DOWNLOAD PDF UNDERSTANDING PUBLIC-KEY INFRASTRUCTURE

Chapter 1 : Understanding Public-Key Infrastructure by Carlisle Adams

Public Key Infrastructure Public key infrastructure is an architecture which supports mechanisms like integrity and confidentiality. It is heavily used in e-commerce, where business transactions are frequent and should be much secured.

Renew certificates that have expired Revoke certificates To enable all of the above listed functions, the PKI consists of numerous policies, software and components that manage public and private keys, and certificates that authenticate users and verify data. Each component included in the PKI is discussed in the following section of this Article. Digital Certificates A digital certificate associates a public key with an owner. The certificate verifies the identity of the owner. A certificate cannot be forged because the authority that issued the certificate digitally signs the certificate. Certificates are issued for functions such as the encryption of data, code signing, Web user and Web server authentication, and for securing e-mail. When certificates are issued to a client, it is stored in the Registry and in Active Directory. You can also store certificates on smart cards. The information included in a certificate is determined by the type of certificate being used. Certificates can contain all of the information listed below, or only some of the information listed below: The name of the user. The e-mail address of the user. The host name of the computer. The serial number of the certificate. The time for which the certificate is considered valid. The issuing Certificate Authority CA ensures that the serial number for each certificate is unique. The location of the certificate revocation list CRL. The CRL is a list which stores the details of certificates which have been revoked. Information on the policy which was used to initially authenticate the user of the certificate. It describes a certificate as the means by which the distinguished name of the user can be associated with the public key of the user. The distinguished name of the user is defined by a naming authority. The distinguished name is used by the issuing Certificate Authority CA as the unique name of the user. The information included in an X. This is the version of the certificate. A unique identifier assigned by the CA to the certificate. This is the hashing algorithm used for the digital signature of the certificate, and is usually MD5 or SHA This is the Certification Authority that issued the certificate. The date which the certificate was issued Valid To: This is the expiry date of the certificate. This is the distinguished name of the owner of the certificate. This is public key which is associated with the private key. This is the algorithm used to create the certificate hash. The hash of the certificate which is used for positive identification of the certificate. Certificate Authorities CA A certificate authority CA is the trusted entity that issues digital certificates to users, computers or a service. An organization can have multiple CAs, which are arranged in a logical manner. A CA can be a trusted third party entity such as VeriSign or Thawte, or it can be an internal entity of the organization. Windows Server Certificate Services can be used to create certificates for users and computers in Active Directory domains. The tasks performed by a CA are listed below: Accepts the request for a certificate from a user, computer, application, or service. Authenticates the identity of the user, computer or service requesting the certificate. The CA utilizes its policies, and incorporates the type of certificate being requested; to verify the identity of the requestor. Creates the certificate for the requestor. Digitally signs the certificate using its own private key. The process by which a user, computer, or service identifies itself to the CA is called registration. Registration can be automatically performed during the certificate enrollment process, or it can be performed by another trusted entity. An example of a trusted entity would be a smart card enrollment station. Certificate enrollment is the terminology used to refer to the process by which a user requests a certificate from a CA. There are basically two type of CAs. The CA types are distinguished by the location in which they store their certificates: Enterprise CAs automatically responds to any certificate requests. This basically enables clients to access and obtain certificates, and locate them in their own local certificate stores. Because of these characteristics, enterprise CAs should not be used to issue certificates to any clients external to the enterprise. A standalone CA stores information on its certificate locally, in a shared folder which can be accessed through a web URL. Standalone CAs depends on an Administrator to manually approve or deny any request sent for a certificate, by default. A standalone CA is typically used to issue certificates to users who are external to the organization. CAs can be categorized into different trust models: Single CA trust model: The CA server is basically a stand-alone server that does not exchange information with any other CA servers. The public key of the CA is distributed to users who need to use the CA for certificate requests. Hierarchical CA trust model: A hierarchical CA model exists when there is multiple CAs within the organization. In a hierarchical CA trust model, each CA is one of the following: The root CA functions as the authority over all subordinate CAs located beneath it. It is basically the parent that issues certificates to the subordinate CAs beneath it. The root CA creates a self-signed certificate for itself. Thisis a certificate where the issuer and subject of the certificate are identical. With a hierarchical model, when a client trusts the root CA, it has to trust each subordinate CA located beneath the root CA. This is because they are issued certificates by the particular root CA. There are two types of subordinate CAs in the hierarchical CA model, namely: The function of an intermediate CA is to issue certificates to leaf CAs. The function of a leaf CA is to issue to certificates to users, servers and services who request CAs. A certificate trust list CTL is a list that documents the trusted certificates of the enterprise. It is a list of root CAs which is trusted within the enterprise. The benefit of using the Windows Server CTL is that you can automatically check certificates to this list. There is however occasions when the CA can end the validity of the certificate through a procedure referred to as certificate revocation. A certificate is typically revoked when information included in the certificate has become invalid or untrusted. When the private key associated with the public key in the certificate is no longer secure or trusted, the certificate should be revoked without delay. The certificate revocation process is performed by the CA issuing the certificate revocation list CRL , and it includes the serial numbers of those certificates which have been revoked. A simple CRL is a single file that grows as more revoked certificates are added to the list of certificates which have been revoked. A simple CRL stores the list of revoked certificates with the following information: After this, periodic updates which are called deltas are sent to the entities. The deltas basically detail any updates that should be included. The OCSP responder sends the response to the party that sent the request. The information included in the response is listed below: The status of the certificate is identified as one of the following: The time when the status of the certificate was last updated. The time when the status of the certificate is expected to be updated next. The time when the response was sent to the party that sent the request. The public key syntax is for certificates and the private key syntax is for the encryption of private keys. The standard describes the Diffie-Hellman Key Agreement, which is a technology used to share secret keys between two entities. The secret key is used to encrypt data transmitted between the pair. This standard describes the encryption of a string with a secret key which stemmed from a password. The output is an eight octet string. This standard describes extended certificates. An extended certificate is an X. This standard is similar to the PKCS 6, Extended-Certificate Syntax Standard, in that it also includes additional attributes, but with public key algorithms, for sending private key information. This standard describes the attribute types which can be used in extended certificates, digitally signed messages, and private key information. PKCS 12 defines the portable format diskettes, smart cards for storing and transmitting the private keys and certificates of users. Certificate Policies A certificate policy can be defined as the rule s which govern the manner in which a certificate can be used.

Chapter 2 : PKI Entities | Understanding the Public Key Infrastructure

A beginner's guide to Public Key Infrastructure PKI can help keep your network secure, but it can be a hard concept to understand. Brien Posey explains how it works.

All access control decisions are driven by an authorization policy, which is itself stored in an X. There is also a Privilege Allocator, which is a tool that constructs and signs attribute certificates and stores them in an LDAP directory for subsequent use by the ADF. Show Context Citation Context This paper assumes the reader is already familiar with the general concepts of PKIs, and these will not be repeated here. The primary data structure in a PMI is an X. Signed by attribute authority This strongly binds a set of attribute Smith - International Journal of Information Security , " A programmable secure coprocessor platform can help solve many security problems in distributed computing. These solutions usually require that coprocessor applications be able to participate as full-fledged parties in distributed cryptographic protocols. Thus, to fully enable these solutions, a gen Thus, to fully enable these solutions, a generic platform must not only provide programmability, maintenance, and configuration in the hostile fieldâ€"it must also provide outbound authentication for the entities that result. A particular application on a particular untampered device must be able to prove who it is to a party on the other side of the Internet. To be effective, a secure outbound authentication service must closely mesh with the overall security architecture. Our initial architecture only sketched a rough design for this service, and did not complete it. This paper presents our research and development experience in refining and implementing this design, to provide PKI-based outbound authentication for the IBM Model 2 secure coprocessor platform. The Validate algorithm usually e. However, it is well known that BGP is vulnerable to a variety of attacks, and that a single misconfigured or malicious BGP speaker could result in large scale service disruption. We first summarize a We believe psBGP trades off the strong security guarantees of S-BGP for presumed-simpler operations, while requiring a different endorsement model: This work contributes to the ongoing exploration of tradeoffs and balance between security guarantee, operational simplicity, and policies acceptable to the operator community. In pervasive computing environments, powerful handheld devices with wireless connections create opportunities for many new nomadic applications. We propose a new service discovery model, called Splendor, supporting nomadic users and services in public environments. Splendor emphasizes security and s Splendor emphasizes security and supports privacy. Location awareness is integrated for location dependent services discovery and is used to lessen service discovery network infrastructure requirements. We analyze the Splendor system performance and provide our experimental results. A public key c Otenko - in Security in the Information Society: This paper describes a role based access control policy template for use by privilege management infrastructures where the roles are stored as X. There is a brief description of the X. A future version will specify it as an XML schema. These are summarised in Table 1 below. It is well known that the Border Gateway Protocol BGP , the IETF standard interdomain routing protocol, is vulnerable to a variety of attacks, and that a single misconfigured or malicious BGP speaker could result in large-scale service disruption. In this paper, we present Pretty Secure BGP psBGP â€"a proposal for securing BGP, including an architectural overview, design details for significant aspects, and preliminary security and operational analysis. A user friendly policy management tool is also being built that will allow non-technical managers to easily specify PERMIS authorisation policies. The Java API is simple to use, comprising of just 3 methods and a constructor. The notion of trust is presented as an important component in a security infrastructure for mobile agents. A trust model that can be used in tackling the aspect of protecting mobile agents from hostile platforms is proposed. We dene several trust relationships in our model, and present a trust We dene several trust relationships in our model, and present a trust derivation algorithm that can be used to infer new relationships from existing ones. An example of how such a model can be utilized in a practical system is provided. Conventional public key infrastructure: It has been conventional wisdom that, for e-commerce to fulfil its potential, each party to a transaction must be confident about the identity of the others. Digital signature technology, based on public key cryptography, has been claimed as the appropriate means of achieving this aim. This paper examines that form of PKI architecture, and concludes that the reason for its failure is its very poor fit to the real needs of cyberspace participants. Its key deficiencies are its inherently hierarchical and authoritarian nature, its unreasonable presumptions about the security of private keys, a range of other technical and implementation defects, confusions about what it is that a certificate actually provides assurance about, and its inherent privacy-invasiveness. Alternatives to conventional Software protection and application security: Understanding the battleground by A. We provide a state-of-the-art explication of application security and software protection. The relationship between application security and data security, network security, and software security is discussed. Three simplified threat models for software are sketched. To better understand w To better understand what attacks must be defended against in order to improve software security, we survey software attack approaches and attack tools. A simplified software security view of a software application is given, and along with illustrative examples, used to motivate a partial list of software security requirements for applications. However, providing security within networked information systems goes far beyond protecting data, cryptographic keying material, and credentials. The transition from a mainframe-based computing infr

Chapter 3 : CiteSeerX â€” Citation Query Understanding Public Key Infrastructure

The Public Key Infrastructure (PKI) is a system for giving you confidence that the certificate your browser has downloaded verifies that the website you're seeing is owned and operated by your bank.

Email Last week we took a look at how public key encryption systems work, and how anyone can send you an encrypted messageâ€”which only you can readâ€”if they have access to your public key. It turns out that the process of getting your public key to people who need to use it is a complex task that involves a combination of trust, third parties, and various other factors which together are known as public key infrastructure. You could make it available for download on your Web site, you could distribute it on a memory stick, or you could simply e-mail it to people. But in practice there is a big problem with that. But if Mallory were smart he would re-encrypt the messages intended for you after he had read them with your real public key and send them on. So how can this problem be overcome? And how can you be sure that any public keys you get hold of really do belong to the people that you think that they do? He then signs a certificate which he attaches to the key that says that he, Solomon, can personally vouch for the fact that the attached key belongs to you. In practice the procedure is slightly different in that it uses something called a hashing function, but the principal is exactly the same. But surely this just pushes the problem back one stage? But how can they know that it is? This means that as long as a public key that you receive is signed by a CA that you have a root certificate for, then you can be sure that the public key belongs to the person it says that it doesâ€"if you are sure that the pre-installed root certificate you have is genuine and you deem the CA to be trustworthy. If the root certificate was included with a software package, then you have to decide whether you trust the maker of the software to have included a genuine root certificate or not. Likewise, you can look at the details of any CA and decide whether you trust them. In this model, you meet face to face with people you know, and get them to sign your public key with their private key confirming that your public key is really yours. The principal then is this: When you get it, you might see that it has been signed as genuine by Bill. If you know Bill and have a copy of his public key that you got from him when you met him face to face, you can easily decide that the key does belong to Carol, because Bill says so and you trust him. Of course the filament of trust could be longer: The more people you trust who confirm that the key is genuine, either directly or indirectly, the better. The important point to remember in the end is that although public key ciphers are extremely secure â€"as far as we knowâ€"public key infrastructure relies on an element of trust:

Chapter 4 : Understanding the Role of the PKI

Public key cryptography, in contrast, uses pairs of keys: a public key that is widely available, and a different private key known only to the person, application or service that owns.

For more information about encryption and pubic key cryptography, see Chapter 7 in my book, Scene of the Cybercrime published by Syngress Media , www. A PKI is not an authentication method; rather it is an infrastructure that uses digital certificates as an authentication mechanism and is built to better manage certificates and their associated keys. A digital certificate is itself a way to reliably identify the user or computer claiming to be the owner of a specific public key. Public key encryption, also called asymmetric encryption, is popular because it is more secure than secret key symmetric encryption. Two mathematically related keys, a public key and a private key, work together, with one used for encrypting and the other for decrypting which one is used for which purpose depends on whether your goal is confidentiality of the data or authentication of the sender. The public key is made known to everyone who wants to engage in encrypted communications with the owner of the key pair. The private key never has to be shared with anyone; it is known only to its owner. This makes for a more secure system than the secret key method in which the same key is used for encrypting and decrypting and thus must be shared between the two communicating parties. The problem with public key encryption is the difficulty of knowing whether a public key is really owned by the person it is claimed to belong to. Thus, a method was needed for verifying the identity of the holder of key pairs. A trusted third party, called a certification authority, issues a certificate associated with a key pair to a user or computer whose identity it has already verified. This works somewhat like the issuance of identification cards by governmental entities or employers. Managing digital certificates and their associated keys is complex, so the PKI was created to provide a framework for the issuance, renewal, revocation and management of certificates. Industry standard PKIs and their certificates are built on the X. A PKI can be implemented within an organization, for the use of the users on its network, or it can be a commercial entity that issues certificates to Internet users, for example. Either way, the PKI consists of the following components: At least one certification authority CA to issue certificates. Policies that govern the operation of the PKI. The digital certificates themselves. Applications that are written to use the PKI. A CA is simply a server that runs some sort of certificate services software. An example is the Microsoft Windows certificate services, which is included with the Windows server operating systems. This CA issues certificates to other CAs, but best practices dictate that it not issue certificates directly to users. Lower level CAs, called subordinate CAs, perform the daily task of issuing user and computer certificates. The root CA is the most trusted, so it should be kept in a very secure physical location or even taken off line when it is not in use. Administrators can assign policies to the CA s that will be used in verifying the identity of users and computers that request certificates. In some cases, certificates are requested by the system without user action; for example, the first time a user attempts to encrypt data on the disk using EFS, an EFS certificate is transparently requested and issued. PKI Policies PKI policies lay out rules governing key security, the process for issuing, renewing and revoking certificates, default certificate lifetimes, and so forth. The CPS addresses issues such as how keys are generated and stored, issuance and revocation of certificates, etc. Issuance, Management and Revocation of Certificates The process involved in issuing and managing certificates is complex. When a request for a certificate is made to a CA, a key pair must be created and signed by the requestor, then the public key is sent to the CA. The CA must verify the signature and identity based on its policies. This creates the certificate, which is then sent back to the requestor. The certificate can then be published. Certificates can be issued for a number of different uses, including smart cards, IPSec, EFS, logon authentication, web authentication, email, and more. You also can export the certificate to back it up, and then import it to restore it. There are many reasons a certificate might need to be revoked. You can think of this as similar to the list of revoked credit cards that credit card companies distributed to merchants in the past. Now the process is usually computerized; the card number is entered and the computer checks it against a database of revoked cards that is updated automatically. Applications Applications must be PKI-aware in order to work with the certificates and use them for authentication purposes. Summary The Public Key Infrastructure is a framework for using digital certificates and their associated keys to verify the identity of users and computers to other users, computers and applications. PKIs are important elements in network and Internet security because many communications, such as business and e-commerce transactions, are dependent on a reliable method to identify the parties to the transaction. The PKI is not itself an authentication method, but is a system for issuing, managing and revoking the digital certificates and key pairs that are used to authenticate users and computers within a private network or across the Internet. The components of a PKI include special servers called certification authorities, policies governing how the CAs issue, manage and revoke certificates and store keys, digital certificates and their keys, and applications that are able to use the PKI. This is only a basic overview of how a standard PKI works. In a multi-CA enterprise network, the infrastructure may be complex, with several levels in the CA hierarchy and different CAs specializing in the issuance of specific types of certificates.

Chapter 5 : Digital Certificates with PKI Overview - TechLibrary - Juniper Networks

Understanding Public-Key Infrastructure has 7 ratings and 0 reviews. This book is a tutorial on, and a guide to the deployment of, Public-Key Infrastruct.

Next page A PKI is the service framework needed to support large-scale public keybased technologies. The PKI is a set of all the technical, organizational, and legal components needed to establish a system that enables large-scale use of public key cryptography to provide authenticity, confidentiality, integrity, and nonrepudiation services. Two very important terms need to be defined when talking about a PKI: A certificate is a document that, in essence, binds the name of the entity and its public key that has been signed by the CA, so that every other entity will be able to trust it. Note Certificates are not secret information and do not need to be encrypted in any way. The idea is not to hide anything but to ensure the authenticity and integrity of the information contained in the certificate. The CRL is a list of certificates that should not be trusted anymore. Examples of when a certificate is added to the CRL "revoked" include exposure or loss of the private key. A PKI user who receives a certificate should verify the CRL to ensure that the received certificate is not on the list of revoked certificates. Many vendors offer CA servers as a managed service or as an end-user product: This format is already extensively used in the infrastructure of the Internet. The most important pieces of information contained in the certificate are these: Validity period of the certificate Signature Algorithms Self-Signed Certificates In a PKI system, all public keys are distributed in a form of a certificate, including the certificate of the trusted introducer, the CA. The obvious question is: Who signs the certificate of the CA, if it is itself the signer of all other certificates? In reality, the CA also issues a certificate to itself, just to have a consistent format for distributing its public key. This process is how the end entities obtain the public key of the CAby obtaining its self-signed certificate. The signature of a self-signed certificate of the CA cannot be verified using the standard method verification by using the public key of the signer because that public key should actually be protected by the signature. Therefore, other methods such as manual verification are needed to ensure the authenticity of a CA certificate. Sometimes end entities also sign their own certificates. For example, a web server could generate a private and public RSA key and sign its public key with its private key to create a self-signed certificate. However, how does the web browser verify the presented certificate, if it was not issued signed by a known CA for which the web browser has a locally available certificate? This web server certificate, therefore, cannot be accepted automatically, but needs to be verified using some other method such as the manual, out-of-band verification that is also used in pre-PKI protocols. Organizations will typically use self-signed certificates internally to save on the continual cost of obtaining certificates from a publicly known Certificate Authority CA. This causes web browser clients to initially display warning messages when connecting to a server using a self-signed certificate.

Chapter 6 : News, Tips, and Advice for Technology Professionals - TechRepublic

Understanding public key infrastructure (PKI) As the name suggests a Public Key Infrastructure is an infrastructure that uses digital certificates as an authentication mechanism and is designed to manage those certificates and their associated keys.

Understanding the Public Key Infrastructure Richard Sinn This article introduces the concepts of Internet security, secret key and public key encryption, digital signatures, the Public Key Infrastructure PKI , and certificate definition. Then, an API example is provided to show how developers can obtain a Web certificate using a program. Internet security A lot of e-business functions can now be done online. Many users have set up accounts to check their credit card statements, shop, or pay various bills online. For most of us, as long as the Web sites are brand names and we have a user ID and password, we log in and do our online transactions. But how secure are Internet transactions? The following are some of the common Internet security issues: Data thus remains intact, but privacy is compromised. Credit card, social security, and account numbers can be stolen through eavesdropping. Data tampering A device is placed to intercept the data in transit. Data is then altered or replaced before it is sent to the recipient. For example, someone can alter the transfer amount to and from a bank account. Entity repudiation Data is passed to a person who poses as the intended recipient. Thus, all data is sent to this newly configured machine. A set of well-established techniques and standards known as public key cryptography helps avoid the Internet security complications mentioned above. To communicate data confidentially between two persons, one of the common ways is to perform transformation of data to gibberish-encryption. Anyone eavesdropping on the data would not be able to understand the gibberish. There are two kinds of encryption: Secret key encryption Secret key encryption is also called symmetric key encryption. In this encryption method, a shared secret the secret key is given to both the sender and the recipient before the data transit. The secret key specifies exactly how the transformation to and from gibberish is to be accomplished. The transformation to gibberish is called encryption and the transformation back to the original text is called decryption. The entire encryption and decryption algorithm is called a cipher, and the encryption and decryption process uses the same secret key. Figure 1 shows the secret key process. Secret key process 2. Public key encryption Although symmetric ciphers have advantages such as a small implementation size and fast encryption and decryption speeds, they suffer from significant drawbacks in the Internet environment. They are as follow: The need for secret key exchange for unknown entities Symmetric ciphers rely completely on the fact that both the sender and the recipient have the same secret key. Thus, the secret key has to be shared before the secure Internet communication happens. This additional step can be extremely difficult or highly inconvenient in most Internet environments. For example, how do business to consumer B2C Web sites exchange a secret key for the first-time consumer two previously unknown entities? As a matter of fact, this additional step of sharing a secret key poses the main barrier for Internet communication between two unknown entities. Security scalability In a community of symmetric cipher users, each individual has to keep secret keys to communicate with all the users. Ninety-nine keys are for decryption from other users; one key is for the individual to encrypt data. In other words, we have to keep track of a lot of keys when using symmetric cipher for communication. Fortunately, there is a technology called public key encryption also called asymmetric encryption that can solve both of these problems. Public key encryption involves a pair of keys a public and a private key associated with an entity that needs to authenticate its identity electronically or to sign or encrypt data. The public key is published, and the corresponding private key is kept secret somewhere in your computer. Microsoft stores it in the Windows registry, and Netscape stores the key in its certificate store. Data encrypted with your public key can be decrypted only with your private key and vice versa. Public key process Figure 2 demonstrates that you can freely distribute a public key. Only you, with the private key, are able to read data encrypted using the corresponding public key. In general, to send encrypted data over the Internet, you perform the following steps: Using these guidelines, you can send secret data to individuals on the Internet without exchanging anything before the transaction. Public key encryption is great, but it still has its own shortcoming. Compared with symmetric key encryption, public key encryption is based on algorithms created by RSA Data Security, which require a lot more computation and might not be appropriate for transferring large amounts of data. So, how do we take the advantages of both the symmetric key and the public key encryption? The Secure Socket Layer SSL protocol approach is to use the slower public key encryption to send a symmetric key a small piece of data between two communication parties; the parties then use the symmetric key a faster way to encrypt additional data that flows between them. Digital signatures Encryption and decryption address the problem of data eavesdropping that we mentioned before. Digital signatures can be used to address entity repudiation and data tampering. Digital signature is based on one-way hash, which is a mathematical function that provides the following properties: The hashed data provides a unique value. Any change in the original data even one character results in a different hash value. The content of the hashed data cannot be deduced from the hash. Thus, the hashing procedure is "one-way" only. Using one-way hash, producing a digital signature is a two-step process: The signer one-way hashes the data to a fixed-size value. The signer then subjects the hashed value to a private key encryption. Verification is a similar process: The verifier uses the same one-way hash algorithm on the transmitted data to generate a fixed-size hash value. If the two hash values in step 1 and 2 match, signature verification is successful. If they do not match, signature verification fails. PKI advantages An infrastructure is a foundation or underpinning for a large environment. One good example is the electric power infrastructure. The power plant, power grid, wiring, and other devices form the electric power infrastructure that enables a user to just plug in electronic equipment to get the voltage and current needed for operation. Thus, the principle is that the infrastructure provides services so that entities can simply tap into and use it on an as-needed basis. PKI is an infrastructure built using public key cryptography that allows users to tap in and take advantage of the security PKI offers. PKI provides three primary services: Authentication - The assurance to the recipient that the sender is who the sender claims to be. This is achieved by means of digital signature. Integrity - The assurance to the recipient that data has not been altered during Internet communication. Confidentiality - The assurance to a sender and recipient that no one can read a particular piece of data except the intended recipient. This is achieved by means of encryption. What is a Web certificate? A Web certificate is an electronic document used to identify an individual, a company, or any other entity. In the Internet world, most certificates follow the X. A trusted third party called Certificate Authority CA issues certificates. A Web certificate is digitally signed by the issuer CA and is valid for a certain period mostly one year. Figures 3 and 4 show what a certificate looks like using Windows graphical interfaces. General certificate information Figure 4: Detailed certificate information PKI uses certificates to address the problem of entity repudiation impersonation. Certificates help prevent the use of fake public keys for impersonation. Only the public key associated and certified by the certificate works with the corresponding private key possessed by the entity identified by the certificate. How do we get one? First, we have to find a CA that issues certificates. In the Internet model, we have to find a public CA, instead of one that only works in a private network. This is the request for a certificate. The output will be in PKCS7 format. This article gives an introduction to different technologies used in PKI and some pointers on where to look for additional information.

Chapter 7 : Understanding and Designing a Public Key Infrastructure

An Introduction to the Public Key Infrastructure (PKI) It has grown more important to ensure the confidentiality and integrity for data communication where an organization's network contains intranets, extranets, and Internet Web sites.

Design[edit] Public key cryptography is a cryptographic technique that enables entities to securely communicate on an insecure public network, and reliably verify the identity of an entity via digital signatures. The PKI creates digital certificates which map public keys to entities, securely stores these certificates in a central repository and revokes them if needed. Methods of certification[edit] Broadly speaking, there have traditionally been three approaches to getting this trust: When the CA is a third party separate from the user and the system, then it is called the Registration Authority RA , which may or may not be separate from the CA. According to NetCraft, [12] the industry standard for monitoring Active TLS certificates, states that- "Although the global [TLS] ecosystem is competitive, it is dominated by a handful of major CAs â€” three certificate authorities Symantec , Comodo , GoDaddy account for three-quarters of all issued [TLS] certificates on public-facing web servers. The top spot has been held by Symantec or VeriSign before it was purchased by Symantec ever since [our] survey began, with it currently accounting for just under a third of all certificates. A single sign-on server will issue digital certificates into the client system, but never stores them. Users can execute programs, etc. It is common to find this solution variety with X. Web of trust An alternative approach to the problem of public authentication of public key information is the web-of-trust scheme, which uses self-signed certificates and third party attestations of those certificates. The singular term "web of trust" does not imply the existence of a single web of trust, or common point of trust, but rather one of any number of potentially disjoint "webs of trust". If the "web of trust" is completely trusted then, because of the nature of a web of trust, trusting one certificate is granting trust to all the certificates in that web. As time goes on, you will accumulate keys from other people that you may want to designate as trusted introducers. Everyone else will each choose their own trusted introducers. And everyone will gradually accumulate and distribute with their key a collection of certifying signatures from other people, with the expectation that anyone receiving it will trust at least one or two of the signatures. This will cause the emergence of a decentralized fault-tolerant web of confidence for all public keys. Simple public key infrastructure[edit] Another alternative, which does not deal with public authentication of public key information, is the simple public key infrastructure SPKI that grew out of three independent efforts to overcome the complexities of X. SPKI does not associate users with persons, since the key is what is trusted, rather than the person. SPKI does not use any notion of trust, as the verifier is also the issuer. This is called an "authorization loop" in SPKI terminology, where authorization is integral to its design. Since blockchain technology aims to provide a distributed and unalterable ledger of information, it has qualities considered highly suitable for the storage and management of public keys. A Certificate Authority paradigm has these issues regardless of the underlying cryptographic methods and algorithms employed, and PKI that seeks to endow certificates with trustworthy properties must also address these issues. Blockchain as a technology has its own restrictions such as a low throughput which cause possibility of a long response time and high transaction fees. Building an independent distributed PKI protocol with custom consensus and cryptocurrency economy could solve the issue [17]. This section does not cite any sources. Please help improve this section by adding citations to reliable sources. Unsourced material may be challenged and removed. January Learn how and when to remove this template message Developments in PKI occurred in the early s at the British intelligence agency GCHQ , where James Ellis , Clifford Cocks and others made important discoveries related to encryption algorithms and key distribution. The public disclosure of both secure key exchange and asymmetric key algorithms in by Diffie , Hellman , Rivest , Shamir , and Adleman changed secure communications entirely. With the further development of high-speed digital electronic communications the Internet and its predecessors , a need became evident for ways in which users could securely communicate with each other, and as a further consequence of that, for ways in which users could be sure with whom they were actually interacting. Assorted cryptographic protocols were invented and analyzed within which the new cryptographic primitives could be effectively used. With the invention of the

World Wide Web and its rapid spread, the need for authentication and secure communication became still more acute. Commercial reasons alone e. Vendors and entrepreneurs saw the possibility of a large market, started companies or new projects at existing companies , and began to agitate for legal recognition and protection from liability. An American Bar Association technology project published an extensive analysis of some of the foreseeable legal aspects of PKI operations see ABA digital signature guidelines , and shortly thereafter, several U. Consumer groups raised questions about privacy , access, and liability considerations, which were more taken into consideration in some jurisdictions than in others. The enacted laws and regulations differed, there were technical and operational problems in converting PKI schemes into successful commercial operation, and progress has been much slower than pioneers had imagined it would be. By the first few years of the 21st century, the underlying cryptographic engineering was clearly not easy to deploy correctly. Operating procedures manual or automatic were not easy to correctly design nor even if so designed, to execute perfectly, which the engineering required. The standards that existed were insufficient. PKI vendors have found a market, but it is not quite the market envisioned in the mids, and it has grown both more slowly and in somewhat different ways than were anticipated. Uses[edit] PKIs of one type or another, and from any of several vendors, have many uses, including providing public keys and bindings to user identities which are used for: In both of these, initial set-up of a secure channel a " security association " uses asymmetric key â€”i. Mobile signatures are electronic signatures that are created using a mobile device and rely on signature or certification services in a location independent telecommunication environment [20] Internet of things requires secure communication between mutually trusted devices. A public key infrastructure enables devices to obtain and renew X certificates which are used to establish trust between devices and encrypt communications using TLS Open source implementations[edit] OpenSSL is the simplest form of CA and tool for PKI. It is a toolkit, developed in C, that is included in all major Linux distributions, and can be used both to build your own simple CA and to PKI-enable applications. It can be used to set up a CA both for internal use and as a service. XCA is a graphical interface, and database. Criticism[edit] Some argue that purchasing certificates for securing websites by SSL and securing software by code signing is a costly venture for small businesses. Presently Symantec holds a major share in PKI certificate market which sold one third of all certificates issued globally in Current web browsers carry pre-installed intermediary certificates issued and signed by a Certificate Authority. This means browsers need to carry a large number of different certificate providers, increasing the risk of a key compromise. When a key is known to be compromised it could be fixed by revoking the certificate, but such a compromise is not easily detectable and can be a huge security breach. Browsers have to issue a security patch to revoke intermediary certificates issued by a compromised root certificate authority.

Chapter 8 : Public key infrastructure - Wikipedia

Public Key Infrastructure is a foundation for trusted communication. When we talk about SSL, Certificate Authorities, browser trust, and all the other regular topics on this blog, we are talking about components of a system called the Web PKI.

These fields are described in the following sections. This optional configuration allows certificate validation to succeed only if the certificate chain received from the peer contains at least one policy OID that is configured on the SRX Series device. On the SRX Series device, policy OIDs are configured in an IKE policy with the policy-oids configuration statement at the [edit security ike policy policy-name certificate] hierarchy level. You can configure up to five policy OIDs. Note that the policy-oids field in a certificate is optional. A certificate can contain one or more certificate policy OIDs. For policy validation to succeed, there must be a common policy OID in the certificate chain. Because the policy OID P2 is common to the certificates being validated, policy validation succeeds. Policy Validation with requireExplicitPolicy Field The optional skipCerts field in an intermediate CA certificate indicates the number of certificates, including the current CA certificate, that are to be excluded from policy validation. If skipCerts is 0, policy validation starts from the current certificate. If skipCerts is 1, the current certificate is excluded from policy validation. The value of the skipCerts field is checked in every intermediate CA certificate. If a skipCerts value is encountered that is lower than the current number of certificates being excluded, the lower skipCerts value is used. However, the skipCerts value is checked in every intermediate CA certificate in the chain. The skipCerts value in Int-CA-2 is 2, which is lower than 12, so now 2 certificates are skipped. The number of intermediate CAs can grow depending upon the deployment scenario. Path length validation provides a mechanism to limit the number of intermediate certificates involved in certificate validation. The value of path-length indicates the number of non-self-signed intermediate CA certificates allowed for certificate validation. The last certificate, which is generally the EE certificate, is not included in the path limit. If the root certificate contains a path-length value of 0, no intermediate CA certificates are allowed. If the path-length value is 1, there can be 0 or 1 intermediate CA certificates. The path length validation always begins with the self-signed root certificate. The path limit is decremented by 1 at each intermediate certificate in the chain. If an intermediate certificate contains a path-length value less than the current path limit, the new limit is enforced. On the other hand, if the path-length value is larger than the current path limit, it is ignored. The path-length value in Root-CA is 10, therefore the initial path limit of non-self-signed intermediate CA certificates allowed for certificate validation is At Int-CA-1, the path limit is or 9. The path-length value in Int-CA-1 is 4, which is less than the path limit of 9, so the new path limit becomes 4. At Int-CA-2, the path limit is or 3. The path-length value in Int-CA-2 is 5, which is larger than the path limit of 3, so it is ignored. At Int-CA-3, the path limit is or 2. The path-length value in Int-CA-3 is 20, which is larger than the path limit of 2, so it is also ignored. EE Certificates For EE certificates, if the key usage field is present but the certificate does not contain digitalSignature or nonrepudiation flags, the certificate is rejected. If the key usage field is not present, then key usage is not checked. Certificate Signature Validation The keyCertSign flag indicates that a CA certificate can be used for certificate signature validation. If this flag is not set, certificate validation is aborted. Downloaded CRL files must be verified before they are downloaded into the device. If this flag is not present, the CRL is discarded. Attribute values encoded in different ASN. Attribute values encoded in PrintableString types are not case-sensitive. These attribute values are compared after removing leading and trailing white spaces and converting internal substrings of one or more consecutive white spaces to a single space. Attribute values encoded in types other than PrintableString are case-sensitive.

Chapter 9 : Understanding the Public Key Infrastructure

A public key infrastructure (PKI) is a set of roles, policies, and procedures needed to create, manage, distribute, use, store, and revoke digital certificates and manage public-key encryption.

Bangalore Public Key Infrastructure Public key infrastructure is an architecture which supports mechanisms like integrity and confidentiality. It is heavily used in e-commerce, where business transactions are frequent and should be much secured. In fact business transaction starts only after the two parties authenticate each other and are assured about each other. This, no doubt, involves risk of an unwanted entity getting access to the key and thus breaking the security. However, in PKI, there are two keys involved- one is meant to be publicly distributed, called as Public key and the other one â€”Private key, which should be held only by the correct user. The public key and the private key are mathematically related to each other. It is not feasible to derive the corresponding Private Key from the public key and vice versa. The firm is supposed to encrypt the information using the public key of the target customer. This ensures the data is provided by the correct user since he only possesses the private key. This is the difference between encryption mechanism, where the public key is needed and signature process which needs private key. In case of any modifications or tampering of the data, the digital signature is no longer valid. This ensures that the data is received as it is and is not corrupt. The RSA public key consists of modulus n The public exponent e. Public exponent is needed for encryption. The private key consists of modulus n The private exponent d. Private exponent is needed for decryption. Briefly, the algorithm involves selecting and multiplying two prime numbers and through added mechanisms, deriving a set of two numbers that are the public key and the private key. RSA is comparatively much slower than the symmetric key algorithms. Thus it is not preferred to use for encrypting or decrypting the large amount of data. In my forthcoming articles, we shall discuss the implementation of the previous mentioned algorithms using the Bouncy Castle APIs.