countries, such as Japan, are actively adopting. Many current technologies, such as firewalls or IDS sensors, cannot handle IPv6.
- Highly Flexible: Honeypots are extremely adaptable, with the ability to be used in a variety of environments, everything from a Social Security Number embedded into a database, to an entire network of computers designed to be broken into.
- Minimal Resources: Honeypots require minimal resources, even on the largest of networks. A simple, aging Pentium computer can monitor literally millions of IP addresses.

For insider threats, we will be leveraging these advantages. However, like all technologies, honeypots share several disadvantages. We have to understand these disadvantages to catch our insider.

- Risk: Honeypots are a security resource you want the bad guys to interact with, there is a risk that an attacker could use a honeypot to attack or harm other non-honeypot systems. This risk varies with the type of honeypot used. For example, simple honeypots such as KFSensor have very little risk. Honeynets, a more complex solution, have a great deal of risk.
- Limited Field of View: Honeypots only see or capture that which interacts with them. They are not a passive device that captures activity to all other systems. Instead, they only have value when directly interacted with. In many ways honeypots are like a microscope. They have a limited field of view, but a field of view that gives them great detail of information.

There are two key types of honeypots that play a role in indicating and capturing an advanced insider threat, *Honeynets* and *honeytokens*. We will now take a moment and discuss these two specific types of honeypots.

2.1 Honeynets

Honeynets are one of the most advanced and complex honeypots, their primary purposes is to capture extensive information on threats, both internal and external. Honeynets are complex in that they are entire networks of computers to be attacked. Nothing is emulated. The systems and applications within a Honeynet can be the same systems found in your organization. Within these systems you can place additional information, such as files, records in databases, log entries, any information you want the attacker to interact with. Honeynets have this flexibility because they are not a standardized solution, instead a Honeynet is a specialized architecture that creates a fishbowl, you can then place any targets systems you want within this fishbowl. Just like a fishbowl, you can create your own virtual world; however instead of adding coral and sand, you add Solaris database servers or Cisco routers. Just like a fishbowl, you can watch everything that is going on, however with a Honeynet; the attacker never realizes you are watching them (similar to a one way mirror).

In figure 1 we see a diagram of a Honeynet. The critical element is the Honeywall gateway, a layer two bridging device that controls and captures all of the attacker's inbound and outbound activity. Since the gateway is a layer two bridging device, it has no MAC address, no routing of packets, nor any TTL decrement, making it nearly impossible for an attacker to detect. Any packet sent to a victim system within the Honeynet must flow through the gateway, ensure you can both capture and control their activity. Honeynets have repeatedly demonstrated their intelligence gathering capabilities. Two examples are the paper *Know Your Enemy: Credit Card Fraud[1]* and the Scan of the Month challenge 28[2]. In the credit card paper, Honeynets were used to capture information on automated credit card fraud and identity theft, to include not only how it was done, but who was involved. In the Scan of the Month challenge (a monthly challenge sponsored by the Honeynet Project), we capture the activity of advanced Italian hackers tunneling IPv6 traffic through IPv4 for convert communications. These are the same individuals that were later prosecuted by Italian authorities for breaking into NASA. For further technical details of a Honeynet, please refer to the paper *Know Your Enemy: Honeynets[3]*.

2.2 Honeytokens

Honeytokens represent one of the newest and most interesting implementations of a honeypot. First, they are not a computer; instead they are a digital entity, such as an Excel spreadsheet. Even though they are not a computer, they share the same definition and concept of honeypots, no one should be interacting with them. Any interaction with a honeytoken implies unauthorized or malicious activity. Second, they are extremely flexible, they have the ability to adapt to any environment. The reason for this is simple, a honeytoken can pretty much be anything you want. Examples can include a Word document, login and password, database record, or social security number. For example, lets say we are a large hospital, responsible for maintaining the privacy of millions of patient records. One of the requirements is identifying when a member of hospital staff attempts to exceed their authorization and access patient data they do not have a need to access. A bogus medical record called "John F. Kennedy" is created and loaded into the database. This medical record has no true value because there is no real patient with that name. Instead, the record is a honeytoken, an entity that has no authorized use. If any employee is looking for interesting patient data, this record will definitely stand out. If the employee attempts to access this record, you have an indication of an employee violating patient privacy. It is as simple as that, no fancy algorithms, no signatures to update, no rules to configure. You load the records, monitor it, and if someone accesses it they most likely have violated the system's usage policy.. Honeytokens are extremely flexible, there is no right or wrong way to use them. Due to their flexibility, you can customize them to easily integrate into your environment. To learn more about honeytokens, refer to the paper *Honeytokens: The Other Honeypot[4]*.

3. The Insider

Before we can discuss how honeypots, specifically Honeynets and honeytokens, can catch the insider threat, we need to first define what our goal is, and the threat we face. Our goal is to detect, identify, and confirm insider threats. This means leveraging honeypots to not only indicate that we have an insider, but also confirm their actions, and potentially learn their motives and resources. What makes our goal difficult is the threat we face, the sophisticated insider. What we mean by this is someone who is technically skilled, highly motivated, and has access to extensive resources. For example, this threat may be an employee working for a large corporation, but in reality they are employed by a competitor to engage in corporate espionage. A second example is highly skilled, disgruntled employee motivated to cause a great deal of

damage before they are fired. A third example could be a spy working for a foreign country. Regardless of who the insider is, we are dealing with a highly dangerous threat, one that is extremely difficult to detect. They have access to critical information; they know the structure of the organization. They are most likely after information, not systems. As a result, there may be few attacks and their access to information may even be authorized. It is what they do with that information that comprises the threat. It is our goal to detect and capture the activity of this threat.

For the purposes of this paper, we will take the lessons learned from the ARDA Cyber Indications and Warning workshop. In this workshop, we focused on past spies in the Intelligence community. Examples of such spies include Aldrich Ames, Robert Hansen, and Anna Montes. These individuals were all highly trusted individuals with extensive and critical knowledge to their respective organizations. However, as insiders they were able to cause extreme harm to their organizations, and over long periods of time without being detected.

3.1 The Strategy

Traditionally, honeypots have been used to detect or capture the activity of outsider or perimeter threats. The purpose of these honeypots varied. Some organizations are interested in learning what threats exist and gaining intelligence on those threats, others want to detect attacks against their perimeter, while others were attempting early warning and prediction of new attack tools, exploits, or malicious code. When focusing on such a threat, the strategy for deploying honeypots is relatively simple, deploy the honeypots and the attackers will come. Honeypots, such as Honeynets, would be placed on a perimeter network, or a direct connection to the Internet, such as a DSL or cable modem. Once deployed, security administrators took the attitude 'sit back and wait'. If you build it, they will come. And come they do. An unprotected honeypot deployed on an external network can expect to see 10-30 probes a day. A vulnerable honeypot (such as a default RH 7.2 installation, or unpatched Windows XP computer) can expect to be compromised in less then seventy-two hours. What makes these numbers even more amazing is nothing is done to advertise the honeypots or entice the attackers. These honeypots are not registered in DNS, they have no entries in Google or in any search engines, no one should know about these deployed honeypots.

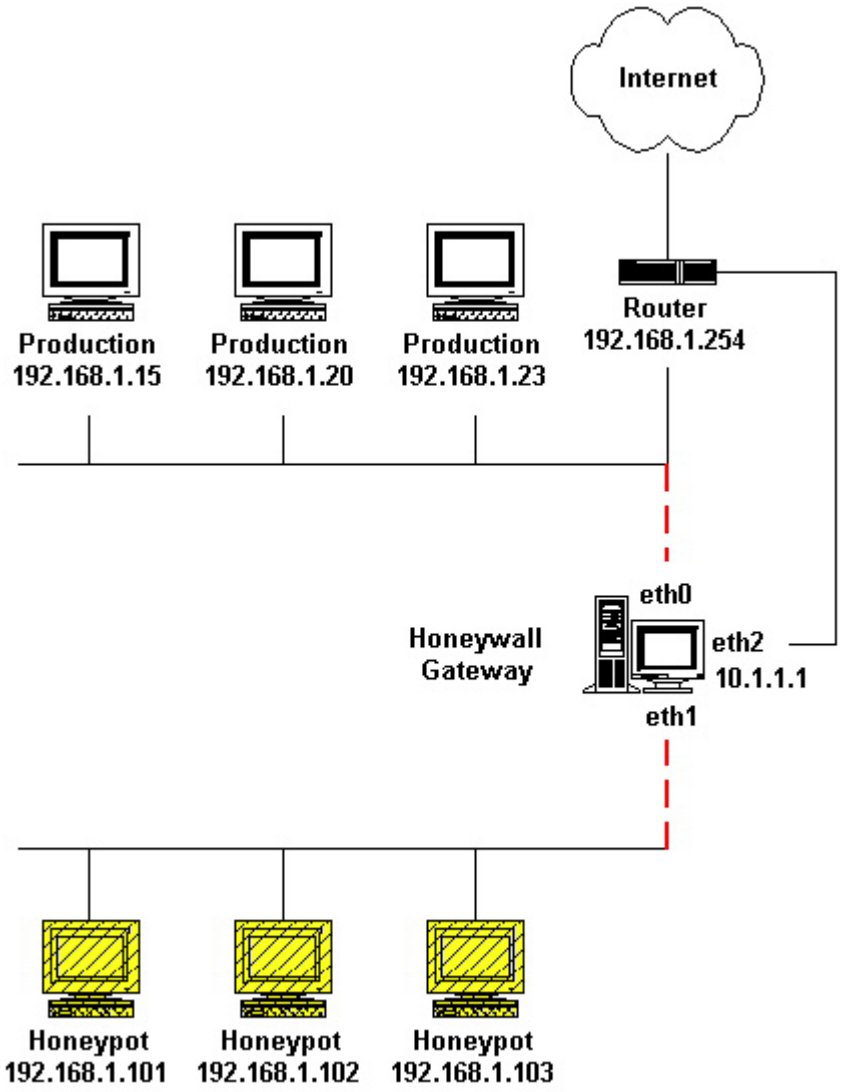

Figure 1

Diagram of a 2nd generation Honeynet. All attacker activity sent to the honeypots (systems in yellow) must go through the Honeywall gateway, a layer two bridging device that controls and captures all of the attackers' activity.

And yet, attackers find and attack these systems repeatedly on their own initiative.

Once you understand the enemy you are dealing with, this is not as amazing as it seems. An extremely large percentage of cyber threats are what we would classify as script kiddies, or automated, random attackers. These individuals targets systems of opportunity. They are not interested in what systems they compromise, but how many. Their goal is to compromise as many computers as possible. Now, their motives for this vary extensively (creating networks to be used for Distributed Denial of Service attacks, networked bots, stealing credit cards, identity theft, scouring for email address to be sold to spammers). However, they share the same goal, literally break into thousands of computers. The Honeynet Project has had honeypots controlled by attackers who own over 15,000 compromised systems.

These attackers do this by simply running automated tools they find on the Internet, or given to them by other blackhats. These automated tools do all the work for the attacker. Once launched, the tools scour the entire Internet, probing every IP address they can find. Once they find a vulnerable system, the tool compromises the box, takes over it, then continues probing. While not an elegant or subtle approach, its effective. The majority of today's attackers are not highly skilled, but they don't need to be. These automated tools do the work for them. One documented example is the Honeynet Project's Scan of the Month 13 challenge[5]. When dealing with this clientele, the strategy for deploying honeypots is simple. As we said before, you just deploy them and they will come. The attackers scan entire blocks of networks. Recent work by V. Yegneswaran and P. Barford from University of Wisconsin and J. Ulrich from SANS supports this. In their publication Internet Intrusions: Global Characteristics and Prevalence[6], they estimate 25 billion intrusion attempts per day, based on 1600 firewall logs collected over a four month period. Because of this brute force method, external threats will also attack and break into honeypots that are on the same networks. However, when attempting to detect and learn about sophisticated insiders, we will need a different strategy.

First, when dealing with insider threats, you will most likely have to move the honeypots from external networks to your internal networks, move the honeypots to where the threat is. Second, we have to address one of the disadvantages of honeypots, the fact they have a limited view, they only see what interacts with them. Simply deploying honeypots on your internal network most likely will not detect the advanced insiders. Such honeypots will detect common threats, such as automated attacks, worms, or insider threats taking a brute force approach, such as scanning internal networks for open shares. These threats represent the same clientele as most external threats, taking a target of opportunity force, sweeping entire networks or actively probing many systems. Regardless of where you deploy your honeypots, they will easily capture such activity. Georgia Tech recently released a paper on how internally deployed Honeynets successfully captured such threats, titled *The Use of Honeynets to Detect Exploited Systems Across Large Enterprise Networks[7]*

However, we have to assume with our sophisticated insider that they will not be so careless, so noisy. This threat will be far more selective, they do not want to be caught. Also, they have better knowledge of the environment, and as such can focus on specific targets. Simply deploying honeypots on your internal network will not do the trick. There are technical ways to increase the likelihood. For example, honeypots such as Honeyd create virtual honeypots that populate all of your internal, unused IP space. Instead of having one honeypot, you have thousands all over your networks. If an attacker attempts to interact with an unused IP address, the honeypot dynamically creates a virtual honeypot that interact with the attacker. This method exponentially increases the likelihood of capturing attacks. But we have to assume our insider knows what systems they are after, and what information they want to compromise. As such, our insider threat will most likely not go after unknown or unused IP addresses or systems.

As a result, we will have to modify our strategy for deploying honeypots. Instead of the threat coming to the honeypot, we have to have some way of directing the attacker to the honeypot, without them knowing it. The honeypots themselves should also be more advanced. To learn more about the attacker, the honeypots cannot simply be basic, emulated services. Instead, the honeypots should be more advanced, real systems with the same applications, data, appearance, and behavior the insider expects. Once the attacker interacts with our honeypot, we have our initial indication that we have an insider threat. Based on what the attacker does with the honeypot, we can then continue to monitor their actions and then potentially confirm if we have an insider, who that threat is, how they are operating and why. So, our strategy for insider honeypots is not one of just building the honeypot. Instead, we have to guide our insider to it, a honeypot realistic enough for the attacker to interact with.

3.2 The Tactics

Now that we have a defined strategy, the next step becomes one of implementing it. There are two problems we must solve. The first one is the redirection of an insider to a honeypot. The second problem is one of creating a realistic honeypot for the insider to interact with. Combined, these two elements can not only be used to indicate an insider threat, but confirm who the threat is,

their identity, motives, and operations. What we will now cover are some theories on how that can be done. To redirect the attacker, we have to better understand the problem. Most insiders are after specific information. In many cases, they already know what that information is, where it is, and potentially even how to access it. Our goal will be to create information that the insider will want, but information that represents an indication of insider attack. This is information that the attacker is not authorized to have, or information that is inappropriate. One of the lessons learned from the ARDA Cyber Indications and Warning workshop was that in many cases insider threats have authorization to access information, but may access information they do not have need to know. One example is of the FBI spy Hansen, who searched for his own name in the FBI active cases database. Though Hansen had authorization to access the database, searching for his own name would most likely be inappropriate. This information will then be our trigger, our honeytoken. If or when the insider accesses the information, we have our first indication of an insider. This information can then direct the insider to our more advanced honeypots, specifically Honeynets.

Our first example will address advanced insiders that are passively monitoring network activity for specific information. In many cases, an insider may use a sniffer to passively monitor and collect sensitive network activity. This approach is very safe as it is difficult to detect, yet it can give the insider tremendous amount of information. Not only can the attacker recover highly sensitive data, but who is using it and how. Also, for many organizations, the more trusted the environment, the less likely you will find advanced security precautions, such as encrypted communications. This makes it very easy for the insider to passively monitor communications, as the insider is part of that trusted environment. Honeytokens can be used to detect such activity. A honeytoken is created, one of perceived value, and inserted into network traffic. If an attacker is monitoring that network, they will most likely capture our honeytoken. As such, our honeytoken needs to have perceived value, one the insider will follow up on. Our honeytoken could be a login and password for a system perceived of high value. The insider recovers this login and password, and attempts to use it on a system. However, since it's a honeytoken, we no one is authorized to use this login/password combination. Any use of this honeytoken on any system is an indication of an insider. We can take this a step further by using different login/password combinations inserted into different networks. Then, not only can we have indication of an insider when the honeytoken is used, but we can determine where the honeytoken was sniffed by matching the different login/password combination to the different networks it was inserted into.

To direct an attacker to a honeypot, we will need to have the honeytokens point to the honeypot. In this case, we can actually login to a honeypot using the honeytoken. When the insider recovers the information from the network, not only will they recover the bogus login and password, but they will see it successfully used on a system (such as a database). What they don't know is that our database is really a honeypot. When the attacker accesses the database with our honeytoken login and password, not only do we have early indication of an insider, but by monitoring their activities on our database honeypot, we can learn more about who the insider is, their motives, etc.

A second example can be used for threats actively looking at documents, or even in emails. An insider may know what resources or individuals are of high value. We can place honeytokens in those environments. Any uses of those honeytokens are indication of an insider. For example, perhaps we are concerned about an insider accessing VP's or senior manager's emails. Inside each of these individuals' mailboxes we create a bogus email, this email is our honeytoken. No one should be reading or accessing it. The contents of the email could be as follows:

```
Date: Wed, 27 Aug 2003 13:48:47 -0400
From: "Bob Davies, Security Admin"
      <bdavies@example.com>
To: "Laura Smith, Big VP"
<lsmith@example.com>
Subject: R&D report

Oh great and powerful Vice President of
Research and Development.  You can find the
documented summary of all our research and
development for 2003 online at the new R&D
server.  Also, your new login and password
for this server is documented below.  Use
this information to download the latest,
highly classified document.  Please do not
share this information with anyone.

   Login:  honeytoken
   Passwd: h0neyt0ken
   System:
http://rd.example.net/docs/2003/rd-
2003.summary.pdf

Your humble security servant,

Bob Davies
Research and Development Administrator
```

What we have done is created an email (our honeytoken) of high perceived value. No one (even our VP Laura Smith) should be using this login or password

combination, no one should be accessing this document. If anyone uses this login anywhere, or if anyone attempts to download the rd-2003.summary.pdf document from any system, we have an indication of an insider threat. Even more, the 'secure' webserver http://rd.example.net is not really a new server, but is our new honeypot. If anyone accesses this honeypot, we have indication of an insider. By monitoring actions on this honeypot, such as where they search or the type of documents they attempt to download, we can learn more about who our threat is and confirm their actions.

A third example is one of search engines. Insiders have access to extensive amounts of information. As a trusted member of your organization, this is expected. One of the most common tools all of your trusted members will use is some type of search engine. At times these individuals will use the search engine as part of their work, to find critical data so they can accomplish their goals. But what if an insider wants to find or access information they should not be doing. Once again, we can plant honeytokens within search engines. These honeytokens are results that have perceived value to insiders, however no one has authorization to access them. Any attempt to access these honeytokens is an indication of an insider threat. For example, perhaps there is work in your organization on prototyping a new UAV (Unmanned Aerial Vehicle). An insider may be interested getting all the latest information on this prototype to share with a competitor. They may do a search on UAV prototypes on the companies internal search engine. The search itself is not an indication of an insider threat, as perhaps this individual has authorization

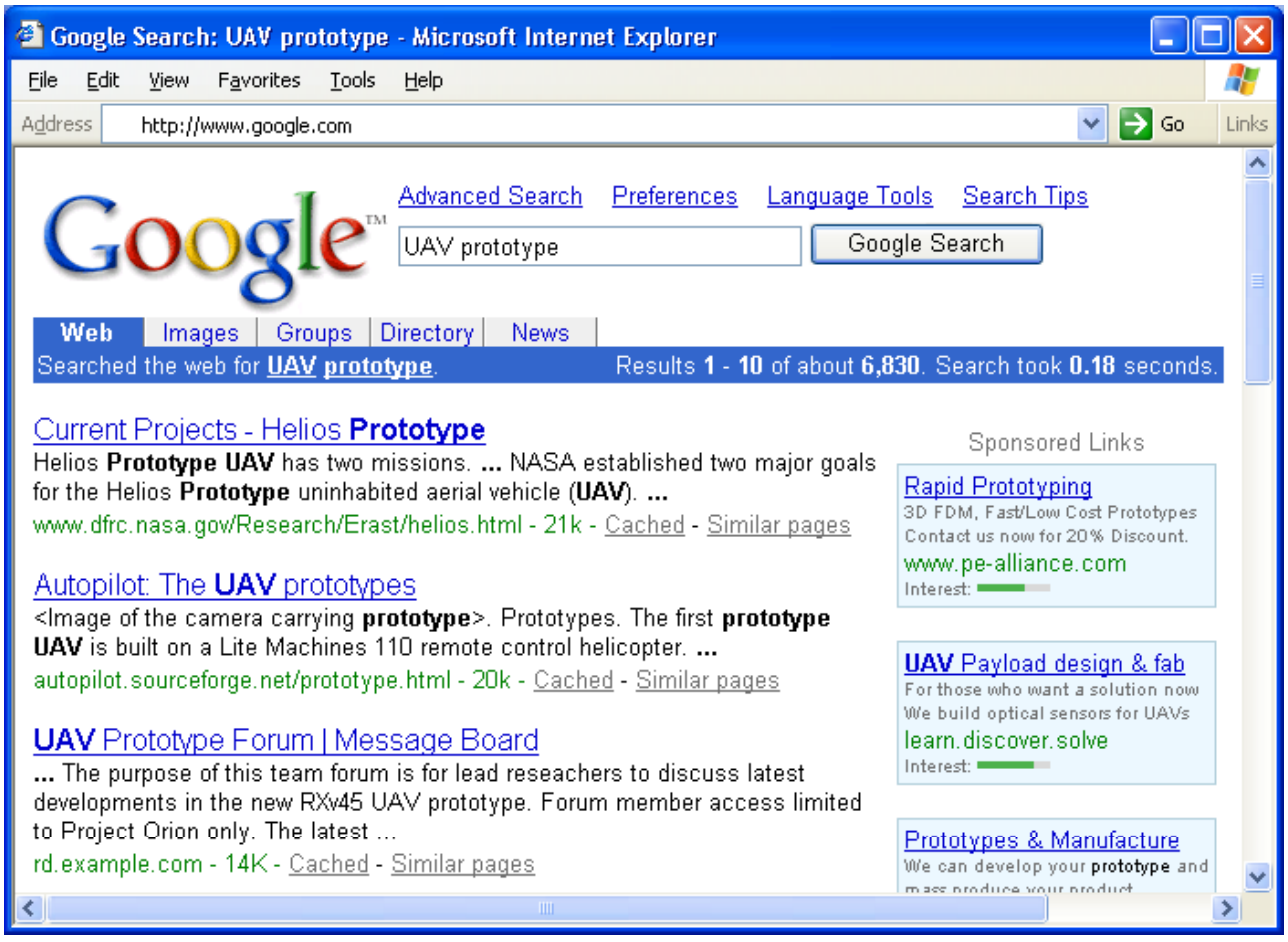

Figure 2

In this search we have embedded a honeytoken link, specifically the UAV Prototype Forum at the bottom of the screen. This forum does not exist, nor is there a Project Orion. As a result, no one has a need to access this forum. Any attempt to access this is indication of an insider threat. Notice the system it resides upon, rd.example.com, our honeypot.

to conduct such a search. What we can do is create a honeytoken link, a hyper link that the user would be attracted to, but has no authorization to use. By having them follow that link, you now do have an indication of an insider. Even more, the honeytoken link can be a link to a honeypot, once again providing bogus information to the attacker. The honeypot can then track the insider's actions with the system, helping to confirm who the attacker is, and their intent. Refer to Figure 2 for one such example of a search resulting in a honeytoken link.

Honeytokens are extremely flexible, we have presented only three examples of their use. They can easily be customized for your environment. The key is creating a honeytoken that is of interest or value to insider threats, but one they are not authorized, or do not have the need to know, to utilize. I feel that honeytokens are especially effective against threats that are interested specifically in information, such as threats in the intelligence community. Honeytokens leverage the fact that this is what the enemy is interested in.

Once we have an indication of an attacker, we will want to redirect them to a more advanced honeypot, specifically a Honeynet. Honeynets can then be used to gather more information, including confirming if the insider has malicious or unauthorized intent, who the insider is, and perhaps their motives. Honeynets have repeatedly demonstrated their ability to capture information on external attackers. We can now apply that capability to insider threats. When the honeytoken directs the insider to the systems within the Honeynet we can then monitor their activity. The Honeynets are crafted to meet the insider's expectations. For example, in our honeytoken example of the system http://rd.example.com, we create a research and development webserver, perhaps complete with bogus files, documentation, log files, and even activity on the system. We can then monitor which files the insider attempts to find, and what they do with them once they download the files. These files on the honeypots can in themselves be additional honeytokens. When the insider downloads them to their system and attempts to open them, these files can call home to a central security operations center, letting administrators know that the honeytoken was downloaded and opened, what system opened it, and so on.

In addition to combining the capabilities of honeytokens and Honeynets is the concept of adaptive behavior. One of the interesting concepts resulting from the ARDA workshop is the idea of dynamically changing honeytokens or Honeynets based on the actions of an insider threat. In the overall scheme of detecting insiders honeypots are not the complete solution. Instead, they are but one of many sensors or data input to detecting insiders. Multiple inputs exist (see Figure 3). All the data collected from various sources can then be directed to a central collection system. Once correlated, indications can be found of insider activity. Honeypots are only one component in that overall architecture. However, honeypots have a unique advantage, the ability to adapt to

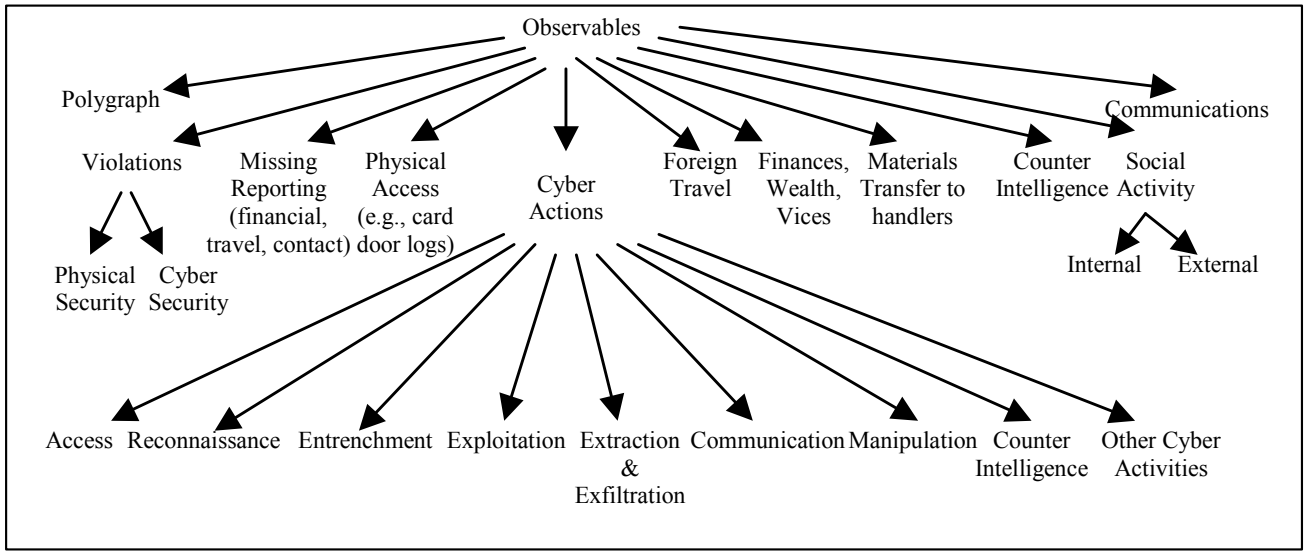

Figure 3

One of the lessons from the ARDA CI&W workshop is that there is no single observable that will always indicate an insider threat. Instead, multiple observables must be used. In this diagram, we see many of the observables that can be used as input for early indications and warning. We also see the category of Cyber Actions broken down into smaller sub-groups.

the threat. By this, we mean once the central collection systems has early indications of an insider threat, honeypots could be adapted to that threat (see Figure 4). For example, perhaps we have adaptive honeytokens in production systems with a feedback mechanism. Specifically, once we have a short list of suspects from a "broad and shallow" search, we can monitor a vital database. If a user not on the suspect list submits a query, the system responds with an unaltered production item. However, if we do have a user that is on the suspect list, then honeytokens can be adapted and introduced into the suspects activity. If suspect A submits a query and, as an additional constraint, that query is tagged as inappropriate, then the system responds with honeytoken A. For suspect B, the system responds with honeytoken B, and so on. Depending upon what the user does with the honeytoken, he or she may be removed from the suspect list. In this case, future queries will return production items rather than honeytokens. Also, Honeynets themselves could be adapted. An insider may be interested in researching a database. Once a suspect has been identified, Honeynets could be adapted to reflect what systems the attacker is interested in, the information those systems should contain.

3.3 Risks

While honeypots represent a powerful tool in our arsenal to fight the insider threat, they are not the only solution. There are several reasons for this. First, the insider threat may not ever use or interact with a honeypot or honeytoken. If that is the case, then honeypots will have little value as an observable. For example, the DIA spy Anna Montes had very few if any cyber indications, as she had trusted access to all the information she needed, and used public pay phones for communication

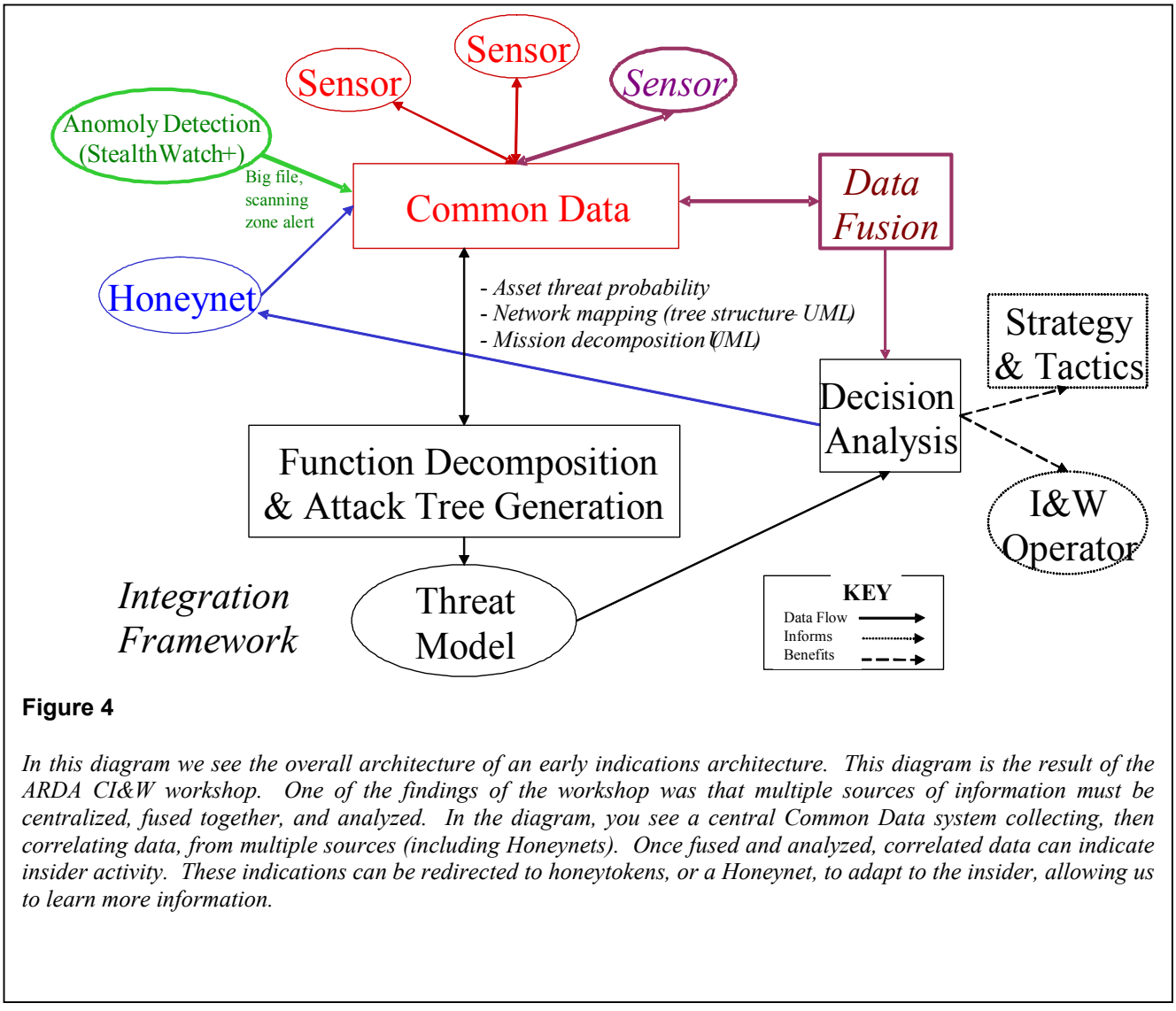

Figure 4

In this diagram we see the overall architecture of an early indications architecture. This diagram is the result of the ARDA CI&W workshop. One of the findings of the workshop was that multiple sources of information must be centralized, fused together, and analyzed. In the diagram, you see a central Common Data system collecting, then correlating data, from multiple sources (including Honeynets). Once fused and analyzed, correlated data can indicate insider activity. These indications can be redirected to honeytokens, or a Honeynet, to adapt to the insider, allowing us to learn more information.

purposes[8]. As such, other observables must also be considered (Figure 2). In contrast, individuals such as Hansen used information technology extensively, including the use of search engines. As such, honeypots have far greater effectiveness against such threats. Second, honeypots will not work if their identity is known or discovered by the insider. The individual will know to avoid the honeypot, and thus avoid an indication of their activity. Potentially even worse, if an insider has discovered an honeypot, they can introduce bogus or false information to it, misleading security organizations. To counter such issues, the use and deployments of honeypots has to be highly controlled information. The fewer people who know its identity, the less likely its identity will be compromised. One of the advantages of honeypots is their identity can easily be changed. Honeypots can monitor different IP addresses, emulate different services, or even different operating systems. Honeytokens can easily be changed as different files, search engine queries, or deployed on different systems. By not only securing the identity of honeypots, but changing its identity, they become more difficult to detect.

3.4 Further Research

The research of honeypots for internal threats, the advanced insider, is still in its infancy. Honeypots are a relatively new technology, with the first serious research begun by Fred Cohen in 1997 with the Deception Toolkit.[4] Since that time, the vast majority of research has been on external threats. Its only recently, in the past year that any work has been published on using honeypots for internal threats, and we have a long way to go. Many of the theories need more research and testing, especially the concepts of honeytokens. Specifically, how to successfully deploy honeytokens against the insider threat, and their relationship with other honeypots. A great deal of the technologies already exist, having been developed for use against external threats. Its not so much the technology that is untested, but its application against the internal threat. The concept of adaptive honeytokens also has great potential, as it can dynamically change based on the threat. This is extremely new, with little if any research or technology in this area.

4. Conclusion

Honeypots are an emerging technology, with extensive potential. They have tremendous advantages that can be applied to a variety of different environments. They dramatically reduce false positives, while providing an extremely flexible tool that is easy to customize for different environments and threats. Traditionally, honeypots have been applied against external threats or common internal threats. However, by combining the capabilities of honeytokens and Honeynets, honeypots contribute to the early indication and confirmation of advanced insider threats. The research in this area is still in the early stages, with the intent of greater testing and development in the future.

5. References

[1] The Honeynet Project “Know Your Enemy: Credit Card Fraud”, 10 July, 2003. http://www.honeynet.org/papers/profiles/cc-fraud.pdf

[2] The Honeynet Project “Scan of the Month Challenge 28”, May 2003. http://www.honeynet.org/scans/scan28/

[3] The Honeynet Project “Know Your Enemy: Honeynets”, January, 2003. http://www.honeynet.org/papers/honeynet/

[4] Lance Spitzner “Honeytokens: The Other Honeypot”, August, 2003. http://www.securityfocus.com/infocus/1713

[5] The Honeynet Project “Scan of the Month13” March, 2001. http://www.honeynet.org/scans/scan13/

[6] V. Yegneswaran, P. Barford, J. Ullrich, “Internet Intrusions: Global Characteristics and Prevalence.” In Proceedings of ACM SIGMETRICS 2003, San Diego, CA, June 2003.

[7] John Levin, Richard Labella, Henry Owen, Didier Contis, Brian Culver, “The Use of Honeynets to Detect Exploited Systems Across Large Enterprise Networks”, IEEE Proceedings, June 2003. http://www.tracking-hackers.com/papers/gatech-honeynet.pdf

[8] Affidavit, Anna Montes, Sep, 2001 http://news.findlaw.com/hdocs/docs/montes/usmontesaff901.pdf

[4] http://www.all.net/dtk/index.html