Global Information Assurance Certification Paper

Intrusion Detection Systems for a Public Key Infrastructure

By
Erin O. Buxton

GSEC Practical Assignment
Version 1.4

Abstract
Trust in a system, especially a network, has been an administrative struggle since the first hackers unleashed their attacks. As we start using and depending on "trusted" environments, we need additional ways to ensure a higher level of protection. A PKI, Public Key Infrastructure, is one such system that requires, by definition, a trusted environment. Our level of trust for this system correlates to its protection, including firewalls, IDS (Intrusion Detection Systems), honeypots, and others. In this practical assignment, the focus will be on how IDS can enhance security for a PKI.

Although sometimes proposed, a one size-fits-all approach to PKI with IDS can be dangerous and will most likely not suit your needs. You can successfully create a solution that fulfills your requirements and hopefully your budget only by carefully considering each factor and component. In order to facilitate fruitful planning, this practical includes some personal lessons learned, including pitfalls and annoyances that you may come across. Due to the lack of major standardization and the recent emergence in industry of PKI, some time will be expended explaining all of the components, choices, and potential problems.

PKI Defined
Functionality
PKI, public-key infrastructure, is a system used to support public-key cryptography applications.[1] It incorporates: "a system of digital certificates, Certificate Authorities, and other registration authorities that verify and authenticate the validity of each party involved in an Internet transaction."[2] While currently there is no single industry standard for establishing a PKI, it generally refers to a trust hierarchy within a single company or linked for several companies.[2] For PKI within a company, the highest level of trust manages certificates issued to users in order for them to access trusted services. For companies that link their trusted authorities, the link is managed as well as the level of trust of each authority.

Basically, PKI works like this: a trusted entity is created based on agreed-upon, documented rules and regulations. This entity is called the CA (Certificate Authority). Once the CA is established and protected, users can then establish trust with the CA via digital certificates. Users are issued a digital certificate based on the rules defined. Users are then able to trust each other via their mutual trust of the CA. Fortunately, most users do not need to understand any of this to use the services it provides.

The Certificate Authority (CA) manages digital certificates by: issuing new certificates, revoking or canceling issued certificates, and sometimes sharing public certificates with other Certificate Authorities. It also evaluates levels of trust and access based on the certificate by determining its validity and what actions are authorized.[1]

Key and potential PKI components
We can think of the PKI network as a small, separate network within your company network with the CA as the most valuable component. Basic components that are also typically found are: security protection devices such as firewalls, network and host-based IDS; security auditing tools including host-based IDS, logging servers, backup / recovery servers; time services to synchronize time of the PKI architecture; and directory services such as LDAP.[3] Some optional components of a PKI architecture are a VPN (Virtual Private Network) Server to enable VPN authentication with the CA, a timestamp server to timestamp certificates as well as logs, and web server to allow users to obtain certificates with a web interface.

PKI business services
By creating a hierarchy of trust, employees or customers can use the PKI to access services for user identification, authentication, authorization, and/or non-repudiation. From a high-level business prospective, it can be used to perform encryption for secure messaging via email, to secure files and folders, to create secure web portals, to digitally and securely sign e-forms, and to secure VPN (Virtual Private Networks) as well as other services.[4] Businesses that currently use PKI include large companies, financial institutions, healthcare industries (providers, researchers, and pharmaceutical companies), and governments.

Why PKI networks must be protected
Reasons to protect a PKI network are basic: to establish / maintain trust of the CA and protect services / sensitive information. Moreover, some industries that would like to use electronic signatures are required to follow government regulations such as HIPAA (Health Insurance Portability and Accountability Act of 1996) requirements applicable to the U.S. healthcare industry healthcare industries. [5] Depending on the services and the level of trust you decide to provide, the level of protection can vary greatly.

For all types of protection, the first component to protect is the CA because it is the foundation of trust for a PKI. The security protection devices are used to detect, analyze, filter, log, allow, restrict, and/ or stop traffic into that network. Firewalls are the most commonly used to serve this purpose, but depending on the level of security and trust required for the CA, this is often not enough. When a higher level of security is needed, intrusion detection systems (IDS) are put in place to analyze traffic passing through the firewall. A network-based IDS will transparently "sniff" traffic in real-time, check for patterns that are known attacks, and display patterns to administrators. A host-based IDS can check for successful attacks on individual systems to verify integrity or detect compromise. Some can also verify that a system has been properly "hardened", i.e. known vulnerabilities have been removed based on security policies established.

Security auditing tools are key to verifying the integrity and trust of the CA. In order for trust to be ensured, it must exist currently and in the past. For example,

imagine if a hacker exploited a vulnerability momentary to compromise the CA and created a certificate illegally. The hacker could cover his tracks and close the hole opened, but the illegal certificate would still exist. Auditing tools offer the proof of integrity of the past and present. Some host-based IDS can be used to substantiate file integrity, even down to binary code, by using "digital snapshots" or a baseline of valid states to compare against current states. [6] Logging servers can be used to compile logs from critical servers to verify that servers have not been compromised. Lastly, backup / recovery servers ensure that copies of data are made and can be restored if data is lost or compromised. They also allow for a return to integrity from a compromised state.

Timeservers are used to synchronize time for all servers. This provides consistency in audit logs that can be used in data integrity forensics. Directory services are used to allow a transparent interaction for the user with digital certificates. Certificates can be stored on the directory and crosschecked with a list of revoked certificates to verify validity and integrity.

While the focus of this document is on the IDS, components are listed and discussed to understand what we are protecting. In order to protect the CA, we must not merely focus on it but also on the components that protect the integrity of the trusted entity. Otherwise, compromising the CA is as easy as compromising the auditing tools or the firewall.

Definition of IDS

Classifying IDS

There are several ways in which you can classify an IDS: misuse detection versus anomaly detection, passive versus reactive, real-time versus manual or scheduled, and host-based versus network-based.[7] The latter will be discussed in more detail in the sections following.

The first two classifications tend to depend on vendor philosophy and their products. An IDS using misuse detection compares data gathered against a database of known attack signatures. This database will constantly need to be updated by administrators provided by the vendor. Also important to note, detection is "only as good as the database of attack signatures that it uses to compare packets against."[7] This means that if new attacks are unleashed on your network, you will not be protected from them. An anomaly detection system uses baselines set by administrators that define the normal state of traffic passing through the network. If suddenly, one protocol appears that is outside this norm, the IDS would detect it.[7] However, many attacks can simulate normal traffic and so may go unnoticed. Also, once it is determined that abnormal traffic is passing through, it may be difficult for the administrator to conclude the type of attack without resources such as a database of known attacks, the purpose of the attack (information gathering or Denial of Service), the risk level, and the potential causes for false-positives

Another difference is whether systems are passive or reactive.[7] A passive IDS will only log and signal alerts, while a reactive IDS will reconfigure a firewall to block the traffic. In later sections, some of the pitfalls of a reactive system are discussed. These systems can create more problems than solutions.

Another classification of systems is real-time versus manual or scheduled. A real-time IDS checks for attack patterns while "sniffing" the network segment and is constantly looking for attacks. Based on configuration, manual or scheduled IDS run scans when prompted or scheduled to do so. Although they do not detect hackers at the time of the attack, this style of detection is better suited for verifying baselines of important files. Usually, a combination of both styles provides the best protection.

Finally, the major difference between IDS is network-based versus host-based systems. This is a fundamental distinction worth further discussion below.

Network-based IDS

A network-based IDS "sniffs" all packets that flow through a particular segment of the network and analyzes packets to detect suspicious patterns designed to be ignored by firewall rules. Analysis is accomplished by comparing packets to a database of known patterns of attacks, called signatures. Three types of signatures include string, port, and header condition.[8] Traffic is not blocked as with a firewall; but behavior is logged, and alarms are sounded. If traffic matches a signature, a log is usually created and an alarm may be activated depending on configuration. A detailed description of how traffic can be "sniffed" is described in a later section.

Host-based IDS

A host-based IDS, by contrast, examines activity on an individual computer or host for vulnerabilities or unintended changes.[7] In general, any tool that monitors activity on a single machine from attacks is a host-based IDS.[7] However, as with a network-based IDS and a firewall, it is important to distinguish between personal firewalls and host-based IDS. Most host-based IDS either detect vulnerabilities based on a database or use file baselining to compare known "good" states to current states.

Using IDS to enhance protection of a PKI

Prioritizing protection for machines

In the PKI network, all machines are important, especially the CA. To prioritize, first major concerns have to be recognized. As listed on the SANS website as "The 7 Top Management Errors that Lead to Computer Security Vulnerabilities," number five states that management "Fail to realize how much money their information and organizational reputations are worth."[9] Depending on the level of trust required for your CA, you may need to have a full audit trail of logs, you may need disaster recovery of data and systems within a specific time frame, or you may need documents as forensic evidence in court cases to prove that your

system was protected or hacked. So, by listing all of your data, applications, and machines, you can start determining:

- What you can lose and what you absolutely cannot lose
- What implications come up if this data is lost or compromised
- How long would it take to discover if data or systems were compromised
- What are your legal concerns, if this data can be restored
- How long it would take to restore
- What happens if services are down.

Obviously, concerns of financial institutions will differ from a healthcare company, so this will need to be customized for your needs. Once you know what you absolutely cannot lose, such as the initial CA root keys, you can set about protecting it. There are multiple ways machines and data can be protected, but the focus here will be on how IDS can help.

The network-based IDS should tell you when a hacker might be attempting to compromise systems so that you can readily protect machines. Certain host-based IDS can be used to ensure only necessary vulnerabilities exist on your machine. For forensic purposes, some host-based IDS actually have "snapshots" of data that are used to determine if changes have been made.[10] It can even help you if you need to restore data and want to make sure restored data can be trusted. The only issue with this type of IDS is that it typically runs scans by manual requests or on a schedule. Therefore, if you need to know real-time when a system is compromised, you will need to run scans often or use multiple tools. Also, if machines have gone down and you need machines up in a limited amount of time, you may need to run scans more often to catch problems. Early detection will help you restore your systems faster.

How, when, and why to choose a host-based IDS

Although some host-based IDS detect attacks in real-time, the majority run based on a schedule or by manual command. Most times this is sufficient since you can adjust the automatic scan schedule based on your requirements of protection. Note that host-based IDS scan only one machine so should at least be installed on every critical machine. For example, in a PKI network, they should be used to protect the CA the backup/recovery machines, and the machine used to interact with the user to have certificates issued (such as a web server). Also, a host-based IDS can be used to ensure consistency of the perimeter protection such as a firewall. It could verify that the firewall has not been compromised and the security policy has not been changed. Often though, since the PKI network is fairly small and each component is important, many companies decide to protect all machines. A good host-based IDS should offer you one of two things: vulnerabilities checks and baselining or file baselining and integrity matching.

The first type is used to check a machine to verify that all currently known vulnerabilities except those necessary for use are removed. For example, scans

are run after a machine has been hardened. The scan checks for known vulnerabilities listed in a database compiled by security specialists and provided by the vendor. A report is created that lists any remaining vulnerabilities, and the administrator either deems the vulnerability necessary for functionality of the machine or removes the vulnerability from the machine based on directions in the report. For example, SNMP may be a vulnerability on the system but may be necessary to run network management tools. However, you probably don't need an email server on the CA. Once all unnecessary vulnerabilities are removed, the machine is base-lined at this state. Future reports will reflect changes to the baseline or new vulnerabilities added to the database that are now seen on the machine. This type of host-based IDS is especially good for newer administrators or ones with less experience in security. It also is a good check to verify that vulnerabilities have been removed properly and no errors have been made. Even if many people put faith in their administrators, humans make errors. If you have multiple administrators on a network controlling security, it is very difficult to have accountability and crosschecking. However, if you have an application that audits changes or has databases of vulnerabilities that are updated by many security specialists, you end up with administrators that can verify their work and be assured that they are up-to-date. Last of all, it allows administrators to see if any vulnerability has suddenly been created, i.e. if a hacker was successful in an attack and has opened up a vulnerability on a machine. Not only should the IDS detect this change, it should tell them what the vulnerability is, what are the repercussions for this vulnerability, what is the risk level of this vulnerability, and how to remove the vulnerability if necessary. Ultimately, the administrator has to decide if the vulnerability is necessary for functionality of the system.

The other type of host-based IDS is file baselining and integrity matching. With this type, the application takes a "snapshot" of files that are determined by administrators to be at a correct state. This "snapshot" will later be used to compare current states to verify file integrity and validity. The files can include executables, registry keys, application files, log files, and other important files. Fortunately, with a good host-based IDS, you can determine which files are important to you and protect in the ways that are appropriate to the file. For example, if an important file is to remain unchanged, you can run scans to verify that the files still exists and has not changed, even to the binary level. However, some files are more complicated and only certain attributes need to be checked for. Application files, for example, may change but their properties should not, or log files may grow but not decrease in size.[6] Another example of functionality is permission checking. Since administrators usually have privileged access to systems, administration behavior can be monitored on a system. This is useful for accountability of administrators. In a PKI environment, this type of host-based IDS can be used to verify that:

- The PKI private key database is not compromised by changing the executable of the database management
- The web interface used by user has not been compromised or changed

- The permissions of the backup server have not been altered to give a hacker access to backup files
- The all application executables, folders, and logs are present with proper permissions.

A file base-lining IDS should check files using this level of sophistication to allow you to have more control over your important files. If you have legal concerns over document integrity, it is suggested that this level of complexity be used and that the vendor has experience using their products as forensic evidence.

Although some vendors will claim to have products that perform both the types of host-based IDS simultaneously, there are several reasons that show why different vendors may be necessary. The main reason is as follows. Vendors tend to specialize in a product then compete with other companies for market share. In order to gain all business, they will try to make a product do everything- "You want our product to protect your systems and make your coffee? No problem!" In some cases, one product is enough, but in a hypersensitive environment where trust is the most important quality and data integrity is vital, choosing a vendor that specializes in one area can save you a lot of frustration and money in the future. In the end, you will have to weigh your cost versus risk to systems to determine if one product will suffice or if multiple are necessary.

How, when, and why to choose a network-based IDS

While host-based IDS protect individual machines, network-based IDS help protect the network and almost all provide real-time detection and warnings. Since it is not a firewall and is only "sniffing" traffic on the network segment, it is transparent to a would-be hacker and should be completely inaccessible except to administrators from the sensor management console. Therefore, if your firewall is brought down, your network-based IDS would probably be the first to signal alarms. If you are looking for the culprit, it will be able to give you necessary forensic data, such as the IP address of the source and destination of attack, type of attack, time and date that the attack started. This forensic data can be used to determine how to remove the vulnerability in the firewall and maybe even whom to unplug from the network. It can also be used to see if someone is trying to attack the CA with an overflow of legitimate protocols or by running port scans destined for the CA to find potential vulnerabilities. If a certain IP address is constantly scanning the CA for holes, the firewall rule base can start filtering this IP address. The network-based IDS will be checking for patterns that may be difficult to see just by reviewing firewall logs. Since the network-based IDS is looking for attacks by checking against a database of exploits or "normal" network behavior, it can help you to see patterns of activity more clearly.

Criteria for a good network-based IDS may differ depending on your environment. For a PKI environment, which tends to have fewer machines with specific roles and protocols and less traffic, most network-based IDS systems can work. You will therefore have a greater choice of products. Major criteria

for selecting an IDS will include: how user-friendly is the product (configuration, GUI interface, alarms, and logs), what kinds of reporting will be provided automatically, how helpful and available is the technical support, can you add your own kind of alarms, can you add attack signatures, does it have a stable database or can a stable database be added, does it work in your general network environment, and does it work on your supported operating systems. Although there are freeware network-based IDS available which can be ideal for shorter periods of time, many are not supported by technical help. If you have legal requirements, freeware applications are not recommended.

Why both may be necessary
Visualize a castle with a king, queen and all their people. In order to protect the king and queen, they first need to be protected from major attacks. These include a moat and gatekeepers (firewalls) and lookouts (network-based IDS) that watch all traffic in and out of the castle. This protects all people in the castle, important or not. However, if someone escapes detection, the king and queen are still vulnerable to attacks. On a schedule, the head of the guards (host-based IDS) verifies that all guardsmen are in place to protect the king and queen and give warning if any are missing or killed. Other guards (host-based IDS) check to make sure that all the royal gold is in place, that the king and queen are still present and well, and that the castle is intact. Without all the guards, holes in security can easily be found and exploited. Each role is vital and unique in the overall protection.

A network-based IDS is important to detect potential attacks in real-time but if an attack passed without detection, your systems could be compromised. On the other hand, host-based IDS may not detect an attack as early as a network-based IDS, and it may be difficult to find out the source or method of the attack. Together, they offer real-time protection, audit trails, and forensic information.

Lessons Learned, Pitfalls, and Annoyances
What you wish vendors told you in the beginning
How to "sniff" traffic[11]
Since the IDS vendors are not necessarily involved in your network design, they do not know how you will be sniffing your network when using a network-based IDS. Therefore, they tend not to talk about it until you are calling technical support and asking why you are unable to see traffic and alarms. If you are using hubs to transfer data, then all information is propagated to all ports, and sniffing your network is as easy as plugging in your IDS to a port. However, if you are using a switched network, which is extremely common nowadays on networks, traffic on the network is not broadcasted to all ports but only to the port for which the traffic is intended. Therefore, you will not be able to sniff without additional configuration. One solution is to use a hub between switches to direct traffic. The obvious problem with this is that collisions will cause your network to be slow and routing loops could be created.

Another solution is to use span port on your switch, which mirrors traffic. The IDS system can then be plugged into the span port to sniff data. Unfortunately, some company's standard committees will not allow span ports on the network so asking administrators before considering this solution is prudent. Also, mirroring multiple ports may not be possible due to high traffic levels, limitations of the IDS sensor, or limitations of the switch.

Network taps can also be used. A tap is placed between two network devices, such as between two switches or between a server and a switch. Two IDS sensors are necessary to capture network traffic in both directions. Taps have several issues: they can be expensive; you may not be able to support features of the IDS such as terminating sessions automatically since taps only sniff and do not send information out onto the network; it requires the IDS sensor to support a promiscuous mode NIC (Network Interface Card) configuration since traffic will not be interactive between the sensor and the network; and tap ports only deliver RX data, so some attacks such as Arp and IPDuplicate or SYNFlood will not be detected. Also, if you want to watch traffic in both directions, which is recommended, and need two IDS sensors, you will need additional hardware and software licenses with can seriously increase your cost. If the traffic in each direction is not to the upper limit of capabilities of the sensor, you will not be fully utilizing and maximizing value of your sensors. This reduces your value to cost ratio.

Another tap solution is to consolidate taps hooked into your network through a switch. A span port would then be set up on this switch that would connect to a sensor. This would reduce the number of IDS sensors, thus reducing cost, while increasing the value of the sensor by maximizing utilization. Issues with this solution though are that span ports can be overloaded and, depending on the sensor capabilities, that it is only advantageous if consolidating links with low utilization. So why not just use a hub to consolidate instead of a switch? Hub will still generate collisions easily, which can cause packets to be lost.

At last, the final solution to this issue, which will solve all your problems but cost you a lot for a small network, is to use a load-balancing device that specializes in working with intrusion detection devices. One such device is a TopLayer [12] switch which allows load-balancing of traffic across multiple span ports, provides for a diversity of port speeds, and allows for load-balancing to two or more sensors in order to maximize usage of the sensors. Issues are that this solution is expensive and fairly new, your company may not support this type of switch, administration / maintenance / training of a new type of switch may be costly to your company, and sensors still need to work with a promiscuous mode NIC.

How to monitor alarms

When using a network-based IDS, it is important to understand what protocols and traffic is going over your network. Practice and time with the network IDS on your network will give you better experience to understanding what violations are,

what attacks look like, and what is normal traffic. Some things you can do to practice are to use freeware security scanners, such as Nessus[13], to imitate attacks so that you can see violations logged on the sensor. This is also a good way to test to see if your IDS will detect the attacks that you are expecting it to detect thus checking your configuration. It can also be used to test your alarms and automatic alert systems if you are using them. Stress tests can also be run to see how many alerts will be sent out and how that will stress your network, email server, network management tool, and firewall depending on what types of alarms you plan on using. It is important to do this in a lab environment so as not to disrupt traffic on your network. Also, you will usually need written permission from the company before performing any scans of their network, and network administrators should be informed of your scanning to prevent downtime.

Many network-based IDS systems will also give details on each violation type and the standard priority level given. It should also list links where you can get additional information. Some applications will also provide information such as how this violation could be a false-positive and how to check, what was the source of the violation, and where was the data destined for. Ultimately, the administrator will have to make the call whether the traffic is legitimate or not and then how to respond. In many cases, the hardest part is imagining what will happen if this traffic is blocked- will blocking traffic create a denial of service? Fortunately, on a PKI network, you will have limited services to each machine so it will be easier to visualize solutions and repercussions. For example, you will normally not need to have a web browser on the CA, therefore blocking all HTTP traffic to and from the CA would not affect performance. Being knowledgeable about all protocols and ports used by each machine will allow you to fine-tune your configuration even more than what you could on a large-scale network.

But now the question, do you still sniff for traffic that is blocked by the firewall? What if the firewall went down? How long would it take for you to notice? What if the hacker was able to get around the firewall and send denial of service attacks from within the network? It is a difficult balance between catching all violations and overwhelming administrators with alerts. It will normally take a minimum of a month to really fine-tune the IDS.

Alarms on a large and small network

Although many vendors boast bells, whistles, and automatic tools, be wary of using these on a large scale, on a high traffic network, or with an IDS that alerts administrators each time a violation is discovered. While they sound exciting and useful, they can cause more damage than good. When performing tests, email servers and network management servers have been brought down within ten minutes of putting an IDS on the network. Essentially, they can create a self-induced denial of service. Testing the IDS with the email and network administrators, so that they can quickly remedy issues, will prevent embarrassment and costly downtime. Some network management tools will allow you to reduce the number of alarms sent by only alerting administrators

after a certain number of violations have been discovered. In the beginning, a large amount of alerts is normal until you can tweak the IDS to fit your network.

For email alerts, the information sent should not violate your security policies, such as giving internal PKI network IP addresses. Security polices within the PKI network may need to be stricter due to the sensitivity of the information on the systems within.

Managing the IDS log database

Although some vendors tout that a database is included in the product, many IDS products need another database application to handle the logs and alarms created. This can increase the cost of your PKI network and create an administration issue of management of your database. For very small networks, another database might not need to be added. Depending on the configuration of the IDS and level of attacks, logs may grow so fast that the database can become corrupt or lost. It is therefore suggested that the vendor be asked what database servers are supported by their product and determine the additional costs.

Pitfalls

Some easy pitfalls to fall upon are money, resources, over-reaction, and apathy. All too often administrators understand what is needed but getting the funds or the time to accomplish this is difficult to impossible. Companies that are willing to pay millions to set up a PKI will be the same ones asking you to cut costs by removing an inexpensive but necessary server. Planning for costs will help but only after testing in a lab will you know truly what the costs will be. Also, creating a hardware and software list with justification of each component is highly recommended to help management to see the light. A risk analysis is even more beneficial.

The people assigned to watch over the system can prove to be a major issue. As listed on the SANS Institute site as the number one "Top Management Errors that lead to Computer Security Vulnerabilities", managers will " Assign untrained people to maintain security and provide neither the training nor the time to make it possible to do the job."[9] For a system in which trust is the most important and necessary, having people that are not trained on using and protecting the system can be dangerous to the company's reputation and costly. If the trust of the CA is compromised and cannot be fixed, another CA will have to be built. CAs can cost up to $100,000 for the root key generation ceremony, not including the cost of hardware and software. Including initial and continued training, on the products used as well as security training, in the budget and justification is recommended to remedy this issue. Outsourcing is another option but needs to be heavily researched to ensure that the outsourcing supplier can offer all the services that you need without creating a security hole. Also, in order for the IDS administrator(s) to have sufficient time to monitor logs and alarms and to research issues, they should not be assigned to administer too many functions.

If you have the IDS administrator assigned as the firewall administrators, you will lose crosschecking of security policies and behavior between groups. By having specialization and separation of duties, your administrators will be better able to perform their duties.

One thing that lack of training can result in is over-reaction to alerts. It takes several months sitting over an IDS system to be aware of what is "normal" and what is a hacker. The administrator also needs to be aware of the network and protocols that are running on the network. Looking at patterns in alerts will allow the administrator to research the types of alerts popping up to determine if the configuration of the IDS needs to be adjusted, if a server or application needs to be reconfigured because it is putting unnecessary traffic on the network, or if a hacker is indeed attacking. If the administrator has not been trained on how to research the alerts (what causes them, what are false-positives that may come up, how their network functions), they will block legitimate traffic, thus causing a self-induced denial of service. This can also create a "crying wolf" administrator that no one believes when an actual attack occurs.

Finally, the over-reacting phase is followed by an apathy phase. If an IDS is configured to notify the administrator too often, the administrator will become apathetic to alarms. The IDS can still be configured to log all suspicious behavior that can be reviewed for patterns and auditing purposes, but alerts should be adjusted to reduce apathy. This is a difficult balance to achieve and usually comes after months of sitting over the IDS, with experience, and with being familiar with the network.

Conclusion

Know what is important to you in the short term and long term

If you know your short term and long-term goals, you will be able to create a system that is flexible enough to accommodate your future needs. Keeping focused on what is most important to you will help you prioritize all your decision-making processes. Once you decide how important something is to you, you can more easily analyze the risks of losing and replacing it. For a PKI network, not having clear short-term and long-term goals can seriously increase costs. PKI services can improve company security and user to data interaction, but this requires development of a vision with budgets included. Once the vision is created, systems can be designed to meet it.

Involve other people to avoid surprises

Any person that may be involved or affected by the construct of a PKI network should be consulted to ensure that no conflicts exist. Although it may be frustrating to have many people involved in the design, major problems can arise from surprises. Involving others early will save time, frustration, and money. For example, say that you have decided to use span ports to sniff for your network-based IDS, but span ports are not allowed on your network. Or, suppose that you would like to use an IDS product that works on operating systems that are

not supported by company standards. With written approval of your PKI design, all involved will be aware, and conflicts can be avoided.

Focusing your attention

As with any system or project, the key to success is thinking and planning before acting. Planning involves drawing up designs, discussing the architecture of the PKI network, confirming the design with the network administrators and standard committees for conflicts, and then building a lab. A lab will allow testing of the configuration. As we have seen some in the section on pitfalls, this will help avoid making many costly mistakes. If configurations are not tested, you can expect major problems. This will also allow you to tweak the configuration, especially if testing the system using realistic and extreme scenarios. This will also let you see how happy you are with the products you have chosen. Although programs and hardware may work together in theory, in real life many do not. Many times, after testing certain hardware with certain applications and finding out that there are conflicts, you will be able to explain to the boss or corporation why buying a slightly more expensive product is necessary. The lab environment should be as close to the final product, or a staging environment can be created. Once tested and refined, the system can be put onto the network where it will surely need more refining.

Know your enemy

The obvious enemy is the outsider, the hackers, but is that really who your enemy is? Most times, internal attacks pose the greatest threat, because users will have a clearer goal of what they want to attack through knowledge of and authorized access to the systems they wish to attack.[14] These attacks are therefore difficult to track and prevent.[14] "Gartner estimates that more than 70% of unauthorized access to information systems is committed by employees, as are more than 95% of intrusions that result in significant financial losses."[15]

So now that we remember who the enemies are, we can make sure to review our network–based and host-based IDS configurations and verify that internal threats are being considered. It is recommended to log all suspicious behavior, external and internal, to prevent being blindsided by internal attacks. Accountability, as well as crosschecking, should be reviewed for administrators. Additional accountability functions have recently been added to many IDS, but we must use them to gain the benefits.

Bibliography

References

1. "4.1.3.1 What is a PKI?" RSA Laboratories' Frequently Asked Questions About Today's Cryptography 4.1. 2002. URL: http://www.rsasecurity.com/rsalabs/faq/4-1-3-1.html (16 Aug. 2002)

2. "PKI." 31 Oct. 2001. URL: http://webopedia.internet.com/TERM/P/PKI.html (16 Aug. 2002).

3. "Public-Key Infrastructure Components." Architecture for Public-Key Infrastructure (APKI). 1998. URL: http://www.opengroup.org/onlinepubs/009219899/chap3.htm#tagcjh_04_08 (16 Aug. 2002).

4. Entrust Homepage. 2002. URL: http://www.entrust.com/index.cfm (16 Aug. 2002).

5. "Frequently asked Questions." Health Insurance Portability and Accountability Act of 1996. 30 Jan. 2001. URL: http://www.hipaa-iq.com/faqs.htm (16 Aug. 2002).

6. Tripwire Homepage. 2002. URL: http://www.tripwire.com (16 Aug. 2002).

7. "intrusion detection system." 2 Aug. 2002. URL: http://inews.webopedia.com/TERM/I/intrusion_detection_system.html (16 Aug. 2002).

8. Northcutt, Stephen. "What is network based intrusion detection?" Intrusion Detection FAQ. 2000. URL: http://www.sans.org/newlook/resources/IDFAQ/network_based.htm (16 Aug. 2002).

9. "The 7 Top Management Errors that Lead to Computer Security Vulnerabilities." 1999. URL: http://www.sans.org/newlook/resources/errors.htm (16 Aug. 2002).

10. "Tripwire Software Protects Data and Network Integrity, Helps Healthcare Systems Meet HIPAA Privacy and Security Standards." 2002. URL: http://www.tripwire.com/literature/white_papers/HIPAA.cfm (20 Aug 2002)

11. Laing, Brian. "Intrusion Detection Systems: How to Guide- Implementing a Network Based Intrusion Detection System." switched.zip. 2000. URL: http://www.iss.net/support/product_utilities/realsecure_tech_center/tips_tricks/index.php (16 Aug. 2002).

12. TopLayer Homepage, URL: http://www.toplayer.com (16 Aug. 2002).

13. Nessus Homepage. URL: http://www.nessus.org (16 Aug. 2002)

14. Bassham, Lawerence E. and W. Timothy Polk. "Threat Assessment of Malicious Code and Human Computer Threats." NISTIR 4939. Oct. 1992. URL: http://security.isu.edu/isl/threat.html (16 Aug. 2002).

15. Hunter, Richard. "Enterprises and Employees: The Growth of Distrust." URL: http://security1.gartner.com/story.php.id.12.s.1.jsp (16 Aug. 2002).

Last Updated: July 28th, 2017

Upcoming Training

SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
Community SANS Pasadena SEC401 @ NASA	Pasadena, CA	Aug 23, 2017 - Aug 30, 2017	Community SANS
Mentor Session - SEC401	Minneapolis, MN	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
Mentor Session - SEC401	Edmonton, AB	Sep 06, 2017 - Oct 18, 2017	Mentor
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
Community SANS Albany SEC401	Albany, NY	Sep 11, 2017 - Sep 16, 2017	Community SANS
Mentor Session - SEC401	Ventura, CA	Sep 11, 2017 - Oct 12, 2017	Mentor
Community SANS Dallas SEC401	Dallas, TX	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS Columbia SEC401	Columbia, MD	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS Boise SEC401	Boise, ID	Sep 25, 2017 - Sep 30, 2017	Community SANS
Baltimore Fall 2017 - SEC401: Security Essentials Bootcamp Style	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
Community SANS New York SEC401	New York, NY	Sep 25, 2017 - Sep 30, 2017	Community SANS
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Copenhagen 2017	Copenhagen, Denmark	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Oct 02, 2017 - Oct 07, 2017	Community SANS
Community SANS Sacramento SEC401	Sacramento, CA	Oct 02, 2017 - Oct 07, 2017	Community SANS
Mentor Session - SEC401	Arlington, VA	Oct 04, 2017 - Nov 15, 2017	Mentor
SANS October Singapore 2017	Singapore, Singapore	Oct 09, 2017 - Oct 28, 2017	Live Event
Community SANS Indianapolis SEC401	Indianapolis, IN	Oct 09, 2017 - Oct 14, 2017	Community SANS