Syracuse University

IST-623 Lab#2

Using Encryption to Enhance Confidentiality and Integrity

ThulasiRam RuppaKrishnan

IST 623

Professor Joon Park

Contents

Lab Environment

Using Encryption to Enhance Confidentiality and Integrity

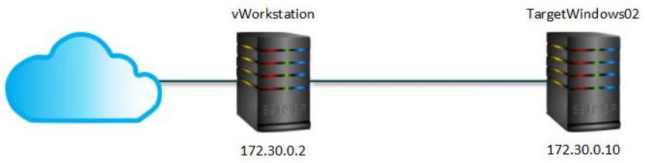
Fundamentals of Information Systems Security - Lab 7

Start	4:36 AM
End	6:36 AM

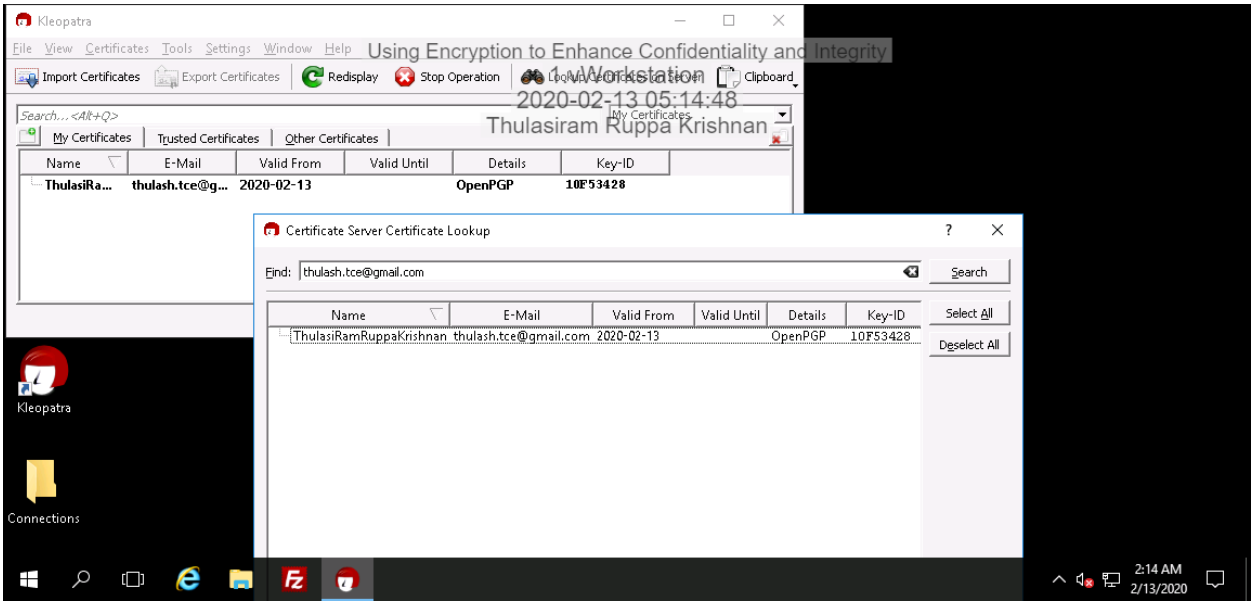
LAB IN USE

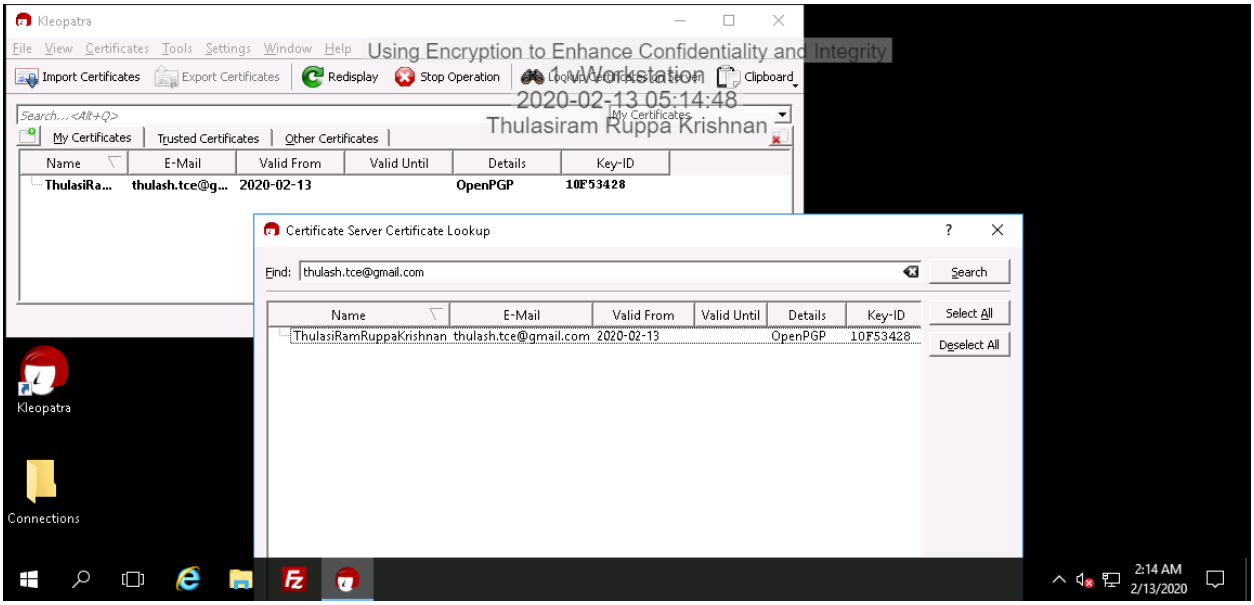
EXTEND LAB END LAB

Topology

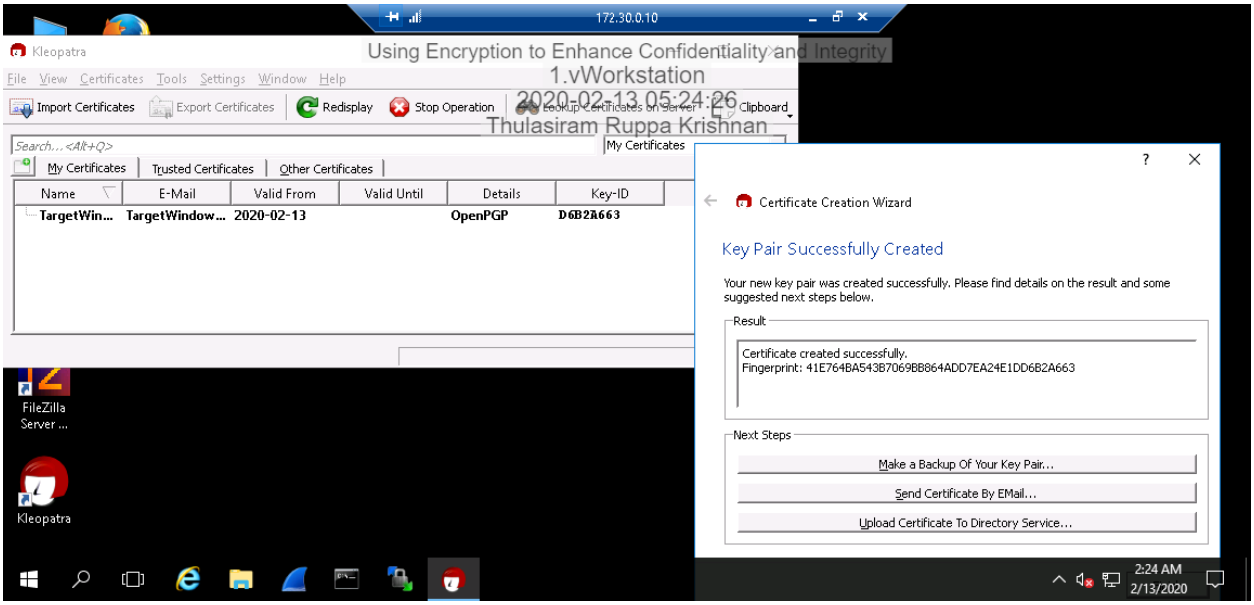


Part 1: Create a Public and Private Key Pair for the Sender



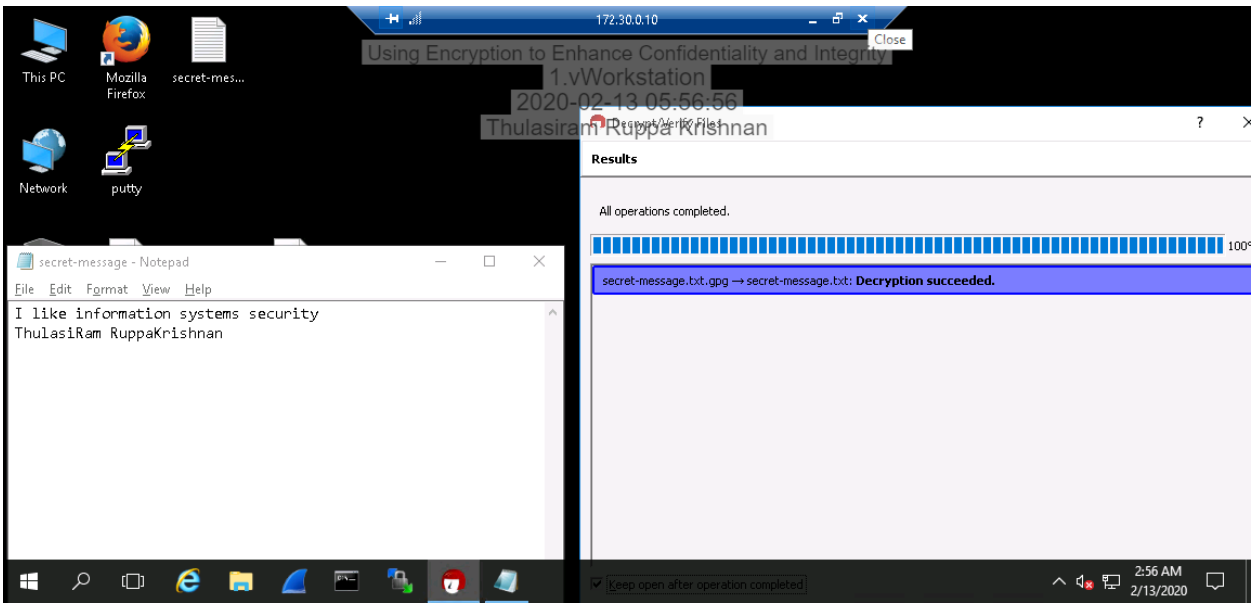


Part 2: Create a Public and Private Key Pair for the Receiver



Part 3: Transfer and Import a Public Key from the Receiver

Part 4: Encrypt and Decrypt a File from the Sender



Part 5: Challenge Question

Difference and tradeoffs between X.509 and PGP certificate types?

PGP certificates differ from X.509 certificates, as seen by the general public, in two ways:

- PGP certificates are issued (signed) by normal people while the popular impression is that X.509 certificates must be issued by a professional CA, and
- PGP implements a security fault tolerance mechanism, called the Web of Trust, that was designed to compensate for the fact that issuers were not specially protected or professional.

X.509

X.509 defines a framework for the provision of authentication services to its users. Each certificate contains the public key of the user and it is signed with the private key of a trusted certification authority (CA). In addition, X.509 defines alternative authentication protocols based on the use of public-key certificates. X.509 is an important standard because of its certificate structure and various authentication protocols defined in X.509 are used in a variety of contexts. X.509 uses the concept of public-key cryptography and digital signatures. An X.509 Certificate is issued by the Certification Authority (CA) and is duly signed with the private key of CA.

It includes:

• Owner's public key

• Owner's name

• Expiration date of the public

• Name of the issuer (the CA that issued the Digital Certificate)

• Serial number of the Digital Certificate

• Digital signature of the issuer

The most widely accepted format for digital certificates is defined by the CCITT X.509 international standard. And the most widely used standard for digital certificates is X.509 certificate. Digital Certificates are the framework for identifying information and bind their identities with public keys. An electronic signature duly issued by the certifying authority that shows the authority of the person who is signing the e-form.

Each Certificate contains the public key of the user and is signed with the private key of the certification authority.

B. Security of document requires:

• Authenticity

• Confidentiality

• Integrity

• Non-repudiation

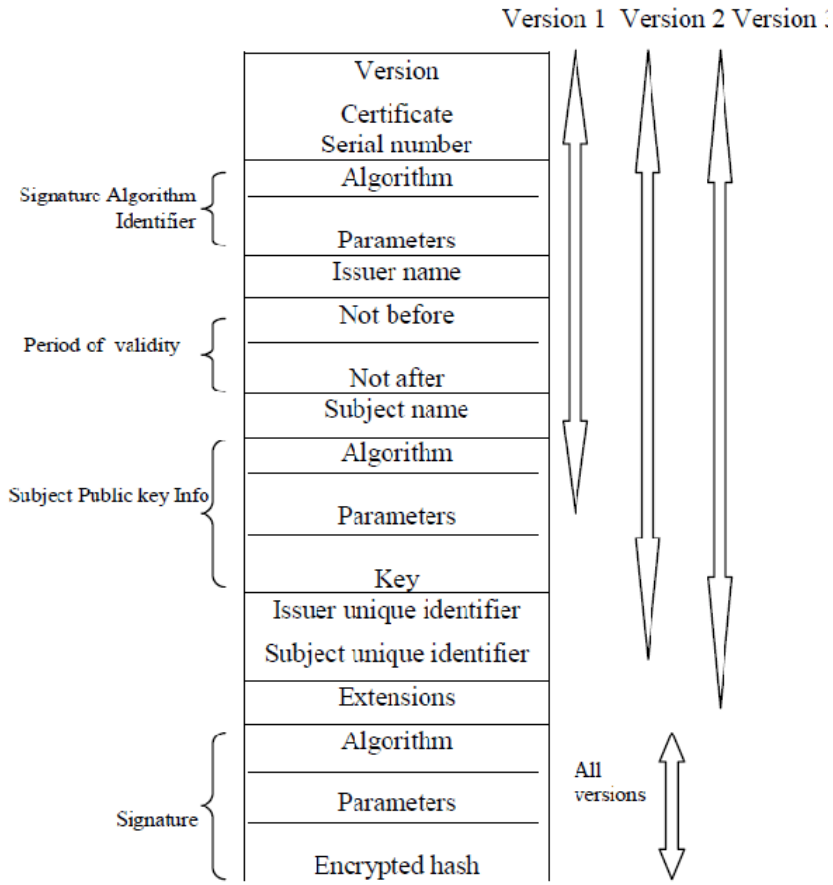
C. Generation of a public-key certificate

The heart of the X.509 scheme is the public-key certificate associated with every user. These certificates are generated by a trusted certification authority (CA) and they are recorded in the directory by the CA or by the user to whom the certificate is issued. The directory server is not responsible for the creation of public key or for the certification function rather it merely provides an easily accessible location for users to obtain certificates.

The assumption here is that a certificate binds a key to a person. That person is identified by a globally unique and globally meaningful name, according to the original X.500 work. However, X.500 achieved that global uniqueness by having a single directory root. That assumption is built into how the certificates are constructed and used. Unfortunately, the single directory root assumption is not valid and probably never will be. Therefore, the identifiers used (Distinguished Names {DNs}) are local names.

The CA hierarchy is designed to bind these assumed global names to keys. The CPS (Certification Practices Statement) is designed, in part, to show to the relying party what quality of job that CA does. [The CPS has evolved to a document that spends a sizable fraction of its content spelling out why it is not the fault of the CA when the relying party's expectations are not met and to many people, that is the only apparent function of the CPS.]

The original design, from X.500, was for a single global naming root. Under the early PEM work, a single certification root was also assumed. Either single root is politically impossible. To accommodate the inevitable multiple certification roots, cross-certification in a variety of forms is being developed. It is not clear how those in favor of cross-certification will resolve the lack of a common naming root when two ID hierarchies are linked by cross-certification. Those two hierarchies could have issued identical names to different keys, belonging to different keyholders.



The assumption behind X.509 (inherited from X.500) is that the global identifier [DN] is somehow inherently bound to a person and that everyone who might use the certificate will correctly map from that name to the named person or from the named person to the name. The probability of error in performing this mapping affects the system wide probability of error, alongside any cryptographic errors. It needs to be extremely small in order not to overwhelm the cryptography used and become the dominant source of error.

It is very hard to do this mapping from name to person when one has a large domain of named people. It is even harder to do the mapping from person to name. Failures in either name mapping are likely to dominate the probability of security error in any system using such names as part of a security decision. Specifically, Carol the CA assigns some constructed name to Bob. Alice is then expected to look at that name and understand which Bob is being named, without any prior communication with either Carol or Bob on this topic. (That usage of names was inherited from the original paper on public key cryptography, Diffie and Hellman, "New Directions in Cryptography", 1976.) We already know that with e-mail names, people often make incorrect guesses about the linkage between name and person and, as a result, e-mail goes to the wrong person. There is no reason to believe that this association between name and person will be any more accurate with X.500 Distinguished Names. In fact, since Microsoft Outlook (Exchange) uses X.500 Distinguished Names for e-mail and we have seen

a high percentage of mistakes in mail addressing under Outlook/Exchange (in domains of as few as 50,000 people), there is reason to believe that the DN yields at least as high an error rate as any other e-mail name, and possibly a significantly higher rate. Full scientific studies of those rates have yet to be done.

PGP

PGP combines symmetric and asymmetric cryptography.

The user generates a pair: (public key, private key) that is associated with his unique ID. Public keys are stored on public key rings and private keys are stored on private key rings. On the sender's side, PGP creates a session key, which is random number generated by the keystroke characteristics of a user. Once the data is encrypted with this key, the session key is encrypted with the recipient's public key and sent together with the cipher text to the recipient. The recipient's copy of PGP uses her private key to recover the session key, which then allows the recipient to decrypt the cipher text. PGP uses pass phrase to encrypt the private key on its owner's machine. Pass phrase is longer and more complicated version of the password. The private key is encrypted on the disc using a hash of the pass phrase as a secret key. In order to use her private key, user has to decrypt it using the pass phrase. The distribution of public keys is usually done by key servers. They are mirrored at various locations around the world. They possess the recipients' public keys and on the demand of sender, they give the sender the recipients' public key.

PGP can also be used for 4 things:

• To encrypt a message or file so that only the intended recipient can decrypt and read it. The sender, after signing with PGP, also provides guarantee to the recipient, that the message have come from the authorized sender and not from any unauthorized person.

• Clear signing a plain text message guarantees that it can only have come from the sender and not an impostor.

• Encrypting computer files so that they can't be decrypted by anyone other than the person who encrypted them.

• Really deleting files (i.e. overwriting the content so that it can't be recovered and read by anyone else) rather than just removing the file name from a directory/folder.

A PGP certificate includes (but is not limited to) the following information:

• PGP version number—PGP version number identifies which version of PGP was used to create the key associated with the certificate.

• Certificate holder's public key— It holds the public key , together with the algorithm of the key such as RSA, Elgamal or DSA.

• Certificate holder’s information— This is the information about the user, such as his or her name, user ID, e-mail address, ICQ number, photograph, and so on.

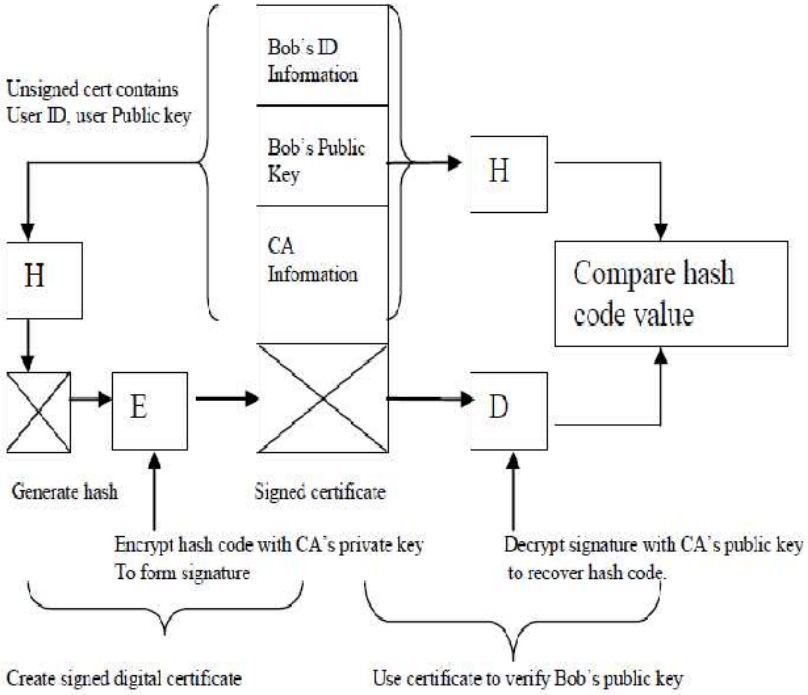
• Digital signature of the certificate owner— It is also called a self-signature. It is the signature which uses the corresponding private key of the public key associated with the certificate.

• Validity period— It is the certificate’s start date/time and expiration date/time which indicates when the certificate will get expire. If the key pair contains sub keys, then this includes the expiration of each of the encryption sub keys as well. Sub keys enable convenient use of separate keys for signing and encryption

• Preferred symmetric encryption algorithm for the key— This indicates the encryption algorithm through which the certificate owner prefers to get information encrypted. The supported algorithms can be CAST, IDEA, Triple-DES, and Blowfish etc.

The assumption here is that a certificate binds a key to a person. That person is identified by a name chosen by the keyholder himself, called the UserID. These names are assumed to be globally unique and meaningful, by including both a common name and an e-mail name in the key's UserID. The e-mail name provides global uniqueness and some measure of meaningfulness. Nothing ensures honesty on the part of the keyholder in choosing his UserID. One must look instead at who besides the keyholder himself signed (attested to) that claimed binding and how many such people have made that attestation, in order to decide how reliable that choice of UserID might be.

Certificates are not issued by CAs that have strict business rules, documented in some CPS. Rather, certificates are issued by normal people who are probably more fallible than one would expect a professional CA to be. To make these certified bindings more reliable, PGP incorporates a security fault tolerance feature called the Web of Trust. Under the Web of Trust, multiple different keyholders sign each certificate (each binding between UserID and key), attesting to the validity of that binding. The assumption is that these different keyholders are independent so that even if one of them makes a bad judgment, they won't all do so. It is also assumed that no false binding will have more than the specified web of trust number of signatures.



People rarely understand or use the Web of Trust. However, it has the interesting design feature that the verifier (what X.509 calls the Relying Party) sets the level of trust in keys -- and can in principle demand some number of independent signatures on a PGP certificate before that binding is considered valid.

Working against this theory of operation is the fact that we cannot prove independence of keys, since we have no means of determining whether two different private keys are controlled by the same one person. The relying party can enforce that independence by assigning trust only to one key per person and only to people who treat certificate signing responsibly, but there is nothing to force a PGP user to go to that trouble.